DECnet–20 System Manager's and Operator's Guide

AA–J678A–TM, AD–J678A–T1

June 1983

This manual describes the operational, control, and monitoring functions of DECnet–20 and the TOPS–20 PSI product. These products together with TOPS–20, provide the DECSYSTEM–2040S/2060 computers with a communications interface to DIGITAL's corporate network, DECnet, and to Public Packet Switching Networks, PPSNs.

OPERATING SYSTEM: TOPS–20 V5.1

SOFTWARE: DECnet–20 V3.0
TOPS–20 PSI V1.0
GALAXY V4.2

Software and manuals should be ordered by title and order number. In the United States, send orders to the nearest distribution center. Outside the United States, orders should be directed to the nearest DIGITAL Field Sales Office or representative.

Northeast/Mid–Atlantic Region

Digital Equipment Corporation
PO Box CS2008
Nashua, New Hampshire 03061
Telephone:(603)884–6660

Central Region

Digital Equipment Corporation
Accessories and Supplies Center
1050 East Remington Road
Schaumburg, Illinois 60195
Telephone:(312)640–5612

Western Region

Digital Equipment Corporation
Accessories and Supplies Center
632 Caribbean Drive
Sunnyvale, California 94086
Telephone:(408)734–4915

First Printing, December 1982
Updated, June 1983

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

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

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

The following are trademarks of Digital Equipment Corporation:

digital™

DEC	MASSBUS	UNIBUS
DECmate	PDP	VAX
DECsystem–10	P/OS	VMS
DECSYSTEM–20	Professional	VT
DECUS	Rainbow	Work Processor
DECwriter	RSTS	
DIBOL	RSX	

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

UPDATE NOTICE

DECnet–20 System Manager's and Operator's Guide

AD–J678A–T1

June 1983

Insert this Update Notice in the *DECnet–20 System Manager's and Operator's Guide* to maintain an up-to-date record of changes to the manual.

Changed Information

The changed pages contained in this update package reflect information on the TOPS–20 PSI product.

The instructions for inserting this update start on the next page.

digital software

INSTRUCTIONS
AD–J678A–T1

The following list of page numbers specifies which pages are to be placed in the *DECnet–20 System Manager's and Operator's Guide* as replacements for, or additions to, current pages.

Title page Copyright page	3–27 3–32	A–13 A–15
iii xiv	7–1 7–2	B–1 B–4
1–7 1–12.1	7–9 7–10	C–3 C–4
2–1 2–18	7–13 7–14	C–11 Blank
3–1 3–4	7–17 7–18	E–1 E–4
3–7 3–14.2	7–49 7–50	Entire Appendix F
3–17 3–22.1	8–1 8–23	G–1 G–12
	A–3 A–6	Entire Index

KEEP THIS UPDATE NOTICE IN YOUR MANUAL TO MAINTAIN AN UP-TO-DATE RECORD OF CHANGES.

TYPE AND IDENTIFICATION OF DOCUMENTATION CHANGES.

Five types of changes are used to update documents contained in the software manuals. Change symbols and notations are used to specify where, when, and why alterations were made to each update page. The five types of update changes and the manner in which each is identified are described in the following table.

The Following Symbols and/or Notations	Identify the Following Types of Update Changes
1. Change bar in outside margin; version number and change date printed at bottom of page.	1. Changes were required by a new version of the software being described.
2. Change bar in outside margin; change date printed at bottom of page.	2. Changes were required to either clarify or correct the existing material.
3. Change date printed at bottom of page.	3. Changes were made for editorial purposes but use of the software is not affected.
4. Bullet (•) in outside margin; version number and change date printed at bottom of page.	4. Data was deleted to comply with a new version of the software being described.
5. Bullet (•) in outside margin; change date printed at bottom of page.	5. Data was deleted to either clarify or correct the existing material.

CONTENTS

CONTENTS (Cont.)

FIGURES

TABLES

PREFACE

This manual describes the operational, control, and monitoring functions of TOPS-20 DECnet-20 Version 3.0. Both tutorial and reference information are included.

The audience addressed is:

- The DECnet-20 System Manager
- The user with OPERATOR privileges who is engaged in network management and operational tasks under the direction of the system manager

Readers are assumed to be experienced in TOPS-20 operations and familiar with computer networks. Previous experience with DECnet is not assumed. However, if you have not yet read the manual, Introduction to DECnet, do so before continuing with this manual.

This manual contains two parts as follows:

Part I, Introduction, contains two chapters.

Chapter 1 briefly describes the DIGITAL Network Architecture (DNA) that serves as a model for the design of networks of DIGITAL computers. Emphasis is on the relationships between the Network Management Layer and other functional layers. The DECnet-20 implementation of DNA is described.

Chapter 2 defines the basic network concepts in relation to the operation of the DECnet-20 network.

Part II, DECnet Operation, contains five chapters.

Chapter 3 includes the minimum information needed by operators responsible for keeping the network up and running. Procedures are explained, and examples are given for such tasks as manual loading of the DN20 front end, controlling line states, monitoring the network, and recognizing potential problems.

Chapter 4 is a tutorial on NCP, the control language that is the operator's and manager's interface to the network control program, NMLT20.

Chapters 5, 6, and 7 describe the syntax and function of NCP commands processed, respectively, by OPR, the local node's NMLT20, and the remote node's Network Management Program.

Chapter 8 describes X.25 NCP commands for using Public Packet Switching Networks (PPSNs) if the Packetnet System Interface (PSI) Version 1.0 software option is included. If the PSI software option is not included, you can use the "TELL" prefix (or SET EXECUTOR command) to send these commands to nodes that support the X.25 facility. If the PSI software is not included and your network does not include nodes that support X.25, you can ignore this chapter.

Six Appendixes complete the manual.

Appendix A summarizes all parameters included in the Network Management V3.0 NCP commands. There is a separate table for each of the four entities: NODE, CIRCUIT, LINE and LOGGING. Event parameters are listed in a separate table. Applicability and restrictions are presented both as defined by Digital Network Architecture and thus relevant to all DIGITAL systems, and as implemented specifically for DECnet-20 V3.0.

Appendix B summarizes the entity counters, including X.25 counters, that may be displayed and zeroed by appropriate NCP commands.

Appendix C describes in detail all messages related to the execution of NCP commands and the state of the network: output responses; informational messages; and error messages.

Appendix D is a table of currently recognized DECnet line devices.

Appendix E describes X.29 files for use with the PSI system, including PAD-parameter files, the X.29 configuration file, and the X.29 herald text file.

Appendix F is an extended bibliography of suggested network-related readings.

Appendix G is a glossary.

Depending on your experience and responsibilities, you may need to refer to one or more of the following manuals that contain more detailed information on procedures or programs related to DECnet-20 V3.0.

DECnet-20 User's Guide, Order No. AA-J579A-TM

This manual contains procedures and examples for:

- The MACRO programmer who will write network application programs
- The nonprivileged terminal user who will use the TOPS-20 Command Language and the utility program, Network File Transfer (NFT), in network-related tasks. The SETHOST program, which provides remote login capabilities, is also described in detail.

DECnet-20 Network Generation and Installation Procedures, Order No. AA-K905A-TM

This manual describes the TOPS-20 network configuration tools, the procedures for creating a host node in a DECnet network, the procedures for configuring a DN20 communications front end, and the optional DN200 Remote Job Entry Station (RJE). Complete network installation procedures, including customer verification of the DECnet software follow the generation procedures.

TOPS-20 DN200 Remote Station Guide, Order No. AA-H786B-TM

This manual provides information on the operation of the DN200 remote station for the operator at the remote station.

TOPS-20 PSI User's Guide, Order No. AA-J533A-TM

This manual contains information on the following:

- Writing a TOPS-20 FORTRAN or MACRO program that uses the TOPS-20 X.25 software to access a PPSN (Public Packet Switching Network).
- Using the TOPS-20 X.29 software to connect a terminal through a PPSN to a TOPS-20 host.

TOPS-20 PSI Installation Guide, Order No. AA-N819A-TM

This manual describes the configuration of the TOPS-20 PSI nodes in a DECnet network and details all generation, verification, and installation steps. It describes configuration tools, procedures to generate DN20 subsystems, X25GEN (X.25 GENeration) program commands and parameters, and illustrates a sample system configuration.

TOPS-10/TOPS-20 SPEAR Manual, Order No. AA-J833A-TK

This manual describes the SPEAR program, a library of functions that analyzes and reports on error and significant event information recorded by the operating system in the ERROR.SYS file. DECnet-20 operating systems record entries for loads, dumps, and startups as well as errors. You use the SPEAR program's functions RETRIEVE and SUMMARIZE to produce reports of errors and events recorded for DECnet-20.

You should also refer to the following non-DIGITAL publications if you have the PSI software option, or if your network includes nodes that support X.25.

CCITT publications that contain complete information on recommendation X.3 (PAD) parameters. (See the TOPS-20 PSI User's Guide for a list of CCITT publications.)

Documentation provided by the user's PPSN (TELENET, for example).

The following suggestions are offered for your effective use of this manual:

1. You are the best judge of what you need to read, and how much hands-on practice you need. Consider the following factors in choosing what to read and what to practice:

   - Your own responsibilities and assigned tasks
   - The depth of your present knowledge of TOPS-20 and DECnet
   - The extent of your experience as an operator or system manager in a network environment
   - The operating systems and the capabilities of the nodes in your network

2. Read the Glossary before you read the main part of the manual. You must understand the "network language" to properly interpret descriptions of concepts and commands. Terms in the Glossary are defined as they are to be interpreted in this manual. The term "monitor", for example, when used in the NCP command SET/DEFINE LOGGING, refers to one possible destination of an event message and not to the program that controls the operating system of the host.

   This preliminary examination of the Glossary is to help you in recall only. Later, when you read a defined word in context in the manual, you can refer to its network-management interpretation as defined in the Glossary for understanding.

3. Beginning with Chapter 4, after you have read each chapter, take the manual to your terminal and enter the commands given in examples, adjusting parameters as needed to conform to your site's implementation and conventions. Try the same commands again without using the manual. Use the "?" feature if you forget the next element to be typed. Use the "?" to review lists of possible entities or arguments for each command. Such practice sessions will be far more helpful than "studying" the examples. (The "?" feature is described in Chapter 4. Several examples are included.)

   With the exception of the SHOW NCP commands, any command that changes parameter values has the potential for interfering with the network activities of another network user. When possible, practice under supervision and always work within the guidelines set by the system manager. You can avoid conflicts by using the OPR command SEND to inform one or all network users of your intentions. Such messages as the following will be helpful to others on the network:

   ```
   15:47:40 From Operator Terminal 23:
   =>Reloading DN20 in 5 minutes

   16:30:25 From Operator Terminal 25:
   =>Restarting Galaxy in 5 minutes
   ```

   You could also use ^Esend to produce messages in the following format:

   ```
   [From PECKHAM on line 16 to all: restarting NMLT20...]
   ```

4. System managers and operators using NCP commands to monitor and control all nodes should read the entire manual. If your network includes DIGITAL operating systems other than TOPS-20, you should also read the appropriate manuals for these operating systems. If you are a central operator in the operations room, you will require a working knowledge of all procedures in Chapter 3. Reading the remainder of the manual is optional.

   If the system manager has restricted your use of NCP commands, concentrate on Chapters 2 through 6 in your initial reading. All operators should refer to Appendix F for additional reading whenever there is a need for further information.

5. This suggestion concerns you only if you have been operating in a Phase II DECnet-20 environment. Phase III NCP commands have two changes in format. The major change is the positioning of the entity (the component to be acted upon) immediately after the command. Another change is using the hyphen to replace the underline symbol. The following example shows both these changes:

   Phase II:

```
NCP>SHOW STATE LINE KDP_0_1
NCP>SET STATE LINE KDP_0_1 OFF
NCP>
```

   Phase III:

```
NCP>SHOW LINE KDP-0-1 STATUS
NCP>SET LINE KDP-0-1 STATE OFF
NCP>
```

CONVENTIONS USED IN THIS MANUAL

The following conventions are used in this manual in command descriptions and in examples of dialogue:

UPPER CASE	Upper case letters indicate actual input required in a command string. You must type an upper case element in full or with an abbreviation acceptable to the system.
lower case	Lower case letters in a command string indicate an input variable type (seconds, for example), not the actual variable determined by the operator (2, for example, for number of seconds).
[ ]	Brackets indicate optional input. (Brackets are not included when you type the command.)
{ }	Braces indicate that one of several enclosed parameters is applicable. (Braces are not included when you type the command.)
spaces	Spaces separate elements of a command. A tab or multiple spaces may also be used. Spaces must be input where shown.
red	Red characters indicate information you, the user typing the command, enter. Black characters indicate system-supplied information.
(RET)	Indicates the place where you press the key labelled RETURN or CR.
(ESC)	Indicates the place where you press the key labelled ESC, ALT, or SEL.
<CTRL/character>	Indicates the place where you type the control character designated by "character".
numbers	All numeric values that appear in this manual are decimal numbers, unless otherwise noted.

The following acronyms are used freely in this manual to avoid wordiness:

DLL	Data Link Layer
DNA	DECnet Network Architecture
NICE	Network Information and Control Exchange protocol
NML	Network Management Layer
NSP	Network Services Protocol
PSI	Packetnet System Interface
PPSN	Public Packet Switching Network

PART I
INTRODUCTION

CHAPTER 1

SYSTEM OVERVIEW

1.1 DIGITAL NETWORK ARCHITECTURE AND DECnet

DIGITAL Network Architecture (DNA) is a model upon which DIGITAL computer networks can be built. DNA presents a layered structure, each layer corresponding to a major network function. A well-defined set of interfaces allows each layer to communicate with its adjacent layers; protocols provide rules and conventions for exchange of information between peer layers in communicating nodes. Refer to Figure 1-1.

DECnet-20 V3.0 for the TOPS-20 system, is the third generation (Phase III) of implementations based on the DNA model. A general knowledge of DNA will be helpful to both system managers and operators with network management responsibilities.

The DNA model specifies functions to be carried out in software modules that implement each layer. How the functions are implemented, and which functions are implemented depends on the operating system. For example, an operating system with a minimum amount of core may choose to implement only the minimum subset of Network Management functions; an operating system with a communications front end (DN20) may choose to implement all Network Management functions. Thus, if you know the functions that are implemented by each node in your network, you can save time and effort. This information is available in the manuals and SPDs for the various DIGITAL operating systems. Personal contact with the system managers of all nodes in your network is highly recommended.

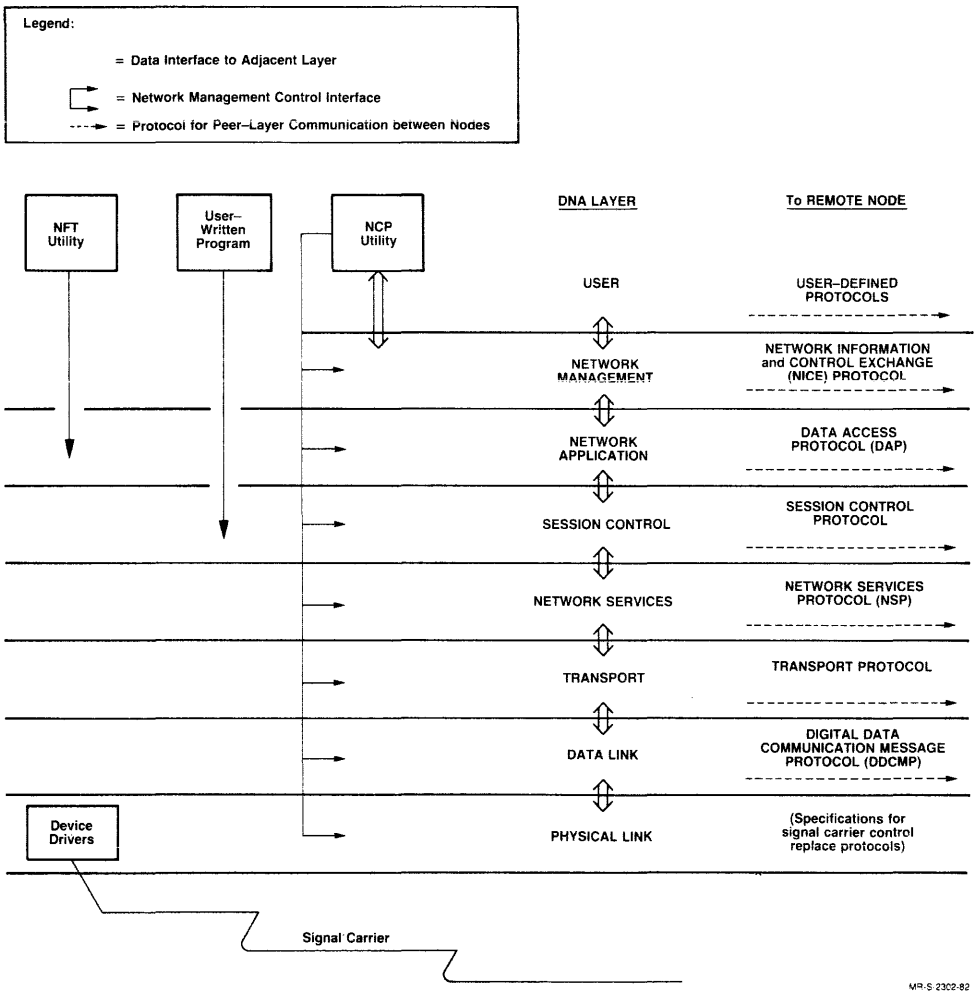


Figure 1-1 Communication between Adjacent and Equivalent Layers of the DIGITAL Network Architecture

The major concepts of DNA are:

- Transparency. In the context of the network, the system manager or operator is the "user" of the DECnet utility, the Network Control Program (NCP). Once NCP is invoked, the "user" is the module or routine that uses the capabilities of the layer beneath it in the functional hierarchy. Once you have typed a command, all processing and routing of your request is a function of the system. You will receive a response informing you of success or failure.

- Local autonomy. The node that is to execute an NCP command retains the right to refuse to do so. Such a refusal is accompanied by a reason in the DECnet-20 V3.0 implementation. Other operating systems may or may not give a reason. The nature of the reason determines whether or not you repeat the command at a later time.

- High flexibility. The functional orientation of the layered structure lends itself well to the inclusion of new modules and the insertion of new layers. (See Section 1.2.)

- Upward compatibility. A Phase II node can communicate with an adjacent node only, either a Phase II node or a Phase III node. The Phase II node derives no benefits from the routing network, however, until it upgrades to Phase III.

- Adaptability to the needs of individual network sites. Within the limits set by Phase III minimum requirements, each operating system implements only those functions appropriate to that operating system.

- Distributed or central management capability. The Network Management Layer (NML) of each DECnet-20 V3.0 node has access to all other NMLs through the Network Information and Control Exchange (NICE) protocol. Thus, operators at any node can control network management functions at any node in the network, given the required willingness and capability on the part of the remote node. When desired, the management of the network can be restricted to one node.

1.2 THE LAYERED STRUCTURE OF DNA

The layered structure of DNA has, among several other advantages, the advantage of flexibility. Adding new functions, and consequently new layers and modules, is easily accomplished during product development. DECnet-20 V3.0 offers a network of considerably more capability than that offered by V2. The changes are reflected in the architecture which has developed as follows:

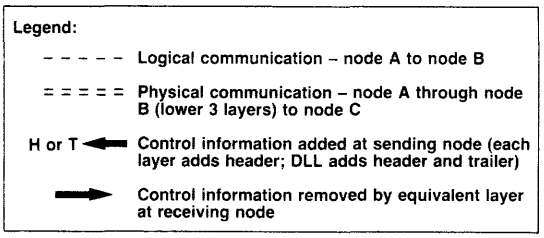
DECnet-20 V2 (Phase II DECnet)	DECnet-20 V3.0 (Phase III DECnet)
User Layer	User Layer Network Management Layer
Network Application Layer	Network Application Layer Session Control Layer
Network Services Layer	Network Services Layer Transport Layer
Data Link Layer	Data Link Layer
Physical Link Layer	Physical Link Layer

As shown in the chart, the layered structure reflects the two major added capabilities of DECnet-20 V3.0:

- increased network management facilities (related modules are in the top three layers)
- route-through capability (modules in the Transport Layer)

Beginning with the highest level: the User Layer, Application Layer, and Network Management Layer carry out user functions; the Session Control Layer, the Network Services Layer, and the Transport Layer carry out network functions; and the Data Link Layer and Physical Link Layer carry out the physical aspects of communication.

A brief description of each of the DNA layers, as implemented for DECnet-20 V3.0, follows. Refer to Figures 1-1 and 1-2 before you continue reading.



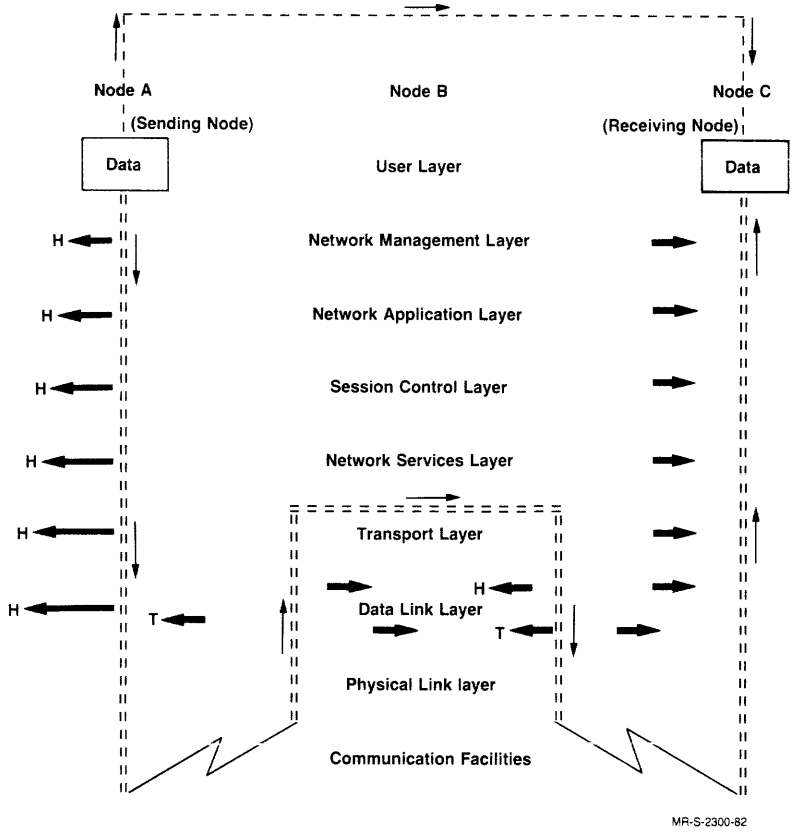


Figure 1-2 Logical Communication and Physical Path of Routed Data

1.2.1 The User Layer

The User Layer supports user services and programs. NCP, the command set of the Network Control Program, is in this layer. User-written programs (MACRO programs using file system monitor calls) also reside in this layer.

1.2.2 The Network Management Layer

NML's interaction with people is through the NCP command interface. NCP commands allow operators at any node in the network to control the network management functions of any other available Phase III node in the network, as well as the local node functions. The NML, unlike other architectural layers which interface only with adjacent layers, has interfaces defined to every other layer. Refer to Figure 1-3 as you read the following description of NCP command processing. All of the action described is transparent to you. If your duties as an operator require only that you enter NCP commands, you will not need to understand transparent processing, but the system manager and operators who are also systems analysts will find such knowledge helpful if they have responsibility for diagnosis and maintenance.

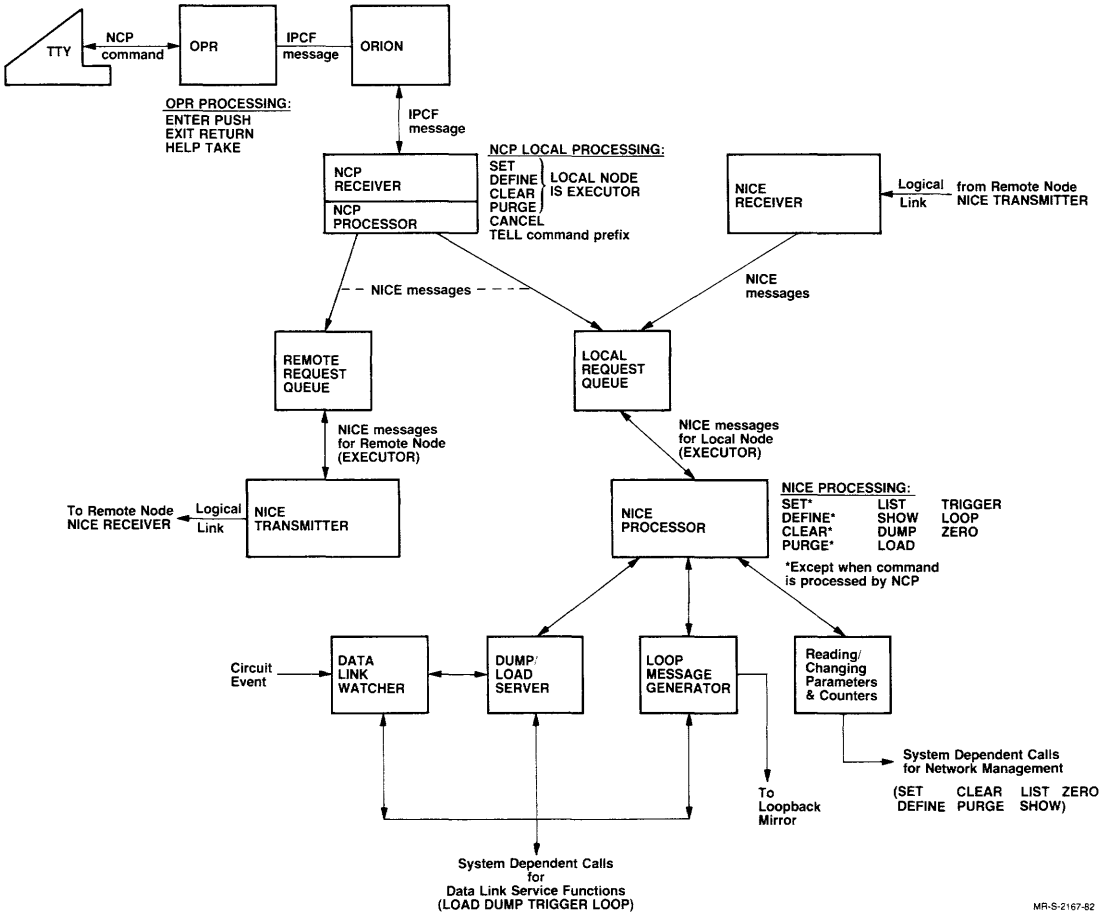


Figure 1-3 OPR/NCP/NML Logical Processing Flow in DECnet-20 V3.0

OPR parses all NCP commands. Correctly formatted commands that can be processed by OPR (no action by NML is necessary) are processed and executed immediately. All other NCP commands, those to be executed by the local node and those to be executed by a remote node, pass to the NCP Receiver in the NML. A few commands have access to required data for processing at this point and these are processed and executed. All other commands are translated from NCP to NICE message format and placed in the Request Queue.

The Request Queue also receives input from a NICE Receiver. The NICE Receiver has received these NICE messages from a remote node for processing and execution at the local node. All NICE messages to be processed and executed by the local node are passed from the Request Queue to the local NML processor. The local NML Processor, using system-dependent calls, directs these messages to appropriate routines where local functions are processed. NICE messages to be processed by a remote node are passed to the NICE Transmitter and sent by logical link from the transmitter to the remote node.

Command responses are output to the terminal of the command node. All NCP commands translated into NICE messages receive a response. Commands that are processed before reaching the Request Queue do not require a response. There is little delay and the timely execution of the command at the local node serves as adequate response. Note that there are two paths for responses from the NML Processor. If the local node (the node pictured in the diagram) was both the command node and the EXECUTOR, the response is from the local node and to the local node. If the command was typed at the local node but the EXECUTOR was set to be a remote node, the response is from the remote node to the local node. The response to a command typed at a remote node for processing at the local node is displayed at the remote node where the command is entered.

Other processes within the Network Management Layer are the following:

The Data Link Watcher monitors the Data Link Layer (DLL). This module can sense when there is a service request on a line from an adjacent node. The Data Link Watcher reads the request (a MOP protocol message) and either services the line itself or passes an appropriate request to the Upline Dumper or Downline Loader process. There is a Data Link Watcher in both the KL10-E (TOPS-20) and the DN20 (MCB). (In DECnet-20, the Data Link Watcher in the MCB front end acts in the manner described. In the TOPS-20 KL10 node, the Data Link Watcher periodically checks to see if the MCB front end is reachable; if it is not, it attempts a dump and load.)

The Event Receiver receives remotely generated event messages and checks these messages for proper syntax. Syntactically correct messages are sent to the Event Recorder.

The Event Recorder examines the mask bits in the event messages and distributes the messages to the indicated event sinks.

The Event Processor reformats raw event messages into processed event messages. Filtering is performed, a date and time stamp is added, and the processed events not filtered out are sent, as appropriate, to the local Event Recorder or to an Event Transmitter to be sent to the remote Event Receiver.

The Event Transmitter uses a logical link to transmit event messages to the Event Receiver of the remote node (the sink node).

The Loop Message Generator is responsible for all loop functions, both those associated with a CIRCUIT or LINE and those associated with a NODE. Node loopback data is dispatched to the Loopback Mirror. The primary function of the Loopback Mirror is to reflect back to the generator the test message or an appropriate error message. CIRCUIT or LINE loopback is accomplished with or without hardware assistance. In a DECSYSTEM 2040S/2060, there is a Loopback Mirror in both the KL10-E and the DN20. DECnet-20 V3.0 supports LOOP NODE and LOOP CIRCUIT.

The Downline Loader, using MOP protocol, accomplishes the loading of DN20s from the host and DN200s connected to a DN20.

The Upline Dumper acts as a receiver of a dump from an adjacent node. The dump is stored on a file in the host node.

The Permanent Data Base Processor processes DEFINE and PURGE requests and supplies information with proper ordering for the SET ALL command.

1.2.3 The Network Application Layer

This layer supports I/O devices and file access. Modules within this layer include the Loopback Mirror, Network File Access, and the File Access Listener.

1.2.4 The Session Control Layer

The functions of the Network Services Layer in DECnet-20 V2 have been expanded for DECnet-20 V3.0. Three DNA layers now support the capabilities associated with logical links. In Phase II, all logical link functions were supported by the Network Services layer. System-dependent functions of communication by logical links are now supported by the Session Control Layer. For example, the Session Control module uses access control information in an incoming connect request to perform system-dependent validation functions.

1.2.5 The Network Services Layer

Modules within this layer, using the Network Services Protocol, control the creation, maintenance, and destruction of the logical links that provide the network's primary user-to-user communication. It is the NSP layer that guarantees delivery, in proper sequence, of all transmitted data.

1.2.6 The Transport Layer

This layer provides routing related processing. The Transport module first determines reachability, that is, it determines if there is a path from the sending node to the specified receiving node. If there is not, you receive an appropriate message. If there is, a table lookup is performed to determine the least-cost path. The message is directed to the first node along the least-cost path. (Routing concepts are explained in more detail in Section 2.4.)

1.2.7 The Data Link Layer

Using the DIGITAL Data Communications Message Protocol, DDCMP, this layer controls message sequencing and data integrity. It does not guarantee delivery, but it does ensure error-free communication between adjacent nodes once communication is established.

An important fact to remember is that within one network the Data Link Block Size must be the same. If this rule is not followed when a node is added to the network, the node will not communicate properly with adjacent nodes.

1.2.8 The Physical Link Layer

This layer is concerned with the physical transmission of bits over a channel. The Physical Link Layer includes the software for driving the communications hardware-interface devices, modems, and communication lines.

1.3 THE DECnet-20 VERSION 3.0 IMPLEMENTATION

Although the KL10-E and the DN20 are two separate nodes, they work together as a system. The KL utilizes the DN20 capabilities to participate fully in a Phase III network. The KL is technically a Phase II node with a Phase III NSP module. The communication front end (DN20) is a full Phase III routing node.

Features implemented in the KL10-E main processor include file transfer, remote batch station features, and a subset of Network Management V3.0. Features implemented in the communications front end include adaptive routing through an intermediate node and exclude X.25 and DMP multipoint functions. RJE-20 Version 2.0 software running in the DN200 is available as a DECnet-20 option.

The two parameter values BUFFER SIZE in bytes (a decimal number in the range 290-576), and MAXIMUM ADDRESS (a decimal number in the range 2-255) are determining factors in the performance of your network. These values are set during network generation. BUFFER SIZE can not be changed by the NCP command SET. MAXIMUM ADDRESS can be changed to a lower address by the SET command with the result that all nodes above MAXIMUM ADDRESS will not be recognized. An attempt to raise MAXIMUM ADDRESS above the value given during network generation will result in an error message.

It is likely that you will repeat the network generation procedure at some time in the future (to add lines or devices, or to change the value of MAXIMUM ADDRESS or BUFFER SIZE). Certain relationships must be understood to ensure that the BUFFER SIZE you select during network generation will meet the requirements of your network.

The Data Link Layer (DLL) block size must be the same throughout the network for communication between all DIGITAL operating systems running DECnet. To accept incoming messages of all possible sizes, an EXECUTOR's BUFFER SIZE should be as large as the BUFFER SIZE of the remote (sending) node. The DECnet-20 default value for BUFFER SIZE, 576 bytes, is the BUFFER SIZE most commonly used by DIGITAL systems that support DECnet Phase III. A node will refuse to accept a message that is larger than its BUFFER SIZE. There is also a minimum requirement for BUFFER SIZE. To accommodate Transport messages, the BUFFER SIZE must be at least (2 * the MAXIMUM ADDRESS) plus 5.

DECnet-20 does not have a modifiable permanent data base and therefore does not support the NCP commands DEFINE, PURGE, and LIST. Data base values established during network generation are read from command files at system startup and become the initial parameter values in the volatile data base. Appendix A summarizes all possible parameters and describes both DNA and DECnet-20 applicability ("applies to EXECUTOR only", for example) and restrictions ("display only", for example).

Currently, the DECSYSTEM-2020 remains a Phase II node. Within the restrictions outlined in the section on compatibility, the 2020 can participate in a Phase III network.

1.4 DECnet-20 OPTIONS

Two options are available for TOPS-20 DECnet on the 2040S/2060.

The RJE-20 Version 2.0 option includes software for the DN200 remote job entry station and provides facilities for the DN200 to be downline loaded, upline dumped, diagnosed, and maintained remotely. Files for RJE-20 loading and dumping are located at the central processor that functions as the host node. The DECnet-20 commands to the RJE station are executed by the adjacent front end (DN20).

The PSI Version 1.0 option includes software for the Packetnet System Interface (PSI) Gateway, which provides access to Public Packet Switching Networks (PPSNs). The PSI Gateway is the interface between a PPSN and a DECnet-20 network. The Gateway software resides with DECnet-20 software in a DN20.

The Gateway:

1. Provides for communication over a PPSN path between two cooperating processes not using DECnet protocols as end-to-end protocols.

2. Permits terminal access to TOPS-20 by a user who dials by phone connection into a PPSN Packet Assembler/Disassembler (PAD) facility (see Appendix E).

NOTE

With DECnet-20 V3.0, you cannot downline load or upline dump a remote node through the DN20 if the PSI software option is running in that DN20. The RJE-20 and PSI options cannot co-exist in the same DN20.

1.5 DECSYSTEM 2040S/2060 HARDWARE - DECnet-20 V3.0

The basic hardware configuration remains much the same as it was for DECnet-20 V2. As in DECnet-20 V2, the DECSYSTEM-2040S/2060 system consists of:

- The KL10-E Central Processing Unit, running TOPS-20 software. The CPU directs the entire operation of the system. Contained in the CPU is the microcode instruction set, the general purpose registers, interrupt and trap facilities, and a subset of DECnet-20 software.

- The console front end, running RSX20F. This front end assists the CPU by handling the operator's console, terminal communications, unit record peripherals, and diagnostics.

- The communication front end, running the Multifunction Communication Base (MCB) software. MCB software, in cooperation with the CPU, implements the V3.0 DECnet-20 functions.

- The DTE hardware interfaces between the main processor and the two front ends.

- The DN200 Remote Job Entry Station running RJE-20 Version 2.0 software is available as a DECnet-20 option.

- Communication devices as required.

The KL10E CPU supports only one DN20-MA communication front end for DECnet-20. It will support one IBM communication front end concurrently with the DECnet front end.

The major hardware change is in line configuration and supporting devices. A maximum of 8 point-to-point synchronous links, with combinations of three communication devices - KMC11/DUP11, DMC11 or DMR11 - are supported. The DUP (alone, not with the KMC) is not supported in Version 3.0.

The PSI Version 1.0 software option restricts the basic hardware configuration. See the Software Product Description (SPD) for more information.

1.6 DECnet IMPLEMENTATIONS

A DECnet network may consist of various DIGITAL operating systems, each with a distinctive implementation. The concepts underlying the implementation will be the same, but the functions implemented may differ.

Although the DECnet Phase (whether Phase II or III) is the major factor that determines the set of network functions available on a particular node, it is not the only factor. Even if all nodes in your network are Phase III full routing nodes, it is very likely that several nodes support different sets of network functions. When any two nodes communicate, their interaction will be limited to the functions common to both.

A Phase III node need only implement the Phase III minimum subset of NCP commands. A node is not required to include any of the many commands outside the subset if the node has no use for them.

DECnet-20 V3.0 implements a subset of the NCP commands in the Phase III DNA Network Management V3.0 Specification. Not all parameters that can be formatted and interpreted are supported. You will not receive an "action response" if a command is not implemented at the receiving end. The response you receive will probably be:

Unsupported Function or Option

DECnet implementations provide the ability to send commands to a remote node for execution at the remote node. Because it is a Central Management node, the DECnet-20 node can format commands supported by other Digital hosts, even though the commands are not supported locally.

1.7 DECnet CAPABILITIES

DECnet-20 capabilities are available to the nonprivileged terminal user, the programmer, and the system manager or operator. The User's Guide (see Preface for complete title and order number) describes the DECnet functions available to nonprivileged terminal users and programmers. This manual is specific to the operational, control, and monitoring functions that are normally the responsibility of the system manager or of an operator designated by the system manager.

The system manager and operator use the commands of the Network Control Program (NCP) to control and monitor the network. It is the system manager's responsibility to inform the staff of critical commands that should be used by only designated operators, or used at off-hours only. Chapters 4 through 7 describe the NCP command set in detail. Chapter 4 summarizes the general functions and features of NCP. The specific function of each command and the meaning and significance of allowed arguments are described in Chapters 5, 6 and 7.

A more general and more inclusive summary of DECnet-20 capabilities follows:

- Adaptive path routing. Phase III routing nodes can send and receive messages by routing through intermediate Phase III nodes. Line and system failures can be detected, and, where an alternate path exists, this path is chosen. Adaptive path routing is transparent to the user. The system manager, or delegated operator, can change a routing path indirectly with an NCP command.

- Homogeneous network command terminals. A user at one node can log into the system at another node in the network if both systems are using the same operating system and both have implemented DECnet Phase III. A Phase II DECnet-20 node can only log into its adjacent Phase III node; it can, however, use its Phase III host to log into another Phase II node adjacent to the same host.

- Support of nodes of varying capabilities and configurations. DECnet-20 allows both end nodes (nodes that can send and receive but not forward) and Phase II nodes to coexist on the network within the restrictions described in Section 2.5.1.

- Because DECnet-20 V3.0.0 nodes are Central Management Nodes, the NM modules understand the complete command set for DECnet Phase III. Thus, using the TELL nodeid prefix, you can direct NCP commands not supported locally to other Phase III nodes on your network. A DECnet-20 V3.0 node can format any command that a remote DECnet Phase III node can execute. This includes multipoint- and X.25-related commands.

- Extensive network management capability. Using NCP, the system manager and operator can control and monitor network activity from one or multiple nodes in the network. Entity states and characteristics can be changed in the permanent or temporary data base. Both current and recent network activity can be displayed. Loopback tests, downline loading and upline dumping are available through NCP commands.

- Logging of significant events. Activity counters that you can reset and display are available. (See Appendix B for details.) Certain events and errors (start-ups, dumps, hardware detected errors, for example) are also entered in the file, ERROR.SYS. You can use the SPEAR program to create reports according to the specifications you select.

- File transfer operation. You can copy, delete, or type a file from a remote node; you can obtain a directory from a remote node; you can submit a batch control file at a remote node.

- Support of up to four remote job entry stations. (RJE-20 option.)

- The X.25 facility for using Public Packet Switching Networks (PPSNs). (PSI option.)

1.8 OVERVIEW OF NCP COMMAND KEYWORDS

There are just 20 NCP command keywords that specify the action you request. The command keyword is always the first part of the command that you type. Command keywords may or may not be followed by an entity (a controllable network element such as a node or circuit) and one or more parameters (values indicating characteristics or status of an entity). Most commands to remote nodes include both entity and parameter(s). Commands addressed to the local node may have entity or parameter or both implied. The complete command is a function of the command keyword, the entity, and the parameter.

Each of the following commands is complete.

Command keyword only:

```
NCP>EXIT
NCP>HELP
```

Command keyword and entity:

```
NCP>SHOW EXECUTOR (SUMMARY is implied)
```

Command keyword, entity, and parameter:

```
NCP>SET EXECUTOR NODE nodeid
NCP>ZERO CIRCUIT cktid COUNTERS
```

A complete list of NCP command keywords follows. The order is alphabetic for ease of reference. Because the action requested by a command keyword is related to both the entity and parameter that follow the command keyword, the functions are defined in non-specific terms. Refer to the section indicated for a more specific description of function.

Table 1-1
NCP Command Keywords

Command	Function	Complete Description
CANCEL	Removes a command from the Request Queue before processing begins.	Section 6.6
CLEAR	Removes a value previously entered in the volatile data base.	Section 6.3 (EXECUTOR) Section 7.6 (All entities)
DEFINE	Enters a value in the permanent data base.	Section 6.2 (EXECUTOR) Section 7.5 (All entities)
DUMP	Stores a copy of a target node's memory image in a dump file at the host node.	Section 7.8.3
ENTER	Typed to OPR to access a subset of OPR commands.	Section 5.2

Table 1-1 (Cont.)
NCP Command Keywords

Command	Function	Complete Description
EXIT	Terminates an NCP session and returns control to TOPS-20 Command level.	Section 5.2
HELP	Returns the function and major keywords for all NCP commands.	Section 6.8
LIST	Displays on the user's terminal information from the permanent data base.	Section 7.7
LOAD	Allows one node to load the system image file to a remote node adjacent to the EXECUTOR	Section 7.8.2
LOOP	Requests a loopback test.	Section 7.8.4
PURGE	Removes a value or values from the permanent data base.	Section 6.4 (EXECUTOR)
PUSH	Changes control from NCP to TOPS-20 Command level.	Section 5.2
RETURN	Changes control from NCP to Operator Command level.	Section 5.2
SET	Enters a value or values in the volatile data base.	Section 6.1 (EXECUTOR) Section 7.5 (All entities)
SHOW	Displays on the user's terminal information from the volatile data base.	Section 6.5 (SHOW QUEUE) Section 7.7 (SHOW entity)
TAKE	Retrieves and executes a file of NCP commands.	Section 5.2
TELL (prefix)	Directs the command that follows to a remote node for execution.	Section 5.2
TRIGGER	Requests the target node to send a load request.	Section 7.8.1
WAIT	Used in Batch programs, delays processing the next command for the specified number of seconds.	Section 5.2
ZERO	Logs counters and then zeroes them.	Section 7.9

CHAPTER 2

NETWORK CONCEPTS

2.1 NODES AND DATA LINKS

Nodes and data links are the two fundamental elements in the network. As used in this document, a network node is an implementation that can pass data, using DECnet, to one or more remote nodes. Network nodes are classified in several ways. One basic classification divides nodes into local and remote. From your point of view as an operator typing NCP commands, the local node is the node where your terminal is located (the TOPS-20 node in DECnet-20). All other nodes, including the MCB front end, are remote. From Network Management's point of view, the local node is the node acting as EXECUTOR; this is usually, but not always, the node at which the command is typed. Any active node in the network is remote to any other active node in the network. Thus, by definition, the central processor is remote to its connected MCB front end. An adjacent node is one that is physically connected to the node that it is termed adjacent to. The MCB front end is adjacent to the KL central processor.

Nodes are also classified according to their network function. In descriptions of network functions, you will frequently see the terms command (or control) node, executor node (or executor), target node, and host node. The command node is the node where the command is typed. The executor is the node where the command you type will be executed. The target node is the node where the action requested by the command will take place. The host node is the node where needed facilities for effecting the command may exist (storage space for files, for example). These functional classifications are not mutually exclusive. For example, a node may serve as executor, command, host node, and target node. Refer to Figure 2-1.

Logical links are logical connections. A physical link connects node A to node B, for example; a logical link connects Mary at one terminal to Bob at a remote terminal, or connects a user process at one node to a utility process at another node. The logical link encompasses the end-to-end transmission of data. Lines are the physical media for logical links. In a logical link from node A to node C by way of node B, there is one logical link but there are two physical links. The term "circuit" is synonymous with a logical point-to-point connection. Circuits are described in detail in Section 7.2.5.

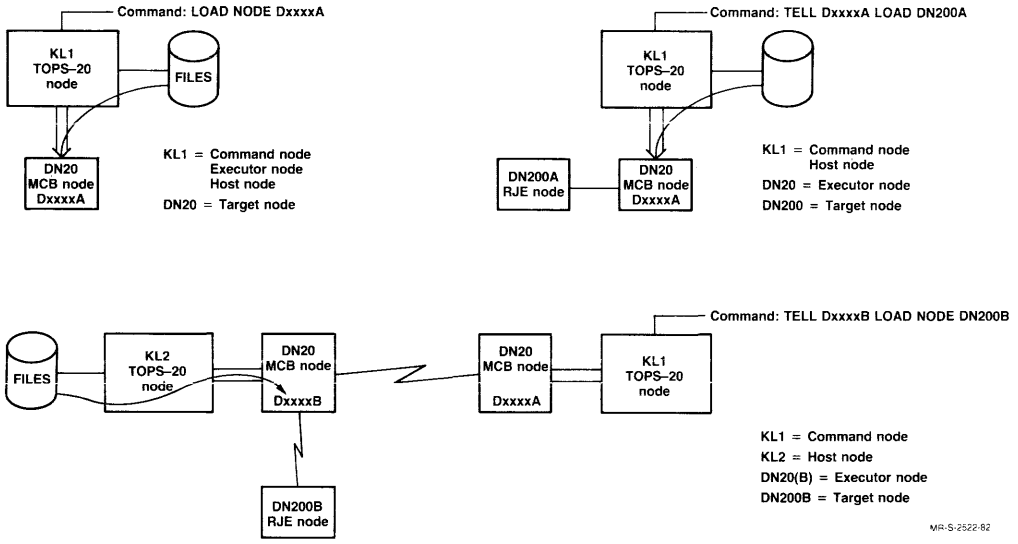


Figure 2-1 Types of Nodes by Network Function

2.2 THE PSI GATEWAY

The PSI Gateway interfaces a DECnet network to an X.25 PPSN over a leased data communications line. The internationally recommended protocol for such an interface has been defined by CCITT Recommendation X.25. The Gateway's main responsibility is to generate and interpret the protocol messages necessary to communicate with the X.25 node at the opposite end of the data link to the PPSN.

The PSI option also includes software to permit access to the Gateway, and thus the X.25 network, from a TOPS-20 DECnet Version 3.0 node. These software units include the Gateway Access Routines and the X.29 Server. Figure 2-2 illustrates the placement of the TOPS-20 PSI Gateway Access Routines and the TOPS-20 Gateway Software in the central (TOPS-20) processor and the communications processor (DTE).

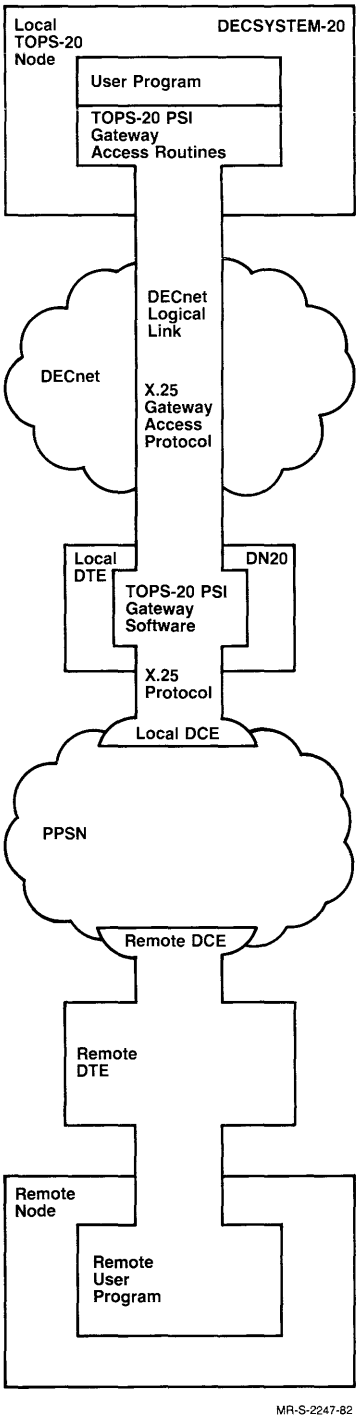


Figure 2-2 TOPS-20 PSI Gateway Software Components

The Gateway Access Routines are a collection of subroutines that the user invokes to perform X.25 functions (see the TOPS-20 PSI User's Guide). These routines are linked with the user's software to form a single-job image. They maintain communication with the Gateway by using the Gateway Access Protocol over a DECnet logical link. This permits users to directly manipulate the X.25 protocol at the Gateway's interface to the X.25 network, even though the user may be on a DECnet node that is geographically distant from that interface.

The X.29 feature permits a user at an asynchronous terminal to log in to a TOPS-20 system that has Gateway access to an X.25 network. The terminal is connected to a PAD, either through a direct asynchronous connection or a modem. The X.29 Server uses the X.29 protocol to control the terminal session through communication with the PAD. The X.29 Server uses the Gateway Access protocol to communicate with the Gateway. It communicates with the NRT Server in the TOPS-20 host using the Network Remote Terminal Protocol (NRTP). Figure 2-3 illustrates the components needed for such a connection.

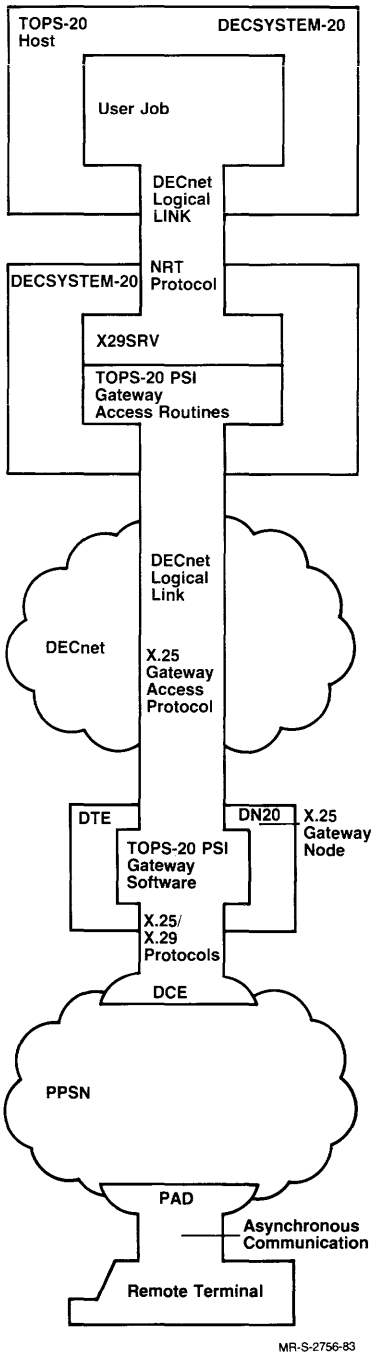


Figure 2-3 X.29 Terminal Access to a TOPS-20 Host

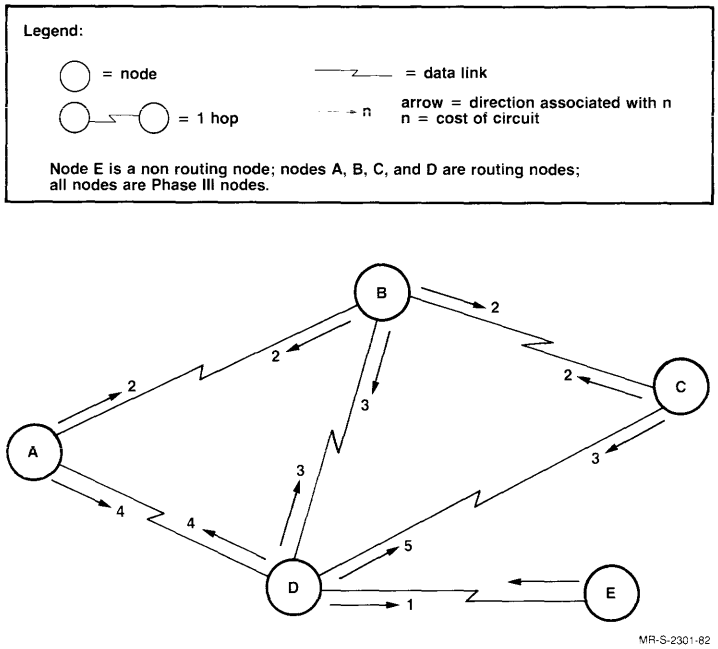
2.3 ENTITIES

The term "entity" is used in reference to NCP commands. An entity is the element on which the command is to act: the node, for example, in the command SET NODE nodeid HOST hostname; or the active lines in the command SHOW ACTIVE LINES COUNTERS. An entity in the singular form is one of: CIRCUIT, LINE, LOGGING, NODE, and MODULE. ACTIVE and KNOWN preceding CIRCUITS, LINES, LOGGING, NODES, and MODULES are allowed plural forms. NODE has a third plural form, LOOP NODES. Entities are described in detail in Chapter 7.

The MODULE entity is provided for users of the TOPS-20 X.25 software. The MODULE entity takes one of three forms: MODULE X25-ACCESS, MODULE X25-PROTOCOL, and MODULE X25-SERVER. The MODULE entity is described in detail in Chapter 8.

2.4 ROUTING CONCEPTS

Routing, the primary function of the Transport Layer, is the determination of the physical path that packets will follow between nodes. This path is determined at each node. At each intermediate node, the transport module reads the header of the packet to determine the destination, then reads the routing table to determine the least-cost path. Within the limits set by the configuration, the routing table is automatically updated when a path is no longer an acceptable choice. The path may also be changed indirectly by the system manager or operator using the appropriate NCP command (SET CIRCUIT cktid COST cost). Refer to Figure 2-4.



	From Node	To Node	Path	Path Cost	Least Cost Path	Path Length
All links up	B	D	B→D B→A→D B→C→D	3 6 7	B→D	1 2 2
	D	B	D→B D→A→B D→C→B	3 6 7	D→B	1 2 2
Link between B and D down	B	D	B→A→D B→C→D	6 5	B→C→D	2 2
	D	B	D→A→B D→C→B	6 7	D→A→B	2 2

NCP Commands for Control:

FOR CIRCUIT	FOR EXECUTOR
SET CIRCUIT DMR-0 COST 3	SET NODE D2136A MAXIMUM COST 30
SET CIRCUIT KDP-0-2 COST 4	SET NODE D2136A MAXIMUM HOPS 4
	SET NODE D2136A MAXIMUM VISITS 8

Figure 2-4 Routing for DECnet-20 V3.0 (Phase III nodes)

2.4.1 Node Routing Types

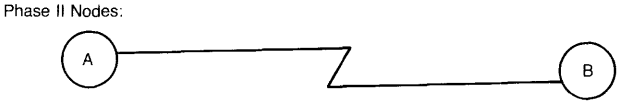
There are three types of nodes supported by DECnet-20 V3.0 routing: routing nodes, nonrouting nodes, and Phase II nodes. The characteristics and capabilities of the three major types of nodes follow.

Routing Nodes

Routing nodes allow communication between nodes that are not adjacent. This capability is first realized in Version 3.0.

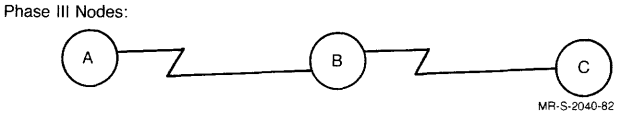
Diagrammatically:

Phase II nodes:



Communicating path is A to B.

Phase III nodes:



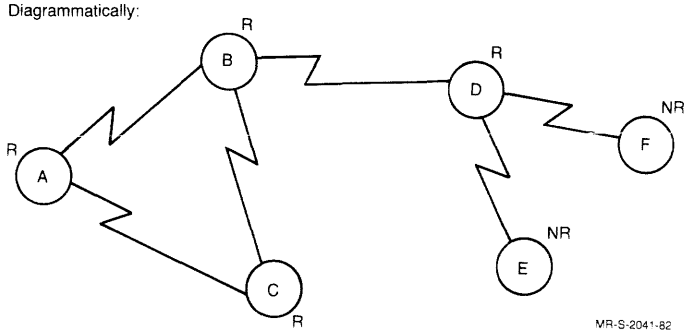
Communicating path is A to C through B.

Under normal conditions, neither A nor C is aware of B's participation in the routing above the DNA Transport level. This means that the operator perceives the connection as a direct A to C connection. If all nodes in the network are routing nodes, then each node can communicate with each other node in the network.

Nonrouting Nodes

Nonrouting nodes are Phase III nodes that have implemented a subset of Phase III capabilities. They must be located adjacent to a Phase III routing node. Although they can send messages to any Phase III node in the network, they can not perform "route-through" for an incoming message not addressed to themselves. A nonrouting node must be placed as an end node in the network, connected to the network by only one link. The nonrouting node can then both send and receive messages.

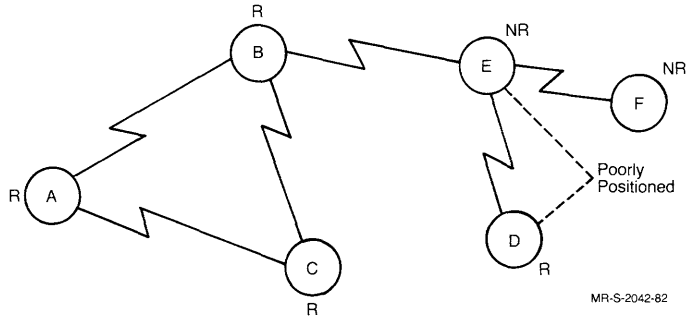
Diagrammatically:



Assume nodes A, B, C, and D are routing nodes; nodes E and F are nonrouting nodes; all nodes are Phase III nodes. E can send messages to A, B, C, D, F, and self, and can receive messages from each of them. Similarly, F can communicate with A, B, C, D, E, and self. As long as each node in the path from E or F to the destination node is a routing node, messages can continue through the network for a maximum of six hops. Messages from any node to E or F can also continue in this manner.

The placement of the nodes diagrammed suggests that D does more communicating with E and F than with A, B, and C. Note that if B goes off line, D (and E and F) can not send to or receive from A, B, or C. The network shown also suggests that A, B, and C expect to communicate with each other more than with D. Note that each of these nodes has a direct path to each of the others: one direct one-line or "hop" path; and one two-hop path: for example, A to B and A to B through C. You considered routing implications when you planned the topology of your network. Certain factors may, however, suggest changes that will increase efficiency. Actual traffic in a functioning network, new network nodes, and new or relocated hardware, may all suggest topological changes. NCP commands are available to accomplish this. Such changes should be made by the system manager, or someone delegated by the system manager. (Sections 2.4.3 and 2.5.1 are also related to routing.)

Now let us reverse the positioning of D and E as shown below:

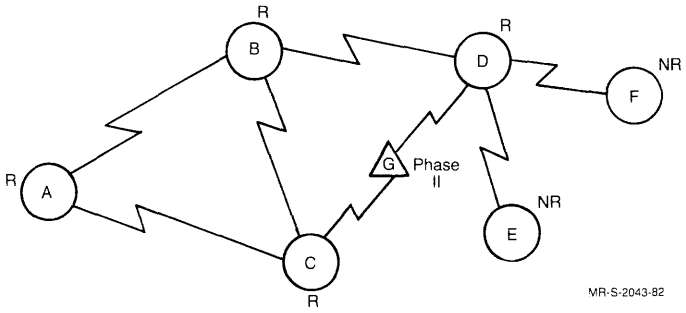


With this topology, both D and F can exchange messages only with E. E loses nothing in the repositioning: there is a direct line to D and F; all other nodes are routing nodes. With this positioning, we have three links into a nonrouting node. As previously mentioned, a nonrouting node must be connected to the network by only one link. The topology shown is illegal.

Phase II Nodes

Phase II nodes can participate in a Phase III network, but a Phase II node gains none of the DECnet capabilities added for DEcnet Phase III. The Phase II node does not support full routing: it can send packets only to an adjacent node to which it is physically connected; it cannot forward packets it receives to any other node in the network. Therefore, a Phase II node is logically located as an end node within a Phase III network. Phase III NCP commands will not be understood by the Network Management process in a Phase II node; Phase II NCP commands will not be understood by the NML in a Phase III node. An appropriate error message will be output if such commands are sent or received. With the exception of NCP, however, a Phase II node can communicate with an adjacent Phase III node: NFT, SETHOST, and user-written programs in MACRO will be understood by both Phase II and Phase III nodes.

Refer to the first diagram of nodes A, B, C, D, E, and F. Now, imagine a Phase II node, G, positioned between C and D, as shown:



Node G needs a point-to-point link to communicate. Both physically and logically, node G has only two links for communication, the two lines from its two adjacent nodes. G perceives the "network" as made up of three nodes: C, D, and itself. D perceives itself as having a link to G, but is (functionally) unaware of G's link to C. Likewise, C perceives its link to G, but does not perceive the D to G link. Both C and D perceive all nodes, but for both the path to G is the end of communication. Although we have added a node and two lines of communication, A, B, E, and F perceive the network exactly as shown in the diagram without node G.

2.4.2 The Adaptive Least-Cost Routing Algorithm

Routing and related functions are accomplished in the Transport Layer of the DN20 communications front end. Although routing is transparent, you, as the system manager or operator can, through the NML interface to Transport, obtain information about and control the routing operation. Therefore, you need to understand how routing is determined. More specifically, you need to understand the significance of parameter changes in the NCP commands that you use. The "Transport language" includes a few common terms with transport-specific meanings. Figure 2-4 is designed to define these terms and, at the same time, provide you with a basic understanding of the DECnet-20 routing algorithm. Be sure to read the legend first, and refer to the diagram as you mentally follow the examples at the bottom of the figure.

2.4.3 Congestion and Packet Delivery

The Transport Layer attempts to keep network traffic through its node at a manageable level. It accomplishes this through a stringent assignment of buffers. When buffer resources are exhausted, Transport refuses packets from NSP and discards packets from other nodes.

If the data link remains open, NSP guarantees eventual delivery (once it accepts the message) between Phase III nodes in a network. Timely delivery occurs under normal traffic loads.

2.5 SUMMARY OF PHASE II/PHASE III COMPATIBILITY

There are three types of nodes permitted in a Phase III network:

Phase III Full - a Phase III Full routing node has software that satisfies the route-through capability. This node has no restrictions in connecting to other Phase III nodes.

Phase III Nonrouting - A Phase III Nonrouting node implements a subset of Phase III capabilities. It can always communicate with its adjacent Phase III node. In addition, when located adjacent to a Phase III Full node, the Full Routing node will accomplish the "route-through" for the nonrouting node.

Phase II - this is a DECnet Phase II node with nothing added. Phase II nodes can route to self, or to an adjacent node.

2.5.1 Topological Restrictions

To achieve maximum performance within the network, certain types of nodes must be strategically placed within the network. A Phase III Full routing node is not restricted in placement within the network. Because a Phase II node has no route-through capability, it can communicate only to itself and adjacent nodes. Therefore, the optimum location for a Phase II node in a Phase III network is as an end node. A Phase III nonrouting node must be adjacent to a Phase III full routing node. A nonrouting node is permitted only one link into the network, and is positioned as an end node.

2.5.2 Implementation Restrictions

Although the DECnet Phase (II or III) is certainly the major factor that determines the set of network functions available at a particular node, it is not the only factor. Even if all nodes in your network are Phase III nodes, it is very likely that several nodes will support different sets of network functions. When any two nodes communicate, their interaction is limited to functions common to both.

A Phase III node need only implement the Phase III Minimum Subset of NCP commands. A node is not required to implement any of the many commands outside the subset if the node has no use for them. You will not receive a "completed successfully" response to a command that has not been implemented by the executor of that command. The response will be "Failed" followed by the reason for the failure (error message). The commands listed below are the Minimum Subset for Phase III nodes. Commands in the minimum subset should be available for all Phase III nodes. The required capability is described first. The NCP commands that are required to implement the capability follow.

- Network Services Layer counters on at least one node of any pair of nodes that will communicate using logical links:

  ```
  SHOW NODE nodeid COUNTERS
  ZERO NODE nodeid COUNTERS
  ```

- Display minimal information about a point-to-point communications link and node:

  ```
  SHOW CIRCUIT cktid SUMMARY
  SHOW NODE nodeid SUMMARY
  ```

- Test communication using the node level logical link loop test:

  ```
  SET NODE nodeid NAME nodename CIRCUIT cktid
  CLEAR NODE nodeid NAME CIRCUIT
  SHOW NODE nodeid SUMMARY
  LOOP NODE nodeid [accessctl] [WITH blocktype]
                               [COUNT count]
                               [LENGTH length]
  ```

- Disable a point-to-point communication link:

  ```
  SET CIRCUIT cktid STATE OFF
  ```

- Disable a node:

  ```
  SET NODE nodeid STATE OFF
  ```

  This command is given for the local node or the executor node only.

- Display and zero Data Link Layer counters on all multipoint control nodes and on at least one end of each point-to-point link:

```
SHOW CIRCUIT cktid COUNTERS
SHOW LINE lineid COUNTERS
ZERO CIRCUIT cktid COUNTERS
ZERO LINE lineid COUNTERS
```

- Display and zero Transport Layer counters and display and control of Transport Layer events for all routing nodes:

```
SHOW CIRCUIT cktid COUNTERS
SHOW NODE nodeid COUNTERS
ZERO CIRCUIT cktid COUNTERS
ZERO NODE  nodeid COUNTERS
SET LOGGING sinktype EVENTS eventlist [sourcequal] [sinknode]
                    NAME sinkname
                    STATE sinkstate
CLEAR LOGGING sinktype EVENTS eventlist [sourcequal] [sinknode]
                      NAME
SHOW LOGGING sinktype SUMMARY [sinknode]
```

- For central management nodes, send NCP commands to remotely managed nodes:

```
TELL nodeid [accessctl] NCP command
SET EXECUTOR NODE nodeid [accessctl]
CLEAR EXECUTOR NODE
```

DECnet-20 V3.0 can parse the complete Phase III NCP command set (except for the commands excluded in Section 1.6), and can format all possible responses. This does not mean that all commands with all possible parameters can be executed. (The complete set of NCP commands is described in detail in Chapters 5, 6, and 7.)

The following DIGITAL computer systems all support some set of DECnet functions. Because one or more of these systems may be in your network now, or in the future, the basic texts for each operating system are listed. Manuals that describe the capabilities of nodes in your network should be available at your site. (There is additional documentation for these systems. The manuals listed represent the minimum you should have on hand.)

RSX DECnet	RSX DECnet User's Guide RSX DECnet System Manager's Guide
DECnet-IAS	RSX/IAS DECnet User's Guide RSX/IAS DECnet System Manager's Guide
DECnet-VAX	DECnet-VAX User's Guide DECnet-VAX System Manager's Guide
DECnet/E	RSTS/E DECnet/E System Manager's Guide
DECnet/RT	RT-11 DECnet/RT User's Guide

Because network software is periodically updated, it is suggested that the local system manager contact the system managers at the sites of nodes in your network that are other than TOPS-20 systems. They will be able to give you the order numbers of the manuals appropriate for the software that they are currently using.

2.6 EVENT LOGGING

The Event Logger is one of the modules in the Network Management Layer. This module provides the capability for logging significant events - such occurrences as a line shutting down, errors in packet headers, or error thresholds reached. Logged events help in maintaining the network, recovering from failures, and planning for the future. See Tables 7-1 and 7-2 for a complete list of events that can be logged from each of the DNA functional layers.

The operator or system manager can control event logging by using the NCP command SET LOGGING EVENT eventlist. Once the events to be logged have been selected and set and the logging sinks determined, the operator or manager can examine the event logger's output periodically. Several avenues that might be followed in utilizing the logged data follow. These are only suggestions. Familiarity with the network, gained in daily operations, will suggest many more ways to capitalize on the event logging feature.

- Network application programs can be tested by setting specific counters to zero, running the application program and then examining the event output to observe how the program affected node and/or network performance.

- Counters specific to the Transport Layer (circuit counters that monitor terminating and transit congestion loss and circuit downs, for example) can be used to identify potential problems.

- An examination of the times and dates when potentially useful information is lost can be helpful. There are many queues and their lengths cannot be infinite. Circumstances will occur when an event will no longer fit on the queue. This will be recorded as an "event-lost" event, not as an error. All processed events are time stamped and identified as to source node and entity name, if any is indicated. On the basis of these factors certain changes can be made: the logging sink can be moved, peak loads can be adjusted by schedule changes, or particular events can be dropped from logging. Any changes so made should be evaluated by a reexamination of the same event counts that suggested the adjustment.

2.7 OPERATOR FUNCTIONS AND RESPONSIBILITIES

An "operator", in the context of this manual, is any terminal user with OPERATOR or WHEEL privileges who is given the responsibility for entering NCP commands or running system or network-related programs (requiring OPERATOR privileges) for the purpose of monitoring or controlling the network. An operator's duties may consist of well defined, simple, and repetitive tasks, or may consist of complex tasks that can not be predefined. Some operators will have responsibilities that combine both types of duties. Because of the many variations in personnel and site management, your responsibilities as an operator may be quite different from those of another operator. However, whatever these responsibilities are, it is important that you:

- Have a clear idea of what your responsibilities are

- Have (or gain) both the software and hardware knowledge needed to perform your assigned tasks

- Consult with the system manager, or someone designated by the system manager, whenever you are in doubt about a procedure.

A few NCP commands have the potential for disrupting the entire network. Several commands will, if formatted with valid but ill-chosen parameters, at least disrupt your own node's performance. No operator should enter an NCP command without a complete understanding of the probable effect of the command. Do not make changes in the permanent data base of a remote node without consulting the system manager at the remote node. All commands to change either the volatile or permanent data base of a remote node must include any required USER and PASSWORD parameters.

A list of functions commonly performed by operators follows. As an operator, you may be responsible for one or several of the following functions:

- Automatic startup of the network

- Manual startup of the network

- Manually loading the DN20 (the DECnet-20 communication front end) if required

- Starting up software or devices needed by the network (GALAXY, printer, for example)

- Loading and starting adjacent nodes

- Monitoring local and remote nodes

- Securing a dump after a crash (local or remote node)

- Changing parameters in the permanent or volatile (temporary) data base

- Shutting down a node or all network operation

- Changing the state of network entities

- Changing the routing path indirectly by changing the line cost

- Gathering (and possibly analyzing) statistics (events, errors, performance)

- Running SPEAR for network events and errors

- Using network command files and possibly creating command files to be executed at scheduled intervals

- Testing network application programs

- Recognizing potential problems (high response time, high error rate, for example)
- Examining the central console log file for breaches of security
- Using diagnostic tools
- Keeping all network-related logs and forms up-to-date

Once you have a clear idea of your responsibilities, concentrate first on the sections of this manual that describe the functions you need. Get working at the terminal as soon as possible. Use the Index to be sure you have read all pertinent information. The Index contains pages that reference examples and Appendix material as well as the pages of primary text in the main section of the manual.

2.8 SYSTEM MANAGER FUNCTIONS AND RESPONSIBILITIES

The DECnet-20 System Manager has final responsibility for the total functioning of the node in the network. The tasks that the manager performs depend on several factors. The principal factors are:

- The degree to which the management of the network is distributed. (Will there be standards and procedures from a central management node? Is the local node responsible for some or all of the control of other nodes in the network?)
- The complexity of the network of which the local node is a member. (Are there non-DECnet-20 nodes in the network? What and where are available resources?)
- The knowledge and experience of the manager's staff. (How can responsibilities best be allocated? What training is needed and how can it best be achieved?)

Ideally, a system manager concentrates on activities that call for planning and require a high degree of knowledge and expertise. Actually, however, you may frequently have to perform some of the operator duties described in the previous section. Listed below are the functions that are the peculiar responsibility of a system manager in a "typical" DECnet-20 environment.

- Generating and installing the DECnet-20 software. A DIGITAL software specialist performs the original generation and installation of the DECnet software. You should, however, follow the procedure and read the appropriate manual. The network may grow in number of nodes and other physical resources. Some of the steps followed in the original generation and installation may be repeated in the future by you as system manager. Some of the parameters established during system generation may require change due to changes in resources. You will become aware of these and be better prepared to make intelligent changes if you use the generation and installation procedures as a learning device.

- Assigning personnel to specific duties. For reasons of security, as well as efficiency, operators should be aware of those functions they are not to perform, as well as those for which they have responsibility. For example, certain NCP commands that have the potential for disrupting services, should be restricted in use.

- Providing appropriate documentation to all operators who will use NM functions.

- Scheduling network services. To the extent possible, network startup and shutdown should be set up automatically at the times dictated by the nature of the node's needs. At least until operators become proficient in their use of the network, the manager should consider a schedule of stand-alone time for demonstrations and practice. Initial schedules should attempt to avoid extremes of traffic, but the ideal schedule can not be built until traffic patterns have been analyzed.

- Handling of physical resources. The manager should provide both hard copy and on line sources of information on the node locations of processors, terminals, peripherals, network application programs, remote stations, and any other facilities to be used in the network. The manager should provide information to all operators on the use of all facilities. All resources should be analyzed for performance. Error and performance data should be kept for maintenance engineers and specialists, as well as for site planning and scheduling.

  Each operator should be able to take simple recovery procedures (manual reloads, for example). Systems analysts, network programmers or the manager should set up command files for procedures that will be repeated frequently. Not only will response be faster, but this procedure can be used to increase the operator's effectiveness in network activity.

- Checking the surveillance log periodically. Whenever OPR is running and you have typed "ENTER NCP", the system automatically records all OPR/NCP activity on the network. This log includes the terminal number where the NCP command was entered and the command itself. The current log has the file name PS:<SPOOL>OPERATOR-SYSTEM.LOG.n. When ORION is started or restarted, this file is copied to PS:<SPOOL>OPERATOR-SYSTEM-LOG.nnn.n where nnn is the next highest generation. This is the file you can examine to observe all network activity by date and time. (Use the VDIRECTORY command to find the date and time you wish to observe because there are frequently multiple files.)

- As the system manager, you should at all times be aware of the operational status of the network - both nodes and devices used in your network activities. Functions you may have to perform include: adjustment to line costs to achieve new routing paths, shutting down network components or circuits, sending messages to all network operators, and analyzing errors and failures.

- Bringing new devices, circuits, and users on line.
- Gathering statistics on network use, traffic congestion, errors, and failures.

You may delegate many of your responsibilities, but you are a key person in the success or failure of your network. You can ensure success by closely monitoring both the system and the manner in which your staff performs. You may have to assume some training duties for inexperienced operators.

PART II
DECnet OPERATION

CHAPTER 3

RUNNING DECnet-20

3.1 DECnet STARTUP PROCEDURES

To run DECnet on the DECSYSTEM-2040S/2060, the communication front end processor (DN20) must be loaded with the appropriate DECnet software. This normally occurs automatically when TOPS-20 is loaded. It can also be performed manually at any time after TOPS-20 is loaded.

3.1.1 Automatic Startup on a DECSYSTEM-2040S/2060

One of the first programs executed when TOPS-20 is loaded is SYSJOB, a system process that starts up other system processes. SYSJOB executes the file PS:<SYSTEM>SYSJOB.RUN. This file contains RUN commands for major system programs (for example, ORION, QUASAR, INFO, MAILER, and LPTSPL). SYSJOB also starts PTYCON which executes the PTYCON.ATO command file.

PTYCON subjobs run the network management program NMLT20, the MCBNRT server, FAL, X29SRV (PSI only) and OPR. The OPR subjob processes command files that start all necessary batch streams, printers, and readers. In addition, the OPR subjob processes the file NCP.CMD. This command file contains the NCP commands required to setup the network management volatile data base.

These commands specify the node names and addresses (node numbers) for the network as well as the parameters required to load and dump the DN20 front end. After OPR processes the NCP command that enables autoservicing for the circuit to the front end, the network management program will check the status of the front end and, if necessary, automatically initiate a down line load.

After you receive the topology message indicating that the node is online, both the TOPS-20 KL10 node and the MCB DN20 node are ready for network activity. If you have an RJE station and wish to start it up at this time, follow the procedures in Section 3.2.4. Two procedures for observing CHK11 output are described in Section 3.7.

Figure 3-1 shows a dialogue to bring up TOPS-20 V5.1 with DECnet-20 V3.0. The CPU in the example is a DECSYSTEM-2060. Startup procedures for your system will be similar, but differences may exist because of the hardware or software at your site. (Because the first command in the PTYCON.ATO file is "SILENCE", not all steps in the startup procedure are visible in the example dialogue.)

```
(Mount the PS:  disk pack on drive 0.  Press the ENABLE rocker  switch
while pressing the DISK rocker switch on the console front end.)

RSX-20F VB14-45 13:51 12-Nov-81

[SY0: REDIRECTED TO DB0:]
[DB0: MOUNTED]
KLI -- VERSION VB12-27 RUNNING
KLI -- KL10 S/N: 2476., MODEL B, 60 HERTZ
KLI -- HARDWARE ENVIRONMENT:
 MOS MASTER OSCILLATOR
 EXTENDED ADDRESSING
 INTERNAL CHANNELS
 CACHE

KLI -- MICROCODE VERSION 275 LOADED
KLI -- ALL CACHES ENABLED

LOGICAL MEMORY CONFIGURATION.
  ADDRESS  SIZE  INT  TYPE CONTROLLER
000000000  512    4   MF20  10

KLI -- CONFIGURATION FILE WRITTEN
KLI -- BOOTSTRAP LOADED AND STARTED


[PS MOUNTED]

System restarting, wait...
ENTER CURRENT DATE AND TIME: 7-21-82 1755

YOU HAVE ENTERED WEDNESDAY, 21-JULY-1982 5:55PM
 IS THIS CORRECT (Y,N) Y
WHY RELOAD? SA
RUN CHECKD? N

RUNNING DDMP


SYSJOB 5(14) STARTED AT 21-JUL-82 1775
RUN SYS:ORION
RUN SYS:QUASAR

****
 21-JUL-82 17:55:12 - TGHA 2(6) RUNNING FIRST TIME.
****
RUN SYS:MOUNTR
RUN SYS:INFO
RUN SYS:MAILER
RUN SYS:MAPPER
RUN SYS:LPTSPL
RUN SYS:LPTSPL
RUN SYS:LPTSPL
RUN SYS:CDRIVE
RUN SYS:SPRINT
```

Figure 3-1 Startup Dialogue for TOPS-20 Version 5.1 DECnet-20 Version 3.0 on a DECSYSTEM-2060

```
JOB 0 /LOG OPERATOR XX OPERATOR
ENA
^ESET LOGIN PSEUDO
^ESET LOGIN CONSOLE
^ESET OPERATOR
PTYCON
GET SYSTEM:PTYCON.ATO
/
JOB 1 /LOG OPERATOR XX OPERATOR
ENA
RUN SYS:BATCON
/
SJ  0: @LOG OPERATOR OPERATOR
SJ  1: @LOG OPERATOR OPERATOR
SJ  0:  JOB 1 ON TTY220 21-JUL-82 17:55:25
SJ  1: @ENA
SJ  0: $^ESET LOGIN PSEUDO
SJ  1:  JOB 2 ON TTY221 21-JUL-82 17:55:25
SJ  1: @ENA
SJ  0: $^ESET LOGIN CONSOLE
SJ  0: $^ESET OPERATOR
SJ  0: $PTYCON
SJ  1: $RUN SYS:BATCON
SJ  0: PTYCON> GET SYSTEM:PTYCON.ATO
SJ  0: PTYCON> SILENCE

[From OPERATOR on line 226 to all: SYSTEM IN OPERATION]
SJ  0: PTYCON> W ALL
SJ  0: NML(0)      3          OPERATOR   NMLT20      RN          0:0:1
SJ  0: NODES(1)    4          OPERATOR   NODES       RN          0:0:0
SJ  0: MCBNR(2)    5          OPERATOR   MCBNRT      RN          0:0:0
SJ  0: FAL(3)      6          OPERATOR   FAL         RN          0:0:1
SJ  0: OPR(4)      7          OPERATOR   OPR         TI          0:0:2
SJ  0: X29(5)     10          OPERATOR   X29SRV      RN          0:0:1
SJ  0: PTYCON>
SJ  0:**** NML(0) 17:56:10 ****
SJ  0:
SJ  0:  [LINK-WATCHER]:     AUTO-DUMP REQUEST #18 ENTERED FOR DTE-0-1
SJ  0: PTYCON> CONN OPR
SJ  0: [CONNECTED TO SUBJOB OPR(4)]
SJ  0: **** NML(0) 17:56:24 ****
SJ  0:
SJ  0:  [NML-PROCESSOR]:    AUTO-DUMP DONE ON DTE-0-1
SJ  0:  [NML-PROCESSOR]:    AUTO-LOAD REQUEST #19 ENTERED FOR DTE-0-1
SJ  0: **** OPR(4) 17:56:24 ****
SJ  0: **** NML(0) 17:57:41 ****
SJ  0:  [NML-PROCESSOR]:    AUTO-LOAD DONE ON DTE-0-1
SJ  0: **** OPR(4) 17:57:41 ****
SJ  0:
SJ  0: 17:57:41         -- NETWORK NODE D2476A IS ONLINE

^C
 DSME 5.1 DECNET TEST SYSTEM, TOPS-20 Monitor 5.1(4766)
@
```

Figure 3-1 Startup Dialogue for TOPS-20 Version 5.1 DECnet-20 Version 3.0 on a DECSYSTEM-2060 (Cont.)

3.1.2 Manual Startup of DECSYSTEM-2040S/2060 Communication Front End

The TOPS-20 and the MCB Operating Systems may crash independently. When TOPS-20 fails, it attempts an automatic restart. If this is successful, the Network Management Program running under TOPS-20 checks to see if the MCB system is running. If the MCB has also failed, it is loaded; if it has not failed, there is no action. Therefore, a manual startup is only required when normal automatic procedures fail.

The following sequence of commands may be used if the DN20 communications front end fails to come up when the TOPS-20 node comes up or following a crash that is not followed by an autoload. Normally, the first step will load the front end. If the load fails, continue with next sequence of commands.

Step 1. Type the following at your terminal:

```
@ena (RET)
$opr (RET)
OPR>enter ncp (RET)
NCP>set cir dte-0-1 service enabled (RET)
NCP>load node d2136a (RET)
```

The circuit identification, DTE-0-1, represents the circuit used to load the DN20. Normally, this will be the DTE-0-1; if it is not, use the identification of the circuit actually used for the load.

Step 2. If the load fails, enter NCP and type an NCP command for each of the required parameters, inserting the values appropriate for your MCB DN20 node. Include as the last command, SET CIRCUIT DTE-0-1 SERVICE ENABLED. The value DTE-0-1 must identify the circuit over which the load will be effected. The command file example in the next section can be used to check format. "D2136A" in the example is the name of the network front end.

The following parameter values are required for the load:

```
Secondary Loader
Tertiary Loader
Service Circuit
CPU = PDP11
Load File
Dump File
Secondary Dumper
Host
```

You can now repeat:

```
NCP>load no d2136a (RET)
```

3.1.3 Manual Startup with Command Files

You will find it helpful to use command files for procedures that may be repeated. You will save unnecessary typing, avoid such errors as misspelling and transposition, and ensure against errors of omission. You will probably find it helpful to keep a set of command files on

your own directory. Following is an example of a command file for loading the front end named D2136A:

```
$type feload.cmd
clear executor node
set node d2136a cpu pdp-11
set node d2136a service circuit dte-0-1
set node d2136a secondary loader sys:dtemps.sys
set node d2136a tertiary loader sys:dtempt.sys
set node d2136a load file sys:d2136a.sys
set node d2136a dump file sys:d2136a.dmp
set node d2136a secondary dumper sys:dtedmp.sys
set node d2136a host kl2136
set cir dte-0-1 service enabled
load node d2136a
$
```

The above command set includes all information needed by the system for a manual load. However, when the system is brought up, several of the commands shown above are processed when the NCP.CMD file is executed. Normally, only the last command is required.

Assume the TOPS-20 system is running. Before using the network, you check the status of the communications front end and receive the following:

```
NCP>show node d2136a status (RET)
NCP>
 9:24:09         NCP
                 Request # 253; Show Node Complete

                 Remote Node = 129 (D2136A)

                   State = Reachable
                   Active Links = 3

.
.
.                (A few NCP commands are entered)
.
```

After a few minutes, the following topology message is output to your terminal:

```
NCP>
 9:30:04         --  Network topology  --

Nodes off-line:
D2136A
```

When you fail to receive output in a few minutes informing you that D2136A is online, you can assume that the DN20 has failed to autoload. You TAKE the file FELOAD.CMD. Input and output follow:

```
$^Esend * I am going to load the front end - in 5 minutes (RET)

[From CIRINO on line 25 to all: I am going to load
  the front end - in 5 minutes
$opr
       (ESC)    (ESC)                (ESC)
OPR>disaBLE outPUT-DISPLAY (of) all-MESSAGES (RET)
OPR>
9:59:51          --Output display for OPR modified--
OPR>ent ncp (RET)
```

```
NCP>take feload.cmd(RET)
NCP>
10:00:34        NCP
                Clear Executor Complete
NCP>
10:00:36        NCP
                Request # 266 Accepted
NCP>
10:00:42        NCP
                Request # 256; Set Node Completed
NCP>
10:00:42        NCP
                Request # 257; Set Node Failed, Component in wrong
state
```

NOTE

This failure will not affect the results. The DTE-0-1 has already been established as the service circuit when the system came up.

```
10:00:42        NCP
                Request # 258; Set Node Completed

10:00:42        NCP
                Request # 259; Set Node Completed

10:00:42        NCP
                Request # 260; Set Node Completed

10:00:42        NCP
                Request # 261; Set Node Completed

10:00:42        NCP
                Request # 262; Set Node Completed

10:00:42        NCP
                Request # 263; Set Node Completed

10:00:42        NCP
                Request #264; Set Circuit Completed

10:02:10        NCP
                Request # 265; Load Node Completed

NCP>show node 129(RET)
NCP>
10:02:26        NCP
                Request # 271; Show Node Summary Completed

                Remote Node = 129 (D2136A)

                  State = Reachable
                  Active Links = 0
NCP>
```

The first command to the OPR program, disables the display of all messages. This allows you to use the network facilities without interruption by numerous messages about batch programs, tapes mounted, and reports printed, for example. You will receive all NCP responses and messages, as well as the information message "--- ORION not running ---." You will not get topology messages unless you include the OPR command ENABLE OUTPUT-DISPLAY USER-MESSAGES after the DISABLE command.

3.2 DECnet-20 RESTART PROCEDURES

3.2.1 Restart Procedure for the Orion Process

The ORION process receives the NCP commands from OPR in the form of IPCF messages and passes these messages to the NCP Receiver in the Network Management Layer. Therefore, if ORION stops running, there can be no response to commands. Refer to Section 4.2 and Figure 1-3.

If the ORION process fails while you are entering NCP commands the following message appears on your terminal:

```
--- ORION not running ---
```

The ORION process can be restarted with the following command sequence:

```
@ENABLE (RET)                              ;enable capabilities
$^ESPEAK (RET)                             ;give ^ESPEAK command
 [PLEASE TYPE SYSJOB COMMANDS - END WITH ^Z] ;
KILL ORION (RET)
RUN ORION (RET)                            ;restart ORION
<CTRL/Z>                                   ;end the ^ESPEAK command
$DISABLE (RET)                             ;disable capabilities
@
```

If the restart is successful, the next step is to restart NMLT20, the Network Management program containing the NCP module among other control modules. (See Section 3.2.2.) This is necessary because the new copy of ORION has no knowledge of NMLT20's location. When NMLT20 is loaded, the initialization routine identifies itself to ORION. If the communication front end has also crashed, you will also need to follow the procedures in Section 3.1.2.

3.2.2 Restart Procedure for NMLT20

If the OPR program can not communicate with NMLT20, you will receive the following message when you type an NCP command:

```
--- NCP not running ---
```

This message may be caused by ORION's inability to communicate with NMLT20 (as when ORION is restarted without restarting NMLT20). NMLT20 must be restarted.

If you receive no response from an NCP command within a few minutes, enter the NCP command SHOW QUEUE. If NML refuses to respond to the command, it may simply be busy loading a node. If there is no response to SHOW QUEUE within a few minutes, assume NMLT20 has stopped running. (Alternatively, you can use your normal procedure for determining the state of a program.) If it is determined that NMLT20 is hung, then a dump of NMLT20 should be taken and, if possible, a description of the environment should be noted.

The NMLT20 program can be restarted with the following command sequence:

```
@ENABLE (RET)
$ADVISE OPERATOR (RET)
 TTY232, EXEC
 TTY2, OPR
 TTY233, FAL
 TTY234, NMLT20
 TTY235, BATCON
 TTY236, PTYCON
TTY:  236 (RET)
 [Pseudo-terminal, confirm]
 Escape character is <CTRL>E, type <CTRL>^? for help
 OPERATOR Job 9 NMLT20

LINK FROM MARTIN, TTY 126
 [Advising]
^X
PTYCON>connect nml (RET)
^C
$ddt (RET)
^C
$save NMLT20.DMP (RET)
$reset (RET)
$run sys:NMLT20 (RET)
^X
PTYCON>connect opr (RET)
$^E
 [Advice terminated]
$OPR (RET)
OPR>enter ncp (RET)
NCP>take system:NCP.CMD (RET)
NCP>set circuit dte-0-1 service enabled (RET)
```

NMLT20 should be successfully restarted.

When NMLT20 is restarted, the volatile data base is lost. Therefore, the procedure to restart NMLT20 includes the NCP.CMD file. The NCP.CMD file sets the node names and addresses (numbers) for the network and sets the values for the TOPS-20 Network Management volatile data base. You are now ready to continue network activity.

3.2.3 Restart Procedure for X29SRV

If the X29SRV program fails, you will receive a fatal error message indicating the system error. (See Appendix C for X29SRV error messages.) Correct the system error, then restart X29SRV. The X29SRV program can be restarted with the following command sequence:

```
@ENABLE (RET)
$ADVISE OPERATOR (RET)
 TTY232, EXEC
 TTY2, OPR
 TTY233, FAL
 TTY234, NMLT20
 TTY235, BATCON
 TTY236, PTYCON
 TTY237, X29
TTY:  237 (RET)
 [Pseudo-terminal, confirm]
 Escape character is <CTRL>E, type <CTRL>^? for help
 OPERATOR Job 9 X29

LINK FROM MARTIN, TTY 126
 [Advising]
^X
PTYCON>connect X29
^C
$reset (RET)
$run sys:X29SRV (RET)
^X
```

X29SRV should be successfully restarted. Otherwise, if the restart fails, you will receive a fatal error message that indicates the system error that caused X29SRV to fail. (See Appendix C for X29SRV error messages.)

3.2.4 Restart Procedure for DN200

If your site includes an RJE station, you constructed the file NCPRJE.CMD during the NIPGEN procedure. This file includes commands that establish RJE-related values in the DN20 volatile data base. Access parameters are required for setting these values. In the example that follows, the "USER OPERATOR PASSWORD SECRETE" parameters (account is defaulted) permit the setting of the specified values in the DN20 volatile data base. The "DECNET KL2102 3R1" access parameters permit access to the load and image files at the KL2102 host node. The "DN200-DUMPS KL2102 3R1" access parameters permit writing the dump files at the KL2102 host node.

If the MCB front end has not been reloaded since you last started the DN200, the RJE-related values are still in the volatile data base. The DN200 can be restarted without using the NCPRJE.CMD. If the MCB DN20 has been reloaded without including DN200 values, then execute the command file with a TAKE command to OPR or NCP.

You can reference the NCPRJE.CMD file by typing

```
$TYP SYSTEM:NCPRJE.CMD
```

on your terminal. Output should be similar to:

```
ENT NCP
SET EXE NOD D2102A USER OPERATOR PASSWORD SECRETE
SET NODE 255 NAME DN200
WAIT 2
SET NODE DN200 SERVICE CIRCUIT KDP-0-3
WAIT 2
SET NODE DN200 CPU PDP-11
WAIT 2
SET NODE DN200 SEC LOA KL2102"DECNET KL2102 3R1"::SYS:DMCMPS.SYS
WAIT 2
SET NODE DN200 TER LOA KL2102"DECNET KL2102 3R1"::SYS:DMCMPT.SYS
WAIT 2
SET NODE DN200 LOAD FILE KL2102"DECNET KL2102 3R1"SYS:DN200.SYS
WAIT 2
SET NODE DN200 DUMP FILE -
    KL2102"DN200-DUMPS KL2102 3R1"PS:<DN200-DUMPS>DN200.DMP
WAIT 2
SET NODE DN200 HOST KL2102
WAIT 2
SET CIRCUIT KDP-0-3 SERVICE ENABLED
CLEAR EXECUTOR NODE
RETURN
```

The DN200 can be reloaded at any time following the above procedure. Either go to the DN200 or call the DN200 operator on the telephone. The DN200 can be reloaded by turning the ON/OFF switch OFF and ON or by pressing the HALT button and CTRL button simultaneously and then pressing the BOOT button and CTRL button simultaneously.

3.3 CONTROLLING NODE, CIRCUIT, DTE, AND LINE STATE

DECnet-20 does not support the setting of node state using the NCP command

```
SET NODE nodeid STATE {OFF        }
                      {ON         }
                      {SHUT       }
                      {RESTRICTED }
```

You can, however, control the state of the circuits connected to your node. Using the OFF parameter in the following NCP command will, if there is no alternate path, cause the adjacent node to become UNREACHABLE. The circuit is nonoperational for network traffic. Therefore, you must consider the position of the node to be set. If the node is a full routing node, you may disrupt network traffic between several nodes.

```
SET CIRCUIT cktid STATE {OFF}
                       {ON }
```

You may wish to set a circuit OFF if traffic over the circuit becomes excessive, or if performance over the circuit becomes unacceptable.

For the PSI option, you can control X.25 access with the following command:

```
                                  {OFF  }  {DTE dte-address}
SET MODULE X25-PROTOCOL STATE     {ON   }  {KNOWN DTES     }
                                  {SHUT }
```

You may wish to turn the DTE off to terminate access to the PPSN.

Lines may also be controlled with the following command:

```
                          {OFF}
SET LINE lineid STATE     {ON }
```

Certain NCP commands require that a line be in the OFF state for execution of the command.

Setting circuit or line state to SERVICE or CLEARED is not supported by DECnet-20.

3.4 MONITORING NETWORK ACTIVITY

DECnet-20 software continually monitors both static and variable characteristics and accumulates statistics for nodes, circuits, and lines. These metrics are available to you in two major ways:

- by means of selected NCP commands
- by means of the SPEAR program

Considerable data is collected automatically. As a system manager, you are responsible for planning the "how" and "when" of collecting, organizing, and analyzing the data being collected by the system. If you take full advantage of the system's automatic monitoring facilities, the network will run more smoothly. Traffic patterns, hardware, and your site's needs will change in time. Parameters can be changed more efficiently when based on observations of actual statistics and characteristics. Periodic analysis will also help you prevent problems before they disrupt the network. Needed routing changes through adjustment of the COST parameter can often be made before performance becomes unacceptable. Before and after comparisons can aid in evaluating the effect of new applications or additions to hardware. Helpful changes in scheduling will be suggested by observing peaks in traffic. It is suggested that the SPEAR program be run daily. See Section 3.4.2 for details. The interval between the monitoring of other display types will depend upon local activity and can only be decided after some experience with your network. Because counters are zeroed when the node where they are located goes down, some data may be lost at that time. How much is lost will depend upon when the last data was captured. Using command or control files will save time and effort.

As an OPERATOR-user, you are responsible for providing the input according to plan and schedule. This requires a knowledge of the format of the appropriate NCP commands; an understanding of display types; familiarity with the specific items displayed according to entity, display type, and arguments and keywords used; and an understanding of the SPEAR program.

3.4.1 NCP Commands for Network Monitoring - Local and Remote

NCP commands that display current attributes and statistics may be used to monitor both the local node and remote nodes that have implemented the commands. To obtain output from a remote node, use either one of the following procedures:

Procedure #1.

```
NCP>set executor node mrvax (RET)

  (Wait for SET EXECUTOR COMPLETE message)

NCP>show ex char (RET)

  (Output will follow)
```

Procedure #2.

```
NCP>tell mrvax sh ex char (RET)
```

All relevant NCP commands for the local node and its communications front end follow:

Command	Entity	Keyword/Argument	
SHOW	NODE nodeid	STATUS	
	KNOWN NODES	CHARACTERISTICS	
	ACTIVE NODES	SUMMARY	
	CIRCUIT cktid	COUNTERS	
	KNOWN CIRCUITS		
	ACTIVE CIRCUITS		
	LINE lineid		
	KNOWN LINES		
	ACTIVE LINES		
	KNOWN MODULES		
	MODULE X25-ACCESS	CHARACTERISTICS	[KNOWN NETWORKS]
		COUNTERS	[NETWORK network-name]
		STATUS	
		SUMMARY	
	MODULE X25-PROTOCOL	CHARACTERISTICS	[DTE dte-address]
		COUNTERS	[KNOWN DTES]
		STATUS	[GROUP group-name]
		SUMMARY	[KNOWN GROUPS]
	MODULE X25-SERVER	CHARACTERISTICS	[KNOWN DESTINATIONS]
		COUNTERS	[DESTINATION destination-name]
		STATUS	
		SUMMARY	

The TOPS-20 node (KL10-E) keeps only limited information for the DTE-0-1 circuit and line. To get a display for any other circuit or line, or for plural entities, you must set the executor to the communications front end. To get any information for the DTE-0-1 except state and service enabled or disabled, set the executor to the communications front end.

To determine which of the above commands are available at non-DECnet-20 nodes, it is necessary to consult the documentation for the remote node's operating system. When you do this, check also for the SHOW LOGGING commands. These are not implemented for DECnet-20 V 3.0, but may be available on another DIGITAL operating system.

NOTE

Using the NCP commands in this manual to communicate between DECnet-20 V 3.0 nodes and other nodes on the network, requires that the other nodes have implemented Network Management Version 3.0 or later.

3.4.2 Using the SPEAR Program for Network Monitoring

All network errors and significant events are written by the system to the file SERR:ERROR.SYS (SERR: is a logical name for PS:<SYSTEM-ERROR>). The SPEAR program can be used to read these entries. You do not need privileges to run SPEAR. However, check any files to be used for SPEAR input or output. If accessing a file requires certain conditions, these should be met before you run the SPEAR program.

When the prompt SPEAR> appears on the terminal, you type the mode of the SPEAR program you wish to run. You will probably find "RETRIEVE" and "SUMMARIZE" modes most useful for monitoring the network. SPEAR next asks for specific information. Each request is followed by a default answer in parentheses. To take the default, you press the return key. Normal responses for both reports appear in the following examples. To select output of all network events and only network events (no monitor, tape, disk, and the like), you specify "Selection type" as "error" and "error class" as "network". The output will then include network errors and all other significant network events.

When SPEAR finishes selecting data as you have directed, SPEAR informs you that retrieval is complete. If you select the output default (DSK:RETRIE.RPT for the RETRIEVE mode and DSK:SUMMAR.RPT for SUMMARY mode), you can now type the report from your connected directory to your terminal, or send it to be printed.

The SPEAR program creates the appropriate output file (RETRIE.RPT or SUMMAR.RPT) in your connected directory. The next time you run SPEAR an existing file may be overwritten, depending on the file-generation-retention count in effect. Rename the previous file if you wish to save it.

The run time for the same SPEAR report can vary from a few minutes to over an hour. Run time will be greatly increased when records are allowed to accumulate in the ERROR.SYS file. Run time will be somewhat longer when the load average is high. If getting the report is critical, use ^ESET RUN-TIME-GUARANTEE to reduce run time. You may use CTRL/T to check both the run time and the fact that SPEAR is still running during the interval of retrieval. (Use of CTRL/T is shown in the examples.)

When there are many thousands of records to be searched, runtime may become excessively long. Copying the ERROR.SYS file to tape every 30, 60, or 90 days may not ensure a satisfactory runtime: fairly minor problems with network or system hardware or software can result in a proliferation of records sent to ERROR.SYS. As system manager, you or a delegated operator should monitor the file to prevent time-consuming SPEAR reports. Both the page count of the ERROR.SYS file (use the TOPS-20 command VDIR) and the record sequence of retrieved entries will help you determine when you should copy the ERROR.SYS file to tape.

All ERROR.SYS records should be kept on disk (current) or tape (historical). This complete copy is in addition to regular system backup.

Example 1a. SPEAR:RETRIEVE mode; Report format selection: FULL

```
@spear (RET)
Welcome to SPEAR for TOPS-20, Version 1(37)
Type "?" for help.

          (ESC)
           ↓
SPEAR> retRIEVE (RET)

RETRIEVE mode
-------------
     Event or packet file (SERR:ERROR.SYS): (RET)

     Selection to be (INCLUDED): (RET)

          Selection type (ALL): errOR (RET)

               Error class (ALL): netWORK (RET)

                    Event class (ALL): (RET)

               Next error class (FINISHED): (RET)

     Time from (EARLIEST): 14-Jul-82 9:00:00 (RET)

     Time to (LATEST): 14-Jul-82 11:00:00 (RET)

     Output mode (ASCII): (RET)

          Report format (SHORT): full (RET)

     Output to (DSK:RETRIE.RPT): (RET)

Type <cr> to confirm (/GO): (RET)
INFO - Retrieving selected entries from SERR:ERROR.SYS
 11:33:11 SPEAR Running at 412323 Used 0:02:02.2 in 21:09:50, Load  9.99
 12:08:33 SPEAR Running at 477624 Used 0:04:53.4 in 21:45:12, Load 10.33

INFO - Retrieval Complete       Total Entries = 13.

SPEAR> exit (RET)
$type retrie.rpt (RET)
```

```
SPEAR Version 1(37). Retrieval from SERR:ERROR.SYS
  Report generated 14-Jul-82 11:31:37-EDT
  As directed by user
  Selected window: 14-Jul-82 09:00:00-EDT to 14-Jul-82 11:00:00-EDT.
  Selected records are included
  Selection type is ERROR
  Report sent to DSK:RETRIE.RPT

**********************************************
PHASE III DECNET ENTRY
 LOGGED ON 14-Jul 09:01:09      MONITOR UPTIME WAS 1 DAY(S) 3:35:12
        DETECTED ON SYSTEM # 2137.
        RECORD SEQUENCE NUMBER: 37376.
**********************************************

Event type 4.7  Circuit down, circuit fault
From node 123. (D2137A), uptime was 1 day(s) 3:26:16


Circuit = KDP-0-0

  REASON = Line synchronization lost

**********************************************
PHASE III DECNET ENTRY
 LOGGED ON 14-Jul 09:08:55      MONITOR UPTIME WAS 1 DAY(S) 3:42:58
        DETECTED ON SYSTEM # 2137.
        RECORD SEQUENCE NUMBER: 37378.
**********************************************

Event type 4.10  Circuit up
From node 123. (D2137A), uptime was 1 day(s) 3:34:03


Circuit = KDP-0-0

  NODE = 16

.
.         (Record sequence #s 37382 and 37383 duplicate
.          above at 09:43:49

**********************************************
PHASE III DECNET ENTRY
 LOGGED ON 14-Jul 10:00:28      MONITOR UPTIME WAS 1 DAY(S) 4:34:30
        DETECTED ON SYSTEM # 2137.
        RECORD SEQUENCE NUMBER: 37384.
**********************************************

Event type 4.1  Node unreachable packet loss
From node 123. (D2137A), uptime was 1 day(s) 4:25:34


Circuit = KDP-0-1

  PACKET HEADER = 2/ 96 / 49203 / 3

.         (Eight more records duplicate first two
.          entries of "circuit down, circuit up". See
.          Example 1b for times.)
```

Example 1b. SPEAR:RETRIEVE mode: Report format selection: SHORT

```
@spear (RET)
Welcome to SPEAR for TOPS-20, Version 1(37)
Type "?" for help.

SPEAR> retRIEVE (RET)

RETRIEVE mode
-------------
     Event or packet file (SERR:ERROR.SYS): (RET)

     Selection to be (INCLUDED): (RET)

          Selection type (ALL): errOR (RET)

               Error class (ALL): netWORK (RET)

                    Event class (ALL): (RET)

               Next error class (FINISHED): (RET)

     Time from (EARLIEST): 14-Jul-82 9:00:00 (RET)
```

```
     Time to (LATEST): 14-Jul-82 11:00:00 RET

     Output mode (ASCII): RET

          Report format (SHORT): RET

     Output to (DSK:RETRIE.RPT): RET

Type <cr> to confirm (/GO): RET
INFO - Retrieving selected entries from SERR:ERROR.SYS
 12:21:00 SPEAR Running at 437063 Used 0:05:19.6 in 21:57:39, Load 11.19
 12:49:38 SPEAR Running at 421272 Used 0:07:08.9 in 22:26:16, Load 13.45

INFO - Retrieval Complete       Total Entries = 13.

SPEAR> exit RET
$type retrie.rpt RET

SPEAR Version 1(37). Retrieval from SERR:ERROR.SYS
  Report generated 14-Jul-82 12:15:41-EDT
  As directed by user
  Selected window: 14-Jul-82 09:00:00-EDT to 14-Jul-82 11:00:00-EDT.
  Selected records are included
  Selection type is ERROR
  Report sent to DSK:RETRIE.RPT

37376. 09:01:09 DECNET Event type 4.7  Circuit down, circuit fault
                     From node 123. (D2137A)
                     uptime was 1 day(s) 3:26:16
37378. 09:08:55 DECNET Event type 4.10  Circuit up
                     From node 123. (D2137A)
                     uptime was 1 day(s) 3:34:03
37382. 09:43:49 DECNET Event type 4.7  Circuit down, circuit fault
                     From node 123. (D2137A)
                     uptime was 1 day(s) 4:08:55
37383. 09:43:49 DECNET Event type 4.10  Circuit up
                     From node 123. (D2137A)
                     uptime was 1 day(s) 4:08:56
37384. 10:00:28 DECNET Event type 4.1  Node unreachable packet loss
                     From node 123. (D2137A)
                     uptime was 1 day(s) 4:25:34
37387. 10:17:45 DECNET Event type 4.7  Circuit down, circuit fault
                     From node 123. (D2137A)
                     uptime was 1 day(s) 4:42:49
37388. 10:17:45 DECNET Event type 4.10  Circuit up
                     From node 123. (D2137A)
                     uptime was 1 day(s) 4:42:50
37389. 10:22:15 DECNET Event type 4.7  Circuit down, circuit fault
                     From node 123. (D2137A)
                     uptime was 1 day(s) 4:47:22
37391. 10:27:25 DECNET Event type 4.10  Circuit up
                     From node 123. (D2137A)
                     uptime was 1 day(s) 4:52:30
37393. 10:34:37 DECNET Event type 4.7  Circuit down, circuit fault
                     From node 123. (D2137A)
                     uptime was 1 day(s) 4:59:44
37394. 10:34:39 DECNET Event type 4.10  Circuit up
                     From node 123. (D2137A)
                     uptime was 1 day(s) 4:59:45
37396. 10:46:09 DECNET Event type 4.7  Circuit down, circuit fault
                     From node 123. (D2137A)
                     uptime was 1 day(s) 5:11:15
37397. 10:46:09 DECNET Event type 4.10  Circuit up
                     From node 123. (D2137A)
                     uptime was 1 day(s) 5:11:15
$
```

Compare the "FULL" and "SHORT" forms, sequence number by sequence number. Because runtime depends primarily on search time for user-selected records, the short form saves little time and has certain disadvantages. If you need any of the following information, select the long form:

- detail on reason for error, if event was an error

  Example: "Line synchronization lost"
  for "circuit down, circuit fault"

- identification of the data link associated with an error

  Example: "Circuit = KDP-0-1" for "circuit down" and "circuit up"

- for a "circuit up" event, the node at the remote end of the circuit

  Example: "Circuit = KDP-0-0

  NODE = 16"

The above list may not be complete. Other events may reveal additional missing items in the short form.

The short form may be useful, because of its simpler format, for locating specific events. Once found you can then limit your "selected window" for the full report.

Example 2. SPEAR:SUMMARIZE mode

```
$spear(RET)
Welcome to SPEAR for TOPS-20. Version 1(37)
Type "?" for help.

SPEAR> summARIZE(RET)

SUMMARIZE mode
--------------
     Event file (SERR:ERROR.SYS):(RET)

     Time from (EARLIEST): 14-Jul-82 9:00:00(RET)

     Time to (LATEST): 14-Jul-82 11:00:00(RET)

     Report to (DSK:SUMMAR.RPT):(RET)

Type <cr> to confirm (/GO):(RET)
 09:08:25 SPEAR Running at 437576 Used 0:01:18.2 in 0:17:45, Load  6.24
 09:22:49 SPEAR Running at 421272 Used 0:03:06.2 in 0:32:10, Load  5.00
INFO - Summarizing SERR:ERROR.SYS
INFO - Now sending summary to DSK:SUMMAR.RPT
INFO - Summary output finished

SPEAR> exit(RET)
$type summar.rpt(RET)
```

File Environment

```
     SPEAR Version 1(37)
     Input file:   SERR:ERROR.SYS Created: 12-Jul-82 08:07:25-EDT
     Output file:  DSK:SUMMAR.RPT

     Date of first entry processed: 14-Jul 09:01:09
     Date of last entry processed:  14-Jul 10:59:58

     Number of entries processed: 29.
     Number of inconsistencies detected in error file: 0.
```

Entry Occurrence Counts:

```
      5. MASSBUS ERROR ...(111)
      4. STATISTICS ...(114)
      6. CONFIGURATION CHANGE ...(115)
      1. FRONT END DEVICE ERROR ...(130)
     13. PHASE III DECNET ENTRY ...(240)
```

Front-end Summary:

```
      1. DLSCAN
```

RH20 Channel/Controller Summary:

	Hard	Soft
# 4	0.	1.
# 5	2.	0.
# 6	0.	2.

TM78 Summary:

	Hard	Soft
S/N 175		
MT600	0.	2.

DX20 Summary:

	Hard	Soft
S/N 19168		
MT502	2.	0.
S/N (N/A)		
DP400	0.	1.

RH20 Breakdown (CONI)

		PAR ERR	EXC	LWC ERR	SWC ERR	CHN ERR	RES ERR	RAE	OVR RUN
DP400	SOFT		1.						
MT502	HARD		2.			2.			
MT600	SOFT		2.						

TM78 Breakdown:

	Interrupt Code	Hard	Soft
S/N 175			
MT600	22 (WRITE)	0.	2.

DX20 Breakdown:

		Error Register						
		MPER	MPST	UPAR	DPAR	CPAR	RMR	ILR	ILF
S/N 19168									
MT502	H	2.							
S/N (N/A)									
DP400	S	1.							

Error distribution

14-Jul-82	Main-frame	Disk	Tape	Unit rec	Comm	Net-work	Soft-ware	Crash	Totals
9:00 - 10:00		1.			1.	4.			6.
10:00 - 11:00			4.			9.			13.
Totals		1.	4.		1.	13.			19.

$

Note that the "13" total network errors under the error distribution chart at the end of the Summary report confirms the 13 entries in the short and long retrieval reports. For help in interpreting the SPEAR output, refer to TOPS-20 documentation of SPEAR and BUGHLTs.

3.5 DOWNLINE LOAD

A manual load may be indicated in the following circumstances:

- the DN20 was not powered up when the KL10 was loaded
- a DN200 was not powered up when the DN20 was loaded
- the DN20 or DN200 became "UNREACHABLE" after being loaded and must be reloaded

In the case of the DN20, an automatic load should be executed by the system. If this does not occur, try to load by using the appropriate command file or by typing the necessary NCP commands on your terminal. Sections 3.1.2 and 3.1.3 contain detailed procedures for loading the DN20 by typing NCP commands and by using a command file, respectively. Section 7.8.2 documents the LOAD command. If your system includes a DN200, set the executor to be the DN20 before setting the parameters needed to load the DN200. See Section 3.2.4 for detailed procedures on loading the DN200. (The NCP command LOAD NODE is not used for the DN200.) To load a node and observe CHK11 output, see Section 3.7.

NOTE

If you have the PSI software option, you cannot downline load the DN200 through the DN20.

3.6 UPLINE DUMP

In DECnet-20, the TOPS-20 KL10-E node serves as both the EXECUTOR and the HOST for an automatic dump/reload procedure for the MCB DN20. For an automatic dump/reload of the RJE DN200, the KL10-E serves as HOST (both the system image file for the DN200 and the command file that sets the volatile data base for the DN200 startup are in the KL10-E). The MCB DN20 node acts as EXECUTOR.

NOTE

If you have the PSI software option, you cannot upline dump the DN200 through the DN20.

3.6.1 Automatic Dump of the MCB DN20

If the DN20 crashes, Network Management software in the KL10-E node senses that communication has been lost. Using dump-related parameter values for the DN20 the KL10 initiates a dump of the entire contents of the DN20's memory image. This is automatic and will normally occur whenever communication between the TOPS-20 and MCB node is lost. The dump is normally followed by a load.

When the automatic dump/load procedure executes successfully, the following message appears on your terminal if you are running the OPR program.

```
NCP>
 9:30:04        -- Network topology --

Nodes on-line:
NODNAM
```

If it appears that the automatic dump/load is not going to occur, type an NCP SHOW QUEUE command. If there is no response to SHOW QUEUE, wait a few minutes. (NCP can not respond if it is in the process of dumping or loading.) Because the system load average has a significant effect on the execution time, you will have to judge wait-time by experience.

If the DN20 is still not online, proceed as follows:

Step 1. With the TOPS-20 KL as executor, use the NCP command:

```
NCP>SET CIRCUIT DTE-0-x SERVICE ENABLED (RET)
```

where DTE-0-x identifies the SERVICE CIRCUIT over which the dump and load will be effected (usually the DTE-0-1).

This should initiate the automatic dump/load procedure. If it does, you are finished. If you do not eventually receive the node-on-line message, give the SHOW QUEUE command; wait again; continue with Step 2 if the dump/load did not occur.

Step 2. Use the NCP command file NCP.CMD:

```
NCP>take system:ncp.cmd (RET)
.
.  (Wait for OPR prompt.)
.
OPR>ent n (RET)
NCP>set cir dte-0-x state service (RET)

(This ensures that there will be no race
condition should the automatic dump/load
procedure start up.)

NCP>dump node DxxxxA (RET)

  (Wait for "Completed" response.)

NCP>Load node DxxxxA (RET)

  (Wait for "Completed" response.)

NCP> set cir dte-0-x state on (RET)

(You must take the circuit out of the
SERVICE state to permit normal use.)
```

You should receive the node-online message. If the message is not received, follow the procedures in Section 3.2.2 (Restart Procedure for NMLT20).

3.7 RECOGNIZING PROBLEMS - POTENTIAL, ACTUAL

In addition to using the SPEAR program, loopback testing and CHK11 output can be used as diagnostic tools.

3.7.1 Using Loopback Tests

DECnet-20 supports the NCP commands LOOP NODE and LOOP CIRCUIT. The LOOP LINE command is supported for the DECnet-20 PSI option. If the PSI option is not included, the LOOP LINE command can be sent by DECnet-20 to a remote node running an operating system that does support LOOP LINE.

The LOOP NODE command is the most useful tool, as well as the simplest to use. LOOP NODE exercises all DECnet layers in the DN20 and in the remote node when the loop is executed local node to remote node. Normal network activity continues with the appropriate protocols and with the normal updating of counters at each level. The LOOP CIRCUIT command, however, does not exercise all levels, uses the MOP Protocol, and does not result in the incrementing of counters.

The LOOP node command can be used to loop messages at a running remote node, at the device controller, or at a turnaround connector (loopback connector) on a physical line. The following examples demonstrate these three uses. The first example is best executed first: if it succeeds there is no need to test further.

The LOOP NODE procedure that loops at a remote node requires that the remote node, as well as the DECnet-20 node, is running Network Management V3.0 or later.

Examples:

1. LOOP NODE - Local to Remote

```
@ena (RET)
$opr (RET)
OPR>ent ncp (RET)
NCP>sh nod d2137a (RET)
NCP>
10:27:04        NCP
                Request # 263; Show Node Summary Completed

                Remote Node = 123 (D2137A)

                  State = Reachable
                  Active Links = 0
NCP>loop node d2137a count 10 length 20 (RET)
NCP>
10:27:46        NCP
                Request # 264 Accepted
10:27:53        NCP
                Request # 264; Loop Node Completed
NCP>exit
$
```

2. LOOP NODE - Using Device Controllers DMR and KDP

In this example, a circuit must be established over which the messages will travel, a loop name must be established, and the controller must be set to loopback mode.

For DMR:

```
$opr (RET)
OPR>ent ncp (RET)
NCP>set exeCUTOR noDE d2136a user operator pasSWORD d2136a (RET)
NCP>
 9:46:47        NCP
                Set Executor Complete
NCP>set cir dmr-0 sta off (RET)
NCP>
 9:47:18        NCP
                Request # 339 Accepted
NCP>
 9:47:18        NCP
                Request # 339; Set Circuit Completed
NCP>set lin dmr-0 sta off (RET)
NCP>
 9:47:50        NCP
                Request # 340 Accepted
NCP>
 9:47:51        NCP
                Request # 340; Set Line Completed
```

```
NCP>set node foo:: cir dmr-0 (RET)
NCP>
 9:48:27         NCP
                 Request # 341 Accepted
NCP>
 9:48:27         NCP
                 Request # 341; Set Node Completed
NCP>set lin dmr-0 conTROLLER loopBACK (RET)
NCP>
 9:49:07         NCP
                 Request # 342 Accepted
NCP>
 9:49:07         NCP
                 Request # 342; Set Line Completed
NCP>sho line dmr-0 charACTERISTICS (RET)
NCP>
 9:49:15         NCP
                 Request # 343 Accepted
NCP>
 9:49:15         NCP
                 Request # 343; Show Line Characteristics Completed

                 Line = DMR-0

                   Device = DMR-0
                   Receive Buffers = 10
                   Controller = Loopback
                   Duplex = Full
                   Protocol = DDCMP-DMC
                   Clock = External
                   Service Timer = 3000
                   Retransmit Timer = 3000
                   Controller Register = 160760
                   Interrupt Vector = 710
                   Interrupt Priority = 5
NCP>sho cir dmr-0 status (RET)
NCP>
 9:49:33         NCP
                 Request # 345; Show Circuit Status Completed

                 Circuit = DMR-0

                   Loopback Name = FOO
                   State = Off
NCP>set line dmr-0 sta on (RET)
NCP>
 9:49:59         NCP
                 Request # 346 Accepted
NCP>
 9:49:59         NCP
                 Request # 346; Set Line Completed
NCP>set cir dmr-0 sta on (RET)
NCP>
 9:50:24         NCP
                 Request # 347 Accepted
NCP>
 9:50:25         NCP
                 Request # 347; Set Circuit Completed
NCP>loop node foo:: count 10 length 20 (RET)
NCP>
 9:50:53         NCP
                 Request # 348 Accepted
NCP>
 9:50:57         NCP
                 Request # 348; Loop Node Completed
```

```
     ***  This controller-loopback test is completed.  ***
     ***  Procedure for restoring normal network       ***
          conditions follow.

NCP>set cir dmr-0 sta off (RET)
NCP>
 9:51:38         NCP
                 Request # 349 Accepted
NCP>
 9:51:38         NCP
                 Request # 349; Set Circuit Completed
NCP>set lin dmr-0 sta off (RET)
```

```
NCP>
 9:51:52        NCP
                Request # 350 Accepted
NCP>
 9:51:52        Request # 350; Set Line Completed
NCP>set lin dmr-0 conTROLLER normal(RET)
NCP>
 9:52:14        NCP
                Request # 351 Accepted
NCP>
 9:52:14        NCP
                Request # 351; Set Line Completed
NCP>clear node foo:: circuit(RET)
NCP>
9:52:32         NCP
                Request # 352 Accepted
NCP>
 9:52:33        NCP
                Request # 352; Clear Node Completed
NCP>set lin dmr-0 sta on(RET)
NCP>
 9:52:54        NCP
                Request # 353 Accepted
NCP>
 9:52:55        NCP
                Request # 353; Set Line Completed
NCP>set cir dmr-0 sta on(RET)
NCP>
 9:53:08        NCP
                Request # 354 Accepted
NCP>
 9:53:08
                NCP
                Request # 354; Set Circuit Completed
NCP>sho lin dmr-0 char(RET)
NCP>
 9:53:36        NCP
                Request # 355 Accepted
NCP>
 9:53:36        NCP
                Request # 355; Show Line Characteristics Completed

                Line = DMR-0

                  Device = DMR-0
                  Receive Buffers = 10
                  Controller = Normal
                  Duplex = Full
                  Protocol = DDCMP-DMC
                  Clock = External
                  Service Timer = 3000
                  Retransmit Timer = 3000
                  Controller Register = 160760
                  Interrupt Vector = 710
                  Interrupt Priority = 5
NCP>sho lin dmr-0 sta(RET)
NCP>
 9:53:45        NCP
                Request # 356 Accepted
NCP>
 9:53:45        NCP
                Request # 356; Show Line Status Completed

                Line = DMR-0

                  State = On
```

```
NCP>sho cir dmr-0 sta (RET)
NCP>
 9:53:55         NCP
                 Request # 357 Accepted
NCP>
 9:53:56         NCP
                 Request # 357; Show Circuit Status Completed

                 Circuit = DMR-0

                   State = On
                   Adjacent Node = 249
                   Block Size = 576
                   User = Node / 129 / D2136A
NCP>cle exe node (RET)
NCP>
 9:54:26         NCP
                 Clear Executor Complete
NCP>
```

For KDP:

This test differs from the test using the DMR in just one respect. For the KDP controller loopback, the CLOCK must be set to INTERNAL.

```
NCP>set exeCUTOR noDE d2136a user operator pass d2136a (RET)
NCP>
 9:56:11         NCP
                 Set Executor Complete
NCP>set cir kdp-0-0 sta off (RET)
NCP>
 9:56:34         NCP
                 Request # 358 Accepted
NCP>
 9:56:34         NCP
                 Request # 358; Set Circuit Completed
NCP>set lin kdp-0-0 sta off (RET)
NCP>
 9:56:41         NCP
                 Request # 359 Accepted
NCP>
 9:56:41         NCP
                 Request # 359; Set Line Completed
NCP>set lin kdp-0-0 conTROLLER loopBACK (RET)
NCP>
 9:56:51         NCP
                 Request # 360 Accepted
NCP>
 9:56:51         NCP
                 Request # 360; Set Line Completed
NCP>set lin kdp-0-0 cloCK intERNAL (RET)
NCP>
 9:57:03         NCP
                 Request # 361 Accepted
NCP>
 9:57:05         NCP
                 Request # 361; Set Line Completed
NCP>set node foo:: cir kdp-0-0 (RET)
NCP
 9:57:16         NCP
                 Request # 362 Accepted
NCP>
 9:57:17         NCP
                 Request # 362; Set Node Completed
NCP>sho lin kdp-0-0 char (RET)
```

```
NCP>
 9:57:27         NCP
                 Request # 363 Accepted
NCP>
 9:57:28         NCP
                 Request # 363; Show Line Characteristics Completed

                 Line = KDP-0-0

                   Device = KDP-0-0
                   Controller = Loopback
                   Duplex = Full
                   Protocol = DDCMP-DMC
                   Clock = Internal
                   Service Timer = 3000
                   Retransmit Timer = 3000
                   Controller Register = 160540
                   Unit Register = 160300
                   Interrupt Vector = 540
                   Interrupt Priority = 5
NCP>set lin kdp-0-0 sta on(RET)
NCP>
 9:57:38         NCP
                 Request # 364 Accepted
NCP>
 9:57:38         NCP
                 Request # 364; Set Line Completed
NCP>set cir kdp-0-0 sta on(RET)
NCP>
 9:57:44         NCP
                 Request # 365 Accepted
NCP>
 9:57:44         NCP
                 Request # 365; Set Circuit Completed
NCP>loop node foo:: count 10 lenGTH 20 (RET)
NCP>
 9:58:00         NCP
                 Request # 366 Accepted
NCP>
 9:58:10         NCP
                 Request # 366; Loop Node Completed

      ***  This controller-loopback test is completed.  ***
      ***  Procedure for restoring normal network       ***
           conditions follow.

NCP>set lin kdp-0-0 sta off(RET)
NCP>
 9:58:26         NCP
                 Request # 367 Accepted
NCP>
 9:58:27         NCP
                 Request # 367; Set Line Completed
NCP>set cir kdp-0-0 sta off(RET)
NCP>
 9:58:37         NCP
                 Request # 368 Accepted
NCP>
 9:58:37         NCP
                 Request # 368; Set Circuit Completed
NCP>clear no foo:: circuit(RET)
```

```
NCP>
 9:58:42         NCP
                 Request # 369 Accepted
NCP>
 9:58:43         NCP
                 Request # 369; Clear Node Completed
NCP>set lin kdp-0-0 conTROLLER norMAL(RET)
NCP>
 9:58:53         NCP
                 Request # 370 Accepted
NCP>
 9:58:54         NCP
                 Request # 370; Set Line Completed
NCP>set line kdp-0-0 cloCK extERNAL(RET)
NCP>
 9:59:07         NCP
                 Request # 371 Accepted
NCP>
 9:59:08         NCP
                 Request # 371; Set Line Completed
NCP>set lin kdp-0-0 sta on(RET)
NCP>
 9:59:17         NCP
                 Request # 372 Accepted
NCP>
 9:59:18         NCP
                 Request # 372; Set Line Completed
NCP>sho lin kdp-0-0 char(RET)
NCP>
 9:59:26         NCP
                 Request # 373 Accepted
NCP>
 9:59:26         NCP
                 Request # 373; Show Line Characteristics Completed

                 Line = KDP-0-0

                   Device = KDP-0-0
                   Controller = Normal
                   Duplex = Full
                   Protocol = DDCMP-DMC
                   Clock = External
                   Service Timer = 3000
                   Retransmit Timer = 3000
                   Controller Register = 160540
                   Unit Register = 160300
                   Interrupt Vector = 540
                   Interrupt Priority = 5
NCP>set cir kdp-0-0 sta on(RET)
NCP>
 9:59:34         NCP
                 Request # 374 Accepted
NCP>
 9:59:34         NCP
                 Request # 374; Set Circuit Completed
NCP>cle exe node(RET)
NCP>
 9:59:50         NCP
                 Clear Executor Complete
NCP>ex(RET)
$
```

3. LOOP NODE - Loopback Connector on Cable

This procedure is similar, but not identical to controller loopback. The LOOP node command is used again to do the testing. Messages are looped back at a manually installed (and later removed) loopback connector. Thus, the controller is not placed in loopback mode.

For DMR:

```
$opr (RET)
OPR>ent ncp
NCP>set exe no d2136a pass d2136a user operator (RET)
NCP>
10:44:23        NCP
                Set Executor Complete
NCP>set cir dmr-0 sta off (RET)
NCP>
10:44:45        NCP
                Request # 383 Accepted
NCP>
10:44:46        NCP
                Request # 383; Set Circuit Completed

            ***  Install loopback connector  ***

NCP>
NCP>set no foo:: cir dmr-0 (RET)
NCP>
10:47:05        NCP
                Request # 384 Accepted
NCP>
10:47:06        NCP
                Request # 384; Set Node Completed
NCP>set cir dmr-0 sta on (RET)
NCP>
10:47:24        NCP
                Request # 386 Accepted
NCP>
10:47:24        NCP
                Request # 386; Set Circuit Completed
NCP>loop no foo:: count 10 length 20 (RET)
NCP>
10:47:34        NCP
                Request # 387 Accepted
NCP>
10:48:14        NCP
                Request # 387; Loop Node Completed
NCP>
NCP>cle nod foo:: circuit (RET)
NCP>
10:49:10        NCP
                Request # 388 Accepted
NCP>
10:49:10        NCP
                Request # 388; Clear Node Completed
NCP>set cir dmr-0 sta off (RET)
NCP>
10:49:36        NCP
                Request # 390 Accepted
NCP>
10:49:37        NCP
                Request # 390; Set Circuit Completed
```

```
***  Remove loopback connector.  ***
***  Restore network connection  ***

NCP>
NCP>set cir dmr-0 sta on (RET)
NCP>
10:50:14        NCP
                Request # 392 Accepted
NCP>
10:50:15        NCP
                Request # 392; Set Circuit Completed
NCP>cle exe node (RET)
NCP>
10:50:20        NCP
                Clear Executor Complete
NCP>ex (RET)
$
```

For KDP:

Procedure is similar to that for the DMR with loopback connector. However, the KDP test requires the setting of the line parameter CLOCK to INTERNAL. Hence, both line and circuit are turned OFF before and ON after parameter changes SET LINE and SET NODE.

```
$opr
OPR>ent ncp (RET)
NCP>set exe no d2136a pass d2136a user operator
NCP>
10:51:37        NCP
                Set Executor Complete
NCP>set cir kdp-0-0 sta off (RET)
NCP>
10:51:46        NCP
                Request # 393 Accepted
NCP>
10:51:47        NCP
                Request # 393; Set Circuit Completed
NCP>set lin kdp-0-0 sta off (RET)
NCP>
10:51:54        NCP
                Request # 394 Accepted
NCP>
10:51:55        Request # 394; Set Line Completed
NCP>set lin kdp-0-0 clock internal (RET)
NCP>
10:52:32        NCP
                Request # 395 Accepted
NCP>
10:52:32        NCP
                Request # 395; Set Line Completed

***  Install loopback connector  ***

NCP>set no foo:: cir kdp-0-0 (RET)
NCP>
10:52:43        NCP
                Request # 396 Accepted
NCP>
10:52:43        NCP
                Request # 396; Set Node Completed
```

```
NCP>set lin kdp-0-0 sta on (RET)
NCP>
10:52:50        NCP
                Request # 397 Accepted
NCP>
10:52:52        NCP
                Request # 397; Set Line Completed
NCP>set cir KDP-0-0 sta on (RET)
NCP>
10:52:59        NCP
                Request # 398 Accepted
NCP>
10:53:00        NCP
                Request # 398; Set Circuit Completed
NCP>loop no foo:: count 10 length 20 (RET)
10:53:09        NCP
                Request # 399 Accepted
NCP>
10:53:14        NCP
                Request # 399; Loop Node Completed
NCP>set lin kdp-0-0 sta off (RET)
NCP>
10:53:59        NCP
                Request # 400 Accepted
NCP>
10:54:00        NCP
                Request # 400; Set Line Completed
NCP>set cir kdp-0-0 sta off (RET)
NCP>
10:54:07        NCP
                Request # 401 Accepted
NCP>
10:54:08        NCP
                Request # 401; Set CIrcuit Completed

***  Remove loopback connector.  ***
***  Restore network connection  ***

NCP>cle no foo:: circuit (RET)
NCP>
10:54:13        NCP
                Request # 402 Accepted
NCP>
10:54:13        NCP
                Request # 402; Clear Node Completed
NCP>set lin kdp-0-0 clock external (RET)
NCP>
10:54:31        NCP
                Request # 403 Accepted
NCP>
10:54:31        NCP
                Request # 403; Set Line Completed
NCP>set lin kdp-0-0 sta on (RET)
NCP>
10:54:41        NCP
                Request # 404 Accepted
NCP>
10:54:42        NCP
                Request # 404; Set Line Completed
NCP>set cir kdp-0-0 sta on (RET)
NCP>
10:54:48        NCP
                Request # 405 Accepted
```

```
NCP>
10:54:50        NCP
                Request # 405; Set Circuit Completed
NCP>cle exe node (RET)
10:55:06        NCP
                Clear Executor Complete
NCP>ex (RET)
$
```

The LOOP CIRCUIT command can also be used local node to remote node (remote node must be adjacent), using controller loopback, and with a loopback connector. The procedure would follow the appropriate node loopback procedure except that the loopback node, FOO, would not be set. (The node loopback procedure is preferred.)

4 LOOP LINE - Loopback Connector on Cable (PSI option only)

This procedure is for the KDP only.

```
NCP>set executor node mrx25 (RET)
NCP>
10:14:29        NCP
                Set Executor Complete
NCP>set line kdp-0-0 state off (RET)
NCP>
10:14:43        NCP
                Request # 20 Accepted
NCP>
10:14:44        NCP
                Request  # 20; Set Line Completed
NCP>

***  Install loopback connector  ***

NCP>set line kdp-0-0 clock internal (RET)
NCP>
10:17:40        NCP
                Request # 21 Accepted
NCP>
10:17:41        NCP
                Request  # 21; Set Line Completed
NCP>set line kdp-0-0 service enabled (RET)
NCP>
10:17:51        NCP
                Request # 22 Accepted
NCP>
10:17:52        NCP
                Request # 22; Set Line Completed
NCP>set line kdp-0-0 state on (RET)
NCP>
10:18:09        NCP
                Request # 23 Accepted
NCP>
10:18:09        NCP
                Request # 23; Set Line Completed
NCP>loop line kdp-0-0 count 120 with mixed length 10 (RET)
NCP>
10:18:30        NCP
                Request # 24 Accepted
NCP>
10:18:36        NCP
                Request # 24; Loop Line Completed
```

```
      ***  This controller-loopback test is completed.  ***
      ***  Procedure for restoring normal network       ***
      ***  conditions follow.                           ***

NCP>set line kdp-0-0 state off (RET)
NCP>
10:18:59        NCP
                Request # 25 Accepted
NCP>
10:18:59        NCP
                Request # 25; Set Line Completed
NCP>

***  Remove the loopback connector.    ***
***  Restore connection to PPSN modem  ***

NCP>set line kdp-0-0 clock external (RET)
NCP>
10:24:28        NCP
                Request # 26 Accepted
NCP>
10:24:28        NCP
                Request # 26; Set Line Completed
NCP>set line kdp-0-0 service disabled (RET)
NCP>
10:24:41        NCP
                Request # 27 Accepted
NCP>
10:24:42        NCP
                Request # 27; Set Line Completed
NCP>set line kdp-0-0 state on (RET)
NCP>
10:24:50        NCP
                Request # 28 Accepted
NCP>
10:24:51        NCP
                Request # 28; Set line Completed
```

3.7.2 Manual Procedure for Observing CHK11 Output

To manually set up for observing CHK11 output, use the following procedure:

```
@ENABLE (RET)
$OPR (RET)
OPR>ENTER NCP (RET)
NCP>LOAD NODE nodeid (RET)
NCP>PUSH (RET)
@TYPE DL1: (RET)
```

The DL1: is used in the procedure only if the DN20 and KL10 are connected by the DTE-0-1 (the normal case). If the connection is the DTE-0-2, substitute DL2: in the type command. Any use of DLn: requires that there be a DL11 line between the DN20 and the console front end.

Following the type command, CHK11 output will be displayed on the terminal.

3.7.3 Procedure for Automatic Output of CHK11

If you wish to collect CHK11 output automatically, and have it logged, then the following procedure may be used. Insert the following lines in the system's PTYCON.ATO file at the place where PTYCON subjobs are being defined. The CHK11 output will be recorded in the PTYCON log file.

```
DEFINE ^$CHK (RET)
CONN CHK (RET)
LOGIN OPERATOR FOO OPERATOR (RET)
TYPE DL1: (RET)
^X (RET)
```

CHAPTER 4

THE NETWORK MANAGEMENT PROGRAM

4.1 OPERATOR INTERFACE

The Network Management Program commands (NCP) provide the only human interface to operational controls and measurements of DECnet-20 V3.0. This manual uses the term NCP to refer to the application-specific subset of commands that are part of NMLT20. NMLT20 is used to refer to the entire program (commands, local processing routines, and dispatching of IPCF and NICE messages).

NCP commands are executed on the TOPS-20 operating system as one set of application commands available through the operator interface program OPR. In this case, the application is DECnet-20 running under TOPS-20 V5.1 or higher.

The DECnet-20 DN20 communications front end has no direct command interface. NCP commands to control the network or obtain network information are executed through the NCP on the DN20's host or the NCP on another network node.

4.2 PROCESSING OVERVIEW

During GALAXY generation, NCP tables are supplied to OPR. These tables provide the information needed by OPR to parse the NCP commands. A command that is syntactically correct is accepted by OPR, formatted as an IPCF packet, and sent to ORION. ORION dispatches the packet to the Network Management Program. The program acknowledges all commands received by returning an IPCF packet for output to the user's terminal. This basic flow has significance for operators: NCP commands cannot be executed if ORION is not running.

How and where the NCP commands will be processed and executed is determined by the Network Control Program's analysis of the command. OPR/NCP generic commands, commands that can be processed locally without access to other architectural layers, commands to be processed locally that do require interlayer communication, and commands to be processed by remote nodes, all follow different paths. You can save time in recovering from abnormal situations if you know the path taken by commands. If, for example, ORION goes down when you are in the process of entering a command to be processed locally by the NCP/NML process, there is no point in looking further for the cause of the problem. The NCP request can not be forwarded to NCP/NML until you, or another operator with that responsibility, restore ORION. The processing path for the generic commands and the locally processed commands requiring a minimum of processing will be described in the next two chapters. In general, these commands are simple in format and easy to understand.

Commands requiring communication with other architectural layers in the local system and commands to remote nodes require additional parameters because of the more complex nature of interlayer and node-to-node communication. These commands are processed by routines in the Network Management Layer (NML) of the local or remote node. Commands processed by the NML are described in Chapter 7.

Both IPCF and NICE messages (used to communicate with NML) are internal and transparent to you as an operator entering NCP commands. As an operator, your responsibility is to type the NCP commands accurately and completely. Sections 4.4 and 4.5 will help you in typing and checking the NCP commands.

Under normal conditions, transparent operations will follow successful completion of NCP commands and the command you entered will be executed. Refer to Figure 1-3 to review the processing flow described above and preview the flow to follow.

4.3 NCP FUNCTIONS

NCP commands perform three major functions:

- Controling
- Monitoring
- Planning - statistics gathered through the SET LOGGING command provide the system manager with data for evaluating future changes or additions to the network.

Control functions that can be performed with NCP include:

- Downline loading of a remote system
- Setting and changing line, node, circuit, and logging parameters
- Changing the network configuration
- Modifying message-traffic patterns
- Initiating and terminating network functions

Monitoring functions that can be performed with NCP include:

- Upline dumping of a remote system
- Displaying information about network entities (their characteristics and/or states) and counters
- Displaying the status of operations on the network, including operations in progress and operations that have failed
- Measuring network performance by displaying the contents of counters and the output of the Event Logger
- Logging of significant network events
- Performing loopback tests

The planning function is served by collecting information from the day-to-day control and monitoring functions. This information, however, must be organized and saved in useable form.

4.4 NCP FEATURES

The NCP command language has many ease-of-use features. Your familiarity with the TOPS-20 Operating System Command Language will help you with NCP, as there are many similarities between TOPS-20 commands and NCP commands.

4.4.1 Typing Commands at the Terminal

There are three ways to type commands:

- Full input mode
- Recognition mode
- Abbreviated mode

The rules for each node are the same as those for the TOPS-20 Command Language. Each of the following commands is acceptable:

Example, using full input:

```
NCP>SHOW KNOWN CIRCUITS SUMMARY (RET)
```

Example, using recognition:

```
          (ESC)  (ESC)  (ESC)        (ESC)
NCP shOW kNOWN cIRCUITS suMMARY (RET)
```

EXAMPLE, using abbreviation:

```
NCP>sh k c su (RET)
```

4.4.2 Editing Commands

The following character commands may be used to edit your input to NCP. Because of the more complex nature of the NCP commands, you may wish to add to your current use of these commands.

- DELETE

  Moving backwards from the last character typed, deletes one character for each delete key typed.

- CTRL/U

  Informs the system that you have made an error and wish to begin a new line. (Used with a lengthy command, this is often faster than other editing procedures. This command deletes all that you have entered for one NCP command.)

- CTRL/W

  Deletes back to the first punctuation character (including space).

- CTRL/F

  Used when typing file specifications. Fills in recognized fields up to the next terminator.

- CTRL/H

  Used after receiving a format-error message. Retypes the command up to the point of error.

- CTRL/R

  Used to retype the current command line. Deleted characters are omitted.

4.4.3 Using Comments

Comments may be included on the command line or on a separate line by prefixing the comment with a semicolon or exclamation point. The semicolon causes the remainder of the line to be considered as a comment; the exclamation point causes only the text up to the next exclamation point or the end of the line to be considered as comment.

4.4.4 Multiple Line Commands

A command may be continued on more than one line by typing hyphen, space, carriage return (- RET). Commands including access information for a remote node will usually require more than one line. For example:

```
NCP>SET NODE RJE1 SECONDARY LOADER - RET
    KL2102"THE-USER GUESS 341"::PS:<SUBSYS>DMCLSC.SYS
```

4.4.5 Getting Help from NCP

Helpful text is output to the user's terminal if a question mark is typed at the beginning of any field.

Note, in the example that follows, how the use of the question mark guides you through the correct format of a command. Assume you wish to check current network activity. You know that the command keyword for display is SHOW, but you are uncertain about the rest of the command. You proceed as shown below, checking for both possible choices and command completion. Command completion is implied when the response is "confirm with carriage return".

```
      (ESC)
NCP>shOW ? one of the following:
 ACTIVE      CIRCUIT     EXECUTOR     KNOWN      LINE      LOGGING
 LOOP        NODE        QUEUE
          (ESC)
NCP>shOW kNOWN ? one of the following:
 CIRCUITS    LINES       LOGGING      NODES
                (ESC)
NCP>shOW kNOWN nODES ? one of the following:
 CHARACTERISTICS     COUNTERS     STATUS     SUMMARY        TO
                           (ESC)
NCP>shOW kNOWN nODES stATUS ? confirm with carriage return
NCP>shOW kNOWN nODES stATUS (RET)
NCP>
14:52:50        NCP
                Request # 116; Show Node Complete

                Remote Node = 88 (KL2530)

                  State = Unreachable
                  Active Links = 0
.
.
.
(Output continues.)
```

4.5 NCP COMMAND OPERATION

4.5.1 Accessing NCP Commands

The Operator Command Language program, OPR, provides the operator with one command language to communicate with TOPS-20 application programs, including the Network Control Program. OPR identifies commands to the Network Control Program (NCP) when you type "NCP". You may type NCP followed by the NCP command as shown below:

```
$OPR (RET)
OPR>NCP SHOW KNOWN NODES STATUS (RET)
OPR>
 8:38:43        NCP
                Request # 273; Show Known Nodes Status Completed

                Executor Node = 113 (KL2137)

                  State = On
                  Active Links = 0

                Remote Node = 16 (EOWYN)

                  State = Unreachable
                  Active Links = 0

                Remote Node = 20 (REX)

                  State = Reachable
                  Active Links = 0

OPR>
```

or, you may type ENTER NCP as shown below:

```
$OPR
OPR>ENTER NCP (RET)
NCP>SHOW KNOWN NODES STATUS (RET)
NCP>
 8:41:27        NCP
                Request #274; Show Known Nodes Status Completed

                Executor Node = 113 (KL2137)

                  State = On
                  Active Links = 0

                Remote Node = 16 (EOWYN)

                  State = Unreachable
                  Active Links = 0

                Remote Node = 20 (REX)

                  State = Reachable
                  Active Links = 0

NCP>
```

The system response to the SHOW KNOWN NODES STATUS commands illustrated several significant particulars of output that might be overlooked. Refer back to the NCP responses to the commands and note the following:

- Each NODE described as to status is prefixed by an adjective that describes it as one of:

  ```
  Remote Node
  Executor Node
  Loop Node
  ```

- Both node number and (when known) node name appear in the output, for example:

  ```
  113 (KL2137)
   16 (EOWYN)
  ```

- The STATUS responses for an EXECUTOR node (ON/OFF) differ from the STATUS response for LOOP or REMOTE nodes (REACHABLE/UNREACHABLE).

- The EXECUTOR has access to the status of known nodes whatever their state.

4.5.2 Access Control

If a node-id in a command represents a node to be connected to, access control information may be necessary or desired. Access control information may consist of one or any combination of the keywords USER, PASSWORD, and ACCOUNT, each keyword followed by a valid value. Access control parameters follow the node-id. The maximum length of each parameter is 39 bytes. Systems other than TOPS-20 may require a format that differs from the TOPS-20 format. Each node in the network may limit the amount of access control data it will accept.

DECnet-20 limits the PASSWORD value to a length of 8 bytes and the USER and ACCOUNT values to a length of 16 bytes.

4.5.3 Terminating NCP

Typing the command EXIT returns control to the TOPS-20 executive. The system prints the TOPS-20 prompt, $, and you can type a TOPS-20 command.

4.5.4 Leaving NCP and Returning to NCP

You use the PUSH command to save your current context and enter a new TOPS-20 command level. Typing POP returns you to the previously saved NCP Command Level. Typing RETURN returns you to Operator Command Level.

```
$OPR (RET)
OPR>ENTER NCP (RET)
NCP>PUSH (RET)

 TOPS-20 Command processor 5(703)
@POP (RET)
NCP>RETURN (RET)
OPR>
```

4.5.5 Restricting NCP Commands

As implemented, NCP requires only OPERATOR capability. Determining the commands that should be further restricted in use is the responsibility of the system manager at each DECnet site. Users (with OPERATOR privileges) who lack the knowledge and experience to use critical commands should be advised of commands with the potential for disrupting the network. An operator using NCP commands should know the restrictions set by the established procedures at his site. All problems related to individual or group access to specific NCP commands should be settled by the system manager, or someone delegated by the system manager.

4.5.6 Command Input

NCP input takes the form of arguments delimited by one or more blanks or tabs.

If a carriage return is typed following the NCP prompt (without typing a command), the NCP prompt is repeated.

Multiple outstanding commands are permitted.

4.5.7 Command Output Response

The most general output content and format in response to NCP commands includes the following:

```
time        NCP
            Request #nn:  Command input status

            Entity type = Entity Identification
            Requested Information or "No Information"
```

where:

time is in hours, minutes, seconds (hh:mm:ss)

Command Input is a brief identification of the request you typed

Status is one of:

Complete - information follows
Accepted - information will follow after brief delay
Failed - reason for failure follows and, optionally, further detail is given.

Entity-type is one of node, circuit, line, or logging (singular or plural form). If the entity-type is node, node must be preceded by Executor, Remote, or Loop. All commands to change either the volatile or permanent data base of a remote node should include the required USER and PASSWORD parameters.

Requested Information will differ according to the action requested, the entity specified, and the arguments given.

The output "No Information" means that no parameters have been set. No failure of the command is implied.

Example #1: Request for characteristics of the TOPS-20 node:

```
$opr (RET)
        (ESC)    (ESC)                  (ESC)
OPR>disaBLE outPUT-DISPLAY (of) all-MESSAGES(RET)
OPR>
 9:27:41        --OUTPUT DISPLAY for OPR Modified--
OPR>enter ncp(RET)
NCP>show node kl2136 charACTERISTICS(RET)
NCP>
 9:28:25        NCP
                Request # 255; Show Node Characteristics
                Completed

                Executor Node = 124 (KL2136)

                  Identification = DECnet-20 V3.0.0
                  Management Version = 3.0.0
                  Loop Count = 1
                  Loop Length = 127
                  Loop With = Mixed
                  NSP Version = 3.2.0
                  Inactivity Timer = 20
                  Delay Weight = 32
                  Delay Factor = 63
                  Retransmit Factor = 5
                  Type = Phase II
NCP>
```

Example #2. Request for status of known circuits (Executor is KL2136):

```
NCP>sh kn cir staTUS (RET)
NCP>
 9:29:16        NCP
                Request # 256; Show Known Circuits Status
                Completed

                Circuit = DTE-0-1

                  No Information


                Circuit = DTE-0-2

                  No Information


                Circuit = DTE-0-3

                  No Information

NCP>
```

Example #3. Request for status of known circuits
(Executor is D2136A)

```
NCP>set executor node d2136a(RET)
NCP>
 9:30:07          NCP
                  Set Executor Complete
NCP>show kn cir staTUS(RET)
NCP>
 9:30:51          NCP
                  Request # 257 Accepted
NCP>
 9:30:53          NCP
                  Request # 257; Shown Known Circuits Status
                  Completed

                  Circuit = DTE-0-1

                    State = On
                    Adjacent Node = 124 (KL2136)
                    Block Size = 576
                    User = Node / 129 / D2136A

                  Circuit = DMR-0

                    State = On
                    Substate = Starting
                    Block Size = 576
                    User = Node / 129 / D2136A

                  Circuit = KDP-0-0

                    State = On
                    Adjacent Node = 123
                    Block Size = 576
                    User = Node / 129 / D2136A

                  Circuit = KDP-0-1

                    State = On
                    Substate = Synchronizing
                    Block Size = 576
                    User = Node / 129 / D2136A

                  Circuit = KDP-0-2

                    State = On
                    Substate = Synchronizing
                    Block Size = 576
                    User = Node / 129 / D2136A

                  Circuit = KDP-0-3

                    State = On
                    Substate = Synchronizing
                    Block Size = 576
                    User = Node / 129 / D2136A

NCP>exit(RET)
```

Appendix C lists and explains, in text or by example, all messages output to the user's terminal, including the CTY or operator's console.

The general output format given in Appendix C does not apply to NCP commands processed by OPR or processed by the local NCP program. All commands translated into NICE messages will follow the content of the format example for DECnet-20 V3.0. The actual order of items and their physical placement may differ for other operating systems.

4.5.8 Specifying the Executor

DECnet-20 V3.0 is designed so that central, fully distributed, or partially distributed control and management are possible. An NCP command does not have to be executed at the node where it is typed. Therefore, the operator must determine and designate the node that is to process and execute each command. Specifying the executor may be done by implication (allowing the default to be effective), by a SET or DEFINE command, or by using the TELL prefix. For a detailed description of how to specify the executor, see the named commands in Chapters 6 and 7.

The operator must be aware of the characteristics (and status) of relevant nodes in the network to effectively determine the ability of a node to act as executor for the command node. Both nodes must be running Network Management V3.0.

(Refer to Chapter 2, the sections entitled COMPATIBILITY and the section entitled RESTRICTIONS for more detailed information.)

4.5.9 Specifying Files in NCP Commands

File specifications used as arguments in NCP commands to be executed by DECnet-20 nodes follow TOPS-20 file specification conventions.

The full file specification for DECnet-20 is:

```
node::dev:<dir>filename.filetype.gen
```

where node:: is nodename"user password"

If the file referenced resides on the executor node, then node:: can be omitted. If the file resides on a node other than the executor, then node:: must be specified.

Example:

```
KL2102"DECNET KL2102"::PS:<SUBSYS>SECDMC.SYS
```

System-wide logical names may be used in file specifications. Load files, for example, generally reside on a device and in a directory defined with the system-wide logical name SYS:. In such case, a file could be specified as SYS:DTEMPS.SYS as well as PS:<SUBSYS>DTEMPS.SYS if the file resided on a KL10 acting as executor.

CHAPTER 5

NCP COMMANDS PROCESSED BY OPR

5.1 OPR/NCP COMMANDS

Some of the commands available in the NCP application subset are provided by OPR for general use by operators, that is, for activities other than network activities. These commands require no action by NCP. They are processed directly by OPR. See Figure 1-3 if you wish to review the flow of these commands. The commands are:

ENTER (command subset) application-name

EXIT (to monitor level)

PUSH (to EXEC level)

RETURN (to Operator Command Level)

TAKE (commands from) fileid /DISPLAY /NODISPLAY

WAIT (for) seconds

5.2 EXAMPLES OF USE OF OPR/NCP COMMANDS

Because these commands are among those used frequently by operators, the description of the input/output examples is specific to network functions. Only the OPR/NCP commands are shown in red so that you may identify them readily.

Assume that you are the user CIRINO and that you are logged in on a DECSYSTEM-2060 host node. First, you enable capabilities and refuse system messages (using the TOPS-20 Command Language). Next, you enter OPR and use the OPR command DISABLE to avoid output to your terminal of OPR messages not relevant to network activity. These commands are never necessary, but they are helpful if you are using your terminal under timesharing and wish to avoid interruptions.

Next, you enter NCP and check the status of the local node. The system reports that "State is On". So far, you have the following input/output:

```
@ena(RET)

   (ESC)    (ESC)
$refUSE syTEM-MESSAGES(RET)
$opr
      (ESC)     (ESC)                  (ESC)
OPR>disABLE outPUT-DISPLAY (of) all-MESSAGES(RET)

 9:37:39        --Output display for OPR modified--
OPR>enter ncp(RET)

     (ESC) (ESC)      (ESC)
NCP>shOW exECUTOR stATUS(RET)
NCP>
 5:29:14        NCP
                Request # 27 Accepted
NCP>
 5:29:15        NCP
                Request # 27 Show Executor Node Status Completed

                Executor Node = 120 (KL2102)

                  State = On
                  Active Links = 1
NCP>
```

You next type the PUSH command to get a new EXEC. You type the INFORMATION DECNET command to check for currently accessible nodes. You have added:

```
NCP>push(RET)

 TOPS-20 Command processor 5(712)
@i dec(RET)
 Local DECNET node: KL2137
 Accessible DECNET nodes are:   ABACUS  ABLE    ADAM    ADE     AJAX
 ALGOL  ALIEN   ALPHA   ALPINE  ANIMAL  APOLLO  ARGON   ARK     ARUBA
 ARWEN  ASDHUB  ATHENA  AURORA  BABEL   BANZAI  BAXTER  BERGIL  BETA
 BISON  BUZARD  CACHE   CADVAX  CAR     CASTOR  CDR     CHAOS   CHIPS
                                .
                                .
                                .
 WOMBAT WOOKIE  WORDS   XENON   YODA    ZAPHOD  ZEKE    ZEPHYR  ZEUS
 ZIP    ZONKER
@
```

You type the POP command to return to NCP, then type the RETURN command to reenter OPR. This order of commands (following) assumes you wish to check the number of operators with jobs on the host node. Satisfied that there can be minimal contention, you reenter NCP. Assume that you now wish to check the DTE counters. You know that the executor must be set to the communications front end (named D2102A in this case) to get this information. Using the SHOW EXECUTOR command, you find that the host node is the executor. You set the EXECUTOR to be the D2102A. You are called away from the terminal and repeat the SHOW EXECUTOR command when you return. Whenever you interrupt a network session, checking the EXECUTOR is desirable. Do not assume that there has been no user-activity at your terminal during your absence. Also, the EXECUTOR you previously set may have gone off-line. You have added:

```
@pop(RET)
NCP>return(RET)

      (ESC)   (ESC)
OPR>shoW operATORS(RET)
OPR>
15:19:07        -- Operators --
    Node      Type     Terminal  Job            User
-----------  ------  --------  --- ----------------------------
KL2102       system      240      7 OPERATOR
KL2102       system        2     46 OPERATOR
KL2102       system       56     65 CIRINO
OPR>enter ncp(RET)
NCP>sh ex(RET)
NCP>
15:19:49          NCP
                  Request # 38; Show Executor Node Summary Completed

                  Executor Node = 120 (KL2102)

                    Identification = DECnet-20 V3.0.0
                    State = On
                    Active Links = 0

          (ESC)     (ESC)
NCP>set exECUTOR noDE d2102a(RET)
NCP
15:41:33          NCP
                  Set Executor Complete
NCP>sh ex(RET)
NCP
15:41:38          NCP
                  Request # 39 Accepted
NCP>
15:41:39          NCP
                  Request # 39; Show Executor Node Summary Completed

                  Executor Node = 121 (D2102A)

                    Identification = DECnet-20 V3.0.0
                    State = On
                    Active Links = 1
```

You have the SHOW COUNTERS command in a file named CNTDTE.CMD and use the TAKE command for indirect input. The /DISPLAY switch displays the command entered. At this point, you continue with whatever network activities you had planned. Finally, you use the EXIT command to return to the EXEC. Your last input/output was:

```
NCP>take cntdte.cmd /display(RET)

NCP>cl ex no(RET)
NCP>
15:42:10        NCP
                Clear Executor Complete

NCP>TELL D2102A SHOW CIRCUIT DTE-0-1 COUNTERS(RET)

15:42:15        NCP
                Request # 43 Accepted

15:42:16        NCP
                Request # 43; Show Circuit Counters Completed

                Circuit = DTE-0-1

                  4059          Terminating Packets Received
                  4669          Originating Packets Sent
                  63            Terminating Congestion Loss
                  39842         Transit Packets Received
                  47361         Transit Packets Sent
                  3             Transit Congestion Loss
                  0             Circuit Downs
                  0             Initialization Failures
                  1532485       Bytes Received
                  2522599       Bytes Sent
                  43993         Data Blocks Received
                  52032         Data Blocks Sent
NCP>EXIT(RET)
$
```

You can now log off or continue with other activities using the EXEC. You have in this short terminal session used each of the OPR/NCP commands except WAIT. You have also moved efficiently among three command sets.

The format of the WAIT command is:

WAIT (for) seconds

where seconds is a number from 1 to 60.

The WAIT command is used in command files and batch control files; it attempts to ensure that the response to one command will be output before the next command is processed. If the WAIT command is not inserted following an NCP command that might take a few seconds to process, it is possible for the command sequence to continue or complete before the response is output. With experience you will learn to approximate the time required for responses in your particular environment. The argument cannot exceed 60 seconds, but the time can be increased by repeating the WAIT command. The WAIT commands are underlined so that you can compare them more easily.

The load average of the system used for the three examples was unusually high.

Command file, first attempt:

```
$opr (RET)
OPR>disaBLE outPUT-DISPLAY (of) all-MESSAGES (RET)
OPR>
16:09:48        --Output display for OPR modified--
OPR>enter ncp (RET)
NCP>take tstwt /Display (RET)

NCP>loop node k12137 coun: 100 length 100 (RET)
$
```

Command file, second attempt:

```
$opr (RET)
OPR>disaBLE outPUT-DISPLAY (of) all-MESSAGES
OPR>
16:18:33        --Output display for OPR modified--
OPR>enter ncp (RET)
NCP>take tstwt2 /display (RET)

NCP>loop node k12137 count 100 length 100 (RET)

NCP>wait 30
$
```

Command file, third attempt:

```
$opr
OPR>disaBLE outPUT-DISPLAY (of) all-MESSAGES (RET)
OPR>enter ncp

16:31:41        --Output display for OPR modified--
NCP>take tstwt3 /display (RET)

NCP>loop node k12137 count 100 length 100

NCP>wait 50 (RET)

16:32:32        NCP
                Request # 259 Accepted

16:32:47        NCP
                Request # 259; Loop Node Completed
$
```

CHAPTER 6

NCP COMMANDS PROCESSED BY THE NETWORK CONTROL PROGRAM

Certain commands are processed by NCP directly by means of an internal data path between NCP and the network data base in the host node (the DECSYSTEM 2040S/2060 KL10 node. Refer back to Figure 1-3 if you wish to review the flow of these commands. Commands processed internally by NCP/NML include:

```
SET EXECUTOR NODE nodeid

DEFINE EXECUTOR NODE nodeid

CLEAR EXECUTOR NODE

PURGE EXECUTOR NODE

SHOW QUEUE

CANCEL QUEUE REQUEST requestid

HELP

              [USER userid]
TELL nodeid   [PASSWORD password] followed by NML-command-line
              [ACCOUNT account]
```

(Everything in front of the NML command is a prefix, not a command. A command to a remote node follows. It is the prefix that is processed by the local node; the remote node processes the command.)

Format, function, arguments (keywords and variable data), examples, and remarks if called for, follow for each of the NCP commands (and prefix) processed locally by NCP.

6.1 SET EXECUTOR

SET EXECUTOR NODE nodeid [USER userid]-
[PASSWORD password] [ACCOUNT account] RET

Function:

The SET EXECUTOR command acts immediately to establish the node specified as the processing node for subsequent commands. The value of nodeid becomes part of the temporary data base. The permanent data base is not affected. The node specified in the command will continue to process commands until the executor is changed with another SET EXECUTOR command or until cleared with the CLEAR EXECUTOR command. If the system goes down, this value is lost. This command can not be executed remotely.

Arguments:

nodeid	This may be the decimal number representing the address of the node, or the node name. If name is used, it must be a name established during network configuration.
userid password account	These arguments provide access control information and may or may not be required. Required format for these values is set by the system identified in nodeid. TOPS-20 systems accept any ASCII string of 8 bytes or less for password and 16 bytes or less for userid and account.

Examples:

Example 1. To display characteristics for the DN20 node:

```
NCP>set ex no d2136a(RET)
NCP>
14:17:17        NCP
                Set Executor Complete
```

Example 2. To change a parameter value:

```
NCP>set ex no d2137a user operator password secret(RET)
NCP>
12:28:32        NCP
                Set Executor Complete
NCP>
```

Remarks:

Note that commands processed locally receive an immediate response of Complete or Failed and are not assigned a request number.

If you specified a value for USER and PASSWORD during the generation procedures for the DN20 MCB front end, these values must be specified in the SET EXECUTOR command if you intend to execute a command that alters the DN20's data base. (ACCOUNT is an optional parameter, but if present, it must match the value set.) If you omit these values subsequent commands that set or change parameter values will fail. If you have not set USER and PASSWORD values, any user at a remote node can manipulate the network by setting the executor to your DN20 MCB node.

USER and PASSWORD values are not required with the SET EXECUTOR command if subsequent commands are to be SHOW commands.

6.2 DEFINE EXECUTOR

DEFINE EXECUTOR NODE nodeid [USER userid]-
[PASSWORD password] [ACCOUNT account]RET

Function:

The DEFINE EXECUTOR command establishes the specified node as the processing node in the permanent data base. This is the default value for the executor. When the Network Control Program is started, the default executor is the node on which the control program is running unless the executor node was previously defined with the DEFINE EXECUTOR command. The values in the permanent data base remain from one initialization to the next. The executor node in the permanent data base can be changed only by the DEFINE EXECUTOR or the PURGE EXECUTOR command.

Arguments:

The arguments are the same as those defined for the SET EXECUTOR command.

Remarks:

Not implemented for DECnet-20.

6.3 CLEAR EXECUTOR

CLEAR EXECUTOR NODERET

Function:

The CLEAR EXECUTOR command accomplishes the reverse of the SET EXECUTOR command. It removes the value that was previously entered in the volatile data base as representing the executor. After this command, the executor defaults to the node on which NCP is running unless a DEFINE EXECUTOR command has previously been given. For operating systems that support a permanent data base, the value in the permanent data base is always the default taken when no value is given. For DECnet-20, the executor will default to the TOPS-20 KL10 node when this command is executed. The CLEAR EXECUTOR NODE command can not be executed remotely.

Arguments:

The keyword NODE acts as the only argument.

Example:

```
NCP>set ex no d2136a user operator password d21362 (RET)
NCP>
 9:30:41        NCP
                Set Executor Complete
NCP>sh ex (RET)
NCP>
 9:30:53        NCP
                Request # 244 Accepted
NCP>
 9:30:54        NCP
                Request   244; Show Executor Node Summary Completed

                Executor Node = 129 (D2136A)

                  Identification = DECnet-20 V3.0.0
                  State = On
                  Active Links = 1

NCP>cl ex no (RET)
NCP>
 9:31:27        NCP
                Clear Executor Complete
NCP>sh ex (RET)
NCP>
 9:31:34        NCP
                Request # 245; Show Executor Node Summary Completed

                Executor Node = 124 (KL2136)

                  Identification = DECnet-20 V3.0.0
                  State = On
                  Active Links = 0

NCP>
```

Remarks:

This command is the simplest way to change the EXECUTOR from the DN20 MCB front end to the TOPS-20 KL10.

6.4 PURGE EXECUTOR

PURGE EXECUTOR NODE (RET)

Function:

The PURGE EXECUTOR command accomplishes the reverse of the DEFINE EXECUTOR command. It removes the value that was previously in the permanent data base as representing the executor. The default executor is then the node where the Network Control Program is running.

Remarks:

Not implemented for DECnet-20.

6.5 SHOW QUEUE

SHOW QUEUE RET

Function:

You use the SHOW QUEUE command to display the NCP-request queue. This command results in either a list of NCP-requests waiting to be processed, or the message, [The queue is empty]. Request numbers are assigned in order as commands are received. Therefore, request numbers are in ascending order (except when numbering begins over with # 1). The numbers are consecutive only when you are the only operator using NCP. To locate the command you entered, refer back to the command. The request number is given immediately preceding "Accepted" and "Completed".

Arguments:

None. (NCP-requests is implied.)

Examples:

```
NCP>sh q RET
NCP>
 9:38:22      NCP
              [The queue is empty]

NCP>

        *****

NCP>sh q RET
NCP>
 9:41:31      NCP
              Request # 249 Active, Remote, Executor = 129 (D2136A)
```

Remarks:

The SHOW QUEUE command is potentially helpful when you receive no response from any command within a few seconds. If there is no response from SHOW QUEUE, the two most likely reasons are: the NMLT20 program is busy loading an adjacent node, or the NMLT20 program is hung. Wait a few minutes, and if there is still no response, restart NMLT20. (See Section 3.2.2 for Restart.)

6.6 CANCEL QUEUE REQUEST

CANCEL QUEUE REQUEST requestid RET

Function:

You use the CANCEL command to remove a request from the queue before processing begins. You can check the status of the request with the SHOW QUEUE command.

Argument:

REQUEST requestid	requestid is the number assigned to a command request by NML; this number is output immediately following all commands to be translated into NICE messages (commands in Chapter 7).

Remarks:

You can not cancel commands processed locally by NCP (commands in this chapter). They are executed immediately and do not appear on the queue. Also, requests that are noted as Active cannot be cancelled.

6.7 TELL PREFIX

TELL nodeid	**[USER userid] [PASSWORD password] [ACCOUNT account]**	**command line**
NCP command prefix	Access control keywords and parameters	NCP command to be executed by node named in nodeid

Function:

You use the TELL nodeid prefix to designate the node that is to issue the command. The prefix is effective only for the single command that immediately follows. (See Section 7.3.2 for help on nodeid if you wish to try these commands.)

The receiving node has the option of requiring any combination of the three access keywords and parameters, or none of them. The receiving node may refuse to execute the command even when access parameters meet requirements. (See Remarks.)

Arguments:

[USER userid] [PASSWORD password] [ACCOUNT account]	You must format access control parameters according to the conventions of the operating system of the node specified in nodeid.

TOPS-20 format:

All three fields may contain any combination of ASCII characters not exceeding 39

Other DIGITAL operating systems:

See documentation for that system.

Examples:

TOPS-20 Access Control Parameters:

```
NCP>tell d2102a user operator password d2102a -
    zero circuit dte-0-1 counters (RET)
```

* * * *

TOPS-20 KL10 to Front End, No Access Control:

```
NCP>tell d2137a sh kn cir sum (RET)
NCP>
 5:45:29   NCP
           Request # 223 Accepted
NCP>
 5:45:31   NCP
           Request # 223; Show Known Circuits Summary Completed

           Circuit = DTE-0-1

             State = On
             Adjacent Node = 113 (KL2137)
             User = Node / 123 / D2137A

           Circuit = DMC-0

             State = On
             Adjacent Node = 116 (KL2116)
             User = Node / 123 / D2137A

           Circuit = DMR-0

             State = On
             Substate = Synchronizing
             User = Node / 123 / D2137A

           Circuit = DMR-1

             State = Off
             User = Node / 123 / D2137A

           Circuit = DMR-2

             State = Off
             User = Node / 123 / D2137A

           Circuit = KDP-0-0

             State = On
             Substate = Synchronizing
             User = Node / 123 / D2137A

           Circuit = KDP-0-1

             State = On
             Substate = Synchronizing
             User = Node / 123 / D2137A
```

Remarks:

Using the TELL NODE prefix is more efficient than a combination of SET EXECUTOR and CLEAR EXECUTOR commands when only one command is to be given to a node. Assume you have set the executor to node CAR for the purpose of monitoring network traffic there for an hour or more. The system manager interrupts with a request to check the status of lines at node BAR. You can use the command TELL NODE BAR SHOW KNOWN LINE STATUS. The TELL prefix command format overrides the previous setting of the executor to node CAR, displays the status of lines at node BAR, and allows you to continue with commands to be executed at node CAR.

Before using this, or any other NCP command requiring that access control parameters be sent to another node, you should have established both the values and the format that will be acceptable to the remote node.

Assuming the access control parameters are correct, the receiving node may still reject the connection for a number of reasons (for example, the limit of logical links has been reached at the receiving node, the state of a required object does not allow processing of the command, or the parameters given by the sender cannot be verified). The receiving node will send the source node a reject message with a reason for the rejected connection. Depending on the reason given, you must determine the likelihood of obtaining a successful connection at a later time.

6.8 USING THE HELP COMMAND

The HELP command provides a convenient check of the function of all NCP commands. Typing HELP lists the commands; typing HELP keyword, where keyword is an NCP command keyword, lists all possible combinations of command keyword and entity keyword.

Examples:

```
NCP>help return(RET)

        RETURN is used to return to the OPR command level.

           ****

NCP>help clear(RET)

     CLEAR is used to clear parameters of an entity from the
     volatile data base. Additional information available.

          CLEAR CIRCUIT
          CLEAR EXECUTOR NODE
          CLEAR LINE
          CLEAR LOGGING
          CLEAR NODE

                    ****

NCP>help clear logging(RET)

     CLEAR LOGGING is used to control the logging of events, both
     the events that will be logged and where they will be
     logged.
```

CHAPTER 7

NCP COMMANDS PROCESSED BY THE NETWORK MANAGEMENT LAYER

7.1 PROCESSING OVERVIEW

All NCP commands not described in Chapters 5 or 6 are processed by routines in the Network Management Layer (NML) and involve transparent conversion of NCP commands to NICE protocol messages. Commands to be executed by the local node are processed by the local node's NML; commands to be executed by a remote node are processed by the remote node's NML.

The NML itself contains most of the logic and data bases needed to process the NCP commands. When needed information resides in other architectural layers, NML uses its interface to the appropriate layer to read or set parameters. Both system managers and operators will find that a knowledge of the processing flow of NCP commands will increase their efficiency in using and maintaining the network.

All communication with the NML that involves an exchange of data between local and remote nodes is by means of the Network Information Communication Exchange (NICE) Protocol. (This includes communication between the 2040S/2060 main processor and the DN20.) As an operator, you are not required to know the details of the NICE protocol. (It may be necessary for a programmer to have a working knowledge of NICE.) The Network Control Program understands NICE, and given a valid NCP command, will reformat the command-content (now in an IPCF message) into a NICE message for the intended NML. There is one NICE message for each NCP command.

When the intended receiver is a remote NML, all DECnet systems use the same method of sending the NICE message. The message is sent over a logical link provided by NSP and Session Control.

When the NICE process is to be processed by the local NML, the NML uses a system-specific call (a monitor call in DECnet-20) to the appropriate routine for processing.

When an NML receives a NICE message from a remote NML, it always sends an acknowledgement of receipt to the sender. Following the receipt of a command, the NML of the node that acted as EXECUTOR sends a response to the NML of the command node. Here NCP/NML formats this response into an IPCF packet; the packet is forwarded to ORION; ORION dispatches the response to OPR; the response is displayed on the operator's terminal.

You have now followed the paths taken by various NCP commands. You may wish to refer to Figure 1-3 for a visual review. Note that your participation was restricted to entering the NCP command at the terminal and receiving an acknowledgement. Later, normally within a few seconds, you will receive a response that informs you of the disposition of the command. When this response indicates no action was taken, the system will, to the extent possible, give the reason.

Both positive and negative responses on the disposition of the command are included in the examples that follow the descriptions of the individual commands beginning with Section 7.4.

7.2 DECNET-20 IMPLEMENTATION AND EXTENSIONS

DECnet-20 V3.0 implements a subset of the NCP commands described by DNA for Version 3.0 of Network Management. Commands specific to X.25 are implemented if you have the PSI software option (see Chapter 8). Commands specific to DMP multipoint are not implemented.

In addition, two extensions have been made to accommodate the distinctive hardware characteristics of DECSYSTEM-20. The extensions to the DNA specification are:

- Remote permanent parameters.

  The communications front end (DN20) has no modifiable permanent memory. Certain MCB (DN20) data base items are set during network generation and are stored in the front-end system image. The NCP commands PURGE and DEFINE will not affect these values after network generation.

- Check, then boot DN20.

  This facility allows the DN20 communications software to continue running while a TOPS-20 system is being reloaded. The TOPS-20 operating system periodically checks the DN20 integrity, and reloads it if necessary.

7.3 NML CONCEPTS AND ORGANIZATION OF PARAMETERS

Some of the NCP commands that are processed by NML have as many as twenty possible parameters. Most of these parameters appear in several commands. The most commonly repeated keywords that accept other keywords as arguments (rather than specific variables like name or line identification) are given in this section. You will save yourself time if you make the required associations at this time. Until you become familiar with the command/entity/keyword/variable combinations, the following descriptions of concepts, variables, and groups of parameters common to specific entities, should help you to choose the correct parameter and interpret the output when you use the "?" feature.

7.3.1 Entities

The major network management keywords are those that immediately follow the command action word (such as SET or LOAD). These keywords specify the element on which the command is to act. All other keywords act as "labels" for variables to be typed or as requests to choose one of a set group of responses.

Most entities may be either singular or plural. Only one entity is permitted in one command. The following list includes all entities in commands processed by the NML for DECnet-20 V3.0:

Singular	Plural
CIRCUIT	ACTIVE CIRCUITS KNOWN CIRCUITS
LINE	ACTIVE LINES KNOWN LINES
LOGGING	ACTIVE LOGGING KNOWN LOGGING
NODE	ACTIVE NODES KNOWN NODES LOOP NODES

7.3.2 Entity Identification

The plural form of an entity defines and restricts itself. (Active lines, for example, are known lines in the ON or SERVICE state.) However, each singular form of an entity must be followed by an identification in a system-specific format to indicate one specific line, circuit, or node, for example. The identification is also referred to as the name of the entity.

The format and an example of the identification of each entity type follow:

lineid device-controller-unit

KDP	1	0
DMC	2	

Lineid would appear in a command as LINE KDP-1-0 for a multiline interface and DMC-2 for a single line interface.

nodeid name or decimal number

Nodeid would appear in a command as NODE KL2102 or NODE 120

cktid device-controller-unit

KDP	2	0
DMC	2	

or

device-controller-unit.tributarynumber

KDP	2	0	1

- for multidrop lines only -

Cktid would appear in a command as CIRCUIT interface and as DMC-2 for a single line interface. For a multidrop KDP interface, cktid would appear as CIRCUIT KDP-2-0.1.

Strictly speaking, the LOGGING entity is followed by an identification of the sinktype, rather than an identification of the logging entity itself. However, sinktype must always follow LOGGING in the command in the same manner that nodeid follows NODE and circuitid follows CIRCUIT. Sinktype identifies the place where a copy of the event is recorded.

Sinktype must be one of the following:

- CONSOLE
- FILE
- MONITOR

The SINK NODE keyword takes as an argument the node name or node address where events are logged. See the appropriate documentation to direct logging commands to operating systems other than TOPS-20.

DECnet-20 logging parameters are not controllable by the OPERATOR or System Manager. Significant events specific to the TOPS-20 (KL) node and the MCB (DN20) node are automatically logged to the TOPS-20 ERROR.SYS file. The SHOW LOGGING NCP commands take the argument FILE, as shown:

```
NCP>SHOW LOGGING FILE SUMMARY(RET)
NCP>
 9:01:33        NCP
                Request # 262; Show Logging Summary Completed

                Logging = File

                  State = On
                  Name = ERROR.SYS

NCP>
```

The EXECUTOR must be the KL node. When the EXECUTOR is the DN20 node, the response is FAILED.

Detailed reports on logged events are available by running the SPEAR program. See Section 3.4.2.

7.3.3 **The Node Entity**

The node is the central entity of the network. All network control and monitoring is based on the concept of the node. From the standpoint of network management, nodes are either local or remote. The local node is the node where you log in. Node functions and parameters are included in the Network Management Layer, the Session Control Layer, the Network Services Layer, and the Transport Layer of DNA. Associated with nodes are several keywords that require you to choose one of several other keywords to qualify or select specific parameters for the first keyword.

TYPE

ROUTING	Full Phase III routing node.
NONROUTING	Phase III node without routing capability.
PHASE II	Phase II node.

STATE

For EXECUTOR node:

ON	Allows logical links.
OFF	Allows no new links, terminates existing links, and stops routing.
SHUT	Allows no new links, does not destroy existing links, and goes to OFF when all logical links are gone.
RESTRICTED	Allows no new incoming logical links from remote nodes.

For Remote nodes:

REACHABLE	A remote node is reachable if the Transport module of the EXECUTOR node believes it has a usable path to the remote node.
UNREACHABLE	A remote node is UNREACHABLE when the path length from the EXECUTOR to the remote node is longer than the maximum hops in the network.

CPU

PDP-8 PDP-11 DECSYSTEM-1020 VAX-11	Identifies the adjacent target node's CPU type for downline loading.

SOFTWARE TYPE

SECONDARY LOADER TERTIARY LOADER SYSTEM	Identifies initial target node program type for downline loading the adjacent node.

All other keyword responses are specific variables (one per keyword), such as number, seconds, fileid and the like.

7.3.4 The Circuit Entity

Circuits provide logical point-to-point connections. Circuit functions and parameters come from the Network Management, Transport, and Data Link layers. Transport is interested in the circuit - not the line - and addresses its routing algorithms to the circuit. A circuit that is a point-to-point circuit will correspond to the line that is the medium of the circuit on a one-to-one basis. Multipoint lines present a different picture, and do not appear the same from the master and slave ends. From the master end, there is one DDCMP control line and each multipoint tributary is a separate circuit. The line is of protocol type DDCMP-CONTROL. Each circuit is also identified as protocol type DDCMP-CONTROL. At the slave or tributary end, there is a one-to-one correspondence of circuit and a DDCMP tributary line.

All circuits have the following choices for the keywords listed:

SERVICE

ENABLED	Indicates that the SERVICE functions, loading, dumping, and line loopback testing, are allowed.
DISABLED	SERVICE state disallowed and SERVICE functions may not be performed.

STATE

CLEARED	The link can not be used, but space is reserved for it. Link data bases and parameters are not present.
OFF	The circuit is not to be used by any network-related software. It is functionally nonexistent. Link data bases and parameters are present.
ON	The circuit is available for normal use by its owner, with the exception of overrides for service.
SERVICE	Applies only to the volatile data base (SET command). The circuit is available for service functions, but is not available for use by the owner (Transport).

7.3.5 The Line Entity

Lines provide physical point-to-point connections. Functions and parameters for the line entity come from the Network Management, Data Link and Physical Link layers.

Lines have the following choices for the keywords listed:

CLOCK

INTERNAL	Clock is supplied by a device instead of a modem. Usually used for external cable loopback.
EXTERNAL	For normal clock operating mode (clock signal normally supplied by the modem).

CONTROLLER

LOOPBACK	Places the device to loopback internally, thus not requiring a loopback connector to turn data around.
NORMAL	Can be used for modem or cable loopback.

NOTE

See Table 7-1 at end of this Section for relationships between the above keywords and the loopback mode of the line.

DUPLEX

FULL	Full duplex mode of the line device.
HALF	Half duplex mode of the line device.

PROTOCOL

DDCMP POINT	This is the line protocol for a point-to-point DDCMP connection.
DDCMP CONTROL	This is the line protocol for the control station (master) end of a DDCMP multipoint group.
DDCMP TRIBUTARY	This is the line protocol for the tributary (slave) end of a DDCMP multipoint group.
DDCMP DMC	This line is in DMC emulation mode.

SERVICE

ENABLED	Indicates that the SERVICE functions, loading, dumping, and line loopback testing, are allowed.
DISABLED	SERVICE state disallowed and SERVICE functions may not be performed.

STATE

CLEARED	The line can not be used, but space is reserved for it. Line data bases and parameters are not present.

OFF — The line is not to be used by any network-related software. It is functionally nonexistent. Line data bases and parameters are present.

ON — The line is available for normal use by its owner, with the exception of overrides for service.

SERVICE — Applies only to the volatile data base (SET command). The line is available for service functions but is not available for use by the owner (the EXECUTOR node implying Transport).

Table 7-1
Relationship of Clock and Controller Parameters to Mode of Line

Mode of Line	Clock	Controller	Operation
Normal	External	Normal	Normal operation
Normal	Internal	Normal	Device supplies clock to simulate synchronous modem
Internal loopback	Internal	Loopback	Device loops data internally
External cable loopback	Internal	Normal	Cable loopback with loopback connector
External modem loopback	External	Normal	Modem loops back data

7.3.6 The Logging Entity

The LOGGING entity is the network component that records information that has potential significance in the operation and/or maintenance of the network. Logging information includes such items as the location of the data recorded, events lost, initialization failures, and circuits down, for example.

The logging function in DECnet-20 V3.0 is under the control of the system software. The operator or System Manager can not set logging parameters. Significant network-related events are recorded in the TOPS-20 file, ERROR.SYS. All users have access to these recorded events by running the SPEAR program. The SPEAR RETRIEVE and SUMMARY reports of network activity are described in detail in Section 3.4.2.

The logging entity has the following choices:

STATE

OFF	The sink is not available. Events destined for sink are discarded.
ON	The sink is available for receiving events.
HOLD	The sink is temporarily unavailable and events are queued.

LOGGING sinktype

CONSOLE
FILE
MONITOR

The arguments for EVENT eventlist in LOGGING commands are typed in the format:

```
EVENT class.event type
```

You type:

```
EVENT 5.3-5,9
```

This means event class 5, event types 3, 4, 5, and 9. Only one event class may be specified in a command. You will need the information in Tables 7-2 and 7-3 to specify the EVENT arguments. Note that a range is shown with a hyphen and a list with a comma.

Table 7-2
EVENT Classes

EVENT Class	Description
0	Network Management Layer
1	Applications Layer
2	Session Control Layer
3	Network Services Layer
4	Transport Layer
5	Data Link Layer
6	Physical Link Layer
7-31	Reserved for other common classes
32-63	RSTS System specific
64-95	RSX System specific
96-127	TOPS-20 System specific
128-159	VMS System specific
160-191	RT System specific
192-479	Reserved for future use
480-511	Customer specific

Table 7-3
Events

Class	Type	Entity	Event text
0	0	none	Event records lost
0	1	node	Automatic node counters
0	2	line, circuit	Automatic data link counters
0	3	line, circuit	Automatic data link service
0	4	line, circuit	Data link counters zeroed
0	5	node	Node counters zeroed
0	6	line, circuit	Passive loopback
0	7	line, circuit	Aborted service request
0	8	any	Automatic counters
0	9	any	Counter zeroed
2	0	none	Local node state change
2	1	none	Access control reject
3	0	none	Invalid message
3	1	none	Invalid flow control
3	2	node	Data base reused
4	0	none	Aged packet loss
4	1	circuit	Node unreachable packet loss
4	2	circuit	Node out-of-range packet loss
4	3	circuit	Oversized packet loss
4	4	circuit	Packet format error
4	5	circuit	Partial routing update loss
4	6	circuit	Verification reject
4	7	circuit	Circuit down, circuit fault
4	8	circuit	Circuit down, software fault
4	9	circuit	Circuit down, operator fault
4	10	circuit	Circuit up
4	11	circuit	Initialization failure, circuit fault
4	12	circuit	Initialization failure, software fault
4	13	circuit	Initialization failure, operator fault
4	14	node	Node reachability change
5	0	line, circuit	Locally initiated state change
5	1	line, circuit	Remotely initiated state change
5	2	line, circuit	Protocol restart received in maintenance mode
5	3	line, circuit	Send error threshold
5	4	line, circuit	Receive error threshold
5	5	line, circuit	Select error threshold
5	6	line, circuit	Block header format error
5	7	line, circuit	Selection address error
5	8	line, circuit	Streaming tributary
5	9	line, circuit	Local buffer too small
5	10	module	Restart (X.25 protocol)
5	11	module	State change (X.25 protocol)
6	0	line	Data set ready transition
6	1	line	Ring indicator transition
6	2	line	Unexpected carrier transition
6	3	line	Memory access error
6	4	line	Communications interface error
6	5	line	Performance error

7.4 GENERIC FORMAT OF NML/NCP COMMANDS

COMMAND ENTITY [KEYWORD] [argument] . . . [KEYWORD] [argument]

Although most commands will have a command entry, an entity entry, and at least one keyword/argument combination, the command itself is the only element that is always present. The [KEYWORD] [argument] fields may also take the form [KEYWORD] [KEYWORD], as in TYPE PHASE-II or STATE OFF. In this form, the second KEYWORD acts as an argument for the first keyword. Under certain conditions, entity is understood. The keyword ALL actually acts as a pluralistic parameter: used with SET/DEFINE or CLEAR/PURGE, ALL indicates all information on the named entity.

There are three pairs of commands - SET and DEFINE, CLEAR and PURGE, and SHOW and LIST - that affect the same entities and accept the same arguments. This means that, except for the command, you type the same information in the same format. For example:

SET LINE KDP-0-0 CLOCK INTERNAL(RET)

DEFINE LINE KDP-0-0 CLOCK INTERNAL(RET)

The first command changes line parameters in the volatile data base; the second changes parameters in the permanent data base.

SET, CLEAR and SHOW refer to parameters in the volatile data base; DEFINE, PURGE, and LIST refer to parameters in the permanent data base. These three pairs are considered together in the detailed descriptions that follow. All other commands are described in separate charts. Remember that neither TOPS-20 nor MCB supports a permanent data base. Therefore, only SET, CLEAR, and SHOW are used for DECnet-20. All commands are given because other operating systems may implement a permanent data base.

7.5 CHANGING NETWORK DATA BASES: SET AND DEFINE COMMANDS

When you generate and install a Phase III DECnet node for your site, certain values can be established in a permanent network data base. Each time the system is brought up, these permanent data base values are read into the current volatile data base. Permanent data base values remain from one system initialization to the next. They take effect whenever the permanent data base is read. The timing is implementation-dependent. To change a value in the permanent data base, you use the appropriate NCP DEFINE command. This newly defined value has no effect on the volatile data base. DECnet-20 V3.0.0 does not support a permanent data base.

In DECnet-20 V3.0, certain initial parameter values are established during network generation. Command files are created that consist of NCP SET commands that use these initial values as arguments. These commands are executed at system startup and the parameter values they set become the initial volatile data base, much like the action of a permanent data base at system startup.

Not all parameter values can be changed or set by the OPERATOR or System Manager. Appendix A (DECnet Parameter Summary) serves as a reference for the applicability (applies to all nodes or to EXECUTOR nodes only, for example) and restrictions (can be displayed, can not be SET, for example) for all node, circuit, line, and logging parameters.

Values displayed in response to the various SHOW commands will be the values in the data base (unless you have changed them). They have been chosen to ensure a viable network. They may not be optimal for your installation because of the many factors upon which network performance depends. However, no value should be changed without good reason and some planning. It is suggested that the System Manager check performance with the SPEAR reports both before and after changes have been made. Whether you have improved or degraded the network performance is best judged by actual results.

To change a value in the dynamic or volatile data base, you use the appropriate NCP SET command. Values established with the SET command are in effect for all commands following the SET command unless changed with another SET command. Volatile data base values are lost when the node is reloaded.

NOTE

NCP commands that are processed by the local Network Management function are described in Chapter 6. Commands processed by the OPR program are described in Chapter 5.

7.5.1 SET/DEFINE Entity ALL

Command	Entity	Keyword/Argument
{SET / DEFINE}	{CIRCUIT cktid / LINE lineid / NODE nodeid}	ALL

Function:

When used with the keyword ALL, the SET command reads all known permanent parameters for the named entity into the volatile data base. Conversely, the DEFINE command reads all known volatile parameters into the permanent data base of the identified entity.

Arguments:

See Section 7.3.2 if you need help on cktid, lineid, or nodeid. The keyword ALL is used with the commands SET/DEFINE and CLEAR/PURGE to indicate all information on the entity identified in the command. Refer to the singular form of each entity for a complete list of all possible keywords and parameters.

Remarks:

DEFINE and SET commands with the keyword ALL cannot be executed by DECnet-20 executor nodes because of the lack of a modifiable permanent data base.

Before attempting to send this command to a non-DECnet node, consult the documentation of that node's operating system. This command depends upon the specific implementation.

7.5.2 SET/DEFINE CIRCUIT

Command	Entity	Keyword/Argument
{SET / DEFINE}	CIRCUIT cktid	{COUNTER TIMER seconds / OWNER ownerid / STATE cktstate / COST cost / LINE lineid / SERVICE srvcntl / TRIBUTARY tribaddr}

Function:

These commands set volatile and permanent circuit parameters for the circuit(s) identified in the command.

Arguments:

Arguments for all circuits:

COUNTER TIMER seconds

Seconds is a decimal integer in the range 1-65535. Counter Timer is a Network Management Timer. When the seconds indicated have elapsed, the circuit counters are recorded as data in a logging event, and then zeroed. The logging begins again, and the log-record-zero cycle continues as long as the node remains up, or until the counter is cleared. If no countertimer value is set, the circuit's counters are not automatically logged.

OWNER ownerid

Ownerid consists of an entity type and entity identification. The only ownerid currently defined is the executor node. (This implies that the circuit is actually reserved for Transport, the DECnet routing module.) Establishing an ownerid merely reserves the circuit. There is no implication as to whether the circuit is open. The owner's exclusive rights to the circuit can always be overridden by Network Management.

STATE cktstate

Cktstate may be CLEARED, OFF, ON, or SERVICE. Refer to Section 7.3.4 for state definitions.

Arguments for DDCMP circuits only:

LINE lineid

Lineid is the Data Link Layer identification of the line to be used by the circuit. Refer to Section 7.3.2 for format.

SERVICE srvcntl

Srvcntl may be ENABLED or DISABLED, indicating respectively that Network Management allows or disallows service operations on a circuit. See Section 7.3.4 for more detail, if needed.

Arguments for all circuits owned by EXECUTOR:

COST cost

Cost is a decimal number in the range 1 to 25. The COST parameter is used in the Transport routing algorithm to determine the most cost-effective routing path. (Refer to Section 2.4 for a more detailed description.)

HELLO TIMER seconds

Seconds is a number in the range 1-65535. The frequency of Transport Hello messages sent to the adjacent node on the circuit is determined by this value.

LISTEN TIMER seconds

Seconds is a decimal number in the range 1-65535. This value determines the seconds that are allowed to lapse before Transport receives either a Hello message or a user message from the adjacent node on the circuit.

NOTE

The DNA Transport Specification contains detailed information on the effect of changing parameter values of timers used by Transport. (See Appendix F for a complete list of DNA Specifications.)

The following arguments are not implemented for circuits in DECnet-20 V3.0. Check the documentation for the operating system of the EXECUTOR to direct commands to remote node.

ACTIVE BASE base

This parameter applies to DDCMP CONTROL circuits only. The value of base represents the base priority to which a tributary is reset after each time it has been polled. This is the base priority for a tributary in the ACTIVE state. Base is a value in the range 0 to 255. If not set, the default is 255.

ACTIVE INCREMENT incrmt

This parameter applies to DDCMP CONTROL circuits only. The value of incrmt represents the increment added to the tributary priority each time the scheduling timer expires. This increment applies to a tributary in the ACTIVE state. Incrmt is a value in the range 0 to 255. If not set, the default is 0.

BABBLE TIMER millisecs

This parameter applies to DDCMP CONTROL circuits only. The value of millisecs represents the number of milliseconds that a selected tributary or remote half-duplex station is allowed to transmit. Millisecs is a value in the range 1 to 65535. If not set, the default is 6000 (6 seconds).

DEAD THRESHOLD count

This parameter applies to DDCMP CONTROL circuits only. The value of count represents the number of times to poll the ACTIVE, INACTIVE, or DYING tributary before changing its polling state to DEAD because of receive timeouts. Count is a value in the range 0 to 255. If not set, the default is 8.

DYING BASE base

This parameter applies to DDCMP CONTROL circuits only. The value of base represents the base priority to which a tributary is reset after each time it has been polled. This is the base priority for a tributary in the DYING state. Base is a value in the range 0 to 255. If not set, the default is 0.

DYING INCREMENT incrmt

This parameter applies to DDCMP CONTROL circuits only. The value of incrmt represents the increment added to the tributary priority each time the scheduling timer expires. This increment applies to a tributary in the DYING state. Incrmt is a value in the range 0 to 255. If not set, the default is 16.

DYING THRESHOLD count

This parameter applies to DDCMP CONTROL circuits only. The value of count represents the number of times to poll the ACTIVE, INACTIVE, or DYING tributary before changing its polling state to DYING because of receive timeouts. Count is a value in the range 0 to 255. If not set, the default is 2.

INACTIVE BASE base

This parameter applies to DDCMP CONTROL circuits only. The value of base represents the base priority to which a tributary is reset after each time it has been polled. This is the base priority for a tributary in the INACTIVE state. Base is a value in the range 0 to 255. If not set, the default is 0.

INACTIVE INCREMENT incrmt

This parameter applies to DDCMP CONTROL circuits only. The value of incrmt represents the increment added to the tributary priority each time the scheduling timer expires. This increment applies to a tributary in the INACTIVE state. Incrmt is a value in the range 0 to 255. If not set, the default is 64.

INACTIVE THRESHOLD count

This parameter applies to DDCMP CONTROL circuits only. The value of count represents the number of times to poll the ACTIVE tributary before changing its polling state to inactive because of no data response. Count is a value in the range 0 to 255. If not set, the default is 8.

MAXIMUM BUFFERS count

This parameter applies to DDCMP CONTROL circuits only. The value of count represents the maximum number of buffers the tributary can use from a common buffer pool. If not set, there is no common buffer pool and buffers are explicitly supplied by the higher level. The value of count is either a value in the range 1 to 255 or the keyword UNLIMITED.

MAXIMUM TRANSMIT count

This parameter applies to DDCMP CONTROL circuits only. The value of count represents the maximum number of data messages that can be transmitted at one time. Count is a value in the range 1-255. If not set, the default is 4.

POLLING STATE polstate

This parameter applies to DDCMP CONTROL circuits only. The value of polstate represents the state of the tributary relative to the multipoint polling algorithm. If not set, the default is AUTOMATIC. The possible states are:

AUTOMATIC - state is allowed to vary according to the operation of the polling algorithm.

ACTIVE/INACTIVE/DYING/DEAD - the tributary is locked in the specified state.

TRANSMIT TIMER millisecs

This parameter applies to DDCMP CONTROL circuits only. The value of millisecs represents the number of milliseconds to delay between data message transmits. Millisecs is a value in the range 0 to 65535.

TRIBUTARY tribaddr

This parameter applies to multipoint DDCMP circuits only. Tribaddr represents the Data Link physical tributary address of the circuit. Tribaddr is a value in the range 0 to 255.

Example:

```
NCP>set ex nod d2137a user operator password d2137a (RET)
NCP>
 6:04:45        NCP
                Set Executor Complete
NCP>set cir dte-0-1 cost 1 (RET)
NCP>
 6:05:11        NCP
                Request # 228 Accepted

NCP>
 6:05:12        NCP
                Request # 228; Set Circuit Completed
```

Remarks:

Section 2.4 describes the COST assignment. Note that the EXECUTOR must be the DN20 MCB node for all SET CIRCUIT commands. If the TOPS-20 node is EXECUTOR, the response will be "Unrecognized component".

Command	Entity	Keyword/Argument
{SET / DEFINE}	KNOWN CIRCUITS	{ALL / COUNTER TIMER seconds / SERVICE srvcntl}

Function:

SET KNOWN CIRCUITS ALL loads all permanent circuit parameter values into the volatile data base. (Not supported for DECnet-20.) Other SET KNOWN CIRCUIT commands set the specified values in the volatile data base of each circuit known to the EXECUTOR. DEFINE KNOWN CIRCUITS ALL has no meaning. Other DEFINE KNOWN CIRCUIT commands set the specified parameter values in the permanent data bases of all circuits known to the EXECUTOR.

Arguments:

Same as for CIRCUIT cktid (singular). See remarks below.

Examples:

(The EXECUTOR is D2137A, the MCB node.)

```
NCP>set kn cir serVICE disABLED (RET)
NCP>
 6:10:21   NCP
           Request # 234 Accepted
NCP>
 6:10:23   NCP
           Request # 234; Set Circuit Completed
             Circuit = DTE-0-1

           Request # 234; Set Circuit Completed
             Circuit = DMC-0

           Request # 234; Set Circuit Completed
             Circuit = DMR-0

           Request # 234; Set Circuit Completed
             Circuit = DMR-1

           .
           .  (output continues for
           .   all known circuits)

NCP>sh kn cir char (RET)
NCP>
 6:10:42   NCP
           Request # 235 Accepted
```

```
NCP>
 6:10:44   NCP
           Request # 235; Show Known Circuits Characteristics
Completed

           Circuit = DTE-0-1

             Service = Disabled
             Cost = 1
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = DTE-0-1

           Circuit = DMC-0

             Service = Disabled
             Cost = 1
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = DMC-0
             Type = DDCMP-DMC

           Circuit = DMR-0

             Service = Disabled
             Cost = 1
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = DMR-0
             Type = DDCMP-DMC

           Circuit = DMR-1

             Service = Disabled
             Cost = 1
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = DMR-1
             Type = DDCMP-DMC

           .
           .  (output continues for
           .   all known circuits)

NCP>set kn cir ser ena (RET)
NCP>
 6:11:08   NCP
           Request # 236 Accepted
NCP>sh kn cir char (RET)

 6:11:10   NCP
           Request # 236; Set Circuit Completed
             Circuit = DTE-0-1
```

```
                Request # 236; Set Circuit Completed
                  Circuit = DMC-0

                Request # 236; Set Circuit Completed
                  Circuit = DMR-0

                Request # 236; Set Circuit Completed
                  Circuit = DMR-1

NCP>
 6:11:17   NCP
           Request # 237 Accepted

 6:11:19   NCP
           Request # 237; Show Known Circuits Characteristics
Completed

           Circuit = DTE-0-1

             Service = Enabled
             Cost = 1
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = DTE-0-1

           Circuit = DMC-0

             Service = Enabled
             Cost = 1
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = DMC-0
             Type = DDCMP-DMC

           Circuit = DMR-0

             Service = Enabled
             Cost = 1
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = DMR-0
             Type = DDCMP-DMC

           Circuit = DMR-1

             Service = Enabled
             Cost = 1
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = DMR-1
             Type = DDCMP-DMC
```

Remarks:

Although all arguments allowed for SET CIRCUIT (described in Section 7.5.2) may be used in the SET KNOWN CIRCUITS commands, many of the arguments are inappropriate for plural entities. The arguments listed are those most likely to be meaningful. The NMLT20 program processes all arguments listed under SET CIRCUIT for the plural CIRCUITS. However, you should consider the probable results carefully before using arguments not listed.

SET STATE for plural entities is not recommended for DECnet-20. Because KNOWN CIRCUITS normally includes the DTE-0-1 circuit, processing the command SET KNOWN CIRCUITS STATE OFF removes the DN20 from the network temporarily and the KL attempts an automatic reload, thus accomplishing nothing. Because the KL always attempts a reload when the DN20 is offline, the command SET KNOWN CIRCUITS STATE ON is unnecessary.

7.5.3 SET/DEFINE LINE

Command	Entity	Keyword/Argument
{SET / DEFINE}	LINE lineid	{CLOCK clock-mode CONTROLLER controller-mode COUNTER TIMER seconds DEVICE deviceid DUPLEX duplex-mode PROTOCOL protocol-name RETRANSMIT TIMER milliseconds SERVICE service-control SERVICE TIMER milliseconds STATE line-state}

Function:

These commands set volatile and permanent line parameters for the line(s) identified in the command.

Arguments:

COUNTER TIMER seconds

Seconds is a decimal integer in the range 1-65535. Counter Timer is a Network Management Timer. When the seconds indicated have elapsed, the line counters are recorded as data in a logging event, and then zeroed. If no countertimer value is set, the line's counters are not automatically logged.

DEVICE deviceid

The deviceid specifies the local hardware to be used. It consists of a device identification (dev) and a controller number (c) if the device supports a single line. For a multiple line controller, the deviceid consists of a device identification, a controller number, and a unit number (u). For example,

dev c u

DEVICE DMR-1
DEVICE KDP-0-2

PROTOCOL

For DECnet-20, this parameter is implemented for the MCB node only and for KDP lines only. Possible values are DDCMP POINT and DDCMP DMC. Other DIGITAL operating systems may permit the values DDCMP CONTROL, DDCMP TRIBUTARY, and LAPB. See the appropriate documentation. See Section 7.3.5 for definitions if needed.

RETRANSMIT TIMER milliseconds

Milliseconds is a decimal integer in the range 1-65535. The RETRANSMIT TIMER is a Data Link timer. When the number of milliseconds specified have elapsed, a block is retransmitted. This timer is used for normal operation of the line.

SERVICE TIMER milliseconds

Milliseconds is a decimal integer in the range 1-65535. The SERVICE TIMER is a Data Link timer that is used during service operations. Milliseconds represents the maximum amount of time allowed to elapse before a receive request is completed.

CLOCK, CONTROLLER, DUPLEX, SERVICE, and STATE: See Section 7.3.5.

Examples:

```
NCP>set ex nod d2137a(RET)
NCP>
11:28:15        NCP
                Set Executor Complete
NCP>sho lin dmc-0 char(RET)
NCP>
11:28:54        NCP
                Request # 219 Accepted
NCP>
11:28:55        NCP
                Request # 219; Show Line Characteristics Completed

                Line = DMC-0

                  Device = DMC-0
                  Receive Buffers = 7
                  Controller = Normal
                  Duplex = Full
                  Protocol = DDCMP-DMC
                  Clock = External
                  Service Timer = 3000
                  Retransmit Timer = 3000
                  Controller Register = 160740
                  Interrupt Vector = 670
                  Interrupt Priority = 5
NCP>set lin dmc-0 counter timer 10(RET)
NCP>
11:30:44        NCP
                Request # 220 Accepted
NCP>
11:30:46        NCP
                Request # 220; Set Line Failed; Privilege violation

NCP>set ex nod d2137a password d2137a user operator(RET)
NCP>
11:31:40        NCP
                Set Executor Complete
NCP>set line dmc-0 counter timer 10(RET)
NCP>
11:32:05        NCP
                Request # 221 Accepted
```

```
NCP>
11:32:07        NCP
                Request # 221; Set Line Completed
NCP>sh line dmc-0 char(RET)
NCP>
11:32:46        NCP
                Request # 222 Accepted
NCP>
11:32:50        NCP
                Request # 222; Show Line Characteristics Completed

                Line = DMC-0

                  Counter Timer = 10
                  Device = DMC-0
                  Receive Buffers = 7
                  Controller = Normal
                  Duplex = Full
                  Protocol = DDCMP-DMC
                  Clock = External
                  Service Timer = 3000
                  Retransmit Timer = 3000
                  Controller Register = 160740
                  Interrupt Vector = 670
                  Interrupt Priority = 5
NCP>
```

Remarks:

For DECnet-20 V3.0, the values for SERVICE TIMER and TRANSMIT TIMER have additional restrictions according to device type. For KDP lines, any multiple of 1000 equal to or greater than 2000 is acceptable. For DMR and DMC lines, only the values 1000 and 3000 are acceptable.

Note in the examples that failure to use password and user parameters defined during network generation in the SET EXECUTOR command allows SHOW commands but gives an error-response for SET commands.

7.5.4 SET/DEFINE NODE

Command	Entity	Keyword/Argument
{SET {DEFINE}	{EXECUTOR {NODE nodename}	{ADDRESS nodeaddr} {CIRCUIT cktid }

Function:

This command sets the volatile or permanent address and circuit for the specified nodename. Nodename must represent the executor node for the ADDRESS nodeaddr parameter. This command cannot be used to set the address of any other node. The CIRCUIT cktid parameter applies to loop nodes and nodes by name only. (See Remarks.)

Arguments:

ADDRESS nodeaddr

Nodeaddr is a unique decimal integer in the range 1 to MAXIMUM ADDRESS. MAXIMUM ADDRESS is the largest address in the network, that is, the number of nodes that can be known. TOPS-20 DECnet-20 sets a limit of 255 supportable nodes. (The addresses may be in the range 1 to 65,535.)

CIRCUIT cktid

Cktid is the identification of the circuit to be used for traffic to the named node which is a loop node.

Examples:

```
NCP>set ex no d2136a(RET)
NCP>
10:07:17        NCP
                Set Executor Complete
NCP>set node loopme cir dmr-0(RET)
NCP>
10:08:07        NCP
                Request # 23 Accepted
NCP>
10:08:08        NCP
                Request # 23; Set Node Completed
NCP>loop no loopme(RET)
NCP>
10:08:30        NCP
                Request # 24 Accepted
NCP>
10:08:32        NCP
                Request # 24; Loop Node Completed
NCP>loop no loopme count 10(RET)
NCP>
10:09:08        NCP
                Request # 25 Accepted
NCP>
10:09:13        NCP
                Request # 25; Loop Node Completed
NCP>
```

Remarks:

DECnet-20 does not support the SET NODE ADDRESS command for the EXECUTOR. The node name/node address relationship is set during generation time. To change the address, the NETGEN program must be repeated.

Command	Entity	Keyword/Argument
{SET / DEFINE}	**NODE nodeaddr**	**NAME nodename**

Function:

This command sets the volatile or permanent parameter for the nodename to be associated with the specified address. A nodename/nodeaddr association is unique.

Argument:

Nodename consists of 1 to 6 alphanumeric characters with at least one alpha character. A node name is associated with one and only one node address.

Examples:

```
NCP>sh ex(RET)
NCP>
 9:35:43   NCP
           Request # 279; Show Executor Node Summary Completed

           Executor Node = 124 (KL2136)

             Identification = DECnet-20 V3.0.0
             State = On
             Active Links = 0

NCP>set node 124 name anynam(RET)
NCP>
 9:36:43   NCP
           Request # 280; Set Node Failed, Component in wrong state

NCP>set no 129 name anynam(RET)
NCP>
 9:43:00   NCP
           Request # 281; Set Node Completed
NCP>sh no 129(RET)
NCP>
 9:43:14   NCP
           Request # 282; Show Node Summary Completed

           Remote Node = 129 (ANYNAM)

             State = Reachable
             Active Links = 0
```

Remarks:

DECnet-20 does not support the SET NODE NAME command for the EXECUTOR. (The failure of Request # 280 in the example illustrates this.)

Node names are of local significance only. If used in a command addressed to the network, the node address will be substituted (using a table lookup) for the name. This mapping is transparent to the user. Thus, although the network recognizes nodes by address only, the operator has the convenience of using either nodename or nodeaddress (except for the few commands where one or the other is specifically noted).

Command	Entity	Keyword/Argument
{SET / DEFINE}	{NODE nodeid / EXECUTOR}	BUFFER SIZE bytes COUNTER TIMER seconds CPU cputyp DELAY FACTOR number DELAY WEIGHT number DUMP ADDRESS number DUMP COUNT number DUMP FILE fileid HOST nodeid IDENTIFICATION idstring INACTIVITY TIMER seconds INCOMING TIMER seconds LOAD FILE fileid MAXIMUM ADDRESS number MAXIMUM BUFFERS number MAXIMUM COST number MAXIMUM HOPS number MAXIMUM CIRCUITS number MAXIMUM LINKS number MAXIMUM VISITS number OUTGOING TIMER seconds RETRANSMIT FACTOR number SECONDARY DUMPER fileid SECONDARY LOADER fileid SERVICE DEVICE devtype SERVICE CIRCUIT cktid SERVICE PASSWORD password SOFTWARE IDENTIFICATION fileid SOFTWARE TYPE sftyp STATE state TERTIARY LOADER fileid TYPE nodtyp
	(EXECUTOR may be substituted for NODE nodeid.)	

Function:

This command sets the volatile or permanent parameters for the node identified as nodeid or set as EXECUTOR.

Arguments:

BUFFER SIZE bytes

This parameter is applicable to the executor node only. This means that entity must be EXECUTOR or nodeid must refer to the executor. If the executor has not been specified, EXECUTOR will default to the node where NCP is running.

Sets the actual size of all circuit buffers, including protocol overhead. Size is a decimal number in the range 1-65535.

For DECnet-20 nodes, the permitted range is 290-576. The default is 576 bytes. This value must be 576 to support all DIGITAL operating systems. The buffer size must be equal or greater than 2 * MAXIMUM ADDRESS + 5.

COUNTER TIMER seconds

This parameter is applicable to all nodes except loop nodes.

Seconds is a decimal integer in the range 1-65535. Counter Timer is a Network Management Timer. When the seconds indicated have elapsed, the node counters are recorded as data in a logging event, and then zeroed. The logging begins again, and the log-record-zero cycle continues as long as the node remains up, or until the counter is cleared. If no countertimer value is set, the node's counters are not automatically logged.

CPU cputyp

This parameter is applicable to adjacent nodes only. See Section 7.3.3 for possible values.

The MCB software in the DN20 node supports loading of RSX-11S and DN200 nodes (executor must be the DN20 node). TOPS-20 supports loading of DN20 nodes (executor must be the KL10-E node). Refer to the appropriate manuals to determine CPU types that can be downline loaded by non-DECnet-20 nodes that are acting as executors in your network.

DELAY FACTOR number

This parameter is applicable to the executor node only.

Sets the number used as a multiplier in an NSP algorithm that determines when to retransmit a message. The delay factor is multiplied by 1/16 of the round trip delay to set the retransmission timer to that node. The default value for DECnet-20 is 2. Number is a decimal number in the range 1-255. The larger the value of number, the longer the delay before retransmission.

DELAY WEIGHT number

This parameter is applicable to the executor node only.

This parameter is used as a weighting factor when updating the estimated round trip delay to a node. Number is a decimal in the range 1-255. This value is initialized during network generation but may be modified by Network Management. Normally, the value is modified only if the round trip delay time changes.

DUMP ADDRESS number

This parameter is applicable to adjacent nodes only.

Sets the memory address where the upline dump of an adjacent node is to begin. This parameter value is not supported for DECnet-20 nodes because a full dump is always taken.

DUMP COUNT number

Sets the default number of memory units to upline dump from the adjacent node specified. This parameter value is not supported for DECnet-20 nodes because a full dump is always taken.

DUMP FILE fileid

This parameter is applicable to adjacent nodes only.

Identifies the file to receive the upline dump.

HOST nodeid

This parameter is applicable to an executor node or an adjacent node.

For the executor, identifies the node from which the executor requests services. For an adjacent node, it is the node providing service. If no host is specified, the default is the executor node.

IDENTIFICATION idstring

This parameter is applicable to the executor node only.

Sets the text identification string for the executor, for example, "Development System" or "Research Lab." Idstring is an arbitrary string of 1-32 characters. An idstring containing blanks or tabs must be enclosed in quotation marks; two adjacent quotation marks indicate a quotation mark within a quoted string.

INACTIVITY TIMER seconds

This parameter is applicable to the executor node only.

The value in seconds determines the maximum time allowed to elapse with no activity in a logical link. When the number of seconds specified have elapsed, NSP will test the link with artificial traffic.

INCOMING TIMER seconds

This parameter is applicable to the executor node only.

Sets a timeout value for incoming connects. If the connect is not accepted or rejected by the user within the seconds specified, Session Control rejects it for the user. Seconds is a decimal in the range 1-65535.

LOAD FILE fileid

This parameter is applicable to an adjacent node only.

Sets the file identification of the file to read when the node is downline loaded. Fileid is in the format required by the executor's file system.

MAXIMUM ADDRESS number

This parameter is applicable to the executor node only.

Sets the greatest node address in the network, and thus determines the maximum number of nodes that can be known about. Number is a decimal value in the range 2-255.

DECnet-20 sets a limit of 255 supportable nodes.

MAXIMUM BUFFERS number

This parameter is applicable to the executor node only.

Sets the total number of buffers allocated to all circuits. Transport uses this number to determine the size of its buffer pool.

This parameter applies to the DN20 MCB node only and can not be set by the OPERATOR or System Manager. It is generated during NETGEN and can be changed only by repeating the NETGEN procedure. MAXIMUM BUFFERS is displayed when you type the NCP command SHOW EXECUTOR CHARACTERISTICS when the EXECUTOR is the MCB front end.

MAXIMUM COST number

This parameter is applicable to the executor node only.

Sets the maximum total path cost allowed from the executor to any node. Path cost is the sum of the circuit costs along a path between two nodes. Number is a decimal in the range 1-1023.

MAXIMUM HOPS number

This parameter is applicable to the executor node only.

Sets the maximum number of hops between any two nodes in the network. Number is a decimal number in the range 1-30.

MAXIMUM CIRCUITS number

This parameter is applicable to the executor node only.

Sets the maximum number of circuits that this node can know about. Number is a decimal in the range 1-65535.

This parameter applies to the DN20 MCB node only and can not be set by the OPERATOR or System Manager. MAXIMUM CIRCUITS is displayed in response to the SHOW EXECUTOR CHARACTERISTICS NCP command when the EXECUTOR is the DN20 MCB node.

MAXIMUM LINKS number

This parameter is applicable to the executor node only.

Sets the maximum allowable active logical link count for the node. Number is a decimal in the range 1-65535.

This parameter applies to the DN20 MCB node only and can not be set by the OPERATOR or System Manager. MAXIMUM LINKS is displayed in response to the SHOW EXECUTOR CHARACTERISTICS NCP command when the EXECUTOR is the DN20 MCB node.

MAXIMUM VISITS number

This parameter is applicable to the executor node only.

Sets the maximum number of nodes a message coming into this node can have visited. If the number specified is exceeded and the destination is not for this node, the message is discarded. Number is a decimal number that may not be less than MAXIMUM HOPS. The upper limit is 255.

OUTGOING TIMER seconds

This parameter is applicable to the executor node only.

Sets a timeout value for outgoing connects. If the time between the request of a connect and the acknowledgement of the request is greater than the specified seconds, Session Control returns an error. Seconds is a decimal in the range 1-65535.

RETRANSMIT FACTOR number

This parameter is applicable to the executor node only.

Sets the number of times the source NSP will restart the retransmission timer. If the value of number is exceeded, NSP reports to Session Control a loss of confidence for this logical link. In DECnet-20, Session Control will then abort the logical link.

SECONDARY DUMPER fileid

This parameter is applicable to adjacent nodes only.

Identifies the secondary dumper file to be used for upline dumping the adjacent node. See Section 4.5.9, "Specifying Files."

SECONDARY LOADER fileid

This parameter is applicable to adjacent nodes only.

Identifies the secondary loader file to be used for downline loading the adjacent node.

SERVICE DEVICE devtype

This parameter is applicable to adjacent nodes only.

Sets the service device type the adjacent node uses when in service slave mode (MOP protocol). The value typed for devtype is one of the standard line device mnemonics, such as DMC or KDP. Appendix D contains a list of all currently recognized DECnet line devices.

DECnet-20 does not require the device type preparatory to a LOAD command; however, if devtype is set, the system will verify that the node is associated with the named device.

SERVICE CIRCUIT cktid

This parameter is applicable to adjacent nodes only.

Sets the circuit to be used to the adjacent node for downline loading or upline dumping. For downline loading the NODE node parameter must be the target (adjacent) node. See Section 7.3.2 for cktid if you need help.

SERVICE PASSWORD password

This parameter is applicable to adjacent nodes only.

Sets the password required to trigger the bootstrap mechanism on the adjacent node. Password is a hexadecimal number in the range 0-FFFFFFFFFFFFFFFF.

SOFTWARE IDENTIFICATION softwareid

This parameter is applicable to adjacent nodes only.

Sets any descriptive string that will identify the software to be loaded when the adjacent node is downline loaded (example, DN20 files). Softwareid contains up to 16 alphanumeric characters.

SOFTWARE TYPE sftyp

This parameter is applicable to adjacent nodes only.

Identifies the initial software to be loaded in a downline load operation as one of:

SECONDARY LOADER
TERTIARY LOADER
SYSTEM

STATE state

For EXECUTOR node:

ON
OFF
SHUT
RESTRICTED
(See Section 7.3.3 for definition of these parameter values.)

For Remote nodes:

REACHABLE
UNREACHABLE
(See Section 7.3.3 for definition of these parameter values.)

TERTIARY LOADER fileid

This parameter is applicable to adjacent nodes only.

Identifies the tertiary loader file to be used for downline loading the adjacent node.

TYPE nodtyp

This parameter is applicable to the executor node only. NMLT20 will parse the command, but TYPE can not be set for DECnet-20. If a node other than a TOPS-20 node is to execute this command, consult the appropriate documentation. Type is one of:

Routing
Nonrouting
Phase II

Examples:

```
NCP>set ex nod d2137a user operator password d2137a(RET)
NCP>
16:28:54   NCP
           Set Executor Complete
NCP>sho ex char(RET)
NCP>
16:36:48   NCP
           Request # 213 Accepted
NCP>
16:36:49   NCP
           Request # 213; Show Executor Node Characteristics
Completed

           Executor Node = 123 (D2137A)

             Identification = DECnet-20 V3.0 Release
             Management Version = 3.0.0
             Host = 113 (KL2137)
             Loop Count = 1
             Loop Length = 127
             Loop With = Mixed
             Incoming Timer = 10
             Outgoing Timer = 30
             NSP Version = 3.2.0
             Maximum Links = 8
             Delay Factor = 2
             Delay Weight = 3
             Inactivity Timer = 30
             Retransmit Factor = 5
             Routing Version = 1.3.0
             Type = Routing
             Routing Timer = 60
             Maximum Address = 255
             Maximum Circuits = 7
             Maximum Cost = 100
             Maximum Hops = 16
             Maximum Visits = 32
             Maximum Buffers = 65
             Buffer Size = 576

NCP>set ex incOMING timER 15(RET)
NCP>
16:38:51   NCP
           Request # 214 Accepted
NCP>
16:38:57   NCP
           Request # 214; Set Node Completed
NCP>set ex outGOING timER 35(RET)
NCP>
16:39:33   NCP
           Request # 215 Accepted
NCP>
16:39:33   NCP
           Request # 215; Set Node Completed
NCP>set ex maximum cost 50(RET)
NCP>
16:40:41   NCP
           Request # 216 Accepted
```

```
NCP>
16:40:42    NCP
            Request # 216; Set Node Completed
NCP>sh ex char (RET)
NCP>
16:41:04    NCP
            Request # 217 Accepted
NCP>
16:41:26    NCP
            Request # 217; Show Executor Node Characteristics
Completed

            Executor Node = 123 (D2137A)

              Identification = DECnet-20 V3.0 Release
              Management Version = 3.0.0
              Host = 113 (KL2137)
              Loop Count = 1
              Loop Length = 127
              Loop With = Mixed
              Incoming Timer = 15
              Outgoing Timer = 35
              NSP Version = 3.2.0
              Maximum Links = 8
              Delay Factor = 2
              Delay Weight = 3
              Inactivity Timer = 30
              Retransmit Factor = 5
              Routing Version = 1.3.0
              Type = Routing
              Routing Timer = 60
              Maximum Address = 255
              Maximum Circuits = 7
              Maximum Cost = 50
              Maximum Hops = 16
              Maximum Visits = 32
              Maximum Buffers = 65
              Buffer Size = 576
NCP>exit (RET)
$
```

Remarks:

As an OPERATOR-user, if you change a parameter value (or several values) to improve a temporary condition or to get a response specific to your own requirements, it is usually good practice to restore the original setting before exiting from NCP. If you feel that the change should remain, consult your system manager or inform other users by message. The default parameter values that are present when the system is brought up are acceptable under normal conditions. Be sure that you understand the probable effect of any change before you make it. For example, it is strongly suggested that you refer to the NSP specification before changing DELAY FACTOR or DELAY WEIGHT.

Command	Entity	Keyword/Argument
{SET / DEFINE}	KNOWN NODES	{ALL / COUNTER TIMER seconds}

Function:

SET KNOWN NODES ALL loads all node parameter values from the permanent data base into the volatile data base. (This command is not supported for DECnet-20.) Other SET KNOWN NODES commands set the specified values in the volatile data base of each node known to the EXECUTOR. Note that many values that can be set for one node do not apply logically to KNOWN NODES.

Arguments:

See SET/DEFINE NODE (singular).

Examples:

```
@ena (RET)
$opr (RET)
OPR>disa out a (RET)
OPR>ent n (RET)

10:52:48  --Output display for OPR modified--
NCP>sho kno nod char (RET)
NCP>
10:54:56        NCP
          Request # 15; Show Known Nodes Characteristics Completed

          Executor Node = 113 (KL2137)

            Identification = DECnet-20 V3.0.0
            Management Version = 3.0.0
            Loop Count = 1
            Loop Length = 127
            Loop With = Mixed
            NSP Version = 3.2.0
            Inactivity Timer = 20
            Delay Weight = 32
            Delay Factor = 63
            Retransmit Factor = 5
            Type = Phase II

          Remote Node = 1 (RED)

            No Information

          Remote Node = 2 (YELLOW)

            No Information

          .

          .

          .

          Remote Node = 123 (D2137A)
```

```
                Service Circuit = DTE-0-1
                CPU = PDP11
                Load File = PS:<NEXT-RELEASE>D2137A.SYS.118
                Secondary Loader = PS:<NEXT-RELEASE>DTEMPS.SYS.3
                Tertiary Loader = PS:<NEXT-RELEASE>DTEMPT.SYS.4
                Dump File = PS:<NEXT-RELEASE>D2137A.DMP
                Secondary Dumper = PS:<NEXT-RELEASE>DTEDMP.SYS.3
                Host = 113 (KL2137)

              .

              .

              .

              Remote Node = 254 (SYS880)

                No Information

              Remote Node = 255 (DN200)

                No Information

NCP>set kn nod counter timer 55 (RET)
NCP>
10:57:35      NCP
              Request # 16; Set Node Completed
                Executor Node = 113 (KL2137)

              Request # 16; Set Node Completed
                Remote Node = 1 (RED)

              Request # 16; Set Node Completed
                Remote Node = 2 (YELLOW)

              .

              .

              .

10:57:38      NCP
              Request # 16; Set Node Completed
                Remote Node = 123 (D2137A)

              .

              .

              Request # 16; Set Node Completed
                Remote Node = 254 (SYS880)

              Request # 16; Set Node Completed
                Remote Node = 255 (DN200)
```

```
NCP>sho kno nod char (RET)
NCP>

11:01:09        NCP
                Request # 17; Show Known Nodes Characteristics
                  Completed

                Executor Node = 113 (KL2137)

                  Identification = DECnet-20 V3.0.0
                  Management Version = 3.0.0
                  Loop Count = 1
                  Loop Length = 127
                  Loop With = Mixed
                  Counter Timer = 55
                  NSP Version = 3.2.0
                  Inactivity Timer = 20
                  Delay Weight = 32
                  Delay Factor = 63
                  Retransmit Factor = 5
                  Type = Phase II

                Remote Node = 1 (RED)

                  Counter Timer = 55

                Remote Node = 2 (YELLOW)

                  Counter Timer = 55

                .

                .

                .

                Remote Node = 123 (D2137A)

                  Service Circuit = DTE-0-1
                  CPU = PDP11
                  Load File = PS:<NEXT-RELEASE>D2137A.SYS.118
                  Secondary Loader = PS:<NEXT-RELEASE>DTEMPS.SYS.3
                  Tertiary Loader = PS:<NEXT-RELEASE>DTEMPT.SYS.4
                  Dump File = PS:<NEXT-RELEASE>D2137A.DMP
                  Secondary Dumper = PS:<NEXT-RELEASE>DTEDMP.SYS.3
                  Host = 113 (KL2137)
                  Counter Timer = 55

                .

                .

                .

                Remote Node = 254 (SYS880)

                  Counter Timer = 55

                Remote Node = 255 (DN200)

                  Counter Timer = 55
```

```
11:02:36        NCP
                [The queue is empty]
NCP>
NCP>ex (RET)
$
```

Remarks:

On the DN20 MCB node, setting COUNTER TIMERS for KNOWN NODES is very demanding on NML resources and should be avoided if not necessary.

7.5.5 SET/DEFINE LOGGING

Command	Entity	Keyword/Argument
{SET / DEFINE}	{LOGGING sinktype / KNOWN LOGGING}	{EVENT eventlist [sourcequal] [sinknode] / KNOWN EVENTS [sourcequal] [sinknode] / NAME sinkname / STATE sinkstate}

Function:

Used to control where events are logged (event sinks) and which events get logged.

Arguments:

Refer to Section 7.3.6 for sinktype, eventlist, and sinkstate. Refer to Table 7-1 in the same section for the relationship between class and entity required for sourcequal.

NAME sinkname

Sinkname further identifies the sinktype as follows:

if sinktype is CONSOLE, enter device name
if sinktype is FILE, enter file name
if sinktype is MONITOR, enter identification string for the monitoring program

sourcequal

This argument selects a specific entity according to event classes. The format is: LINE lineid, NODE nodeid, or CIRCUIT cktid. Sourcequal (source qualifier) may precede or follow EVENT eventlist or KNOWN EVENTS.

sinknode

Specifies a node that receives events in format: SINK NODE nodeid or SINK EXECUTOR.

Examples:

```
NCP>set log file state off(RET)
NCP>
11:10:17        NCP
                Request # 257; Set Logging Completed

                        * * *

NCP>set logging file state on(RET)
NCP>
11:17:28        NCP
                Request # 260; Set Logging Completed
```

Remarks:

DECnet-20 does not support control of logging parameters by the OPERATOR or System manager. The system software, transparent to the user, automatically logs a limited number of events specific to TOPS-20 and MCB (front end) source nodes. The OPERATOR and System Manager have access to recorded events through the ERROR.SYS file and the SPEAR program. The examples shown above are the only NCP SET LOGGING commands that are supported for DECnet-20. DECnet-20 software can parse all SET/DEFINE LOGGING commands. A command sent to a remote node will be processed if the remote node is running an operating system that has implemented the command. A command not implemented by the operating system of the executor will receive a response similar to the following. The example command was given to the TOPS-20 node, KL2136.

```
NCP>set log file event 5.3-5,9(RET)
NCP>
11:13:09    NCP
            Request # 259;  Set Logging Failed,  Unrecognized
            parameter type
```

7.6 CHANGING NETWORK DATA BASES: CLEAR AND PURGE COMMANDS

The CLEAR command clears parameters from the volatile data base; for systems that support a permanent data base, the PURGE command clears parameters from the permanent data base. The entities are the same as the entities for the SET and DEFINE commands. However, not all parameters can be purged or cleared by executing commands that specify individual parameter values. The allowed parameter keywords are, therefore, a subset of the SET/DEFINE keywords. Except for the LOGGING entity, values need not be supplied for keywords.

Refer to the arguments for the SET/DEFINE commands if you need to refresh your memory on restrictions and exact meaning of keywords. Complete information follows for only those commands that require further description.

Command	Entity	Keyword/Argument
{CLEAR / PURGE}	LOGGING sinktype	{EVENT eventlist [sourcequal] [sinknode] / KNOWN EVENTS [sourcequal] [sinknode] / NAME sinkname}

Function:

Together with the SET/DEFINE LOGGING commands, these commands control the logging of events, both the events that will be logged and where they will be logged.

Arguments:

Refer to Section 7.3.6 for details.

Remarks:

DECnet-20 does not support these commands for DECnet-20 nodes. Logging parameters are not controllable by the OPERATOR or System Manager. The implementation of the logging function is system-specific. Consult the appropriate documentation.

Command	Entity	Keyword/Argument
{CLEAR / PURGE}	{CIRCUIT cktid / KNOWN CIRCUITS}	{ALL / COUNTER TIMER / OWNER} For systems supporting multipoint: {ACTIVE/INACTIVE/DYING BASE / ACTIVE/INACTIVE/DYING INCREMENT / BABBLE TIMER / INACTIVE/DYING/DEAD THRESHOLD / MAXIMUM BUFFERS / MAXIMUM TRANSMITS / TRANSMIT TIMER}
{CLEAR / PURGE}	{LINE lineid / KNOWN LINES}	{ALL / COUNTER TIMER} For systems supporting multipoint: DEAD/DELAY/SCHEDULING TIMER
{CLEAR / PURGE}	{NODE nodeid / KNOWN NODES / (EXECUTOR may be substituted for nodeid.)}	{ALL / CIRCUIT / COUNTER TIMER / CPU / DUMP ADDRESS / DUMP COUNT / DUMP FILE / HOST / IDENTIFICATION / INCOMING TIMER / LOAD FILE / NAME / OUTGOING TIMER / SECONDARY DUMPER / SECONDARY LOADER / SERVICE DEVICE / SERVICE CIRCUIT / SERVICE PASSWORD / SOFTWARE IDENTIFICATION / SOFTWARE TYPE / TERTIARY LOADER}

Function:

The CLEAR and PURGE commands remove the volatile and permanent values, respectively, from the specified entity.

Arguments:

None. (You specify the keyword only.)

Examples:

With the exception of CLEAR EXECUTOR to change EXECUTOR from the MCB front end to the TOPS-20 KL node, and the CLEAR NODE command to remove a loopnode NAME, you will have little use for the CLEAR command. (These two uses have been shown.) It is not necessary to CLEAR a parameter value for an entity to change it. Using the SET command to establish the new value is sufficient.

Remarks:

DECnet-20 does not support the PURGE command for any entity when a TOPS-20 node or MCB node is serving as EXECUTOR (there is no modifiable data base). DECnet-20 does not support the CLEAR command for the following parameters when a TOPS-20 node or MCB node is the EXECUTOR:

For CIRCUITS

ALL
OWNER
Multipoint parameters
X.25 parameters

For LINES

ALL
Multipoint parameters
X.25 parameters

For NODES

ALL
DUMP ADDRESS
DUMP COUNT
NAME (not for EXECUTOR)
X.25 parameters
Multipoint parameters

7.7 MONITORING THE NETWORK: LIST AND SHOW COMMANDS

The SHOW and LIST commands are used to display the following types of information:

- characteristics - parameters that remain constant until cleared or reset (timer values and buffer size, for example)
- counters - error and performance statistics (logical links in use, for example)
- events - significant occurrences in the network that are automatically logged; logged events aid in detecting failures and isolating problems (line initialization failures or line down, for example)

- status - dynamic values associated with an entity, including state
- summary - includes the most useful information from the characteristics and status types

No values follow the keywords.

Command	Entity	Keyword/Argument	Qualifier
LIST	LOGGING sinktype	EVENTS	SINK NODE nodeid
SHOW	KNOWN LOGGING	STATUS	KNOWN SINKS
	ACTIVE LOGGING	CHARACTERISTICS	
		SUMMARY	

Function:

The SHOW command displays information from the volatile data base; the LIST command displays information from the permanent data base.

Arguments:

The qualifier KNOWN SINKS is not supported for DECnet-20 V3.0.0. The SINK NODE is always the KL10-E and will be the default.

Examples:

```
NCP>sh log file sum (RET)
NCP>
 6:30:10   NCP
           Request # 255; Show Logging Summary Completed

           Logging = File

             State = On
             Name = ERROR.SYS

                          * * *

NCP>sh log file char (RET)
NCP>
 6:30:57   NCP
           Request # 256; Show Logging Characteristics Completed

           Logging = File

             Name = ERROR.SYS

                          * * *

NCP>sh log file sta (RET)
NCP>
 6:31:28   NCP
           Request # 258; Show Logging Status Completed

           Logging = File

             State = On
```

Remarks:

When the network is brought up, logging file status is "STATE ON." If you request a function not implemented by the executor, the response will be similar to:

```
NCP>sh logging file events (RET)
NCP>
 6:31:47        NCP
                Request # 259; Show Logging Events Failed,
                Unrecognized component

                Logging = File
```

Command	Entity	Keyword/Argument
{LIST} {SHOW}	CIRCUIT cktid	CHARACTERISTICS
	ACTIVE CIRCUITS	COUNTERS
	KNOWN CIRCUITS	STATUS
	LINE lineid	SUMMARY
	ACTIVE LINES	
	KNOWN LINES	
	NODE nodeid	
	ACTIVE NODES	
	KNOWN NODES	
	LOOP NODES	

Arguments:

None. (You can use the display type keyword only.)

Examples:

```
NCP>set ex no d2137a (RET)
NCP>sh kn ci cou (RET)

 6:32:38   NCP
           Set Executor Complete
NCP>
 6:34:14   NCP
           Request # 262 Accepted
NCP>
 6:34:15   NCP
           Request # 262; Show Known Circuits Counters Completed

           Circuit = DTE-0-1

             7272       Terminating Packets Received
             9027       Originating Packets Sent
             0          Terminating Congestion Loss
             6701       Transit Packets Received
             7186       Transit Packets Sent
             0          Transit Congestion Loss
             0          Circuit Downs
             0          Initialization Failures
             852561     Bytes Received
             510678     Bytes Sent
             14361      Data Blocks Received
             16220      Data Blocks Sent
```

```
        Circuit = DMC-0

          0          Terminating Packets Received
          0          Originating Packets Sent
          0          Terminating Congestion Loss
          1615       Transit Packets Received
          1615       Transit Packets Sent
          0          Transit Congestion Loss
          1          Circuit Downs
          0          Initialization Failures
          17910      Bytes Received
          230124     Bytes Sent
          1233       Data Blocks Received
          1615       Data Blocks Sent
          0          Data Error Inbound
          0          Data Errors Outbound
          0          Remote Reply Timeouts
          0          Local Reply Timeouts
          0          Remote Buffer Errors
          0          Local Buffer Errors

6:34:15   NCP

         Circuit = DMR-0

          0          Terminating Packets Received
          0          Originating Packets Sent
          0          Terminating Congestion Loss
          5893       Transit Packets Received
          5406       Transit Packets Sent
          0          Transit Congestion Loss
          1          Circuit Downs
          0          Initialization Failures
          555909     Bytes Received
          1108635    Bytes Sent
          7952       Data Blocks Received
          8638       Data Blocks Sent
          0          Data Error Inbound
          0          Data Errors Outbound
          0          Remote Reply Timeouts
          0          Local Reply Timeouts
          0          Remote Buffer Errors
          0          Local Buffer Errors
          0          Selection Timeouts

        Circuit = DMR-1

          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 Downs
          0          Initialization Failures
          0          Bytes Received
          0          Bytes Sent
          0          Data Blocks Received
          0          Data Blocks Sent
          0          Data Error Inbound
          0          Data Errors Outbound
          0          Remote Reply Timeouts
          0          Local Reply Timeouts
          0          Remote Buffer Errors
          0          Local Buffer Errors
```

```
 6:34:16   NCP

           Circuit = DMR-2

             0          Terminating Packets Received
             0          Originating Packets Sent
             0          Terminating Congestion Loss
             92         Transit Packets Received
             94         Transit Packets Sent
             0          Transit Congestion Loss
             1          Circuit Downs
             1          Initialization Failures
             216929     Bytes Received
             215067     Bytes Sent
             969        Data Blocks Received
             967        Data Blocks Sent
             0          Data Error Inbound
             0          Data Errors Outbound
             0          Remote Reply Timeouts
             0          Local Reply Timeouts
             0          Remote Buffer Errors
             0          Local Buffer Errors
             0          Selection Timeouts

                           * * *

NCP>sh cir dmr-0 cou(RET)
NCP>
 6:34:53   NCP
           Request # 263 Accepted
NCP>
 6:34:54   NCP
           Request # 263; Show Circuit Counters Completed

           Circuit = DMR-0

             0          Terminating Packets Received
             0          Originating Packets Sent
             0          Terminating Congestion Loss
             5893       Transit Packets Received
             5406       Transit Packets Sent
             0          Transit Congestion Loss
             1          Circuit Downs
             0          Initialization Failures
             555909     Bytes Received
             1108635    Bytes Sent
             7952       Data Blocks Received
             8638       Data Blocks Sent
             0          Data Error Inbound
             0          Data Errors Outbound
             0          Remote Reply Timeouts
             0          Local Reply Timeouts
             0          Remote Buffer Errors
             0          Local Buffer Errors
             0          Selection Timeouts
NCP>sh ex(RET)
NCP>
 6:35:19   NCP
           Request # 264 Accepted
```

```
NCP>
 6:35:20   NCP
           Request # 264; Show Executor Node Summary Completed

           Executor Node = 123 (D2137A)

             State = On
             Active Links = 1
NCP>cl ex nod(RET)
NCP>
 6:35:38   NCP
           Clear Executor Complete
NCP>sh ex(RET)
NCP>
 6:35:42   NCP
           Request # 265; Show Executor Node Summary Completed

           Executor Node = 113 (KL2137)

             Identification = DECnet-20 V3.0.0
             State = On
             Active Links = 0

                          * * *

NCP>tell eubie shOW exECUTOR chARACTERISTICS(RET)
NCP>
13:30:40   NCP
           Request # 276 Accepted
NCP>
13:30:52   NCP
           Request # 276; Show Executor Node Characteristics
Completed

           Executor Node = 118 (EUBIE)

             Identification = EUBIE
             Management Version = 3.0.0
             Incoming Timer = 60
             Outgoing Timer = 60
             NSP Version = 3.2.0
             Maximum Links = 16
             Delay Factor = 64
             Delay Weight = 2
             Inactivity Timer = 60
             Retransmit Factor = 10
             Routing Version = 1.3.0
             Type = Routing
             Routing Timer = 600
             Maximum Address = 255
             Maximum Circuits = 3
             Maximum Cost = 1022
             Maximum Hops = 11
             Maximum Visits = 16
             Maximum Buffers = 28
             Buffer Size = 576
             Parameter #2731 = 3
             Parameter #2740 = 0
             Parameter #2751 = 0
NCP>exit(RET)
```

* * *

```
@ena (RET)
$opr (RET)
OPR>disa out all (RET)
OPR>
 16:40:57  --Output display for OPR modified--
OPR>ent ncp (RET)
NCP>set ex nod d2137a (RET)
NCP>
 16:41:07  NCP
           Set Executor Complete
NCP>sho act nod char (RET)
NCP>
16:41:51   NCP
           Request # 307 Accepted
NCP>
16:42:00   NCP
           Request # 307; Show Active Nodes Characteristics
Completed

           Executor Node = 123 (D2137A)

             Identification = DECnet-20 V3.0.0
             Management Version = 3.0.0
             Host = 113 (KL2137)
             Loop Count = 1
             Loop Length = 127
             Loop With = Mixed
             Incoming Timer = 10
             Outgoing Timer = 30
             NSP Version = 3.2.0
             Maximum Links = 8
             Delay Factor = 2
             Delay Weight = 3
             Inactivity Timer = 30
             Retransmit Factor = 5
             Routing Version = 1.3.0
             Type = Routing
             Routing Timer = 60
             Maximum Address = 255
             Maximum Circuits = 8
             Maximum Cost = 100
             Maximum Hops = 16
             Maximum Visits = 32
             Maximum Buffers = 66
             Buffer Size = 576

           Remote Node = 113 (KL2137)

             No Information

           Remote Node = 118 ()

             No Information

NCP>ex (RET)
$
```

Node EUBIE in the example above is a DECnet-VAX node running Network Management Version 3.0. The KL2136 (a DECnet-20 node) is the command node in the DECnet-VAX example. The three parameters at the end of the display are specific to VAX, and can not be translated into text by the DECnet-20 system.

Remarks:

SHOW or LIST entity with no information type (carriage return follows plural entity or entity entityid) will default to SUMMARY.

To successfully display STATE, STATUS, SUMMARY, CHARACTERISTICS, or COUNTERS for all DECnet lines and circuits (except the DTE-0-1), you must set the EXECUTOR to the communications front end (DN20). The KL10 recognizes the DTE-0-1 as the path it uses to load and dump the DN20. The characteristic needed to perform a dump or load is SERVICE ENABLED. The SHOW CIRCUIT (or LINE) DTE-0-1 CHARACTERISTICS command, executed with the KL10 as EXECUTOR, displays SERVICE = ENABLED or SERVICE = DISABLED. To receive other characteristics (those related to the front-end's communication with other nodes), the EXECUTOR must be SET to be the front end, or the prefix, TELL DxxxxA, must precede the SHOW command. Note the differences in response in the examples below:

```
NCP>sh ex (RET)
NCP>
 6:39:47   NCP
           Request # 268; Show Executor Node Summary Completed

           Executor Node = 113 (KL2137)

             Identification = DECnet-20 V3.0.0
             State = On
             Active Links = 0
NCP>sh cir dte-0-1 ch (RET)
NCP>
 6:40:06   NCP
           Request # 269; Show Circuit Characteristics Completed

           Circuit = DTE-0-1

             Service = Enabled
NCP>sh cir kdp-0-1 ch (RET)
NCP>
 6:40:50   NCP
           Request # 270; Show Circuit Characteristics Failed,
Unrecognized component

           Circuit = KDP-0-1
NCP>set ex no d2137a (RET)
NCP>
 6:41:51   NCP
           Set Executor Complete
NCP>sh cir dte-0-1 ch (RET)
NCP>
 6:42:03   NCP
           Request # 272 Accepted
NCP>
 6:42:03   NCP
           Request # 272; Show Circuit Characteristics Completed

           Circuit = DTE-0-1

             Service = Enabled
             Cost = 1
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = DTE-0-1
```

```
NCP>sh cir kdp-0-1 ch(RET)
NCP>
 6:42:52   NCP
           Request # 273 Accepted
NCP>
 6:42:52   NCP
           Request # 273; Show Circuit Characteristics Completed

           Circuit = KDP-0-1

             Service = Enabled
             Cost = 4
             Hello Timer = 10
             Listen Timer = 60
             Owner = Node / 123 / D2137A
             Line = KDP-0-1
             Type = DDCMP-DMC
NCP>
```

7.8 CONTROLLING THE NETWORK: TRIGGER, LOAD, DUMP, LOOP COMMANDS

7.8.1 TRIGGER

Command	Entity	Keyword/Argument
TRIGGER	**NODE nodeid**	**[VIA cktid]**
		[[SERVICE] PASSWORD password]
	VIA cktid	**[[SERVICE] PASSWORD password]**

Function:

The TRIGGER command attempts to cause the target node (the node to be loaded) to send a load request to the executor. The executor must be an adjacent node. If the load request is sent successfully, the load proceeds automatically. (See Remarks.)

Arguments:

[[SERVICE] PASSWORD password]

The entire parameter with its value is optional. If included, SERVICE (for clarity) may be included or excluded as an additional keyword.

Remarks:

The VIA cktid, entered in the entity position, is actually a more descriptive phrase for CIRCUIT cktid and serves to identify the entity.

DECnet-20 V3.0 does not support the TRIGGER command for the DN20 or DN200. The command can be used with TELL prefix if the node addressed supports TRIGGER.

7.8.2 LOAD

Command	Entity	Keyword/Argument
LOAD	{NODE nodeid} {VIA cktid }	[ADDRESS nodeaddr] [CPU type] [FROM loadfileid] [HOST nodeid] [NAME nodename] [SECONDARY [LOADER] fileid] [SERVICE CIRCUIT cktid] [SERVICE DEVICE devtype] [[SERVICE] PASSWORD password] [SOFTWARE IDENTIFICATION softwareid] [SOFTWARE TYPE type] [TERTIARY [LOADER] fileid] [VIA cktid] (Not applicable if command is LOAD VIA cktid.)

Function:

The LOAD command (see Remarks for conditions and restrictions) allows one node in the network to load the system image file to a remote node in the network.

NOTE

All parameters and arguments for the LOAD command are optional. This is because the permanent data base may contain values that will be used as defaults by systems supporting a permanent data base. In DECnet-20, the values that will be used are those initial values established as the volatile data base at system startup. You type only those keywords and values that you need to change. (For example, if the circuit normally used to load the specified node is not operational, you type an alternate cktid.) In DECnet-20, any missing parameters will be indicated in an error message, and can be set at that time.

Arguments:

VIA cktid

Identifies the circuit over which to load.

ADDRESS nodeaddr

Nodeaddr is the address the target node is to use when it initializes.

CPU type

This is the target CPU type. See Section 7.3.3 for predefined arguments.

FROM loadfileid

Identifies the image file to be loaded.

HOST nodeid

Sends the target node the nodeid of the host node that will provide services (such as a file system) for the target node once it is loaded. Note that this may or may not be the same node as the executor. If HOST nodeid is not specified, the executor is assumed to be the host node of the target node.

NAME nodename

Specifies the name of the target node.

SECONDARY [LOADER] fileid

Identifies the secondary loader file, that is, the file to be requested by the primary loader (contained in the target bootstrap).

SERVICE DEVICE devtype

See Appendix D for mneumonics (KDP, DMC, and others).

SERVICE PASSWORD
SOFTWARE IDENTIFICATION

See Section 7.5.4, SET NODE nodeid command for these two keywords.

SOFTWARE TYPE

See Section 7.3.3.

TERTIARY LOADER fileid

Identifies the tertiary loader file (requested by the secondary loader file) for the target node.

Examples:

See Section 3.1 for examples.

Remarks:

See Section 3.1 for loading procedure.

7.8.3 DUMP

Command	Entity	Keyword/Argument
DUMP	**NODE nodeid**	**[VIA cktid]**
	[DUMP COUNT number]	
		[SECONDARY [DUMPER] fileid]
		[SERVICE CIRCUIT cktid]
		[SERVICE DEVICE devtype]
		[[SERVICE] PASSWORD password]
		[TO dumpfileid]

Function:

Stores a copy of the target's memory image in a dump file at the host node. For DECnet-20, the host node is the KL10 for dumping both the DN20 and the DN200 (if one exists). The KL10 also acts as executor for the DN20 dump. To dump the DN200, the executor must be set to be the DN20.

Arguments:

VIA cktid

Identifies the circuit to be used for the dump.

TO dumpfileid

Identifies the host's file that will receive the target's dump.

All other parameters and arguments are described in Section 7.5.3, the SET/DEFINE NODE nodeid command.

Example:

```
NCP>dump node d2137a
```

Remarks:

This command is not needed to obtain a dump when the DN20 is automatically reloaded. The dump will also be automatic. However, there may be times you wish to dump the MCB front end when an automatic DUMP/LOAD procedure is not indicated. It is sufficient to provide file identification. The file is not required to exist before the command is given.

Before dumping the DN20, execute the NCP command SET CIRCUIT DTE-0-1 SERVICE ENABLED, if necessary. (The circuit you enable is the circuit connecting the Host and Target node.)

Command	Entity	Keyword/Argument
DUMP	**VIA cktid**	**[[DUMP] ADDRESS number]** **[[DUMP] COUNT number]** **[SECONDARY [DUMPER] fileid]** **[SERVICE DEVICE devtype]** **[[SERVICE] PASSWORD password]** **[TO dumpfileid]**

Function and arguments as for DUMP NODE nodeid.

Remarks:

DECnet-20 should not require this command. The DTE-0-1 is set to be the SERVICE CIRCUIT when initial values are read from a command file at system startup.

The VIA cktid parameter and argument in the entity position is simply a more descriptive way of indicating the entity circuit.

DECnet-20 V3.0 software normally requires no parameters, that is, the commands shown in the examples will be executed under normal conditions. The node data base for the Host node will contain the needed dump parameter values when the system is brought up. If a needed parameter value is missing, the NCP response will name the missing parameter. Use the SET NODE command to enter the missing parameter and repeat the DUMP NODE command.

7.8.4 LOOP

Command	Entity	Keyword/Argument	
LOOP	NODE nodeid EXECUTOR CIRCUIT cktid LINE lineid	[USER userid] [ACCOUNT account] [PASSWORD password]	[COUNT count] [LENGTH length] [WITH ONES] ZEROS] MIXED]

Function:

This command requests a node level loopback test. The node specified is the node to be connected TO; that is, the node that will loop back the data. This may be the local node or a remote node.

Arguments:

Access information (USER, ACCOUNT, and PASSWORD) refers to these values for the node named in NODE nodeid. If node is remote, the conventions of the remote node are used.

COUNT count

This parameter is the number of times the test data is to be looped.

LENGTH length

This parameter is the length of the test data in bytes.

WITH ONES
WITH ZEROS
WITH MIXED

This parameter specifies the content of the test data.

Examples:

Refer to Section 3.7.1 for several examples.

Remarks:

LOOP NODE (and LOOP CIRCUIT) are described in detail in Section 3.7.1.

If any of the three access parameters (USER, ACCOUNT, or PASSWORD) are typed, they must precede other optional parameters as shown.

7.9 MONITORING NODE ACTIVITY: ZERO COMMAND

Command	Entity	Keyword/Argument
ZERO	CIRCUIT cktid	COUNTERS
	KNOWN CIRCUITS	
	LINE lineid	
	KNOWN LINES	
	NODE nodeid	
	KNOWN NODES	

Function:

The ZERO command generates a counters zeroed event that causes counters to be logged. The counters are then set to zero. The counters affected are those associated with the specified entity.

Arguments:

None - counters are considered as an aggregate. Individual text items, such as line down for line counters can not be addressed.

Examples:

```
NCP>set ex nod d2137a (RET)
NCP>
11:27:30   NCP
           Set Executor Complete
NCP>sh circuit dte-0-1 counters (RET)
NCP>
11:27:57   NCP
           Request # 223 Accepted
NCP>
11:27:58   NCP
           Request # 223; Show Circuit Counters Completed

           Circuit = DTE-0-1

             3213       Terminating Packets Received
             3980       Originating Packets Sent
             1          Terminating Congestion Loss
             24968      Transit Packets Received
             23528      Transit Packets Sent
             0          Transit Congestion Loss
             0          Circuit Downs
             0          Initialization Failures
             1712739    Bytes Received
             750241     Bytes Sent
             26669      Data Blocks Received
             27510      Data Blocks Sent
NCP>zero circuit dte-0-1 counters (RET)
NCP>
11:28:58   NCP
           Request # 224 Accepted
NCP>
11:28:58   NCP
           Request # 224; Zero Circuit Failed,
Privilege violation

NCP>set ex nod d2137a user operator password d2137a (RET)
NCP>
11:29:52   NCP
           Set Executor Complete
```

```
NCP>zero circuit dte-0-1 counters (RET)
NCP>
11:30:17   NCP
           Request # 225 Accepted
NCP>
11:30:22   NCP
           Request # 225; Zero Circuit Completed
NCP>sh circuit dte-0-1 counters (RET)
NCP>
11:30:47   NCP
           Request # 226 Accepted
NCP>
11:30:48   NCP
           Request # 226; Show Circuit Counters Completed

           Circuit = DTE-0-1

             19          Terminating Packets Received
             23          Originating Packets Sent
             0           Terminating Congestion Loss
             9           Transit Packets Received
             10          Transit Packets Sent
             0           Transit Congestion Loss
             0           Circuit Downs
             0           Initialization Failures
             1691        Bytes Received
             948         Bytes Sent
             28          Data Blocks Received
             33          Data Blocks Sent
NCP>
```

Remarks:

The example given illustrates the effect of failure to include USER and PASSWORD for SET commands addressed to the MCB front end.

7.10 NUMERICALLY CODED FIELDS

All entities, parameters, counters, and events are assigned a "Type Number" by the NICE Protocol. Before data items are returned to you in response to your NCP requests, or before events are sent to your system console (CTY), these numbers are translated into meaningful text if they are recognized by the local NML. (All responses are returned to the command node.) Not all DECnet implementations recognize all parameters. When a system does not recognize a parameter, the response output to your terminal, or the event message typed on your CTY, contains the parameter type number rather than the appropriate text.

Given below are two examples of the same event message sent to a CTY and to the ERROR.SYS file. In the first example, the receiving node recognizes the numeric codes and substitutes the appropriate text. In the second example, the receiving NML does not have the necessary conversion data in a data base. Therefore, the type numbers remain. To translate a response or message containing these coded fields, you need at least one set of tables, and possibly two sets. Appendix A contains tables of most node, line, circuit, logging, and event parameters with their parameter type numbers. The tables also contain coded values for keywords, the applicability of the parameter, the restrictions related to the use of the parameter, and the operating system (TOPS-20 or MCB) that executes the commands containing that parameter. System-specific parameters are not included. To translate system-specific parameters, you need access to the documentation of the system that sent you the response or event message. The method of translation is the same. The message below is translated using Table 7-3 (line 1) and Table A-2 (line 3).

Example #1. Recognized type numbers.

```
Event type 0.3, Automatic data link service
From node 21 (VENUS), occurred 29-JUL-81 13:05:10.002
Line KDP-0-1, Service = Load, Status = Requested
```

Example #2. Unrecognized type numbers.

```
Event type 0.3
From node 21 (VENUS), occurred 29-JUL-81 13:05:10.002
Line KDP-0-1, Parameter #0 = 0, Parameter #1 = 0
```

CHAPTER 8

NETWORK MANAGEMENT PACKET SWITCHING INTERFACE

If the PSI Version 1.0 software option is included in your TOPS-20 system, your DECnet network supports the X.25 facility for using Public Packet Switching Networks (PPSNs). You can use the X.25 NCP commands and parameters.

If the PSI software option is not included in your TOPS-20 system, your DECnet network may include nodes of DIGITAL systems that support the X.25 facility for using Public Packet Switching Networks (PPSNs). The Network Management modules of DECnet-20 V3.0 understand the syntax of X.25-related commands and parameters. You can, therefore, by using the TELL nodeid prefix or the SET EXECUTOR NODE nodeid command, direct X.25-related commands to a Phase III (or higher) DECnet node that supports a Packetnet System Interface (PSI).

A complete list of X.25 NCP commands as specified by the DNA is included in this chapter. As was true of the DECnet utility commands documented in previous chapters, the set of commands implemented, and the way in which they are implemented, is a function of the operating system of each node in the network.

If the PSI software option is not included, as a system manager, you will need to cooperate fully with the system manager of the node to which you will direct the X.25-specific NCP commands with the TELL command-prefix or by SET EXECUTOR to a remote node. In addition to the DECnet manuals mentioned in the introductory part of this manual, you will need to obtain complete documentation on the PSIs of all nodes to which you will communicate in this manner. In addition to the documentation, there may be parameter values, schedules, operating procedures, and command files that the manager of the PSI-node will provide. This information must be available to those operators who will use the commands.

If the PSI software option is not included, as an operator given the responsibility for using the X.25-specific commands, be aware that this chapter is for reference only, it is beyond the scope of this manual to provide a tutorial on all X.25 implementations. If there is any question about the values to be entered for specific parameters, consult your system manager or immediate supervisor if you are an operator. The system manager should read the X.25 documentation for systems in the network that support X.25.

NOTE

See Appendix A for X.25 parameters not implemented for the DECnet-20 PSI Version 1.0 software option.

8.1 THE MODULE ENTITY

You have already been introduced to four possible entities (controllable objects) that are addressable in Phase III DECnet: NODES, LINES, CIRCUITS, and LOGGING. With the addition of the X.25 facility, another entity is provided. This is the MODULE entity. The MODULE entity takes one of three forms: MODULE X25-ACCESS, MODULE X25-PROTOCOL, and MODULE X25-SERVER. The general nature of the information contained in the data bases associated with each is as follows:

MODULE X25-ACCESS

The access module data base contains the information needed to connect to one or more PPSNs. The information is indexed by network name. NCP commands that reference this data base must indicate the name of the network to which the command applies unless there is only one name in the data base.

MODULE X25-PROTOCOL

The protocol module data base contains the information needed to maintain switched virtual circuits (SVCs) and permanent virtual circuits (PVCs) through the public data network.

Much of the SVC information is indexed by the local Data Termination Equipment (DTE) address. This address is assigned by the PPSN at the time the line is installed. You must include this address whenever it is a parameter in an NCP command, unless it is the only DTE address in the data base.

MODULE X25-SERVER

The server module data base contains the information needed to map incoming calls to a DECnet process. This information is indexed by destination name. You must associate each destination name with a DECnet node and process identification and with the necessary X.25-related information to recognize the incoming call. NCP commands that reference this data base must indicate to which destination name they apply unless there is only one such name defined in the data base.

8.2 X.25-SPECIFIC COMMANDS

NOTE

SET, CLEAR and SHOW refer to parameters in the volatile data base; DEFINE, PURGE, and LIST refer to parameters in the permanent data base. These three pairs are considered together in the detailed descriptions that follow. All other commands are described in separate charts. Remember that neither TOPS-20 nor the MCB supports a permanent data base. Therefore, only SET, CLEAR, and SHOW are used for DECnet-20 PSI. All commands are given because other operating systems may implement a permanent data base.

Command	Entity	Keyword/Argument
{SET / DEFINE}	{CIRCUIT cktid / KNOWN CIRCUITS}	MAXIMUM BLOCK bytecnt MAXIMUM WINDOW blockcnt USAGE usagtyp BLOCKING blockcntl NUMBER callnum MAXIMUM RECALLS recallcnt RECALL TIMER seconds CHANNEL chnlnum DTE dteaddr TYPE X25

In addition to the parameters that are X.25-specific, the following parameters are common to both DDCMP and X.25 circuits: COUNTER TIMER, OWNER, and STATE. See Chapter 7 for definitions. (Definitions are the same except that there is no SERVICE value permitted for the STATE parameter in X.25 because X.25 circuits do not support any of the Network Management service functions.)

NOTE

TOPS-20 PSI circuits are not owned by the EXECUTOR, and the parameters are set at network generation time (see the TOPS-20 PSI Installation Guide). Thus, you cannot change the following arguments with a SET/DEFINE command:

MAXIMUM BLOCK
MAXIMUM WINDOW
USAGE PERMANENT
CHANNEL
DTE
TYPE X25

Only circuits of USAGE permanent are supported.

Function:

These commands set volatile and permanent circuit parameters, respectively, for the circuit(s) identified by cktid.

Arguments:

MAXIMUM BLOCK bytecnt

For PVCs: the data link maximum X.25 block size permitted. For SVCs owned by the executor node: the block size that transport is to request from the X.25 protocol handler module. Bytecnt is an integer in the range 1-65535. It must be less than or equal to the maximum block size allowed within the X.25 protocol handler.

MAXIMUM WINDOW blockcnt

For PVCs: the data link maximum number of X.25 blocks allowed to exist with outstanding acknowledgements. For SVCs owned by the executor node; the block count that transport is to request from X.25 for the circuit. Blockcnt is an integer in the range 1-255.

USAGE usagtyp

The value of usagtyp is one of the following:

INCOMING

This value is used for SVC incoming calls only and only for circuits owned by the executor node.

OUTGOING

This value is used for SVC outgoing calls only and only for circuits owned by the executor node.

PERMANENT

This value is used for all PVCs (circuits permanently connected to the same remote station).

BLOCKING blockcntl

This parameter applies to circuits owned by the executor node. The value you enter determines whether or not messages will be blocked before being sent over the circuit. The possible values for blockcntl are:

ENABLED: Perform blocking
DISABLED: No blocking

NUMBER callnum

This parameter applies to outgoing X.25 circuits owned by the executor node. The value you enter in callnum is the full remote DTE address used to call out on the circuit. The address value must be an integer of one to sixteen digits.

MAXIMUM RECALLS recallcnt

This parameter applies to outgoing X.25 circuits that are owned by the executor node. The value you enter determines the number of automatic call retries. Recallcnt is an integer in the range 0-255. If no value is set, there is no maximum.

RECALL TIMER seconds

This parameter applies to outgoing X.25 circuits that are owned by the executor node. The value you enter determines the number of seconds between automatic retries. Seconds is an integer in the range 1-65535. If no value is set, there is no wait.

CHANNEL chnlnum

This parameter applies to PVCs only. The value that you enter for chnlnum is the channel number to be used in running the X.25 protocol on the circuit. The channel number is an integer in the range 0-4095.

DTE dteaddr

This parameter applies to PVCs only. The value you enter is the data link X.25 local DTE address to which the circuit belongs. DTE address is an integer from one to sixteen digits.

TYPE X25

For X.25 circuits, this parameter must be entered as TYPE X25. If this entry is not made, the circuit will not be useable.

Command	Entity	Keyword/Argument
{SET / DEFINE}	**MODULE X25-ACCESS**	{**ALL network-qualifier** / **network-options** / **network-qualifier**}

NOTE

The use of SET MODULE X25-ACCESS ALL is meaningless if there is no permanent data base. Check the PSI documentation for the DIGITAL system that will execute the command. If there is a permanent data base, this SET command will move all permanent X.25 ACCESS module data base values to the volatile data base. DEFINE MODULE X25-ACCESS ALL is never meaningful with the plural entity KNOWN NETWORKS. DEFINE MODULE X25-ACCESS ALL with NODE nodeid is currently undefined.

Function:

The SET and DEFINE MODULE X25-ACCESS command sets volatile and permanent data base parameters, respectively, needed to connect to the X.25 server for one or more networks. See Appendix A for a list of parameters supported by TOPS-20 PSI.

Arguments:

network-qualifier

Network-qualifier indicates the network name to which the command applies. If only one network is known, it is the default. Network name is the name of the PPSN: TELENET, TRANSPAC, or PSS, for example. The possible values for network-qualifier are:

KNOWN NETWORKS
NETWORK network-name

network-options

Network options may be one or more of the following:

ACCOUNT account

The value of the access control account field to be used by the access routines in connecting to the PSI gateway. This value is used if the user does not supply an account value when calling the access routines (see the TOPS-20 PSI User's Guide for more information on access routines). If no account is set, none is included in the access control on connect by the access routines. Account is a string of one to thirty-nine characters.

NODE node-id

The identification of the node containing the X.25 gateway module that gives access to the associated PPSN. Node-id is a standard Network Management node identification.

PASSWORD password

The value of the access control password field to be used by the access routines in connecting to the PSI gateway. This value is used if the user does not supply a password when calling the access routines (see the TOPS-20 PSI User's Guide for more information on access routines). If no password is set, none is included in the access control on connect by the access routines. Password is a string of one to thirty-nine characters.

USER user

The value of the access control user field to be used by the access routines in connecting to the PSI gateway. This value is used if the user does not supply a user identification when calling the access routine (see the TOPS-20 PSI User's Guide for more information on access routines). If no user is set, none is included in the access control on connect by the access routines. User is a string of one to thirty-nine characters.

NOTE

ACCOUNT and USER are not used in DECnet-20 V3.0 implementation.

Examples:

```
NCP>set module x25-access node mrx25:: network telenet (RET)
NCP>
14:59:36        NCP
                Request # 1; Set Module Completed
NCP>set module x25-access password x25-gate (RET)
NCP>
15:00:42        NCP
                Request # 2; Set Module Completed
```

Command	Entity	Keyword/Argument
{SET / DEFINE}	MODULE X25-PROTOCOL	{ALL dte-qualifier / group-qualifier / GROUP group-name / group-options / CALL TIMER seconds / CLEAR TIMER seconds / DEFAULT DATA bytecnt / DEFAULT WINDOW blockcnt / MAXIMUM DATA bytecnt / MAXIMUM CLEARS retrycnt / MAXIMUM RESETS retrycnt / MAXIMUM RESTARTS retrycnt / MAXIMUM WINDOW blockcnt / NETWORK network-name / RESET TIMER seconds / RESTART TIMER seconds / dte-options dte-qualifier}

Functions:

The SET and DEFINE MODULE X25-PROTOCOL commands control the parameters necessary to maintain switched and permanent virtual circuits through a public packet switching network over the assigned X.25 lines of the protocol module.

Arguments:

ALL

Refer to NOTE for SET/DEFINE MODULE X25-ACCESS.

ALL dte-qualifier

Dte-qualifier indicates the local DTE address to which the command applies. Dte-qualifier may be one of:

KNOWN DTEs
DTE dte-address

If only one DTE address is known to the X.25 protocol module, that address is the default. If more than one DTE address is known, the applicable address must be given.

ALL group-qualifier

Group-qualifier indicates the closed user group or bilateral closed user group to which the command applies. It may be one of the following:

KNOWN GROUPS
GROUP group-name

Group name is an ASCII string of one to sixteen characters. It may have any value decided upon by the managers of the communicating systems. (See the TOPS-20 PSI User's Guide for a description of the user groups.)

ALL has the same interpretation already defined.

GROUP group-name group-options

Group-name is defined under the previous argument. Group-options is both of the following:

DTE dte-address
NUMBER group-number
TYPE group-type

Dte-address

The PPSN DTE address associated with the group-name.

Group-number

This is the closed user group number (defined by the PPSN). It must be an integer in the range 0-9999.

Group-type

If specified, the only value permitted is BILATERAL.

CALL TIMER seconds

The value you set in seconds determines the elapsed time between receiving no response on an outgoing call from the DTE and the sending of a clear packet. (The sending of the clear packet is transparent to the user.)

CLEAR TIMER seconds

The value you enter for seconds determines the elapsed time between retransmissions of clear packets. Seconds is an integer in the range 1-255. If no value is set, there is no retransmission.

DEFAULT DATA bytecnt

This is the default packet size for SVCs. The value you enter must be an integer in the range 1-65535, which is used when there is no negotiation for a packet size (see the TOPS-20 PSI User's Guide for more information on the flow control parameter negotiation facility).

DEFAULT WINDOW blockcnt

This is the number of unacknowledged transmitted blocks on an SVC that may be sent before an acknowledgment is received. Once you set this value, it remains in effect until changed. The value that you enter must be an integer in the range 1-255, which is used when there is no negotiation for a window size (see the TOPS-20 PSI User's Guide for more information on the flow control parameter negotiation facility).

MAXIMUM DATA bytecnt

The value that you enter for this parameter establishes the maximum packet size for all X.25 circuits. Bytecnt must be an integer in the range 1-65535, which is the maximum value you can specify when negotiating for a packet size (see the TOPS-20 PSI User's Guide for more information on the flow control parameter negotiation facility).

MAXIMUM CLEARS retrycnt

The value that you enter for this parameter determines the maximum number of times that the X.25 protocol handler is to retry the sending of a clear packet for SVCs. The value must be an integer in the range 1-255. If you do not set a value, there is no maximum.

MAXIMUM RESETS retrycnt

As for preceding parameter, but sets the maximum times for retrying the sending of a reset packet. If you do not set this value, there is no maximum. Retrycnt is an integer in the range 1-255.

MAXIMUM RESTARTS retrycnt

The value you enter for retrycnt determines the maximum number of times that the X.25 protocol handler retries the sending of a restart. If you do not set this value, there is no maximum. Retrycnt is an integer in the range 1-255.

MAXIMUM WINDOW blockcnt

The value you enter in blockcnt determines the maximum number of unacknowledged transmitted blocks permitted on an SVC. The value of blockcnt must be an integer in the range 1-255, which is the maximum value you can specify when negotiating for a window size (see the TOPS-20 PSI User's Guide for more information on the flow control parameter negotiation facility).

NETWORK network-name

Previously defined.

RESET TIMER seconds

The value you enter for seconds determines the number of elapsed seconds between a transmission and a retransmission of an unacknowledged reset packet. Seconds must be an integer in the range 1-255. If no value is set, there is no retransmission.

RESTART TIMER seconds

The value you enter for this parameter determines the elapsed seconds between transmission and retransmission of an outgoing restart packet from the local DTE. Seconds must be an integer in the range 1-255. If no value is set, there is no retransmission.

dte-options

DTE-options, indexed by the local DTE address, include the following:

COUNTER TIMER seconds

This is a Network Management timer. The value that you enter determines the time interval between successive recordings and zeroing of protocol module counters. If no value is set, there is no automatic logging of counters.

CHANNELS list

List is one or more logical channel numbers that can be used for outgoing calls or possibly taken by incoming calls. If more than one channel number is specified, separate by commas.

LINE lineid

Identification of the line associated with the DTE address. It cannot be set or defined by a Network Management command.

STATE dte-state

The value you enter must be one of:

ON - DTE is allowed to operate normally
OFF - DTE is not operating; all virtual circuits terminated immediately
SHUT - local DTE will not allow new virtual circuits; existing virtual circuits continue

Examples:

```
NCP>set executor node mrx25 (RET)
NCP>
15:01:24        NCP
                Set Executor Complete
NCP>set module x25-protocol maximum data 128 (RET)
NCP>
15:06:56        NCP
                Request # 8 Accepted
NCP>
15:06:57        NCP
                Request # 8; Set Module Completed
NCP>set module x25-protocol channels 10-1 dte 311030301234 (RET)
NCP>
15:08:15        NCP
                Request # 12 Accepted
NCP>
15:08:16        NCP
                Request # 12; Set Module Completed
```

Command	Entity	Keyword/Argument
{SET / DEFINE}	MODULE X25-SERVER	{ALL destination-qualifier / COUNTER TIMER seconds / MAXIMUM CIRCUITS count / destination-options destination-qualifier}

Function:

The SET/DEFINE MODULE X25-SERVER commands control the parameters related to the server module data base. Some parameter values entered provide the information needed to map incoming X.25 calls to a DECnet process; other parameters are independent of the destination name but are needed for more general functions related to incoming calls, for example, event recording.

Arguments:

ALL

ALL has the previously defined meaning, related to the server module data base.

destination-qualifier

Indicates the destination to which the command applies. May be one of:

KNOWN DESTINATIONS - all destinations known to the server module DESTINATION destination-name

destination-name

The name of a specific destination. Name is an ASCII string of from one to sixteen characters. It is the responsibility of the system manager to assign destination names for input to his node. Because such names must be unique, this will involve cooperation between managers of communicating nodes.

COUNTER TIMER seconds

This counter timer is the Network Management timer that determines the elapsed time between the logging of a server module logging event. Seconds is an integer in the range 1-65535.

MAXIMUM CIRCUITS count

The value you enter for count indicates the number of circuits that the server module can have open at one time. Count must be an integer in the range 1-65535.

destination-options

You can add new destination names at the initial reference to the destination name in an NCP SET command. You can remove old names from the data base using the CLEAR command. Parameters of an existing destination can be modified. The parameters accessed by destination name are described below. These parameters are divided into two groups:

- Parameters that are applied to a field in the incoming call packet for the purpose of determining a match.
- Parameters that define the DECnet characteristics necessary to initiate a logical link connect to a DECnet object.

The parameters that are applied to a field of the incoming call packet are:

CALL MASK hexadecimal string

This is a string of up to 32 hexadecimal characters that is used to determine the destination of the incoming call based on the content of the user call data field of the incoming call packet. This value is ANDed byte-by-byte with the user call data field of the incoming call packet. The resultant string is compared with the CALL VALUE.

CALL VALUE — hexadecimal string

This is a string of up to 32 hexadecimal characters that is used to determine the destination of an incoming call based on the content of the user call data field of the incoming call packet. This value is compared byte-by-byte with the string resulting from the CALL MASK operation described above to determine the destination of an incoming call.

GROUP — ascii-string

This is the closed user group or bilateral closed user group name used to determine the destination of an incoming call based on user group membership. If given, only incoming calls with this user group name will be routed to this destination.

NUMBER — ascii-string

This is the full remote DTE address used to determine the destination of an incoming call based on the calling DTE address in the incoming call packet. This value must be a string of digits and/or asterisks (*) from one to sixteen characters.

PRIORITY — decimal number

This is the priority associated with a given entry in the DESTINATION data base. The highest priority is 255 and the lowest is 0. If an incoming call maps to more than one destination, the one with the highest priority is chosen. If there is more than one match with equal priority, the first is chosen.

SUBADDRESSES — range

This is the range of the local DTE subaddresses used to identify the destination of an incoming call. If given, only those calls with the called DTE subaddresses in the specified range will be routed to this destination. A subaddress is a decimal number from 0-999.

Examples:

```
NCP>set executor node mrx25 (RET)
NCP>
15:01:24        NCP
                Set Executor Complete
NCP>set module x25-server call mask 01 destination x29srv (RET)
NCP>
15:12:19        NCP
                Request # 17 Accepted
NCP>
15:12:39        NCP
                Request # 17; Set Module Completed
NCP>set module x25-server call value 01 destination x29srv (RET)
NCP>
15:12:19        NCP
                Request # 18 Accepted
NCP>
15:13:00        NCP
                Request # 18; Set Module Completed
NCP>set module x25-server object 34 destination x29srv (RET)
NCP>
15:12:19        NCP
                Request # 19 Accepted
NCP>
15:13:01        NCP
                Request # 19; Set Module Completed
```

The following parameters are necessary to initiate a logial link connection to a DECnet destination object:

ACCOUNT ascii-string

This is the DECnet access control account string used when the Server module connects to the destination of an incoming call.

NODE nodeid

This is the standard Network Management node name used when the Server module connects to the destination of an incoming call. This is a required parameter.

OBJECT objectid

This is the DECnet object identification used when the Server module connects to the destination of an incoming call. The value objectid can be one of the following:

An ASCII object name or numeric object name in quotes, of 1 to 16 characters

An object number, from 1 to 255

PASSWORD ascii-string

This is the DECnet access control password used when the Server module connects to the destination of an incoming call; must be a string of 1 to 39 characters.

USER ascii-string

This is the DECnet access control user identification to be used when the Server module connects to the destination of an incoming call.

Remarks:

To determine the destination for an incoming call, the information in the incoming call packet is compared with the CALL MASK, CALL VALUE, GROUP, NUMBER, and SUBADDRESSES parameters in each of the DESTINATION entries to see if there is a match. If there is a match, the ACCOUNT, NODE, OBJECT, PASSWORD, and USER parameters are used to initiate a logical link connect. If there is more than one match, the match with the highest PRIORITY is taken. If multiple matches have the same PRIORITY, the first entry in the data base is used.

If any of the fields used to identify the DECnet destination of an incoming call are missing, then no values are used for the missing fields in the identification process. It is the responsibility of the receiving node to take care of the distribution of messages with missing values in these fields (destination-options).

A match exists only if all the conditions described below are true:

1. The user call data field of the incoming packet, logically ANDed byte-by-byte with the CALL MASK, equals the CALL VALUE. If the user call data field is longer than the CALL MASK, then the strings need only be equal to the end of CALL MASK. If the CALL MASK is longer than the user call data field, a match can only be obtained if all bytes in the CALL MASK beyond the length of the user call data field are zero.

2. The GROUP parameter equals the closed user group from the incoming call packet, or no GROUP parameter is defined. If a GROUP parameter is defined and there is no group present in the incoming call packet, there is no match.

3. The NUMBER parameter matches the calling DTE address in the incoming call packet. The match is determined by scanning both ASCII strings from left to right. The NUMBER parameter can be any combination of ASCII characters from 0 to 9 and the wild-card character "*". A match exists if the following is true:

   - There is no NUMBER parameter defined; or
   - Up to the length of the shorter string, each character in the NUMBER parameter string contains an "*" or the same decimal digit as the DTE address string; and

     a. If the NUMBER parameter string is the shorter of the two strings, and the last character in the NUMBER parameter string is an asterisk; or

     b. If the NUMBER parameter string is the longer of the two strings, the excess characters are all considered to be asterisks.

   - If there is no calling DTE address, then a match can occur only if the NUMBER parameter string consists of one or more asterisks.

4. The local DTE subaddress from the incoming call packet is in the range defined by SUBADDRESSES, or there is no SUBADDRESSES range defined.

For a job to receive incoming calls, it must set itself up as a target task. The procedure for doing this is described in the DECnet-20 User's Guide. Below are some examples showing the relation between the TOPS-20 target task file descriptor and the DESTINATION parameters.

You can specify two types of target tasks. The example below shows these and the values of the DESTINATION parameters required to connect to the target.

1. Specify a numbered object in the TOPS-20 program: If the user program specifies a target of the form SRV:128, set the DESTINATION parameters:

```
OBJECT  :  128
NODE    :  TOPS-20 node number
```

2. Specify a named object in the TOPS-20 program: If the user program specifies a target of the form SRV:.TASK0, set the DESTINATION parameters:

```
OBJECT  :  TASK0
NODE    :  TOPS-20 node number
```

   The parameters USER, PASSWORD, and ACCOUNT are optional. If the user program requires those parameters, the system manager must set them.

Command	Entity	Keyword/Argument
{SET / DEFINE}	{NODE nodeid / KNOWN NODES / EXECUTOR}	See Chapter 7 for all applicable parameters and values.

Arguments:

There is one additional parameter specific to X.25:

SUBADDRESSES range

Previously defined (see destination-options for SET MODULE X25-SERVER). The subaddresses parameter distinguishes between calls destined for transport and calls destined for users. (This parameter is not implemented for TOPS-20 PSI.)

All DECnet nodes in which Transport will use X.25 circuits must include the SUBADDRESSES parameter.

Command	Entity	Keyword/Entity
{**SET** / **DEFINE**}	**LINE lineid**	Specific to LAPB (X.25 lines):
		HOLDBACK TIMER seconds
		MAXIMUM BLOCK byte-count
		MAXIMUM RETRANSMITS block-count
		MAXIMUM WINDOW block-count
		Common to DDCMP and LAPB(X.25) lines:
		CLOCK clock-mode
		CONTROLLER controller-mode
		COUNTER TIMER seconds
		DEVICE deviceid
		DUPLEX duplex-mode
		PROTOCOL protocol-name
		RETRANSMIT TIMER milliseconds
		SERVICE service-control
		STATE line-state

Arguments:

With the exception of PROTOCOL, keywords and arguments common to DDCMP and LAPB lines are adequately defined in the previous chapter. For PROTOCOL, the protocol name for lines using X.25 protocol must be LAPB.

HOLDBACK TIMER seconds

The value you enter for seconds determines the length of time an acknowledgement is held back to wait for a data message with which it can be included. If you do not specify a value, there is no wait. (This argument is not implemented for TOPS-20 PSI.)

MAXIMUM BLOCK byte-count

The value you enter determines the data link maximum byte count of the block size on the line. Byte count must be an integer in the range 1-65535.

MAXIMUM RETRANSMITS block-count

The value you enter in block-count determines the data link maximum number of retransmissions of a block on the line. Block-count must be an integer in the range 1-255. If you do not set a value, there is no maximum.

MAXIMUM WINDOW block-count

The value you enter for block-count determines the maximum number of unacknowledged transmitted blocks permitted on a line. Block-count must be an integer in the range 1-255.

Example:

```
NCP>tell mrx25 set line kdp-0-0 maximum retransmits 30 (RET)
NCP>
15:33:00        NCP
                Request # 40 Accepted
NCP>
15:33:01        NCP
                Request # 40; Set Line Completed
```

Command	Entity	Keyword/Argument
{SHOW / LIST}	MODULE module-name	Specific to named module; see "Arguments:" for keywords and values of each module name.
	ACTIVE MODULES	CHARACTERISTICS / COUNTERS / STATUS / SUMMARY
	KNOWN MODULES	CHARACTERISTICS / COUNTERS / STATUS / SUMMARY

Function:

The SHOW command displays information from the volatile data base; the LIST command displays information from the permanent data base.

Arguments:

Arguments for the entities ACTIVE MODULES and KNOWN MODULES are defined in Section 7.7. Arguments for named modules are:

Entity	Keyword/Argument	Qualifier
MODULE X25-ACCESS	CHARACTERISTICS	[KNOWN NETWORKS]
	COUNTERS	[NETWORKS network-name]
	STATUS	
	SUMMARY	
MODULE X25-PROTOCOL	CHARACTERISTICS	[DTE dte-address]
	COUNTERS	[KNOWN DTES]
	STATUS	[GROUP group-name]
	SUMMARY	[KNOWN GROUPS]
MODULE X25-SERVER	CHARACTERISTICS	[KNOWN DESTINATIONS]
	COUNTERS	[DESTINATION destination-name]
	STATUS	
	SUMMARY	

All X.25-specific parameters for named modules are previously defined in this chapter.

Examples:

```
NCP>show module x25-access characteristics known networks (RET)
NCP>
15:01:11        NCP
                Request # 3; Show Module Characteristics Completed

                Module = X25-ACCESS


                  Network = TELENET
                  Node = 129 (MRX25)
                  Password = X25-GATE
NCP>set executor node mrx25 (RET)
NCP>
15:01:24        NCP
                Set Executor Complete
NCP>show module x25-protocol characteristics known dtes (RET)
NCP>
15:05:24        NCP
                Request # 6 Accepted
NCP>
15:05:25        NCP
                Request # 6; Show Module Characteristics Completed

                Module = X25-PROTOCOL


                  DTE = 311030300170
                  Line = KDP-0-0
                  Channels = 20-1
                  Maximum Channels = 20
NCP>show module x25-protocol characteristics (RET)
NCP>
15:05:35        NCP
                Request # 7 Accepted
NCP>
15:05:35        NCP
                Request # 7; Show Module Characteristics Completed

                Module = X25-PROTOCOL


                  Network = TELENET
                  Default Data = 128
                  Default Window = 2
                  Maximum Data = 256
                  Maximum Window = 2
                  Maximum Clears = 6
                  Maximum Resets = 6
                  Maximum Restarts = 6
                  Call Timer = 180
                  Clear Timer = 180
                  Reset Timer = 180
                  Restart Timer = 180
NCP>show module x25-protocol counters known dtes (RET)
NCP>
15:14:17        NCP
                Request # 24 Accepted
```

```
NCP>
15:14:18        NCP
                Request # 24; Show Module Counters Completed

                Module = X25-PROTOCOL


                  DTE = 311061700084
                  0             Seconds Since Last Zeroed
                  25783         Bytes Received
                  872951        Bytes Sent
                  17029         Data Blocks Received
                  28397         Data Blocks Sent
                  69            Calls Received
                  23            Calls Sent
                  0             Fast Selects Received
                  0             Fast Selects Sent
                  12            Maximum Switched Circuits Active
                  4             Maximum Channels Active
                  0             Received Call Resource Errors
                  8             Locally Initiated Resets
                  2             Network Initiated Resets
                  0             Remotely Initiated Resets
                  1             Restarts
NCP>show module x25-server characteristics known destination (RET)
NCP>
15:16:12        NCP
                Request # 25 Accepted
NCP>
15:16:15        NCP
                Request # 25; Show Module Characteristics Completed

                Module = X25-SERVER


                  Destination = X29
                  Node = 0 (120)
                  Object = 34 ()
                  Priority = 1
                  Call Mask = 01
                  Call Value = 01

                  Destination = X25
                  Node = 0 (120)
                  Object = 0 (X25TST)
                  Priority = 0

                  Destination = KL2137
                  Node = 34 (KL2137)
                  Object = 34 ()
                  Priority = 1
                  Call Mask = 01
                  Call Value = 01
                  Subaddresses = 1-1
NCP>show module x25-server counters (RET)
NCP>
15:16:37        NCP
                Request # 26 Accepted
```

```
NCP>
15:16:37        NCP
                Request # 26; Show Module Counters Completed

                Module = X25-SERVER

                   0     Seconds Since Last Zeroed
                  12     Maximum Circuits Active
                   0     Incoming Calls Rejected, No Resources
                   0     Logical Links Rejected, No Resources
```

Command	Entity	Keyword/Argument
ZERO	**MODULE X25-SERVER**	**[COUNTERS]**
	KNOWN MODULES	
	MODULE X25-PROTOCOL	**COUNTERS [KNOWN DTES] [DTE dte-address]**

Function:

The zero command used with the module entity generates a counters zeroed event that causes counters to be logged and then zeroed. The counters zeroed are those the executor node supports for the entity that you specified, and possibly qualified, in the command.

Arguments:

All arguments previously defined in this chapter.

Remarks:

The command ZERO MODULE X25-PROTOCOL COUNTERS can be used only if there is no more than one DTE address on the executor node. If there is more than one DTE address, use the qualifier DTE dte-address, that is, ZERO MODULE X25-PROTOCOL COUNTERS DTE dte-address. You may also use the command in the form ZERO MODULE X25-PROTOCOL COUNTERS KNOWN DTES.

Examples:

```
NCP>set executor node mrx25 (RET)
NCP>
15:01:24        NCP
                Set Executor Complete
NCP>zero module x25-server counters (RET)
NCP>
15:16:47        NCP
                Request # 27 Accepted
NCP>
15:16:49        NCP
                Request # 27; Zero Module Completed
NCP>zero module x25-protocol counters known dtes (RET)
NCP>
15:17:20        NCP
                Request # 28 Accepted
NCP>
15:17:26        NCP
                Request # 28; Zero Module Completed
```

Command	Entity	Keyword/Argument
CLEAR	**CIRCUIT cktid**	**MAXIMUM RECALLS**
PURGE	**KNOWN CIRCUITS**	**RECALL TIMER**

Function:

All CLEAR and PURGE commands clear parameters from the volatile and permanent data bases respectively.

Example:

```
NCP>clear circuit kdp-0-1 recall timer (RET)
NCP>
19:24:32        NCP
                Request # 65 Accepted
NCP>
18:24:32        NCP
                Request # 65; Clear Circuit Completed
```

Command	Entity	Keyword/Argument
{CLEAR / PURGE}	{LINE lineid / KNOWN LINES}	{ALL / COUNTER TIMER / HOLDBACK TIMER / MAXIMUM RETRANSMITS}

Arguments:

ALL and COUNTER TIMER: see Section 7.6. HOLDBACK TIMER and MAXIMUM RETRANSMITS are X.25-specific and previously defined under SET LINE lineid in this chapter.

Example:

```
NCP>tell mrx25 clear line kdp-0-0 maximum retransmits (RET)
NCP>
15:24:45        NCP
                Request # 38 Accepted
NCP>
15:24:46        NCP
                Request # 38; Clear Line Completed
```

Command	Entity	Keyword/Argument
{CLEAR / PURGE}	MODULE X25-ACCESS	{ALL network-qualifier / network-options network qualifier}

Arguments:

network-options may be one or more of: ACCOUNT, PASSWORD, USER.

All arguments are previously defined. See SET/DEFINE for MODULE X25-ACCESS.

Example:

```
NCP>clear module x25-access password network telenet (RET)
NCP>
15:22:18        NCP
                Request # 33; Clear Module Completed
```

Command	Entity	Keyword/Argument
{CLEAR / PURGE}	MODULE X25-PROTOCOL	ALL dte-qualifier group-qualifier GROUP group-name group-options CALL TIMER CLEAR TIMER MAXIMUM CLEARS MAXIMUM RESETS MAXIMUM RESTARTS RESET TIMER dte-options dte-qualifier

Arguments:

All arguments previously defined in this chapter. See SET/DEFINE for MODULE X25-PROTOCOL.

Example:

```
NCP>tell mrx25 clear module x25-protocol maximum clears (RET)
NCP>
15:23:10        NCP
                Request # 35 Accepted
NCP>
15:23:10        NCP
                Request # 35; Clear Module Completed
```

Command	Entity	Keyword/Argument
{CLEAR / PURGE}	MODULE X25-SERVER	ALL destination-qualifier COUNTER TIMER seconds destination-options destination-qualifier

Arguments:

Previously defined in this chapter. See SET/DEFINE MODULE X25-SERVER.

Example:

```
NCP>tell mrx25 clear module x25-server subaddr dest k12137 (RET)
NCP>
15:17:41        NCP
                Request # 20 Accepted
NCP>
15:17:41        NCP
                Request # 20; Clear Module Completed
```

APPENDIXES

APPENDIX A

DECnet PARAMETER SUMMARY

Table A-1
Node Parameters

	Digital Network Architecture			DECnet–20 V3.0 Implementation		
Param. Type No.	NCP Keywords	Applicability	Restrictions	TOPS–20	MCB	Restrictions
0	STATE 0 = ON(EXECUTOR) 1 = OFF(EXECUTOR) 2 = SHUT(EXECUTOR) 3 = RESTRICTED (EXECUTOR) 4 = REACHABLE (REMOTE) 5 = UNREACHABLE (REMOTE)	All nodes except loop nodes		X	X	Can not be set Can be displayed
100	IDENTIFICATION	EXECUTOR only		X	X	
101	MANAGEMENT VERSION	EXECUTOR only	Display only			Can not be set Can be displayed
110	SERVICE CIRCUIT	Adjacent only		X	X	
111	SERVICE PASSWORD	Adjacent only		X	X	
112	SERVICE DEVICE	Adjacent only		X	X	
113	CPU 0 = PDP–8 1 = PDP–11 2 = DECSYSTEM–1020 3 = VAX	Adjacent only		X	X	
120	LOAD FILE	Adjacent only		X	X	
121	SECONDARY LOADER	Adjacent only		X	X	
122	TERTIARY LOADER	Adjacent only		X	X	
125	SOFTWARE TYPE 0 = SECONDARY LOADER 1 = TERTIARY LOADER 2 = SYSTEM	Adjacent only		X	X	
126	SOFTWARE IDENTIFICATION	Adjacent only				
130	DUMP FILE	Adjacent only		X	X	
131	SECONDARY DUMPER	Adjacent only		X	X	
135	DUMP ADDRESS	Adjacent only				
136	DUMP COUNT	Adjacent only				
140	HOST Node address Node name (if any)	Adjacent and EXECUTOR	Display only	X	X	Can not be set Can be displayed
141	HOST	Adjacent and EXECUTOR		X	X	Can be set, displayed as parameter #140
150	LOOP COUNT	EXECUTOR only	Used with LOOP command only	X	X	Used with LOOP command only
151	LOOP LENGTH	EXECUTOR only	Used with LOOP command only	X	X	Used with LOOP command only
152	LOOP WITH 0 = zeroes 1 = ones 2 = mixed	EXECUTOR only	Used with LOOP command only	X	X	Used with LOOP command only
160	COUNTER TIMER	All nodes except loop nodes		X	X	
500	NAME	All nodes except loop nodes	Special format	X	X	Can not be set for EXECUTOR, can be set for other nodes. Will be displayed in node-id
501	CIRCUIT	Loop nodes and nodes by name only			X	
502	ADDRESS	EXECUTOR only	Special format	X	X	Can not be set Displayed in node–id
510	INCOMING TIMER	EXECUTOR only			X	
511	OUTGOING TIMER	EXECUTOR only			X	
600	ACTIVE LINKS	All nodes except loop nodes	Display only	X	X	Can not be set Can be displayed

Table A-1 (Cont.)
Node Parameters

	Digital Network Architecture			DECnet–20 V3.0 Implementation		
Param. Type No.	NCP Keywords	Applicability	Restrictions	TOPS–20	MCB	Restrictions
601	DELAY	All nodes except EXECUTOR and loop	Display only		X	Can not be set Can be displayed
700	NSP VERSION	EXECUTOR only	Display only	X	X	Can not be set Can be displayed
710	MAXIMUM LINKS	EXECUTOR only			X	Can not be set Can be displayed
720	DELAY FACTOR	EXECUTOR only		X	X	
721	DELAY WEIGHT	EXECUTOR only		X	X	
722	INACTIVITY TIMER	EXECUTOR only		X	X	
723	RETRANSMIT FACTOR	EXECUTOR only		X	X	
810	TYPE 0 = ROUTING 1 = NONROUTING 2 = PHASE–II	Adjacent only	Display only	X	X	Can not be set Can be displayed
820	COST	All nodes except EXECUTOR and loop.	Display only		X	Can not be set Can be displayed
821	HOPS	All nodes except EXECUTOR and loop	Display only		X	Can not be set Can be displayed
822	CIRCUIT	All nodes except EXECUTOR and loop	Display only		X	Can not be set Can be displayed
900	ROUTING VERSION	EXECUTOR only	Display only		X	Can not be set Can be displayed
901	TYPE 0 = ROUTING 1 = NONROUTING 2 = PHASE–II	EXECUTOR only		X	X	Can not be set Can be displayed
910	ROUTING TIMER	EXECUTOR only			X	
911	SUBADDRESSES	EXECUTOR only X.25 only				
920	MAXIMUM ADDRESS	EXECUTOR only			X	
921	MAXIMUM CIRCUITS	EXECUTOR only			X	Can not be set
922	MAXIMUM COST	EXECUTOR only			X	
923	MAXIMUM HOPS	EXECUTOR only			X	
924	MAXIMUM VISITS	EXECUTOR only			X	
930	MAXIMUM BUFFERS	EXECUTOR only			X	Can not be set Can be displayed
931	BUFFER SIZE	EXECUTOR only			X	Can not be set Can be displayed

Table A-2
Line Parameters

	Digital Network Architecture			DECnet-20 V3.0 Implementation		
Param. Type No.	NCP Keywords	Applicability	Restrictions	TOPS-20	MCB	Restrictions
0	STATE 0 = ON 1 = OFF 2 = SERVICE 3 = CLEARED	All circuits		X	X	Code 2 in MCB display only Code 3 not implemented
1	substate (NOT a keyword) 0 = STARTING 1 = REFLECTING 2 = LOOPING 3 = LOADING 4 = DUMPING 5 = TRIGGERING 6 = AUTOSERVICE 7 = AUTOLOADING 8 = AUTODUMPING 9 = AUTOTRIGGERING 10 = SYNCHRONIZING 11 = FAILED	All circuits Display only		X	X	
100	SERVICE 0 = ENABLED 1 = DISABLED	DDCMP circuits only		X	X	
110	COUNTER TIMER	All circuits		X	X	
200	CONNECTED NODE	X.25 circuits only				
201	CONNECTED OBJECT	X.25 circuits only				
400	LOOPBACK NAME	Circuits owned by EXECUTOR node only	Display only from circuit standpoint		X	(This is the Session Control nodename set by the SET NODE nodeid CIRCUIT cktid command.)
800	ADJACENT NODE node address node name (if any)	Circuits owned by EXECUTOR node only	Display only		X	Can not be set Can be displayed
810	BLOCK SIZE	Circuits owned by EXECUTOR node only	Display only		X	Can not be set Can be displayed
900	COST	Circuits owned by EXECUTOR node only			X	
906	HELLO TIMER	Circuits owned by EXECUTOR node only			X	
907	LISTEN TIMER	Circuits owned by EXECUTOR node only			X	
910	BLOCKING 0 = ENABLED 1 = DISABLED	X.25 only				
920	MAXIMUM RECALLS	X.25 only				
921	RECALL TIMER	X.25 only				
930	NUMBER	X.25 only				
1000	USER entity type (node = 0) node address node name (if any)	All circuits EXECUTOR node only	Display only (The only USERs currently defined are MODULE X25 PROTOCOL and EXECUTOR.)		X	Can not be set Can be displayed
1010	POLLING STATE 0 = AUTOMATIC 1 = ACTIVE 2 = INACTIVE 3 = DYING 4 = DEAD	DDCMP CONTROL circuits only	If not set default is AUTOMATIC			Not implemented

Table A-2 (Cont.)
Line Parameters

	Digital Network Architecture			DECnet–20 V3.0 Implementation		
Param. Type No.	NCP Keywords	Applicability	Restrictions	TOPS–20	MCB	Restrictions
1011	Polling substate (NOT a keyword) 1 = ACTIVE 2 = INACTIVE 3 = DYING 4 = DEAD	DDCMP CONTROL circuits only	Display only			Not implemented
1100	OWNER entity type (node = 0) node address node name (if any)	All circuits EXECUTOR node only			X	Can not be set Can be displayed
1110	LINE	DDCMP circuits			X	Can not be set Can be displayed
1111	USAGE 0 = PERMANENT 1 = INCOMING 2 = OUTGOING	X.25 only			X	Only USAGE PERMANENT IS supported Can not be set Can be displayed
1112	TYPE 0 = DDCMP POINT 1 = DDCMP CONTROL 2 = DDCMP TRIBUTARY 3 = X25 4 = DDCMP DMC	DDCMP circuits	Display only		X	Can not be set Can be displayed
1120	DTE	X.25 only			X	Can not be set Can be displayed
1121	CHANNEL	X.25 only			X	Can not be set Can be displayed
1122	MAXIMUM DATA	X.25 only			X	Can not be set Can be displayed
1123	MAXIMUM WINDOW	X.25 only			X	Can not be set Can be displayed
1140	TRIBUTARY	DDCMP circuits only				Not implemented
1141	BABBLE TIMER	DDCMP CONTROL circuits only				Not implemented
1142	TRANSMIT TIMER	DDCMP CONTROL circuits only				Not implemented
1145	MAXIMUM BUFFERS 1 to 254 = no. of buffers 255 = unlimited	DDCMP CONTROL circuits only				Not implemented
1146	MAXIMUM TRANSMITS	DDCMP CONTROL circuits only				Not implemented
1150	ACTIVE BASE	DDCMP CONTROL circuits only				Not implemented
1151	ACTIVE INCREMENT	DDCMP CONTROL circuits only				Not implemented
1152	INACTIVE BASE	DDCMP CONTROL circuits only				Not implemented
1153	INACTIVE INCREMENT	DDCMP CONTROL circuits only				Not implemented
1154	INACTIVE THRESHOLD	DDCMP CONTROL circuits only				Not implemented
1155	DYING BASE	DDCMP CONTROL circuits only				Not implemented
1156	DYING INCREMENT	DDCMP CONTROL circuits only				Not implemented
1157	DYING THRESHOLD	DDCMP CONTROL circuits only				Not implemented
1158	DEAD THRESHOLD	DDCMP CONTROL circuits only				Not implemented

Table A-3
Circuit Parameters

	Digital Network Architecture			DECnet–20 V3.0 Implementation		
Param. Type No.	NCP Keywords	Applicability	Restrictions	TOPS–20	MCB	Restrictions
0	STATE 0 = ON 1 = OFF 2 = SERVICE 3 = CLEARED	All lines			X	Codes 2 and 3 display only
1	substate (NOT a keyword) 0 = STARTING 1 = REFLECTING 2 = LOOPING 3 = LOADING 4 = DUMPING 5 = TRIGGERING 6 = AUTOSERVICE 7 = AUTOLOADING 8 = AUTODUMPING 9 = AUTOTRIGGERING 10 = SYNCHRONIZING 11 = FAILED	All lines	Display only		X	
100	SERVICE 0 = ENABLED 1 = DISABLED	All lines used for service			X	
110	COUNTER TIMER	All lines			X	
1100	DEVICE	All lines			X	Can not be set Can be displayed
1105	RECEIVE BUFFERS	All lines			X	
1110	CONTROLLER 0 = NORMAL 1 = LOOPBACK	All lines			X	Not for DTE * (state off)
1111	DUPLEX 0 = FULL 1 = HALF	All lines			X	Not for DTE Can not be set
1112	PROTOCOL 0 = DDCMP POINT 1 = DDCMP CONTROL 2 = DDCMP TRIBUTARY 4 = DDCMP DMC 5 = LAPB	All lines			X	Can be set for KDP lines only; only DDCMP, DDCMP DMC, and LAPB are allowed as values. Can be displayed within same limits Can not be set
1113	CLOCK 0 = EXTERNAL 1 = INTERNAL	All lines			X	Can be set and displayed for KDP lines only. * (state off)
1120	SERVICE TIMER	All DDCMP lines			X	Parameter value must be in multiples of seconds
1121	RETRANSMIT TIMER	All lines			X	Not for DTE KDP: restricted to seconds DMC: restricted to 1000 or 3000 ms
1122	HOLDBACK TIMER	LAPB lines				Not implemented
1130	MAXIMUM BLOCK	LAPB lines			X	Specific to X.25 (Display only)
1131	MAXIMUM RETRANSMITS	LAPB lines			X	Specific to X.25
1132	MAXIMUM WINDOW	LAPB lines			X	Specific to X.25
1150	SCHEDULING TIMER	DDCMP CONTROL				Not for DECnet–20
1151	DEAD TIMER	DDCMP CONTROL				Not for DECnet–20
1152	DELAY TIMER	DDCMP CONTROL				Not for DECnet–20
1153	STREAM TIMER	DDCMP CONTROL				Not for DECnet-20

* Some parameters can be set only in line state off

Table A-3 (Cont.)
Line Parameters

System–Specific Line Parameters				
Param. Type No.	NCP Keywords	TOPS–20	MCB	Restrictions
2650	CONTROLLER REGISTER		X	All line devices except KDP Can be displayed
2651	UNIT REGISTER		X	All line devices except KDP
2655	INTERRUPT VECTOR		X	All line devices except KDP Can be displayed
2660	INTERRUPT PRIORITY		X	All line devices except KDP Can be displayed
2665	RECEIVE SPEED		X	All line devices except KDP
2666	TRANSMIT SPEED		X	All line devices except KDP
2670	PAUSE TIMER		X	DTE only

None of the system-specific line parameters can be set by the user.

Table A-4
Logging Parameters

	Digital Network Architecture			DECnet–20 V3.0 Implementation		
Param. Type No.	NCP Keywords	Applicability	Restrictions	TOPS–20	MCB	Restrictions
0	STATE 0 = ON 1 = OFF 2 = HOLD	The EXECUTOR node's logging state for the specified logging sink.		X		Implemented for logging file only. (See NOTE below.)
100	NAME	The specific sink specified by the EXECUTOR as the event–receiver: device name, file name, or process identification for the sink types CONSOLE, FILE, and MONITOR.				Not implemented (See NOTE below.)
200	SINK NODE node address node name (if any)	The node that the EXECUTOR specifies as the event–receiver				Not implemented (See NOTE below.)
201	EVENTS entity type –1 = no entity 0 = node 1 = line 3 = circuit node address (if entity type is node) node name (if entity type is node) identification (lineid or cktid as indicated by entity type) entity class 0 = single class 2 = all events for class 3 = KNOWN EVENTS event class (if entity class is 0 or 2) event mask (if entity class is 0)	The events at the source node to be recorded at the SINK NODE.				Not implemented

NOTE

Logging parameters are not controllable by the OPERATOR or the System Manager. A limited number of events specific to TOPS–20 and MCB source nodes or lines will be automatically logged to the TOPS–20 ERROR.SYS file. This is transparent to the OPERATOR and under the control of the system software. The OPERATOR or System Manager has access to recorded EVENTS through the ERROR.SYS file and the SPEAR program.

Table A-5
Event Parameters

Network Management Layer Parameters - Class 0

Type	Keywords
0	SERVICE 0=LOAD 1=DUMP
1	STATUS Return code 0=REQUESTED >0=SUCCESSFUL <0=FAILED Error detail (if error) Error message (optional)
2	OPERATION 0=INITIATED 1=TERMINATED
3	REASON 0=Receive timeout 1=Receive error 2=Line state change by higher level 3=Unrecognized request 4=Line open error

Table A-5 (Cont.)
Event Parameters

Session Control Layer Event Parameters - Class 2

Type	Keywords
0	REASON 0=Operator command 1=Normal operation
1	OLD STATE 0=ON 1=OFF 2=SHUT 3=RESTRICTED
2	NEW STATE 0=ON 1=OFF 2=SHUT 3=RESTRICTED
3	SOURCE NODE
4	SOURCE PROCESS
5	DESTINATION PROCESS
6	USER
7	PASSWORD (0 means password set; n) parameter means not set)
8	ACCOUNT

Network Services Layer Event Parameters - Class 3

Type	Keywords
0	MESSAGE Message flags Destination link address Source link address Data
1	CURRENT FLOW CONTROL 0=no flow control 1=Segment flow control 2=Message flow control

Table A-5 (Cont.)
Event Parameters

Transport Layer Event Parameters - Class 4

Type	Keywords
0	PACKET HEADER Message flags Destination node address (not for control packet) Source node address Forwarding data (not for control packet)
1	PACKET BEGINNING
2	HIGHEST ADDRESS
3	NODE
4	EXPECTED NODE
5	REASON 0=Line synchronization lost 1=Data errors 2=Unexpected packet type 3=Routing update checksum error 4=Adjacent node address change 5=Verification receive timeout 6=Version skew 7=Adjacent node address out of range 8=Adjacent node block size too small 9=Invalid verification seed value 10=Adjacent node listener received timeout 11=Adjacent node listener received invalid data
6	RECEIVED VERSION
7	STATUS 0=REACHABLE 1=UNREACHABLE

Table A-5 (Cont.)
Event Parameters

Data Link Layer Event Parameters - Class 5

Type	Keywords
0	OLD STATE 0=HALTED 1=ISTRT 2=ASTRT 3=RUNNING 4=MAINTENANCE
1	NEW STATE 0=HALTED 1=ISTRT 2=ASTRT 3=RUNNING 4=MAINTENANCE
2	HEADER
3	SELECTED TRIBUTARY
4	PREVIOUS TRIBUTARY
5	TRIBUTARY STATUS 0=Streaming 1=Continued send after timeout 2=Continued send after deselect 3=Ended Streaming
6	RECEIVED TRIBUTARY
7	BLOCK LENGTH
8	BUFFER LENGTH
9	DTE
10	REASON
11	(Reserved)
12	(Reserved)
13	PARAMETER TYPE
14	CAUSE
15	DIAGNOSTIC

Physical Line Layer Parameters - Class 6

Type	Keywords
0	DEVICE REGISTER
1	NEW STATE 0=OFF 1=ON

Table A-6
Module X25-ACCESS Parameters

	Digital Network Architecture			DECnet–20 V3.0 Implementation		
Parameter Number	NCP Keywords	Applicability	Restrictions	TOPS–20	MCB	Restrictions
320	NODE node address node name (if any)	Qualified parameter		X		
330	USER	Qualified parameter		X		Unused in TOPS–20
331	PASSWORD 0 = Password set	Qualified parameter	Set only	X		Can be displayed in TOPS–20
332	ACCOUNT	Qualified parameter		X		Unused in TOPS–20
1110	NETWORK	Qualifying parameter		X		

Table A-7
Module X25-PROTOCOL Parameters

	Digital Network Architecture			DECnet-20 V3.0 Implementation		
Parameter Number	NCP Keywords	Applicability	Restrictions	TOPS-20	MCB	Restrictions
0	STATE 0 = ON 1 = OFF 2 = SHUT	Qualified by DTE			X	
100	COUNTER TIMER	Qualified by DTE			X	Non-qualified parameter in MCB
1000	ACTIVE CHANNELS	Qualified by DTE	Display only		X	
1010	ACTIVE SWITCHED	Qualified by DTE	Display only		X	
1100	DTE	Qualifying parameter			X	Display only in MCB
1101	GROUP	Qualifying parameter			X	
1110	NETWORK				X	Display only in MCB
1120	LINE	Qualified by DTE			X	Display only in MCB
1130	CHANNELS range beginning range end (none if same as beginning)	Qualified by DTE			X	* (state off)
1131	MAXIMUM CHANNELS	Qualified by DTE	Display only		X	
1140	DEFAULT DATA				X	
1141	DEFAULT WINDOW				X	
1150	MAXIMUM DATA				X	
1151	MAXIMUM WINDOW				X	
1152	MAXIMUM CLEARS				X	
1153	MAXIMUM RESETS				X	
1154	MAXIMUM RESTARTS				X	
1160	CALL TIMER				X	
1161	CLEAR TIMER				X	
1162	RESET TIMER				X	
1163	RESTART TIMER				X	
1170	DTE	Qualified by GROUP			X	
1171	NUMBER	Qualified by GROUP			X	
1172	TYPE 1 = BILATERAL	Qualified by GROUP			X	

* Some DTE parameters can be set only when the DTE is off.

Table A-8
Module X25-SERVER Parameters

	Digital Network Architecture			DECnet-20 V3.0 Implementation		
Parameter Number	NCP Keywords	Applicability	Restrictions	TOPS-20	MCB	Restrictions
100	COUNTER TIMER				X	
200	ACTIVE CIRCUITS		Display only		X	
300	DESTINATION	Qualifying parameter			X	
310	MAXIMUM CIRCUITS				X	Display only in MCB
320	NODE node address node name (if any)	Qualified parameter			X	
330	USER	Qualified parameter			X	
331	PASSWORD 0 = Password set	Qualified parameter	Set only		X	
332	ACCOUNT	Qualified parameter			X	
340	OBJECT object number object name	Qualified parameter			X	
350	PRIORITY	Qualified parameter			X	
351	CALL MASK	Qualified parameter			X	
352	CALL VALUE	Qualified parameter			X	
353	GROUP	Qualified parameter			X	
354	NUMBER	Qualified parameter			X	
355	SUBADDRESSES range beginning range end (none if same as beginning)	Qualified parameter			X	

APPENDIX B

COUNTER SUMMARY

This appendix contains the entity counters that may be zeroed or displayed by the appropriate NCP commands (ZERO and SHOW). Section B.1 describes DECnet-20 specific counters, and Section B.2 describes DECnet-20 PSI specific counters.

B.1 DECnet COUNTER SUMMARY

This counter summary is specific to DECnet-20 V3.0. See the appropriate documentation for other DIGITAL operating systems for counter text displayed and counters zeroed.

Table B-1
DECnet-20 Specific Counters

ALL COUNTERS (NODE, LINE, AND CIRCUIT)	
Maintained By	Counter Text
Network Management	Seconds since last zeroed
NODE COUNTERS	
Maintained By	Counter Text
Network Services	Bytes received Bytes sent Messages received Messages sent Connects received Connects sent Response timeouts Received connect resource errors Maximum logical links active (EXECUTOR only)
Transport (EXECUTOR node only)	Local aged packet loss Local node unreachable packet loss Local node out-of-range packet loss Local oversized packet loss Local packet format errors Local partial routing update loss Local verification reject

Table B-1 (Cont.)
DECnet-20 Specific Counters

LINE COUNTERS (for DDCMP lines)	
Maintained By	Counter Text
Physical Link	Remote Process Errors Local Process Errors
CIRCUIT COUNTERS	
Maintained By	Counter Text
Transport	Terminating Packets Received Originating Packets Sent Terminating Congestion Loss Transit Packets Received Transit Packets Sent Transit Congestion Loss Circuit Downs Initialization Failures Bytes Received
Data Link	Bytes Sent Data Blocks Received Data Blocks Sent Data Error Inbound, including: NAKs Sent, Header Block Check Error NAKs Sent, REP Response Data Errors Outbound Remote Reply Timeouts Local Reply Timeouts Remote Buffer Errors Local Buffer Errors Selection Intervals Elapsed Selection Timeouts
	NOTE The "Seconds Since Last Zeroed" function is not implemented for DECnet-20 Version 3.0. Counter text displayed for DECnet-20 depends on the executor of the command (KL10 or DN20).

B.2 X.25 SPECIFIC COUNTERS

The following tables describe X.25 specific counters for the PSI Version 1.0 software option.

The following table specifies the Data Link counters that apply to permanent X.25 circuits.

Table B-2
Data Link Circuit Counters for Permanent X.25 Circuits

Type Number	Bit Width	Standard Text
0	16	Seconds since last zeroed
1000	32	Bytes received
1001	32	Bytes sent
1010	32	Data blocks received
1011	32	Data blocks sent
1240	8	Locally initiated resets
1241	8	Remotely initiated resets
1242	8	Network initiated resets

The following table specifies the Data Link counters for LAPB lines.

Table B-3
Data Link Line Counters for LAPB Lines

Type Number	Bit Width	Standard Text	Bit Number Standard Text
0	16	Seconds since last zeroed	
1000	32	Bytes received	
1001	32	Bytes sent	
1010	32	Data blocks received	
1011	32	Data blocks sent	
1020	8	Data errors inbound	3 Block too long 4 Block check error 5 REJ sent
1021	8	Data errors outbound	3 REJ received
1030	8	Remote reply timeouts	
1031	8	Local reply timeouts	
1040	8	Remote buffer errors	2 RNR received, buffer unavailable
1041	8	Local buffer errors	2 RNR sent, buffer unavailable
1100	8	Remote process errors	4 Invalid N(R) received 5 FRMR sent, header format error
1101	8	Local process error	2 Transmit underrun 4 Receive overrun 5 FRMR received, head format error

The following table specifies the X.25 protocol module local DTE counters.

Table B-4
X.25 Protocol Module Counters

Type Number	Bit Width	Standard Text
0	16	Seconds since last zeroed
1000	32	Bytes received
1001	32	Bytes sent
1010	32	Data blocks received
1011	32	Data blocks sent
1200	16	Calls received
1201	16	Calls sent
1210	16	Past selects received
1211	16	Past selects sent
1220	16	Maximum switched circuits active
1221	16	Maximum channels active
1230	16	Received call resource errors
1240	8	Locally initiated resets
1241	8	Remotely initiated resets
1242	8	Network initiated resets
1250	8	Restarts

The following table specifies the X.25 server module counters.

Table B-5
X.25 Server Module Counters

Type Number	Bit Width	Standard Text
0	16	Seconds since last zeroed
200	16	Maximum circuits active
210	8	Incoming calls rejected, no resources
211	8	Logical links rejected, no resources

APPENDIX C

NETWORK RELATED MESSAGES

When you are using NCP commands, messages related to the execution of the commands and to the condition of the network will be output. Some appear on the user's terminal, and some appear on the CTY. If you are the System Manager, you should check the network-related messages on the CTY, or delegate this responsibility. If you are an operator-user, you should resolve those messages output to your terminal. Only those messages output to the user's terminal are described in this appendix (see also Section 4.5.7).

C.1 OPR/NCP COMMAND SYNTAX MESSAGES

All NCP messages are first examined by OPR. OPR uses the NCP table and the COMND JSYS to check the command syntax. The parsed command is then sent in an IPCF packet to ORION and QUASAR where further errors or informational messages may be added. The completely checked command returns to OPR and OPR directs all messages on each command to the terminal where the NCP command was typed.

The syntax messages output by OPR all begin with a question mark (?) and are in readable text only (no codes or abbreviations). In these messages the "?" does not indicate a fatal error: it questions the accuracy of your input. You correct the command by retyping up to the point of error (use CTRL/H) and substituting or adding the correct keyword or value. If you are unsure of what is required, type a question mark to get a list of possible arguments.

Possible syntax error messages follow in alphabetic order. Several examples of NCP commands with errors, showing the desired corrections, follow the alphabetic list.

? Ambiguous

? Does not match switch or keyword: "word"

? Filename was not specified

? File not found

? First nonspace character is not a digit

? Invalid character in number

? Invalid device terminator

? Invalid guide word

? Invalid node name

? Invalid wildcard designator

? Negative number improper

? No help available for "word"

? No such file type

? Node name doesn't contain any alphabetic characters

? Not confirmed

? Null switch or keyword given

? Too many characters in node name

Examples:

```
NCP>set exector node d2102a(RET)
? Does not match switch or keyword: "exector"
NCP>set executor node d2102a(RET)
NCP>
10:21:26        NCP
                Set Executor Complete

        **********

NCP>set node 124 name 2136(RET)
?Node name doesn't contain an alphabetic character
NCP>set node 124 name KL2136(RET)
NCP>
11:21:05         NCP
                 Set Node Complete

        **********

NCP>set node kl2136 inactivity timer seconds 20(RET)
?First nonspace character is not a digit
NCP>set node kl2136 inactivity timer20(RET)
? Does not match switch or keyword
NCP>set node kl2136 inactivity timer 20(RET)
NCP>
11:35:35         NCP
                 Set Node Completed
```

C.2 OPR/ORION/QUASAR INFORMATIONAL MESSAGES

Informational messages report major system changes: they cannot be "REFUSED" or "DISABLED". Depending on your current needs, these messages may or may not concern you, but you should read them.

Such messages are always time-stamped and have the general format:

```
NCP>
hh:mm:ss        --      text      --
```

Examples of informational messages of interest to network users follow:

Examples:

```
NCP>
14:25:53        -- Network topology --

Nodes on-line:
POLLUX

       **********

NCP>
16:17:44        -- NCP is not running--

       **********

NCP>
16:32:48        -- Network topology --

Nodes off-line:
DELPHI
```

NCP commands that are syntactically correct (no errors reported through OPR) are executed immediately if they do not require use of the NICE protocol. This includes the OPR/NCP commands (described in Chapter 5) and the commands processed locally by NCP (described in Chapter 6). These commands that are not sent in NICE protocol messages receive only the messages described in C.1 and C.2.

C.3 NCP COMMAND RESPONSE MESSAGES

All commands that are processed by Network Management routines are assigned a request number by NCP. They are then translated into NICE messages and are forwarded to NML in the central processor for local processing or to NML in the remote node for remote processing. Results of this processing return to the local NCP by means of NICE messages which are then formatted into response messages.

The generic format of NCP command response messages is:

```
hh:mm:ss     NCP
             Request # nnn; command entity status, [error information]
```

where:

hh:mm:ss	is the time in hours, minutes, and seconds
nnn	is the request number assigned
command	is a command indicator
entity	is a specific entity
status	is one of Complete, Accepted, or Failed
error information	is displayed only if status is Failed

C.3.1 Status Messages

Status is reported as Complete if the command has executed successfully. Status reported as Accepted implies only that the command is semantically (and syntactically) correct; the command will receive a further response of Complete or Failed if the link is maintained. If the status is reported as Failed, an error message follows.

Examples:

```
NCP>sh kn no sta (RET)
NCP>
 9:37:29      NCP
              Request # 96: Show Node Status Completed

              Remote Node = 88 (KL2530)

                State = Unreachable
                Active Links = 0

           **********

NCP>Clear exec node (RET)
NCP>lo no d2102a (RET)
NCP>
 11:18:16     NCP
              Clear Executor Complete

 11:18:16     NCP
              Request # 2 Accepted

 11:18:19     NCP
              Request # 2; Load Node Complete

           **********

NCP>sh no XXYZZ ch (RET)
NCP>
 11:00:17     NCP
              Request # 13; Show Node Failed, Unrecognized component
```

C.3.2 Error Messages from Network Management Software

When the status is Failed, an error message gives the reason for the failure. This reason is followed by detail if such detail is available. The detail, in turn, may be followed by additional explanatory text.

NCP error messages are listed below in alphabetic order. A brief explanation is given; where possible, appropriate action is suggested. The following NCP error messages are standard for most DECnet implementations.

Error Message Text	Description/Procedures
Bad loopback response	The message returned in a loopback test did not mirror the message sent. Repeat the loopback test. If the error persists, use the various loopback tests to try to isolate the part of the line that is bad. Follow your site's procedures for notifying your Field Service Representative.
Component in wrong state	There is a problem with the state of the entity to which the command applies, or to the secondary entity (line, for example, in a LOAD command). Check the state of the primary and secondary entities. Are nodes or lines On, Off, or in Service state as required by the function being requested? Correct and try again. This error occurs if you try to do a downline load or upline dump and have not set the service circuit to SERVICE ENABLED.
File I/O error	A hardware error occurred while trying to read or write a file required to execute the command. The error detail indicates the problem is in one of the following: DUMP FILE LOAD FILE Permanent Data Base SECONDARY DUMPER SECONDARY LOADER TERTIARY LOADER Volatile Data Base Repeat the command. If the error persists, follow your site's standard procedure for reporting hardware errors.
File open error	A file needed for processing could not be opened. Error detail specifies same files as for "File I/O error". System-specific detail, if present, may suggest procedure. If no detail is given, check directory or file restrictions.
Hardware failure	The hardware needed to satisfy the request could not perform the function requested. Try again. If the error message is repeated, check the hardware for obvious errors (not up, no paper in printer, and the like). If devices appear to be in order, follow your standard rules for contacting your Field Service Representative.

Error Message Text	Description/Procedures
Incompatible Management version	The function requested cannot be performed because the version skew between the NMLs in the source and destination nodes is too great. This problem cannot be solved. Make a record of the incompatibility for the future.
Invalid file contents	The error detail indicates the file that contains invalid information. The file is one of: Permanent Data Base LOAD FILE SECONDARY LOADER TERTIARY LOADER Although DECnet-20 does not currently support a permanent data base, you may receive this message if you have another node acting as EXECUTOR. If the file in error is a remote file, the System Manager at the remote site should be notified. If the file is local, the user should check to be sure the file typed is the correct file. If it was, next check the Directory. Files that contain errors are preceded by an asterisk. The TOPS-20 Commands Reference Manual describes the procedures to follow.
Invalid identification	The format of the primary or secondary entity identification is invalid. The error detail indicates identity type. Correct the error.
Invalid message format	NML has received a message that is not properly formatted. This could indicate a problem with the software on either the transmitting or receiving end. Try the command again. If the error persists, follow procedures for a Software Performance Report.

Error Message Text	Description/Procedures
Invalid parameter grouping	A request to change multiple parameters included an entity parameter group that could not be changed - the command was not executed. Check all entity-parameter combinations for validity, completeness, and an in-bounds value. Are there any contradictions or duplications? NMLT20 does not accept multiple-parameter commands. Each parameter must be in a separate command. Repeat specifying one parameter per command.
Invalid parameter value	The error detail indicates the name of the invalid parameter. Check length, size, and number values as to range allowed. Is the value permitted with the entity to which it refers?
Line communication error	This error occurs only on direct use of a line such as for looping, downline load, or upline dump. The error can be in transmitting or in receiving. If further detail is given, the device may be specified. Check (or have the system operator check) the operating condition of the indicated device. If the problem remains, follow procedures for reporting to your Field Service Representative.
Line protocol error	This error can refer to the Data Link protocol or the service operation protocol. The error is given only on direct use of a line (as for a line loop or downline load). Repeat the command. If the error persists, follow procedures for a Software Performance Report.

Error Message Text	Description/Procedures
Listener link connect failed	A connect to the NM Listener could not be completed. Error detail will vary depending upon the operating system being connected to. Be guided by detail, if present. There are several possible conditions that could cause this error: The remote node name of the access control information is incorrect. Correct the error and repeat the command. The remote node or the local node is in the process of shutting down and will accept no more logical link connections. Try later. Either the local or remote node has insufficient network resources to connect the logical link. Try later. There is no path to the remote node. Check the status of the remote node (use the SHOW NODE nodeid STATUS command). If status is "Unreachable" wait until it is "Reachable" and try again. A remote node you wish to connect to may not have a Network Management Listener A connection is not possible. Note for future reference. Procedure following this message will depend on detail given. If the detail indicates an easily correctable cause, missing or invalid parameters, for example, correct and repeat the command. If the detail given suggests a transient problem, wait appropriately and try again later ("appropriately" might be when a remote comes back on line, depending on the reason for the failure to connect).
Listener link disconnected	A successful connection to the NM Listener was made, but the logical link then failed. Both the error detail and the procedure are as described for "Listener link connect failed".

Error Message Text	Description/Procedures
Management program error	A software error in the NM software was detected. A system-level message may supply detail. Follow procedures for an SPR.
Mirror connect failed	A connect to the NM Loopback Mirror could not be completed. Error detail and procedure as for "Listener link connect failed".
Mirror link disconnected	A successful link was made to the Loopback Mirror, but the link was then disconnected. Error detail and procedure as for "Listener link connect failed".
No room for new entry	Insufficient room exists in some data base for an entry required by the requested function.
Operation failure	A requested operation failed. NML supplies no detail; there may or may not be system-level or system-specific detail. In the absence of detail, check the following; correct if possible, and try again. If the error persists, follow procedures for contacting your Software Specialist. Is there a required parameter for this command that is missing or invalid? Is the EXECUTOR in the proper state? Are there any helpful messages on the CTY?
Oversized management command message	A message size was too long. The NICE message was too long for the Management Listener to receive. Follow procedures for a Software Performance Report.
Parameter missing	You failed to include a required parameter. The error detail will specify the name of the missing parameter. Repeat the command with the required parameter.
Parameter not applicable	You have included a parameter that is not allowed with the entity to which it refers. The parameter is named in the error detail. Check the parameter description of arguments for limits and the restrictions in the applicable table in Appendix A. Correct and repeat the command.

Error Message Text	Description/Procedures
Parameter value too long	You have included a parameter that is too long to be accepted by the implementation. The error detail identifies the parameter type. If the parameter identified is a file specification, be aware that maximum file length is system-specific (VAX/VMS, for example, allows a maximum of 64 characters for a file specification; RSX-11M allows a maximum of 34.) If the EXECUTOR is not your local node, you may need to refer to documentation for the EXECUTOR's operating system.
Privilege violation	You do not have, or have not enabled, the privilege required for the function you have requested.
Resource error	A resource required to perform the requested function was not available. Repeat the command later.
System-specific Management function not supported	System-specific functions return this message if the EXECUTOR node is running an operating system other than the one for which the function was implemented.
Unrecognized component	An entity (component) is not known to the system. The error detail indicates the entity. Usually caused by an error in typing. Try the command again.
Unrecognized function or option	The requested function or option is not implemented by the EXECUTOR.
Unrecognized parameter type	The parameter indicated in the error detail is not implemented in the EXECUTOR.

C.4 X29SRV MESSAGES

The following messages are sent by X29SRV to the console terminal to indicate unusual events and errors. The warning messages are preceded by a % (such as, % X29SRV: ...). They are for information purposes only. You may choose to ignore them.

Fatal error messages are preceded by ? (such as, ? X29SRV: ...). These error messages are followed by an optional system error message, which indicates the system error that causes X29SRV to fail. The error messages indicate that X29SRV has stopped and you should investigate the error then restart it manually.

% X29SRV: Cannot find SYSTEM:X29SRV.INI, no configuration was performed

X29SRV could not locate the configuration file X29SRV.INI. The default configuration parameters are defined for X29SRV.

% X29SRV: Unrecognized configuration command

X29SRV has detected an illegal configuration command. The command is ignored.

% X29SRV: Failed to initialize GALAXY, no configuration was performed

X29SRV failed to initialize the GALAXY library. You should examine the versions of the GALAXY software to ensure compatibility with the X29SRV program. If they are not compatible, follow the instructions in the installation procedures to re-link X29SRV (using the batch control file X29SRV.CTL) with a compatible version of the GALAXY library. (See the TOPS-20 PSI Installation Guide.)

? X29SRV: Failed to create server, [Optional system error message]

X29SRV failed to create subjobs to receive incoming calls. The optional system error message will indicate the reason X29SRV failed to create subjobs.

? X29SRV: Failed to get server image file, [Optional system error message]

X29SRV failed to get its own executable image file in order to create the subjobs. X29SRV expects to find the file X29SRV.EXE in directory SYS:.

? X29SRV: Failed to map server image, [Optional system error message]

X29SRV failed to map the executable image file in order to create the subjobs. The optional system error message will indicate the reason X29SRV failed to map the image file.

? X29SRV: Failed to start server, [Optional system error message]

X29SRV failed to start the subjobs. The optional system error message will indicate the reason X29SRV failed to start the subjobs.

APPENDIX D

DECnet LINE DEVICES

The following table contains all of the currently recognized DECnet line devices. Devices followed by an asterisk are those supported in DECnet-20 V3.0.0.

Mnemonic	Multi-line	Multi-point	Type
DA	No	No	Parallel
DL	No	No	Async
DLV	No	Yes - slave	Async
DMC*	No	No	Sync
DMR*	No	No	Sync
DTE*	Yes	No	Parallel
DU	No	Yes	Sync
DUP	No	Yes	Sync
DUV	No	Yes - slave	Sync
DV	Yes	Yes	Sync
DZ	Yes	Yes	Async
KDP*	Yes	Yes	Sync
KDZ	Yes	Yes	Sync
PCL	No	Yes	Parallel

APPENDIX E

X.29 FILES

All X.29 files should reside in the SYSTEM: directory. Section E.1 describes X.29 PAD-parameter files, Section E.2 describes the X.29 configuration file, and Section E.3 describes the X.29 herald text file.

E.1 X.29 PAD-PARAMETER FILES

DEF.PAD is the file that defines the complete set of default PAD parameters for your system. In addition, you can create files that describe other PAD parameter sets specified by the System Manager.

The file name should have the form *.PAD. For example, the file FDX.PAD contains a PAD parameter set that an X.29 terminal user may use. You use this parameter set by typing /FDX following the host name when instructing X29SRV to connect the terminal to that system. See the TOPS-20 PSI User's Guide for more information on how to use the PAD parameter sets.

For the complete specification of all the parameters you can set for your PAD facility, see Recommendation X.3, "Packet Assembly/Disassembly Facility (PAD) in a Public Data Network" (not supplied by DIGITAL). The entries you can insert in DEF.PAD are defined in Section 3, "List of PAD parameters and possible values." For example, "PAD recall using a character" is called Reference 1 in the X.3 document; it is defined as 1:x in the sample shown below.

The format of a file that specifies PAD parameter sets is of the following form:

- An "!" in the first column indicates a comment line. Blank lines are ignored.
- Each line of text contains a PAD parameter definition, in the format:

  <PAD parameter>:<PAD value>

  where <PAD parameter> is the PAD parameter number and <PAD value> is the value you set. All values are decimal. For example:

  3:2

  indicates that the PAD forwards data to the host only when the user types <RET>.

- The file has one section containing CCITT recommended PAD parameters and, optionally, a section of PPSN-specific parameters. The sections are separated by a special entry in the form:

  0:0

  If the CCITT section is empty, start your file with the 0:0 entry.

- The number of PAD parameters defined in the PPSN section should not exceed 45.

For example, the following shows PAD-Parameter definitions that might be used for TELENET in the United States.

```
! 2-Oct-82
! TOPS-20 Line mode PAD parameter definitions

! PAD recall using character DLE
1:1

! Echo is enabled
2:1

! Forward data on control characters
3:126

! Idle timer delay is 1 second
4:20

! Terminal to PAD XON/XOFF is enabled
5:1

! Transmission of PAD service signals is enabled
6:1

! Normal data delivery
8:0

! No padding after carriage-return
9:0

! No line folding
10:0

! Parameter 11 is read only

! PAD to terminal XON/XOFF is enabled
12:1

! Insert line-feed after echo of carriage-return to terminal
13:4

! No padding after line-feed
14:0

! Use local editing
15:1

! Character delete editing character is DEL
16:127
```

```
! Line delete editing character is control-U
17:21

! Line display editing character is control-R
18:18

! Separator
0:0

! Automatic hangup upon disconnection is enabled
32:1

! Eight bit transparent (no parity checking) is enabled
63:1
```

E.2 X.29 CONFIGURATION FILE

The file X29SRV.INI in the SYSTEM: directory contains definitions of the network name, the maximum number of simultaneously active X.29 circuits, the default host system connection mode and reverse charging restriction. These parameters are set every time X29SRV is started.

Command	Parameters	Values
DEFINE	**MAXIMUM CIRCUITS**	**count**
	NETWORK	**network-name**
	NETWORK CLASS	**network-class**
	DEFAULT HOST	**control**
	REVERSE CHARGING	**restriction**

MAXIMUM CIRCUITS count

The maximum number of simultaneously active X.29 circuits that you allow on the system. The value of count ranges from 1 to 20.

Default: 8

NETWORK network-name

The name of the PPSN that X29SRV is communicating with. X29SRV recognizes the following PPSN names:

DATEX-P	Federal Republic of Germany
PSS	The United Kingdom
TELENET	The United States
TRANSPAC	France

If your PPSN name is not on the above list, contact your DIGITAL software specialist for your PPSN's network class.

Default: TELENET

NETWORK CLASS network-class

The class of the PPSN that X29SRV is communicating with. Contact your DIGITAL software specialist for the value of **network-class.**

Default: 0

DEFAULT HOST control

This parameter, if enabled, allows the X.29 terminal to be connected to the TOPS-20 host system automatically without specifying its DECnet node name at the initial connection. If your PSI gateway node services more than one TOPS-20 host system, and this parameter is enabled, the X.29 terminal will be connected to the system that has X29SRV running.

The values of **control** are:

{**DISABLED** }
{**ENABLED** }

Default: **DISABLED**

REVERSE CHARGING restriction

This parameter allows X29SRV to accept incoming calls with the reverse charge facility (such as calls that originate from public PAD facilities of the PPSN). Otherwise, those incoming calls are rejected.

The values of **restriction** are:

{**ALLOWED** }
{**DISALLOWED**}

Default: **ALLOWED**

E.3 X.29 HERALD TEXT FILE

The file X29NAM.TXT in the SYSTEM: directory contains the herald text printed at the beginning of the X.29 terminal session.

APPENDIX F

BIBLIOGRAPHY

The following DIGITAL documents contain detailed information on the protocols used in DECnet.

Each of the following subtitles is preceded by "DECnet DIGITAL Network Architecture" and followed by "Functional Specification" or (for DDCMP) "Specification".

General Description — AA-K179A-TK

This is an overview of the architecture that provides an introduction to each of the following documents.

Data Access Protocol (DAP) — AA-K177A-TK

Digital Data Communications Message Protocol (DDCMP) — AA-K175A-TK

Maintenance Operation Protocol (MOP) — AA-K178A-TK

Network Management Protocol — Version 3.0

Network Services Protocol (NSP) — AA-K176A-TK

Network Services Protocol (NSP) Update #1 — AD-K176A-T1

Transport — AA-K180A-TK

Session Control — AA-K182A-TK

The following published books are a source of general network information. The list is both arbitrary and abbreviated and is intended to be a catalyst to stimulate the reader's interest.

Title	Author	Publisher
Application Design Handbook for Distributed System	Patrick, Robert L.	CBI Publishing Co.
Basics of Data Communication	Karp, ed.	McGraw-Hill
Communications Networks for Computers	Davies and Barker	Wiley and Sons
Computer Networks and Distributed Processing: Software, Techniques, and Architecture	Martin, James	Prentice-Hall

BIBLIOGRAPHY

Data Communications	Doll	Wiley Interscience
Design and Strategy for Distributed Data Processing	Martin, James	Prentice-Hall
Handbook of Data Communications	Editors	National Computing Centre Publications (United Kingdom)
Introduction to Data Communications	Murphy and Kalis	DIGITAL
Introduction to Mini-Computer Networks	---	DIGITAL
Introduction to Teleprocessing	Martin, James	Prentice-Hall
Technical Aspects of Data Communications	McNamara, John E.	DIGITAL

APPENDIX G

GLOSSARY

The definitions and exposition in this glossary are specific to their use in this manual.

active — A designation for plural entities that limits them to those that meet an activity criterion. The criterion is defined individually for each entity.

active lines — Known lines in the ON or SERVICE state.

active logging — All known sink types that are in the ON or HOLD state.

active nodes — All reachable nodes as perceived from the executor node.

adjacent node — A node connected to the executor node by a single physical line.

aged packet — A packet that has exceeded the maximum number of visits.

Bilateral Closed User Group (BCUG) — An optional PPSN facility that restricts a pair of DTEs to communicating with each other. The basic BCUG also prevents this pair from accessing or being accessed by other DTEs. Additions to the BCUG facility allow one or both of the DTEs to access or be accessed by DTEs outside the group. These additions are known as BCUG with Outgoing Access and BCUG with Incoming Access respectively.

CCITT — Comite Consultatif International Telegraphique et Telephonique. An international consultative committee that sets international communications usage standards.

Central Management Node	Any Phase III (or higher) node that is capable of parsing all DNA Phase III NCP commands, including those that the Central Management Node can not or does not implement for its operating system. A Central Management Node is capable of managing the entire network of which it is a member; it has the capability of setting parameters and initiating and terminating the operation of nodes, circuits, lines, and events and altering the logical configuration of the network.
characteristics	Parameters that are generally static values in volatile memory or permanent values in a permanent data base. A Network Management information type. Characteristics can be set or defined. Examples of characteristics are the values assigned in response to the keywords NAME, TYPE, and LOAD FILE.
circuit	A logical point-to-point connection. A circuit is identified by device, controller, unit, and (if present) tributary. For example, KDP-2-0.1.
Closed User Group (CUG)	An optional PPSN facility that restricts two or more DTEs in the same group to communicating with each other. The basic CUG also prevents these DTEs from accessing or being accessed by other DTEs outside the group. Additions to the basic CUG facility allow one or more DTEs to access or be accessed by DTEs outside the group. These additions are known as CUG with Outgoing Access and CUG with Incoming Access respectively.
command node	The node where an NCP command originates.
controller	That part of a line identification that denotes the control hardware for a line. For a multiplex device that controller is responsible for one or more units. For example, in the line identification KDP-0-1, the controller is KMC-0. (The KDP is a KMC controller with up to four DUPs.)
counters	Error and performance statistics based on a node's network activity.
data link	A physical connection between two nodes. In the case of a multipoint line, there can be multiple data links for one physical connection.
data transmission	The sending of data from one computer to another over a physical link, or from one task to another over a logical link.

DCE	Data Communications Equipment (DCE) is an X.25 PPSN node to which a DTE is connected with a leased data communications line. A DCE can be a modem.
DDCMP	Digital Data Communications Message Protocol. A formal set of conventions designed to provide error-free, sequential transmission of data over physical links.
DMC11	A single line microprocessor-based interface to the network. The DMC11 is a synchronous Direct Memory Access (DMA) device.
DMR11	A single line microprocessor-based interface to the network. The DMR11 is a synchronous Direct Memory Access (DMA) device.
DN20	The DECnet-20 communication front end.
DN200	A remote station, based on a PDP-11/34A, that can be connected over a synchronous line to a host system running TOPS-20. The RJE-20 software can be loaded by DECnet-20 V3.0 and the DN200 can then participate in a DECnet-20 network as a full routing node.
downline loading	Transmitting a program's memory image over a physical link to an adjacent node and starting execution of it.
DTE	Data Terminal Equipment (DTE) is a host processor or communications processor directly connected to an X.25 PPSN with a leased data communications line.
DTE20	The hardware interface between the main processor in a DECSYSTEM-2040/50/60 and the PDP-11 processor in the DN20 communication front end.
duplex	Simultaneous independent transmission in both directions. Also referred to as full-duplex.
entity	CIRCUIT, LINE, LOGGING, or NODE. These are the major Network Management keywords. Each entity has its own parameters and options. CIRCUIT, LINE, and NODE have counters also. Allowed plural forms are KNOWN and ACTIVE CIRCUITS, LINES, and NODES. NODES may also take the form LOOP NODES. Entities are components that can be displayed and controlled.
event class	A subset of events, currently consisting of groups of events associated with each of the DNA layers or with specific operating systems.

event type	A particular type of event, unique within an event class.
executor node	The node where the active Local Network Management Function is running (that is, the node actually executing the command); the active network node physically connected to one end of a line being used for a load, dump, trigger or line loop test.
frame	A unit, delimited by flags, that includes a header used by the link level to exchange packets as well as control and error information between the DTE and DCE.
filter	A set of on/off states for a logging event class that indicates whether or not each event type in that class is to be recorded.
full-duplex	See duplex.
full routing node	A node that allows communication between non-adjacent nodes. If all nodes in a network are full routing nodes, then each node can communicate with each other node in the network.
global filter	A filter that applies to all entities within an event class and type.
half-duplex	Transmission in either direction, but not in both directions simultaneously.
hold state	A logging state in which the sink is temporarily unavailable and events for it should be queued.
hop	A hop is one unit of path length. When a request is routed from node A to node C by way of Node B, two hops are involved: node A to node B is one hop; node B to node C is another hop. (See path cost.)
host computer	A computer at a network node that primarily provides services such as computation, data base access, special programs, or programming languages to other nodes in the network.
host node or host	The node that provides services for another node (for example, the storage of files needed for a downline load).
information type	One of CHARACTERISTICS, COUNTERS, EVENTS, or SUMMARY. Used with the NCP command SHOW to control the type of information returned. Each entity parameter and counter is associated with one or more information types.

KDP	A DECnet term that refers to a combination of a KMC11 (controller) and one to four DUP11s. With the KMC11, a microprocessor-based system, the DUP11 functions as a direct memory access device. The interface to the network is synchronous.
KNOWN	The classification for a plural entity that includes all perceived occurrences of the entity type.
KNOWN LINES	All lines addressable by Network Management in the appropriate data base (volatile or permanent) on the EXECUTOR node. They may not all be in a usable state.
LAPB line	A line that uses the X.25 level 2 data link protocol LAPB. (LAPB protocol in X.25 is analogous to DDCMP in DECnet.) The acronym stands for Link Access Protocol Balanced.
line	A distinct physical data path. Line is a Network Management entity.
line cost	An arbitrary positive integer value assigned to a physical path. Because the routing algorithm selects the least-cost path to a destination, an operator can dynamically affect the path to be taken by changing line costs.
line identification	The device, controller, unit and/or tributary assigned to a line. Examples are KDP-0-1, DMC-1.
line level loopback	Testing a specific data link by sending a repeated message directly to the data link layer and over a wire to a device that returns the message to the source.
load	To read data into memory.
Local DTE	The communications node that connects to the public network.
local node	From the user's standpoint, a relative term indicating the node at which your terminal is logged in. From Network Management's standpoint, the node at which your requested task is executing. If at your local node you SET the executor to another node, that other node is the local node to Network Management and the node where your terminal is attached is a remote node.
Local user	A user of the User Access Module who communicates with the public network through the node running TOPS-20 PSI Gateway Software. Also called user.

logging	Recording information from an occurrence that has potential significance in the operation and/or maintenance of the network in a potentially permanent form. Logging is the Network Management entity that routes event data to logging sinks such as a console or file. Logging sinks can be accessed by persons and/or programs.
logging console	A logging sink that is to receive a human-readable record of events, for example, a terminal or printer.
logging event type	The identification of a particular type of event, for example, a line restarted or a node down.
logging file	A logging sink that is to receive a machine readable record of events for later retrieval.
logging identification	The sink type associated with the logging entity (file, console, or monitor).
logging sink	A place that a copy of an event is to be recorded.
logging sink flags	A set of flags in an event record that indicate the sinks on which the event is to be recorded.
logging sink node	A node to which logging information comes.
logging source process	The process that recognized an event.
logical connectivity	The ability of nodes to communicate with each other.
logical link	A node-to-node connection that is established and controlled by the session control, network services, and transport layers.
loopback	A mode of operation in which data transmitted by a network task is reflected at some point along the communication path and is returned to the originating task.
loopback node	A special name for a node that is associated with a line for loopback testing purposes. The SET NODE LINE command sets the loopback node name. A loopback node is treated as if it were a remote node. All traffic to the loopback node is to be looped over the associated line.

maximum cost	The greatest total cost the path to a node may have if the node is to be reachable. When the cost associated with transmitting a message from node to another exceeds the maximum cost, the transmission is not made.
maximum hops	The maximum number of hops a path to a node may be made up of if the node is to be reachable. Currently, the maximum hops supported is six.
maximum visits	The maximum number of nodes a message coming into this node can have visited. If the number specified as MAXIMUM VISITS is exceeded, and the destination is not this node, then the message is discarded.
maximum node address	The largest node address with which the local node can communicate.
maximum path cost	The greatest value assigned to the least-cost path between any two nodes in the network is the maximum path cost.
monitor	An event sink that is to receive a machine-readable record of events for possible real-time decision making.
network diameter	The reachability distance between the two nodes on the network having the greatest reachability distance, where reachability distance is the length of the shortest path between a given pair of nodes.
node	An implementation that supports DECnet. Used alone in this manual, node implies support of Transport, NSP, and Session Control. A node that does not support these three layers is referred to as a Phase II node. Node is a Network Management entity.
node address	The required unique numeric identification of a specific node.
node identification	Either a node name or a node address. In some cases an address must be used as a node identification. In other cases, a name must be used. A table is maintained for converting one to the other.
node level loopback	Testing a logical link using repeated messages that flow with normal traffic through the session control, network services, and transport layers within one node or from one node to another and back. In some cases node level loopback involves using a loopback node name associated with a particular line.

node name	An alphanumeric identification associated with a node address in a strict one-to-one mapping. No name may be used more than once in a node. The node name must contain at least one letter.
nonrouting node	A Phase III DECnet node that contains a subset of routing modules and can deliver and receive packets, but cannot route through.
NSP	Network Services Protocol. A formal set of conventions used in DECnet to exchange messages over logical links. NSP also refers to the program that implements the NSP protocol. (In the text, NSP refers to the program; NSP Protocol is used to refer to the protocol.)
off state	Applied to a node: a state where network traffic will no longer be processed. Applied to a line: a state where the line is unavailable for any kind of traffic. Applied to logging: a state where the sink is not available for receiving events, and any events for the sink should be discarded. Applied to circuits: the state where the circuit is not in use by any network-related software.
on state	Applied to a node: a state of normal network operation. Applied to a line: a state of availability for normal usage. Applied to logging: a state where a sink is available for receiving events.
OPR	OPR is the Operator Command Language program that provides the operator with one command language to communicate with several TOPS-20 components. OPR processes commands for syntax and passes syntactically correct commands to ORION. (See ORION.)
originating packet	A packet that originated in this node's Network Services layer.
ORION	ORION is the operator control program that accepts commands from OPR and forwards the commands to the proper program for execution. ORION forwards NCP commands to the Network Control Program.
packet	A group of bits, comprising data and control information, which is transmitted as a composite whole over a physical link. The data, control, and possibly error information are arranged in a specific format. (The basic transmission unit of DDCMP.)

path — The route a packet takes from source to destination node.

path cost — The sum of the line costs along a path between two nodes. See line cost.

path length — The sum of the hops along a path between two nodes.

permanent data base — Default information, about entities, that is kept in permanent memory (for example, files). Generally established at network generation. A permanent data base may be changed with the NCP commands PURGE and DEFINE; they may be modified for use in the volatile data base with CLEAR and SET commands.

Permanent Virtual Circuit (PVC) — A virtual circuit always associated with the same remote DTE address. The software references a PVC by its logical channel number (LCN). The public network vendor establishes the correspondence between a PVC and its LCN.

Phase III end node — A Phase III node that has implemented a subset of Phase III capabilities. A small routine node can send messages to and receive messages from adjacent nodes only. It has no "route through" capability, and must rely on an adjacent full routing node for communication with nonadjacent nodes.

physical connectivity — The condition of nodes being attached to each other by active lines.

physical link — A communication path between two adjacent nodes. This can be in the form of a dial-up line, radio, satellite link, or a channel-to-channel connector such as a DTE.

Port — The resources required to manage a virtual circuit.

PPSN — Public Packet Switching Network.

plural entity — A set of entities classified as known or active, or loop (nodes only).

processed event — An event after local processing, in final form.

protocol — A formal set of conventions or rules governing the format and relative timing of message exchange.

raw event — An event as recorded by the source process, incomplete in terms of total information required.

reachable node	A node to which the executor node's transport process believes it has a usable communication path.
Remote DTE	The DTE on the public network with which the local user in a DECnet network wishes to communicate. The remote DTE may give access to another network in addition to the DECnet network of the local user, and the public network with which the local user communicates.
remote node	A node in a network that is not your local node.
remote task	A task executing in a remote node.
Remote User	A user at the remote DTE with whom the local user communicates over a virtual circuit.
restricted state	A node state where no new logical links from other nodes are allowed.
routing node	A Phase III DECnet node that contains the full set of Transport modules, and can deliver, receive, and route through packets.
server task	An alternate designation for a task that has declared itself willing to accept a network connection, usually to provide some system service.
service password	The password required to permit triggering of a node's bootstrap ROM.
service slave mode	That mode where the processor is taken over and the adjacent, executor node is in control, typically for execution of a bootstrap program for downline loading or for upline dumping.
service state	A line state where such operations as downline load, upline dump, or line loopback are performed. This state allows direct access to the line by Network Management.
shut state	A node state where existing logical links are undisturbed, but new ones are prevented.
singular entity	A specific entity, for example, a line, a logging sink type, a node, or a circuit.
sink	See logging sink.
sink node	A node where logging sink types are located.

sink type	A particular final destination for logging events, either a file or a console.
source node	The node at which the request for a connection is initiated or from which a message is transmitted.
source task	The task in which the request for a connection is initiated or from which a message is transmitted.
specific filter	A filter that applies to a specific entity within an event class and type.
station	A physical termination on a line, having both a hardware and software implementation, that is, a controller and/or a unit. A station is part of a circuit identification.
status	Dynamic information relating to entities, such as their state. Status is a Network Management information type. Also, a message indicating whether or not an NCP command succeeded.
substate	An intermediate line state that appears as a tag on a line state display.
summary	An information type meaning the most useful information for an entity; the default if no information type is requested.
Switched Virtual Circuit (SVC)	A temporary logical association between two DTEs connected to a PPSN, analogous to connection by a dial-up line. An SVC is set up only when there is data to transmit; the SVC is cleared when the data transfer is complete.
target node	The node that receives a memory image during a downline load, generates an upline dump or loops back a test message.
terminating packet	A packet whose destination is this node.
topology	The physical arrangement and relationship of interconnected nodes and lines in the network. A legal topology satisfies all DNA requirements.
transit packet	A packet arriving at this node from a source node and destined for another node.
tributary	A physical termination on a multipoint line that is not a control station. Part of the line identification for a multipoint line.

unit	Part of a line/circuit identification. Together with the controller forms a station. In the line identification KDP-2-0, "KDP" identifies the device (KMC11/DUP11), "2" identifies the controller , and "0" identifies the unit.
unreachable node	A node is unreachable when the path length to that node is longer than the maximum number of hops in the network.
upline dumping	Transmitting a copy of a memory image over a line to a file at the host node.
Virtual Call	The process of establishing a virtual circuit between two DTEs.
Virtual Circuit	A logical path between the user process in a DECnet node and a cooperating process in a remote DTE. A virtual circuit is similar to a DECnet logical link.
volatile data base	The volatile, or temporary data base consists of dynamic values currently in memory. Parameters can be status, such as line or node state, or characteristics, such as CPU type or timer values, that remain constant until cleared or reset. Volatile data base values are lost when the system shuts down. When the system is first brought up, data from the permanent data base is read into the volatile data base. The OPERATOR may change certain of these values with the SET NCP command. Changes made to the permanent data base with the DEFINE command are not reflected in the volatile data base until the system is brought up again.
X.3	A CCITT recommendation that specifies the Packet Assembly/Disassembly (PAD) facility in a public data network.
X.25	A CCITT recommendation that specifies the interface between Data Terminal Equipment and Data Circuit-terminating Equipment for equipment operating in the packet mode on public data networks.
X.29	A CCITT recommendation that specifies procedures for the exchange of control information and user data between a packet-mode DTE and a packet Assembly/Disassembly (PAD) facility.

INDEX

READER'S COMMENTS

NOTE: This form is for document comments only. DIGITAL will use comments submitted on this form at the company's discretion. If you require a written reply and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Did you find this manual understandable, usable, and well-organized? Please make suggestions for improvement.

Did you find errors in this manual? If so, specify the error and the page number.

Please indicate the type of reader that you most nearly represent.

- ☐ Assembly language programmer
- ☐ Higher-level language programmer
- ☐ Occasional programmer (experienced)
- ☐ User with little programming experience
- ☐ Student programmer
- ☐ Other (please specify) _____

Name _____ Date _____

Organization _____ Telephone _____

Street _____

City _____ State _____ Zip Code _____
or Country

Do Not Tear - Fold Here and Tape

digital

No Postage
Necessary
if Mailed in the
United States

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

SOFTWARE PUBLICATIONS
200 FOREST STREET MRO1–2/L12
MARLBOROUGH, MA 01752

Do Not Tear – Fold Here and Tape