60480000

CONTROL DATA
CORPORATION

NETWORK PRODUCTS

NETWORK ACCESS METHOD VERSION 1 NETWORK DEFINITION LANGUAGE REFERENCE MANUAL

CDC® OPERATING SYSTEMS:
NOS 1

REVISION RECORD

Revision	Description
A (12/01/76)	Original release. PSR level 439.
B (04/01/77)	Revised at PSR level 446 for technical corrections.
C (04/28/78)	Completely revised for NAM Version 1.1 release at PSR level 472 to include support of remote and foreign NPUs, asynchronous and HASP TIPs, virtual terminals, IAF, and TVF.
D (08/15/78)	Revised at PSR level 477 for technical corrections.
E (12/18/78)	Revised at PSR level 485 to include autorecognition up to 1200 baud and for technical corrections.
F (08/10/79)	Revised to reflect release of NAM Version 1.2. Included are descriptions of special editing support, extended APL support, and various technical corrections.
G (05/23/80)	Revised at PSR level 517 to reflect support of the 714-30 terminal class, and of 714-10/20 and 714-30 line printers. Various technical corrections are also included.
H (10/31/80)	Revised at PSR level 528 to document support of the X.25 protocol.
J (05/29/81)	Revised to reflect release of NAM Version 1.3 at PSR level 541 to document support of the PRU interface and the BSC protocol (2780/3780 terminal support). Also includes various technical corrections.

REVISION LETTERS I, O, Q, AND X ARE NOT USED

Address comments concerning this manual to:

CONTROL DATA CORPORATION
Publications and Graphics Division
215 MOFFETT PARK DRIVE
SUNNYVALE, CALIFORNIA 94086

or use Comment Sheet in the back of this manual

LIST OF EFFECTIVE PAGES

New features, as well as changes, deletions, and additions to information in this manual are indicated by bars in the margins or by a dot near the page number if the entire page is affected. A bar by the page number indicates pagination rather than content has changed.

Page	Revision
Front Cover	-
Title Page	-
ii	J
iii/iv	J
v/vi	J
vii	J
viii	J
ix/x	J
xi	J
1-1 thru 1-5	J
2-1 thru 2-7	J
3-1 thru 3-17	J
4-1 thru 4-35	J
5-1	J
5-2	J
6-1 thru 6-3	J
7-1 thru 7-18	J
A-1 thru A-41	J
B-1 thru B-23	J
C-1 thru C-5	F
D-1	J
D-2	J
E-1	J
E-2	J
F-1	C
Index-1	J
Index-2	J
Comment Sheet	J
Mailer	-
Back Cover	-

PREFACE

This manual describes the Network Definition Language (NDL) for the CDC® Network Access Method, Version 1.3. It assumes that the reader is an installation administrator familiar with the Network Operating System (NOS) and other software in the networks product set.

As described in this publication, the Network Access Method Version 1.3 operates under control of the NOS 1 operating system for the CONTROL DATA® CYBER 170 Series; CYBER 70 Models 71, 72, 73, and 74; and 6000 Series Computer Systems.

The Network Definition Language Processor is a compiler used by a network administrator to create and maintain the files that define the physical and logical structure of the network for other network software, which in turn establishes, initializes, and operates the network.

RELATED MANUALS

The NDL user can find additional pertinent information in the following Control Data Corporation manuals. The NOS manual abstracts is an instant-sized manual containing a brief description of the contents and intended audience of every manual available for NOS and its product set. The abstracts manual can be useful in determining which manuals are of greatest interest to a particular reader. The Software Publications Release History serves as a guide in determining which revision level of software documentation corresponds to the Programming Systems Report (PSR) level of installed site software.

The following publications are of primary interest:

Publication	Publication Number
CYBER Cross System Version 1 Build Utilities Reference Manual	60471200
Network Products Network Access Method Version 1 Reference Manual	60499500
NOS Version 1 Reference Manual (Volume 1)	60435400
NOS Version 1 System Maintenance Reference Manual	60455380

The following publications are of secondary interest:

Publication	Publication Number
Network Access Method Version 1 Terminal Interface Guide	60480600
Network Products 255x Series Communications Control Program Version 3 Reference Manual	60471400
Network Products Interactive Facility Version 1 Reference Manual	60455250
NOS Version 1 Installation Handbook	60435700
NOS Version 1 Manual Abstracts	84000420
NOS Version 1 Operator's Guide	60435600
Software Publications Release History	60481000

CDC manuals can be ordered from Control Data Corporation, Literature and Distribution Services, 308 North Dale Street, St. Paul, Minnesota 55103.

This product is intended for use only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features or parameters.

CONTENTS

APPENDIXES

INDEX

FIGURES

TABLES

NOTATIONS

NOTATION USED

Unless otherwise specified, the following conventions are used throughout this manual in statement formats and diagnostic messages:

HOST — Uppercase letters identify NDL reserved words (see appendix D) in statement formats. These words must be spelled correctly and cannot be used in any network definition statement other than the ones in which this manual presents them. Uppercase words are also used in examples to indicate programmer-supplied values.

value — Lowercase letters in formats identify variables for which values are supplied by the NDL user, or by the NDL Processor as output. Lowercase letters are not used in examples.

{ } — Braces enclose entities from which one must be chosen. If the entities are arranged vertically, any of the entities can be used but one must be used.

[ ] — Brackets enclose entities that are optional. If omission of any entity causes the use of a default entity, the default is underlined. If the entities within the brackets are arranged horizontally, any one or more of the entities can be used or all can be omitted. When the entities are arranged vertically, only one of the entities can be used but all can be omitted.

... — Ellipsis indicates that omitted entities repeat the form and function of the entity last given.

Δ — The delta is used to indicate one or more blanks. A blank is a legal separator in NDL and can be substituted for a comma. Only the first blank in a string of blanks is significant; subsequent blanks, as well as blanks preceding or following a comma, are ignored.

___ — Underlining shows defaults.

Unless otherwise specified, all references to numbers are to decimal values.

INTRODUCTION 1

As an installation administrator, you will use the CDC Network Definition Language to formally define your network. You do so by using a Network Definition Language (NDL) program to create the two network definition files, which describe the physical and logical configuration of the network's hardware and software elements. These files are the Network Configuration File (NCF) and the Local Configuration File (LCF). The NCF contains information about the physical and logical configuration of the network. The LCF contains information about application programs accessed via the network. It can also contain login information for terminals that can connect to the network. Your program will be processed by the Network Definition Language Processor (NDLP), which will create the two files, as shown in figure 1-1.

NETWORK DEFINITION LANGUAGE PROCESSOR

In addition to the two network definition files, the NDLP produces a job listing file. The NDLP provides you with several options for the print output:

- A source input listing
- An error summary listing
- Summaries of the NCF and LCF contents
- A list of all legal DEFINE statements (statements used like macros to equate a character string with a single identifier)
- A source input listing with DEFINE statements expanded

These options are described in section 6, Job Structure.

The NDLP executes as a batch preprocessing compiler; it does not interact with the network during network operation.

BASIC NETWORK CONCEPTS

To ensure efficient functioning of the network, you must configure it properly. Using the Network Definition Language, you must provide the information necessary to identify more than just the hardware and software elements of the network. You must also establish certain logical relationships among the hardware and software elements. The following basic concepts are essential to an understanding of those relationships. A more detailed description of the elements of a CDC network is presented in section 2, The Network.

Figure 1-2 shows the network configuration file and the local configuration file, used by supervisory programs within the host computer to initialize, monitor, and control network operations. The network supervisor (NS) uses information from the NCF primarily for starting network operation and for error monitoring. The communications supervisor (CS) uses information from the LCF in overseeing message traffic during network operation. The network validation facility (NVF), another supervisory program, uses the LCF in conjunction with the VALIDUz system validation file to determine which network resources terminal users will be allowed to access.

All messages pass through the Network Access Method (NAM) to ensure their integrity as they are routed between your terminals and the host computer. NAM consists, for the most part, of three interface programs in the host computer: the Peripheral Interface Program (PIP), the Network Interface Program (NIP), and the Application Interface Program (AIP). The relationship between the three programs is shown in figure 1-3. Each of these programs is essential to data communication within the network.

The PIP runs in a dedicated peripheral processor. The NIP runs in the central processor at a system control point; it is thus able to communicate with other programs running at other control points. A copy of the AIP runs in the field length of every network application program. Supervisory messages are passed by the NIP to the PIP and the AIPs to ensure proper routing of your data messages. The application programs shown could be the supervisory programs from figure 1-2, or they could be programs tailored to your needs. You can provide some of your own programs, or you can use CDC-written programs such as the Interactive Facility (IAF) or the Remote Batch Facility (RBF).

SIMPLE NETWORK

Figure 1-4 shows a possible network, Simnet. In this simple network, a single host computer, a CYBER 170 Model 176, is connected directly to a network processing unit (NPU), which is actually a small independent processor. This connection consists of a communication line connected to a coupler, a hardware module in the NPU. An NPU connected to the host in this manner is known as a front-end NPU. The front-end NPU in Simnet is NPU1, a CDC 2550 communications processor. The front-end NPU can support both terminals and other NPUs, known as remote NPUs. Remote NPUs can also support terminals. The only remote NPU in this network is NPU2, a CDC 2551 communications processor.

In the figure, terminal T1 is the system console. The system console is not considered a network terminal, and therefore is not defined in an NDL program. The system console operator can access the network with special commands. The front-end network processing unit NPU1 supports terminal T2, and the remote processing unit NPU2 supports terminals T3 and T4. CDC provides support for many different types of terminals, as discussed in section 4, Terminal Support.

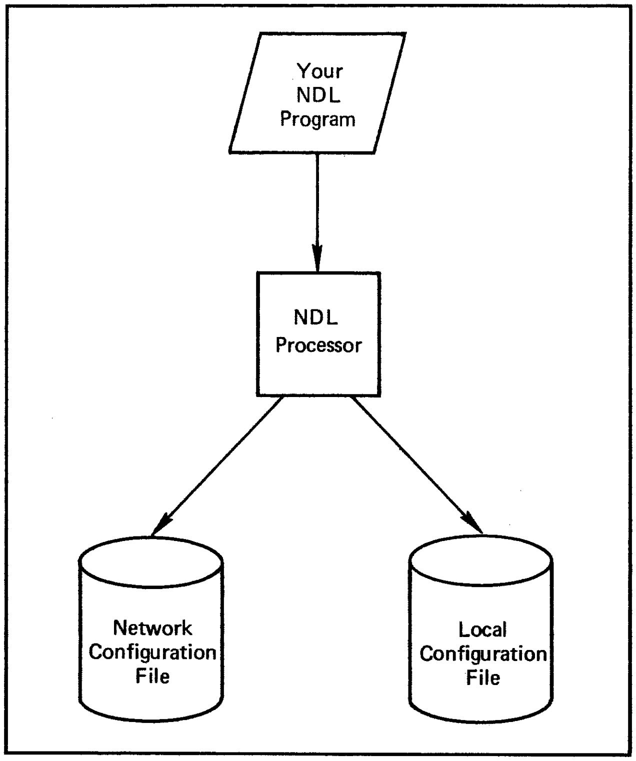


Figure 1-1. Creation of Network Definition Files

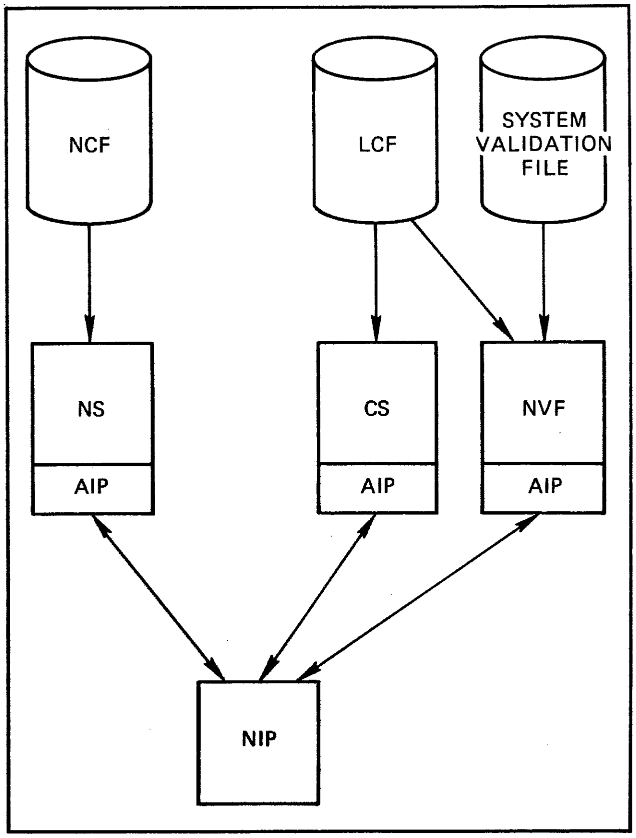


Figure 1-2. Use of NCF and LCF by Supervisors

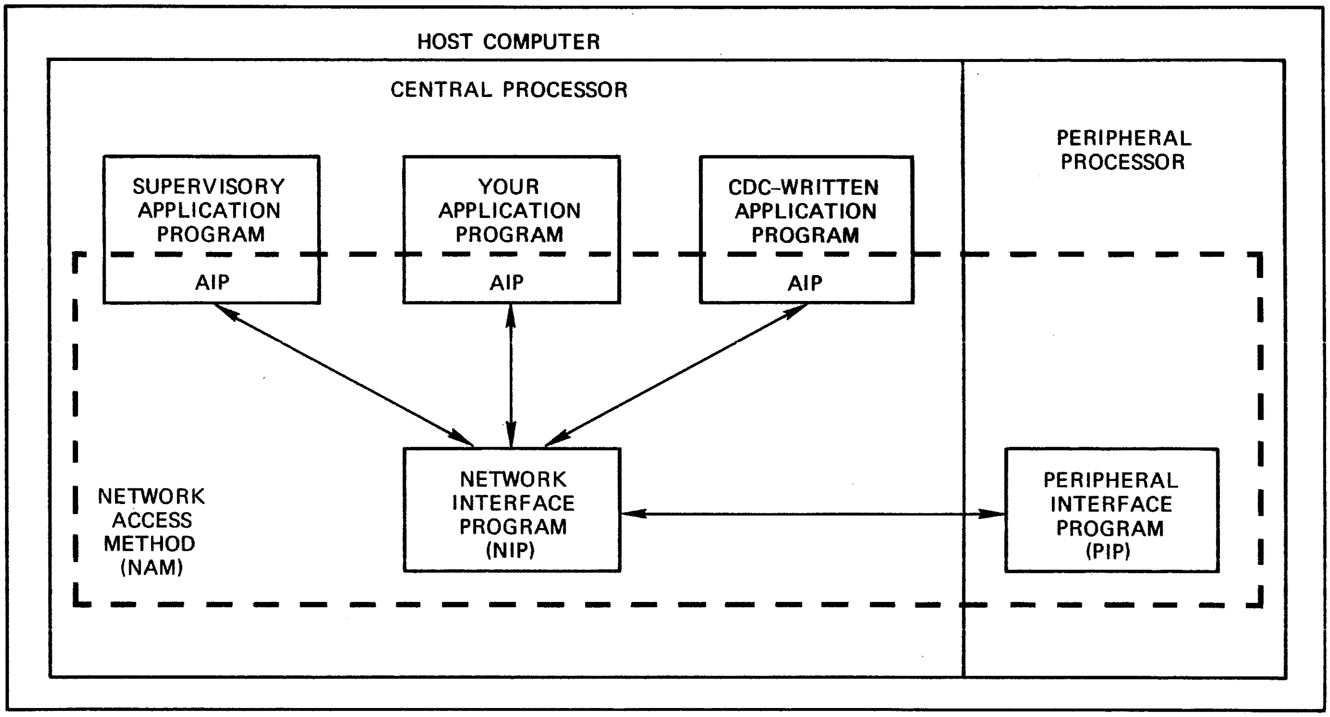


Figure 1-3. The Network Access Method

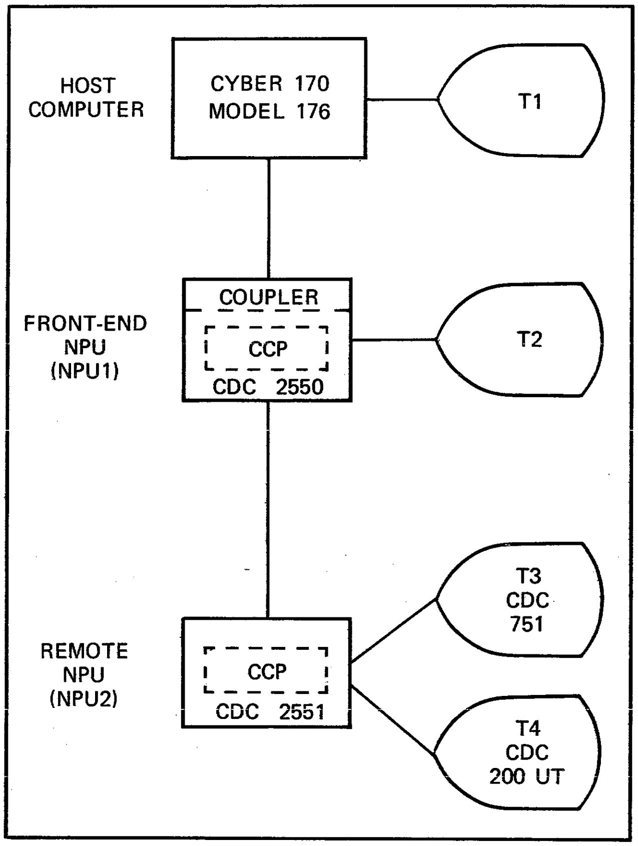


Figure 1-4. Simnet - A Simple Configuration

A network has an administrator who is designated the network operator. This operator can be located at the host computer console or at a terminal serviced by the network. For every host computer there is also an administrator designated the local operator. This operator can also be located at either the host computer console or a terminal serviced by the host through the network. The local operator can be the network operator, the system console operator, both, or neither. In Simnet, the network operator is located at the system console, terminal T1, and the local operator is located at terminal T2. The distinction between the two operators becomes more apparent in the multiple-host environment, presented later in this section.

Each NPU, since it is an independent processor, has its own software. The software packages in the NPU are known collectively as the Communication Control Program (CCP). The version of CCP loaded into a particular NPU will depend on various aspects of the network to be configured, including the number and type of terminals your network is to support, the application programs you require, and the NPU and host hardware in your network.

Three basic types of interface programs exist in CCP: host interface programs (HIPs), link interface programs (LIPs), and terminal interface programs (TIPs). A HIP links a front-end NPU to a host computer. A LIP links a remote NPU to a front-end NPU. A TIP links terminals to either type of NPU. Figure 1-5 presents another view of Simnet, illustrating the interrelationships between these interface programs. Note that terminals T3 and T4 each connect to a separate terminal interface program. T3 is a CDC 751, an asynchronous terminal, whereas T4 is a CDC 200 User Terminal, a synchronous terminal. Each has different support requirements; therefore, a different interface is required for each. The network software currently supports TIPs for five general terminal protocols. You can also provide the software to support up to four additional protocols.

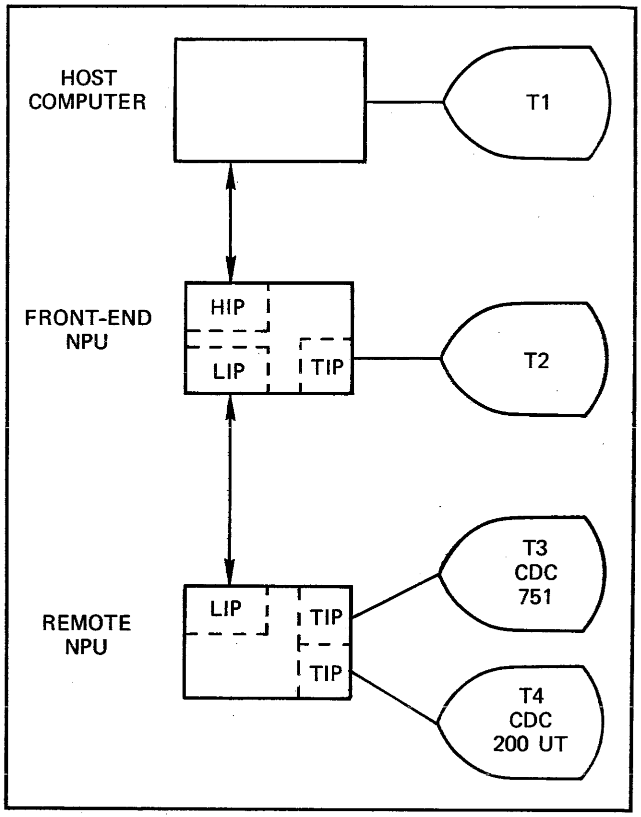


Figure 1-5. CCP Interface Programs

MULTIPLE-HOST NETWORK

Although the current network software allows only one host computer in the network, certain characteristics have been included in the software to allow possible support in the future of networks with multiple host computers. A simple multiple host configuration is illustrated by Dualnet, in figure 1-6. A second host has been added to the network, connecting to a coupler in NPU2. NPU2 thus becomes a front-end NPU, and NPU1 is considered a remote NPU.

A more complicated multiple-host network concept is illustrated by the network Multinet, in figure 1-7. Note that the left side of the figure closely resembles Simnet, from figure 1-4. In effect, another entire network has been added to that simple network. This has been accomplished by connecting the front-end processor NPU1 to another host, HOST11, in the same manner as NPU1 is connected to HOST1.

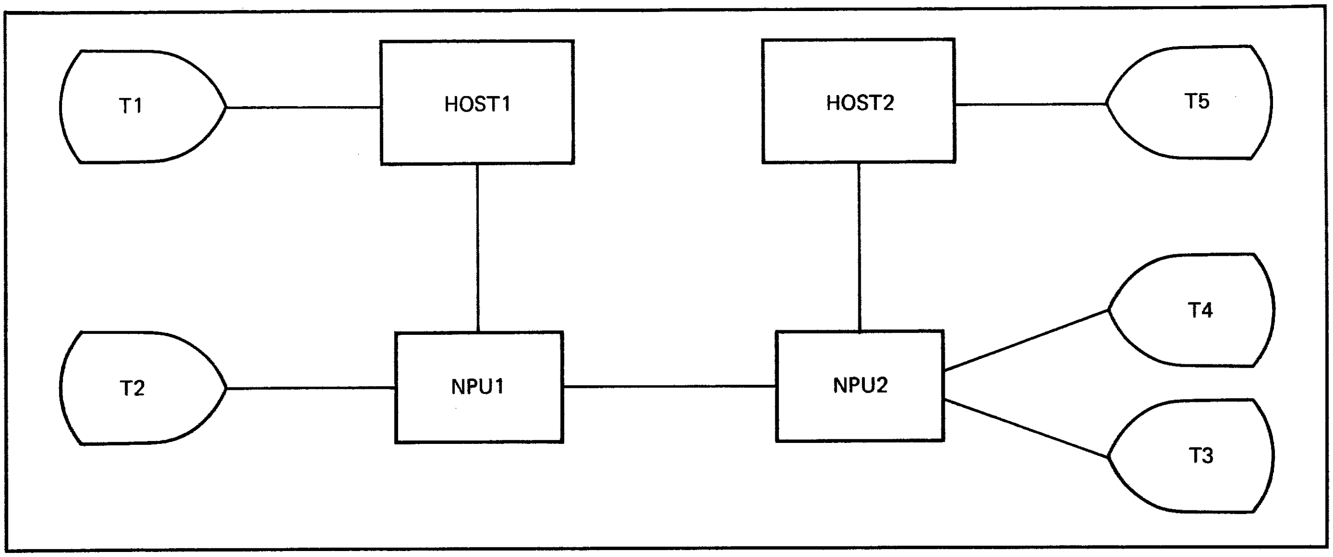


Figure 1-6. Dualnet - A Possible Future Network

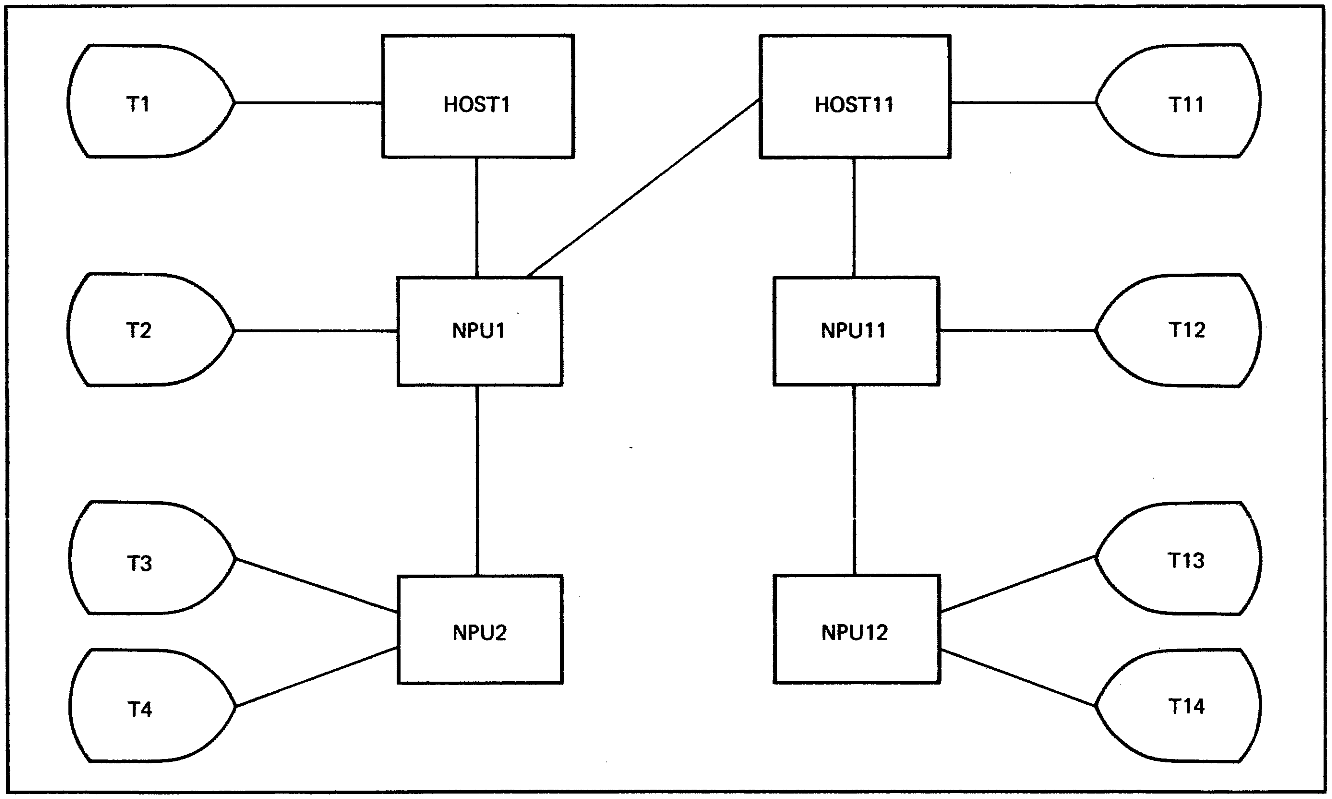


Figure 1-7. Multinet - A Possible Future Network

NETWORK AND LOCAL OPERATORS

The multiple-host configuration is not currently supported, but an understanding of its basic nature helps to clarify the necessary separation of the functions of the network operator and the local operator. In Multinet, as in all current CDC networks, there is only one network operator. The network operator can be located at any of the terminals. The network operator communicates with the network supervisor program to manage elements of the network. His or her functions include

- Initiation of network operation
- Loading of software into the NPUs
- Dumping the memory contents of the NPUs
- Permitting or preventing an NPU's accessing the network
- Halting network operation

There are two local operators, however, in this network - one for each host. Each local operator can perform the following functions for the elements of the network that are assigned to his host.

- Permit or prevent execution of application programs
- Halt executing application programs
- Permit or prevent a terminal's accessing the network
- Turn communication lines logically on or off

A local operator is usually located at the system console. This is the case in Multinet; the local operators are located at terminals T1 and T11. The operator at terminal T1 is also designated the network operator.

PUBLIC NETWORKS

CDC network software provides support for asynchronous terminals connected to public packet-switching networks. This support complies with recommendation X.25 of the International Telegraph and Telephone Consultative Committee (CCITT), for standardizing the interface between data terminal equipment and packet-switching networks.

Figure 1-8 shows the packet assembly/disassembly (PAD) service, one part of the interface between the network and the data terminal equipment. This part of the interface is provided by the packet-switching network (PSN).

A calling terminal is linked with the communication control program in the NPU via a logical path called a virtual circuit. The host must subscribe to the PSN for the desired number of virtual circuits. CDC support of this type of configuration is described in detail in section 4, under X25 Protocol.

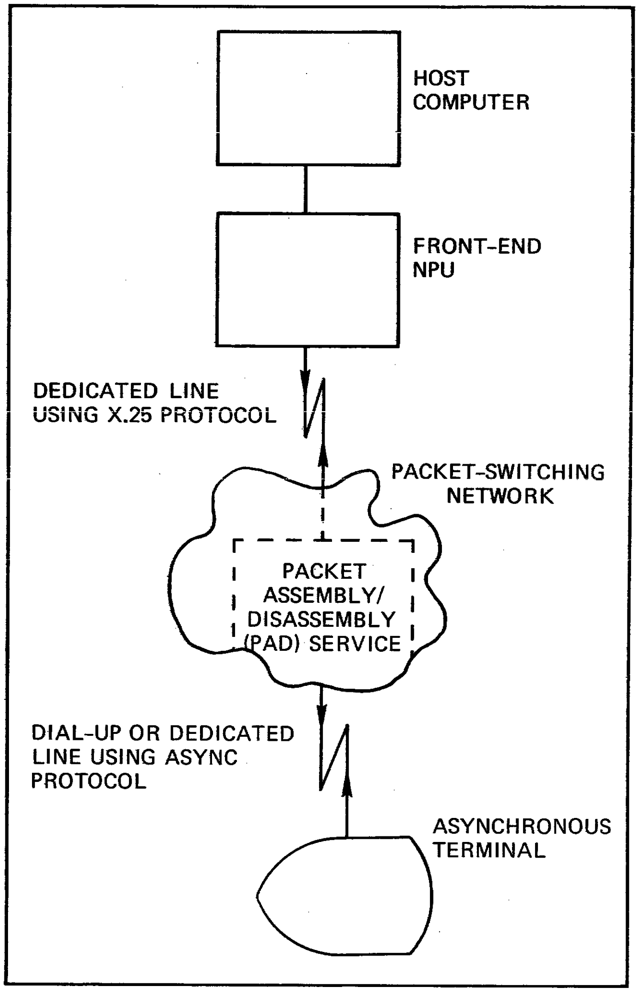

HOST COMPUTER
FRONT-END NPU
DEDICATED LINE USING X.25 PROTOCOL
PACKET-SWITCHING NETWORK
PACKET ASSEMBLY/ DISASSEMBLY (PAD) SERVICE
DIAL-UP OR DEDICATED LINE USING ASYNC PROTOCOL
ASYNCHRONOUS TERMINAL

Figure 1-8. Packet-switching Network Interface

While defining your network, you should pay careful attention to both its physical and its logical characteristics. The Network Definition Language is designed to help you create your definition in a simple and logical manner. This section presents three aspects of the network that must be described in your NDL program: its hardware elements, its software elements, and the logical relationships you must define among them.

HARDWARE ELEMENTS

To ensure proper operation of the network, you must carefully define all of its hardware elements and establish the relationships between them. The configurable hardware elements of the network are shown in figure 2-1. NDL provides a separate statement to configure each type of element. Each type of element is described in detail below.

Data conversion can take place at any of the elements in this figure. For further information about this conversion, see the Terminal Interface Guide listed in the preface.

HOST PROCESSORS

A host processor is any computer mainframe that is linked to the network to run network application programs. Host computers contain the portions of the network software necessary for applications to access the network.

The current version of the network software provides for only one host computer in a single network. A host can be connected by communication lines to one or more network processing units, as was illustrated by the network processing units NPU1 and NPU2 of Simnet (figure 1-4 in section 1).

Message flow in the network is defined from the perspective of the host computer. Messages coming to the host are said to be travelling upline; those moving away from the host are said to be travelling downline. This concept is illustrated by figure 2-2.

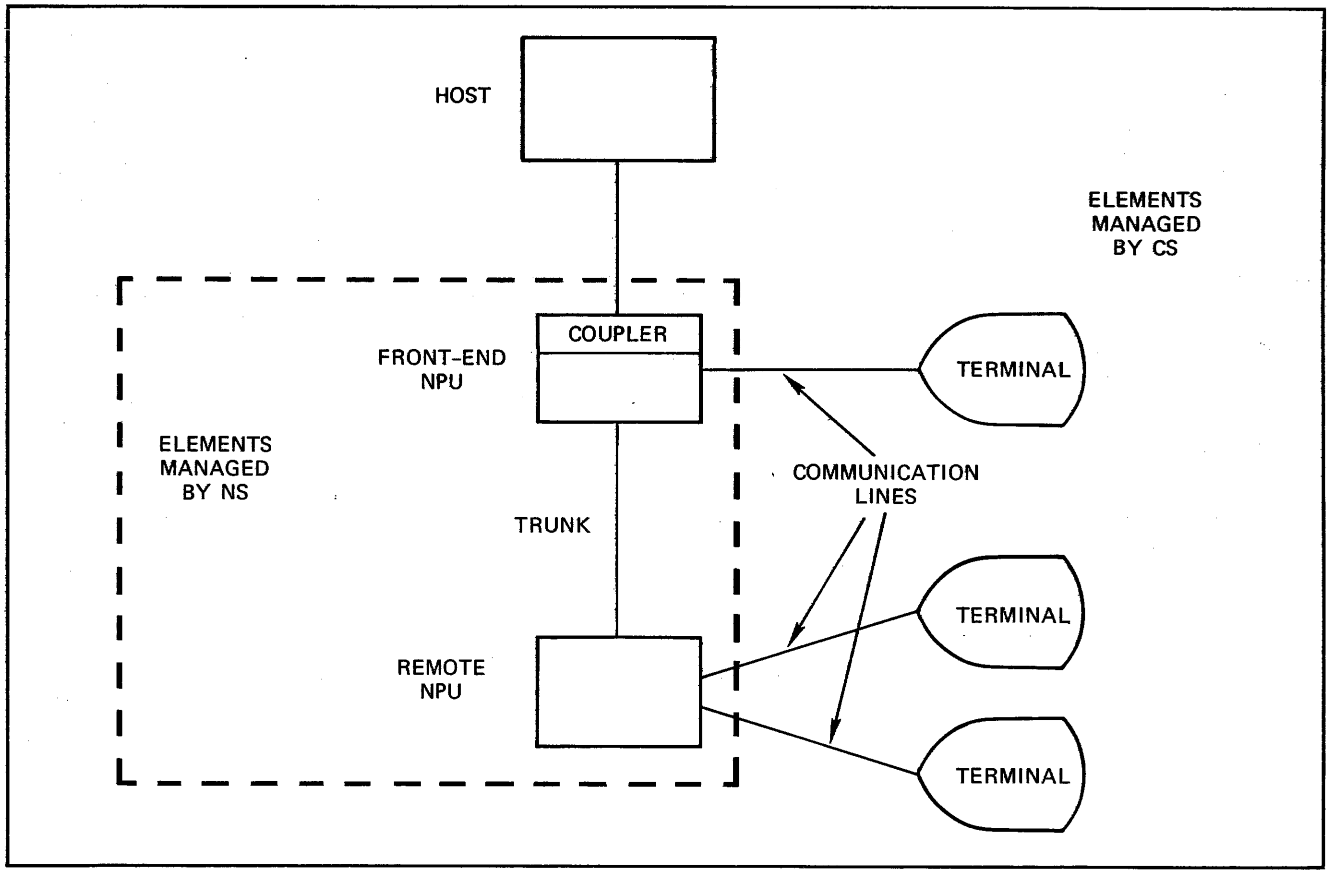


Figure 2-1. Configurable Hardware Elements

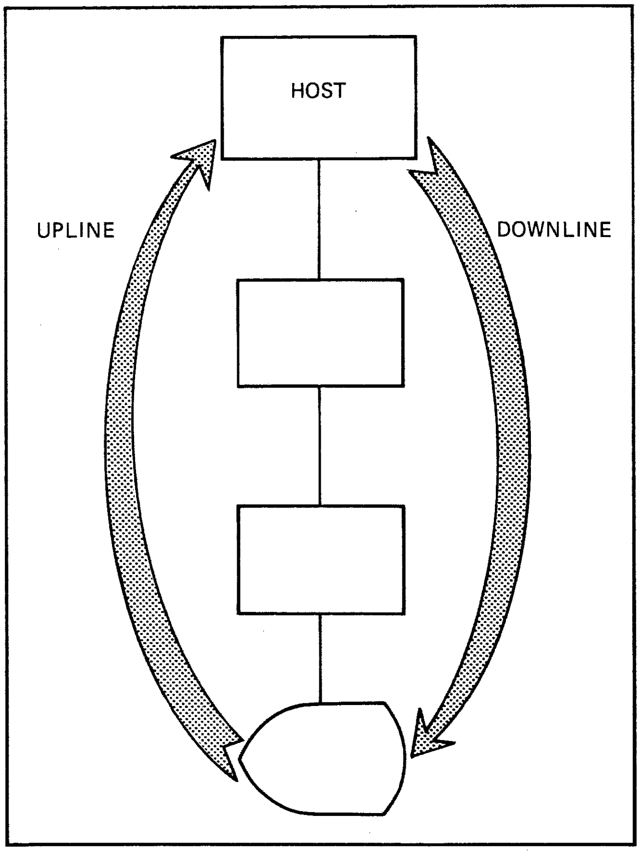


Figure 2-2. Message Flow Direction

COUPLERS

The input/output channel from the host is connected to the front-end NPU via a hardware module known as a coupler. The coupler is physically housed within an NPU, but must be configured separately because some NPUs do not have couplers.

The coupler makes upline data signals compatible with the hardware of the host computer and downline data signals compatible with the NPU. The number and type of couplers configured with a host computer depend, therefore, on the type of host and NPUs to be connected. An NPU can have one or two couplers, each connected to a single, separate host computer. Only one coupler can be active at a time.

NETWORK PROCESSING UNITS

Network processing units can be either CDC communications processors (known by their 255x series numbers) or non-CDC processors with compatible communication procedure software. Unless otherwise qualified, subsequent references to network processing units in this manual apply to both classes of equipment.

A 255x NPU is a 16-bit microprocessor with an expandable macromemory. It functions as a communication switching element, code converter, and input/output concentrator to route and control data flow through the network. Locating these functions in the NPU frees the host processor from performing interface functions, thereby allowing terminal users to more efficiently share the host's resources.

There are two types of 255x network processing units: the 2550 (sometimes referred to as a 2550 Host Communications Processor) and the 2551. A 2550 NPU has at least one coupler. A 2551 can have a coupler but need not. Each 255x NPU can be configured as either a front-end or remote NPU. A 2551 can have twice as many ports as a 2550, but has the same micromemory.

Front-end and remote 255x NPUs can be connected by trunks. You cannot configure two NPUs as connected by more than one trunk. You can, however, configure up to eight trunks to different remote NPUs from a single front-end NPU.

Both front-end and remote NPUs can be connected to terminals through their input/output ports. Terminals can be linked directly to these ports with communication lines, or indirectly through a hierarchy of equipment that permits up to 255 connected communication lines.

If a network processing unit in your network is not CDC equipment, you can configure it as a remote NPU if it is self-loading and runs software using the same communication protocol as CDC NPUs. You can configure it as either a front-end or a remote NPU if the Network Supervisor can load it from the Communications Control Program (CCP) load file.

TRUNKS

A front-end NPU can be connected to a remote NPU by a communication line called a trunk. You can configure only one trunk between any pair of NPUs, but you can configure up to eight trunks connecting different remote NPUs to any single front-end NPU. To enable the Network Supervisor program to keep track of message traffic between the NPUs, the trunk must be configured from the perspective of both NPUs (see Nodes and Logical Links, later in this section). This also eliminates the need for both NPUs to recognize a trunk by the same parameters.

You must make sure that the trunk configuration you define for any NPU is compatible with the CCP macromemory load module of that NPU. For example, if you configure NPU1 with two trunks in your NDL program, the CCP module loaded into NPU1 must also define two trunks.

COMMUNICATION LINES

The method of configuring a communication line between an NPU and a terminal differs from that used to configure a trunk. You define such a line by providing the name of the NPU and the number of the port on that NPU to which the line is connected. That information is stored in the local configuration file for use by the Communications Supervisor during network operation. The port number is the hexadecimal thumbwheel-selected address of the communication line adapter.

You must also configure a communication line in terms of the terminal interface program servicing required for the line, as determined by the protocols of the terminals to be connected to the line.

The network software currently supports lines both for synchronous terminals and for asynchronous terminals. These lines can be switchable (also called dialup) or dedicated (also called hardwired). You can configure a line with a speed of up to 19200 bits per second (bps) as one of six general line types.

TERMINALS

The current network software provides support for four basic types of terminals:

- Asynchronous terminals, such as the CDC 751, utilizing either the ASYNC protocol or the X25 protocol

- Mode 4 protocol synchronous terminals, such as the CDC 200 User Terminal

- Multileaving bisynchronous terminals, such as IBM Corporation's HASP workstations

- Bisynchronous terminals, such as IBM Corporation's 2780 terminal

You can configure up to four more types of terminals with a site-provided TIP for each type.

Terminals are grouped into classes by their specific hardware characteristics. A terminal class parameter is used in the local configuration file to define some physical characteristics of the terminal and to determine the default values to represent characteristics that you choose not to define in your NDL program. The current version of the network software supports 17 terminal classes. These terminal classes are described in greater detail in section 4, Terminal Support.

If you provide your own support software, you can use features of NDL to define up to four additional terminal classes. CDC cannot anticipate the support required for such terminals, and thus cannot document such support. You can find some information pertinent to defining your own terminals in section 4.

SOFTWARE ELEMENTS

Network software exists in two environments: the host and the NPU. Standard CDC software exists in both to support many types of network configurations. You can also provide your own software to support other types of equipment and other uses. The characteristics of software not provided by CDC are unknown, and therefore are not addressed by this manual.

HOST-RESIDENT SOFTWARE

Four types of programs are associated with the host computer. These are the supervisory modules, the network utilities, CDC-written application programs, and application programs that your site provides. For further information on writing your own application programs, see the Network Access Method Reference Manual listed in the preface.

Supervisory Modules

Because of the many potential sources and destinations for data in the network, the network software includes programs that supervise elements and control routing along various data paths. These programs are known as the network supervisory modules; each of them runs at a user control point throughout network operation. The supervisory modules are

- The Network Supervisor (NS)

- The Communications Supervisor (CS)

- The Network Validation Facility (NVF)

NS and CS share the responsibility of managing the network resources. Each has its own domain (as shown in figure 2-1) and controls both the physical and the logical elements within that domain. NVF controls the terminal user's access to application programs.

A network always contains one copy of the Network Supervisor program. The nature of NS is such that it is very busy at network initiation, but is not very active thereafter until the network operator uses it to shut down the network. This module supervises the functioning of the network processing units, couplers, and trunks. It communicates with the network operator, providing status reports about these elements of the network as necessary.

When the network operator initiates the loading of the NPUs, NS creates central memory tables from information in the network configuration file. These tables are analogous to the operating system equipment status table. Then, in accordance with these tables, NS loads the NPUs with software from the CCP load file (created by the Load File Generator utility).

After network operation begins, the network operator communicates with NS to monitor and change the status of the elements in the tables created by NS. Such changes, however, do not affect the network configuration file. NS also writes data to the network dayfile, as necessary, during network operation.

The Communication Supervisor is very active during network operation. Once the network initiation functions have been performed by NS, CS begins configuring the elements that it will supervise. CS constructs memory tables from the information in the local configuration file. It then uses these tables to configure the elements within its domain.

The functions of CS include

- Configuring lines

- Configuring terminals

- Managing connections between terminals and application programs

- Processing commands from the local operator

- Providing the local operator with status reports about the elements it supervises

From the terminal user's point of view, network operation effectively begins when CS has finished its initial configuration functions. The local operator then communicates with CS during network operation to monitor and change the status of elements assigned to his or her host. Such changes affect the central memory tables constructed by the Communication Supervisor; they do not affect the local configuration file.

The Network Validation Facility does not actively manage elements of the network. Its main function is to determine whether a terminal user is allowed to access an application program. Whenever a terminal user requests connection to an application, NVF decides whether that connection will be allowed. That decision is based upon four parameters:

- Family name
- User name
- Password
- Application program name

Initially, you can supply these parameters in the definition of the terminal in your NDL program (see Automatic Login, later in this section). Alternatively, the terminal user can supply them when he or she logs in at the terminal.

When the terminal user wants to be connected to a different application program, he or she provides NVF with any new values required for the parameters above. NVF compares those values with values previously established for the parameters in the system validation (VALIDUz) file.

Since a passive terminal is not equipped to provide the information NVF requires, you must provide that information in your description of the terminal.

NVF also conducts the logout processing for terminals. Upon logout, NVF updates the account dayfile for that terminal.

Utilities

The network utilities are host-resident network programs that perform off-line batch functions related to network operations. There are four network utilities:

- The Network Definition Language Processor (NDLP)
- The CCP Load File Generator (LFG)
- The Network Dump Analyzer (NDA)
- The Debug Log File Processor (DLFP)

NDLP processes your input in the form of an NDL program, as described in section 1. LFG is used to create the Communication Control Program load file, which the Network Supervisor uses to load the NPUs. NDA is used to create a readable listing of an NPU memory dump. DLFP is used to create a readable trace listing from the ZZZZZDN file written by the Application Interface Program (see the Network Access Method Reference Manual listed in the preface).

CDC Network Application Programs

CDC network software contains standard programs to provide processing capabilities for a number of applications. Programs of this type are directly accessible by the terminal user. Five of these programs are currently supported:

- The Interactive Facility (IAF)
- The Remote Batch Facility (RBF)
- The Transaction Facility (TAF)
- The Terminal Verification Facility (TVF)
- The Message Control System (MCS)

IAF provides the terminal user with the ability to create files and programs at the terminal and to execute jobs from the terminal. IAF comprises seven subsystems:

- BASIC, an interactive version of the interpreter for the BASIC programming language
- FORTRAN, an interactive version of the FORTRAN 5 compiler
- FTNTS, an interactive version of the FORTRAN Extended 4 compiler
- EXECUTE, used for running frequently used programs previously stored in the system in object-code form
- NULL, used for quick manipulation of files and clearing of subsystem association
- BATCH, used for entering and executing, one at a time, operating system control statements for any product except other network application programs
- ACCESS, used for communication between two interactive terminals

IAF allows a terminal to be switched easily from one subsystem to another, or from a subsystem to another network application program.

RBF transfers data between remotely located batch terminals and a host computer. It also allows monitoring of a job's progress through the network.

TAF provides on-line transaction processing, allowing the terminal user to manipulate information in a data base directly from the terminal. It is streamlined to require very little dialog to update a data base.

TVF provides the terminal user with the ability to test the terminal's data transmitting and receiving functions.

MCS provides the terminal user with a mechanism for transferring data to a COBOL program or between two COBOL programs.

NPU-RESIDENT SOFTWARE

Each NPU in the network contains a separate copy of the Communications Control Program (CCP) tailored to the specific micromemory and macromemory of that NPU. CCP consists of three major parts:

- The base system, or operating system
- The network communication software for routing messages
- The programs to provide the host-to-NPU, NPU-to-NPU, and NPU-to-terminal interfaces required for your configuration

The base system is a collection of subroutines written in PASCAL. These routines perform several basic operating system functions and manage multiplexing and code conversion.

The communication software performs the routing of data messages and the conversion of those messages into a common (virtual terminal) format acceptable to the host processor. The software also handles service messages for the NPU and its related trunks, lines, and terminals, communicating with the local operator when necessary.

Another important function of the communication software is the buffering of data for the terminals serviced by the NPU. This buffering is done in terms of blocks of data. The amount of buffering a given copy of CCP can perform depends on the size of the NPU's macromemory. To keep your terminals busy, you will specify blocking parameters for each terminal in your NDL program.

If the network processing units in your network are CDC 255x communication processors, you will probably be concerned with only the interface programs. These are the HIP, LIP, and TIP modules depicted in figure 2-3.

Your concern with the HIP and LIP modules is likely to be limited to ensuring that the CCP load file for each NPU contains only the necessary modules. For example, NPU2 is a remote NPU, so it does not need a HIP. Since NPU3 does not service a remote NPU, it does not require a LIP. Since a LIP requires about 8000 words of memory, you would want a CCP variant for NPU3 that does not include a LIP.

The TIPs, however, exist to provide the interface for a wide variety of terminals. Since the terminals supported vary from network to network, and even within a single network, the TIPs required by the network are more subject to change than the other interface modules. Thus, when you define a communication line in your NDL program, you must identify the TIP support required for that line; the LINE and TERMINAL statements include parameters for this purpose.

For more detailed information about CCP, refer to the reference manual listed in the preface.

LOGICAL ELEMENTS

In addition to the hardware and software elements of your network, you must define certain logical elements. These logical elements help establish the association between terminals and application programs, as well as the communication paths between them. Using these logical elements, the network software can perform as though it was independent of the physical organization of the network.

NODES AND LOGICAL LINKS

A logical path concept is used to describe the connections between elements of the network in a manner that is independent of the physical data paths between them.

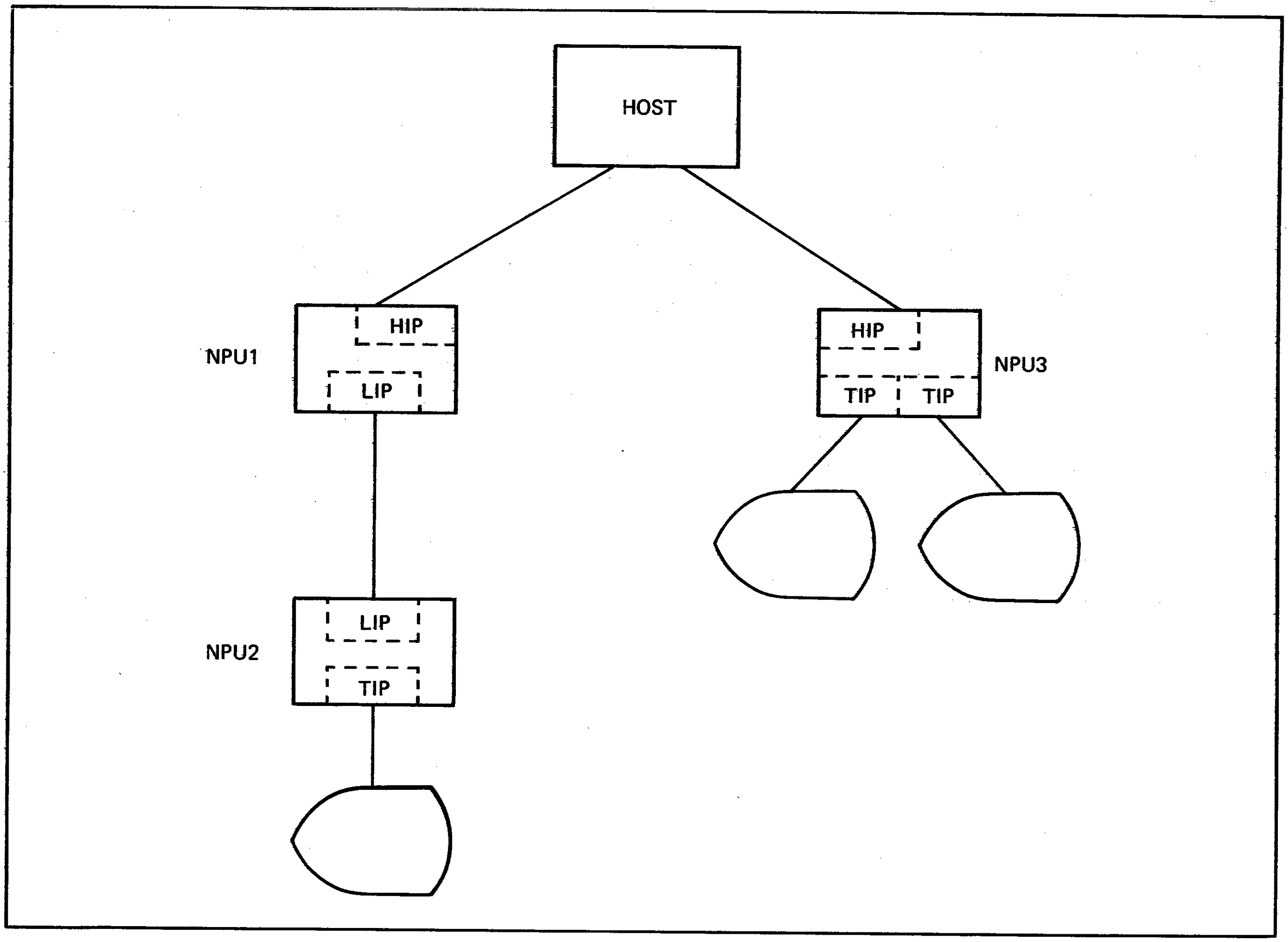


Figure 2-3. CCP Interface Program Configuration

Each junction of possible branches in these logical paths must be known to the network. These junctions are called network nodes. In your NDL program you identify each network node by assigning it a unique node number from 1 to 255. The Communications Control Program uses these numbers to route messages from a source node to a destination node.

Table space for node numbers is allocated sequentially. You can minimize memory resource requirements by assigning node numbers sequentially, giving host nodes lower values than terminal nodes. Figure 2-4 shows the nodes (node numbers are circled) in Simnet, the simple network described in section 1.

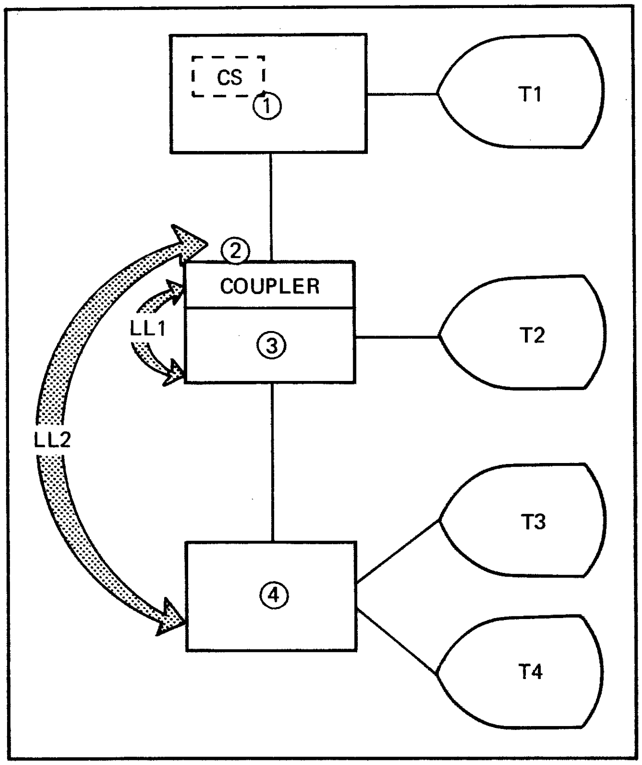


Figure 2-4. Nodes and Logical Links

Because the Communication Supervisor is a point of control for messages to or from the application programs, it is considered a node and is assigned a node number, which is considered the node number of the host. That number is 1, by convention.

Node number 2 is assigned to the coupler. Because of the potential existence of two couplers, each connected to a different host, the number assigned to the coupler is known as the host node. Note that the front-end NPU is a separate node from its coupler. A front-end NPU is serviced by application programs, yet it can also service terminals; thus, the NPU must be separately addressable from its coupler(s).

Node 3, the front-end NPU, is known as either a terminal node or a neighbor node, depending on the logical path being defined. In the logical path from the host to terminal T2, it is a terminal node. In the logical path from the host to terminal T4, it is known as a neighbor node. In this case, the remote NPU (node 4) is the terminal node.

The distinction between a neighbor node and a terminal node is essential to the definition of trunks and logical elements known as logical links. A logical link is a message path between a host node and a terminal node. You define a trunk in your NDL program by providing the neighbor node of the front-end NPU and the terminal node of the remote NPU. To define a logical link, you provide the host node and terminal node of the link.

Figure 2-4 shows the two logical links in the network, LL1 and LL2. A message from terminal T2 travelling to an application has a source node of 3 and a destination node of 2. A downline message to terminal T4 from an application has a source node of 2 and a destination node of 4.

Node numbers are assigned as build-time parameters. NDL and the network software allow node numbers up to 255. The MAXNN installation parameter, however, declares the highest node number permitted by the installation.

Your NDL program establishes the correlation between node numbers and the configured host computer or NPU. The Network Supervisor obtains information about this correlation from the network configuration file. Once a logical path is established, the node numbers are used for addressing as described above.

LOGICAL CONFIGURATION OF HOSTS

You must describe a host computer in your program in terms of the application programs it is to run. A host data section is provided in the NDL program structure for this purpose.

LOGICAL CONFIGURATION OF TERMINALS

In addition to the hardware description of each terminal, you must supply information concerning its ability to access network resources. This information includes

- Operating system access security
- Manual login procedure
- Control by an interactive console (owning console)

You describe the terminal's logical relationship to the network by declaring values for parameters in the TERMINAL statement for that terminal.

Automatic Login

When an NPU reports an operational communication line to the Communications Supervisor, the Supervisor connects all terminals on that line with NVF. If a terminal is a passive device, the Communications Supervisor must find a family name, user name, and application program name to pass to NVF for validation of the terminal. The Communications Supervisor obtains these values from the automatic login family name (ALFAM), automatic login user name (ALUN), and initial application program name (IAPPL) parameters that you specify for the terminal in the local configuration file.

If a terminal is an interactive device, the Communications Supervisor passes to NVF any of the automatic login or initial application program parameters it finds defined for the terminal in the local configuration file; NVF then requires the terminal user to enter any of the parameters not passed to it. NVF ignores any redundant parameters typed in by the terminal operator (through an abbreviated login sequence) at this stage. When NVF has verified all of the parameters it obtains and requires no others, it causes the Communications Supervisor to connect the terminal to the identified application program. The above description holds for the initial login attempt only. When NVF processes additional login attempts for an interactive terminal (because of an unsuccessful initial attempt, or termination of the initial connection), NVF sometimes disregards the auto-login parameters, and requests that they be typed in from the terminal.

Owning Consoles

A passive terminal is incapable of conducting dialog with an application program to control its own functioning. Each passive terminal, therefore, must be defined in the local configuration file as associated with a specific interactive terminal (console) that enters commands controlling it. This association is made by specifying the name of the console terminal that is to own (or control) the passive terminal. The declaration of an owning console terminal (OCTERM) is required for passive terminals. This method permits a single, specific owning console to log in with any legal family name and user name combination. For BSC terminals, you must provide a statement defining a logical terminal as the owning console, even though the physical terminal does not exist.

There are no actual interactive terminals in the BSC protocol. You can, however, simulate an interactive terminal by using a card reader and a line printer.

Passive terminals cannot be declared as owning consoles for interactive terminals. Interactive terminals cannot be declared as owning consoles for other interactive terminals. The interactive terminal defined as an owning console cannot be disabled unless all passive terminals for which it is the owning console are also disabled.

THE NETWORK DEFINITION LANGUAGE 3

This section presents the format and functional syntax of the Network Definition Language. The NDL program statements you will use to create the network definition files are described in separate subsections.

All the examples in this section refer to the hypothetical network CDCnet, shown in figure 3-1.

FUNCTIONAL SYNTAX AND FORMATS

Each NDL statement has the general form given in figure 3-2. Separators, terminators, and format representation conventions are explained on the Notation page at the front of this manual. The formal syntax of the Network Definition Language (in Backus-Naur form) and a concise summary of each statement's syntax and format are given in appendix E.

Unless otherwise indicated in the format descriptions presented later in this section, all values declared for keywords are either unsigned decimal integers or character strings of one through seven letters and digits. Element names are also restricted to one initial letter and up to six more letters and/or digits. All value declarations identified as hexadecimal integers are also unsigned. NOS user names declared as values can contain asterisks.

VALUE DECLARATION FORMATS

Four formats can be used when declaring values. These formats depend upon the presence or absence of keywords and values. Declarations that include the keyword are called explicit declarations, and declarations that omit the keyword are called implicit declarations. The value portion of the declaration can be of two forms: it can be single-valued; or, it can be null-valued, in which case the keyword in both explicit and implicit declarations becomes the value.

Explicit single-value declarations look like this:

```
    LABEL: EXAMPLE KEYWORD1=VALUE1,

        KEYWORD2=VALUE2.
```

Implicit single-value declarations look like this:

```
    LABEL: EXAMPLE VALUE1,VALUE2.
```

Explicit keyword value declarations look like this:

```
    LABEL: EXAMPLE KEYWORD1,KEYWORD2.
```

Implicit keyword value declarations look like this:

```
    LABEL: EXAMPLE,,.
```

ORDERING VALUE DECLARATIONS

The value declaration portion of NDL statements permits you to use three different ordering conventions:

- keyword$_1$=value$_1$,keyword$_2$=value$_2$, keyword$_3$=value$_3$.

- value$_1$,value$_2$,value$_3$.

- keyword$_1$=value$_1$,value$_2$,keyword$_3$=value$_3$.

The first convention is called order-independent and is the one used throughout this manual. The second convention is called order-dependent. The third is a combined convention.

Using an order-independent convention, explicit single-value or keyword value declarations can be made in any order because the keyword identifies each value being declared. With an order-dependent convention, the implicit values that you declare are assigned to particular keywords in a predetermined and fixed order.

Each statement is known by the NDL Processor to have a specific set of keywords with optional default values, and those keywords have a defined sequence of occurrence. The NDL Processor also maintains a keyword pointer. This pointer is set to the appropriate keyword when a keyword is encountered. When a keyword is not specified, the pointer is incremented to identify the next keyword in its internal list of keywords. The value specified is then associated with the keyword identified by the pointer. Thus, the statements shown in both the order-independent and order-dependent conventions above are equivalent when the sequence of value declarations known to the NDL Processor is keyword$_1$,keyword$_2$,keyword$_3$.

To correctly use any order-dependent or combined convention declarations, you must know the order of keywords that will be used by the NDL Processor. The NDL Processor uses the order presented in the statement summaries in appendix E.

Omission of the keyword$_i$=value$_i$ or value$_i$ declaration causes the NDL Processor to substitute a default value (if any) for value$_i$. Omission is accomplished in the first ordering convention by omitting the declaration altogether; in the second, omission is signified by sequential commas.

For example, consider a statement such as:

LABEL: EXAMPLE KEYWORD1=value$_1$,
KEYWORD2=value$_2$,
KEYWORD3=value$_3$.

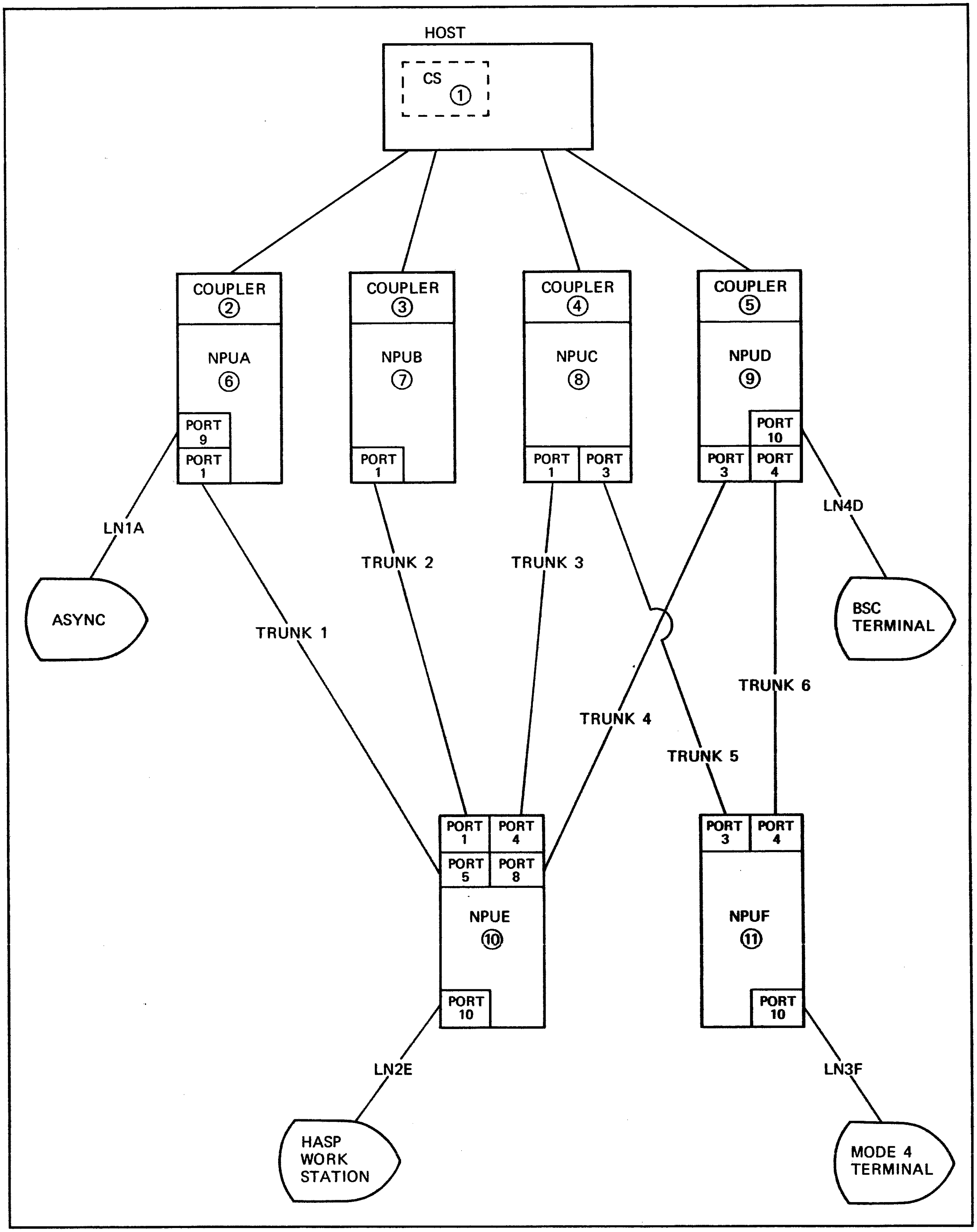


Figure 3-1. CDCnet - Configuration Example

$$\begin{bmatrix}\text{element-name:}\\ :\end{bmatrix}\ [\Delta]\ \text{reserved-statement-name}\Delta \left[\begin{Bmatrix}\text{keyword}_1=\text{value}_1\\ \text{keyword}_1\\ \text{value}_1\end{Bmatrix}, \ldots, \begin{Bmatrix}\text{keyword}_n=\text{value}_n\\ \text{keyword}_n\\ \text{value}_n\end{Bmatrix}\right].$$

element-name — The name given to the entity being specified. When the NDL processor requires an element name, the name supplied must be unique within a division and cannot duplicate any reserved word (see appendix D). Element names comprise one alphabetic character optionally followed by up to six alphabetic or numeric characters. The local and network operators use the element name to monitor and alter the status of the named entity.

: — The colon separates the element name from the reserved statement name. When an element name is specified, a colon must immediately follow it. The colon does not become part of the element name. (A colon can precede the reserved statement name even if an element name is not included in the statement.) A blank can separate the colon from the reserved statement name.

Δ — This character represents one or more blanks. Blanks and commas are separators and are required where shown in the prototype statement formats; blanks can be used where commas are shown, except to delimit an omitted parameter. One blank is sufficient as a separator, and multiple blanks are treated as one. One or more blanks preceding or following a comma are ignored.

reserved-statement-name — A reserved word specifying the statement type. Each statement type has a unique statement name.

$\text{keyword}=\text{value}_i$
keyword_i
value_i — A value declaration of the statement. This has four possible formats, depending upon the keyword, inclusion of the keyword in the declaration, and the network being specified (see Value Declaration Formats).

, — Commas, along with blanks, are separators. Separators are required to separate declarations. A comma preceded or followed by one or more blanks is treated as a single separator. A comma indicates that the NDL Processor should interpret whatever follows it as a new keyword or value; that is, if two consecutive commas are specified, an implicit declaration exists between them. Implicit declarations cannot be made with consecutive blanks.

. — A period is required to terminate all statements.

Figure 3-2. Functional Syntax of NDL Statements

The default for value_2 is 100, but value declarations for KEYWORD1 and KEYWORD3 are required. The statement can be written correctly in any of the following ways:

```
LABEL: EXAMPLE KEYWORD1=10,KEYWORD3=T47.

LABEL: EXAMPLE 10,,T47.

LABEL: EXAMPLE KEYWORD3=T47,KEYWORD1=10.

LABEL: EXAMPLE 10,75,T47.

LABEL: EXAMPLE KEYWORD2=90,T47,
   KEYWORD1=10.
```

All of the above statements are equivalent, with the exception of the value_2 portion, which is 100, 100, 100, 75, and 90, respectively.

The statement

```
LABEL: EXAMPLE,,KEYWORD3=T47.
```

which omits declarations for KEYWORD1 and KEYWORD2, is incorrect because the NDL Processor has no default value for KEYWORD1 and would issue a diagnostic message. The statement

```
LABEL: EXAMPLE KEYWORD1=10,,
   T47,KEYWORD4=7.
```

is incorrect because KEYWORD4 is unknown to the NDL Processor, and only three value declarations are expected for the statement EXAMPLE. This occurrence would also cause the NDL Processor to issue a diagnostic message.

STATEMENT LENGTH AND CONTINUATIONS

NDL statements are assumed to be input from either card readers or terminal keyboards. Because these input sources are physically different, there are no physical restrictions (such as card column or line tabulator position usage) on the format of the statements. You can define your own physical formatting conventions, with the following exceptions:

- NDL statements are divided into units of 80 or fewer characters. Characters 73 through 80 of each unit are reserved for your convenience as a sequence number field and are not interpreted by the NDL Processor. Characters in the sequence number field are included in the count of characters in each statement unit, however. Any statement of more than 72 meaningful characters must be split into units so that the meaningful portion of each unit ends with a legal separator.

- You cannot divide a value declaration over two units.

- A period terminates a statement; failure to terminate a statement with a period causes the NDLP to assume that the next statement is a continuation of the present one, producing unpredictable errors.

- There is no limit on the number of units into which a statement can be broken. Completely blank units are permitted.

- No more than one statement can be declared in a single unit.

These constraints are illustrated by the examples in figure 3-3. There are no field restrictions imposed on an NDL statement; its input image is essentially free-field.

DEFINITION STRUCTURE

The structure of the NDL program that defines your network will parallel the physical and logical structure of the network. An NDL program can contain two formal types of divisions:

- A network division, corresponding to the network configuration file to be produced

- A local division, corresponding to each local configuration file to be produced

Each of these divisions comprises several formal sections.

Initially, you must provide at least one network division and one local division to configure your network. In subsequent programs for reconfiguring your network, it is not unusual to include only one division to redefine the associated network definition file.

SECTION AND DIVISION HIERARCHY

You must specify sections within their divisions in the order shown in figure 3-4, but you can specify the divisions in any order. Divisions are always terminated by END statements.

An NDL input file can consist of several network and local divisions. NDLP creates one network configuration file for each network division it encounters while processing your program. NDLP also creates one local configuration file for each local division in your program.

STATEMENT HIERARCHY

Statements must occur within the proper section and division. All statements following a HOST, NPU, HOSTDATA, NPUDATA, or LINE statement must apply to the entity identified by that statement; these statements serve as subdividers within sections. The statements shown in figure 3-4 must be ordered in the hierarchy shown.

DEFINE statements can occur anywhere within any division but apply only to sections in the same division. DEFINE statements must always precede any references to them within the input stream. These statements are, therefore, best placed immediately following the NFILE or LFILE statements.

COMMENT statements can occur anywhere within a division or between divisions. An END statement can only be the last statement of a division.

The following three-unit statement is correct:

```
GOODTRY:ΔCOUPLERΔNPUNAME=NPU1,Δ. . .Δ

Δ. . .Δ

NODE=2.
```

The following two-unit statement is incorrect because it divides a value declaration with a separator (a blank):

```
BADTRY1:ΔCOUPLERΔNPUNAME=NPUΔ. . .Δ

1,ΔNODE=2.
```

The following two-unit statement is incorrect because it omits the period terminator:

```
BADTRY2:ΔCOUPLERΔNPUNAME=NPU1,Δ. . .Δ

NODE=2
```

The following one-unit statement is incorrect because it contains more than 72 characters, and it is not divided into two units at a separator:

```
COMMENT:ΔTHISΔISΔBADTRY3,ΔAΔSTATEMENTΔNOTΔINDICATINGΔDIVISIONΔBYΔAΔSEPARATOR.
```

Figure 3-3. Statement Continuation Examples

```
Network Division
    NETNAME statement
    NFILE statement
    Host Section
        HOST statement
        COUPLER statement
        LOGLINK statement
    NPU Section
        NPU statement
        TRUNK statement
    END statement

Local Division
    LOCAL statement
    [NFILE statement]
    LFILE statement
    Host Data Section
        HOSTDATA statement
        APPL statement
    NPU Data Section
        NPUDATA statement
            LINE statement
                TERMINAL statement
    END statement
```

Figure 3-4. NDL Program Statement Hierarchy

NETWORK DIVISION STATEMENTS

The network division describes the major components of the entire network (its host computers and network processing units) and the physical and logical connections among them. The network division consists of two sections:

- The host section, which specifies and names the host computers, couplers, and logical links by which the host computers are connected to the data communications network
- The NPU section, which specifies and names the network processing units and the trunks connecting them

You can omit the network division from an NDL program if no network configuration file is to be created; otherwise, one network configuration file is created for each network division you define.

DIVISION IDENTIFICATION

The host section is the first section of the network definition, but statements must precede it to name the network and the network configuration file that is to be created; these statements also signal that the NDL input contains a network division. If you omit these statements, a network configuration file is not created, and any network division statement you include causes an error.

NETNAME Statement

Use the NETNAME statement to name your network. Its format is given in figure 3-5. The name you declare for the network appears in the NAM dayfile when the network is initiated, and in the NDL dayfile messages. An example of a valid NETNAME statement is

```
NETWORK: NETNAME.
```

```
netname:NETNAME.

netname   The name to be assigned to the network.
          This name is required; there is no default
          value.
```

Figure 3-5. NETNAME Statement Format

NFILE Statement

The NFILE statement identifies the network configuration file to be created. Its format is shown in figure 3-6. An example of a valid NFILE statement is

```
NETFILE: NFILE.
```

```
nfile:NFILE.

nfile     The local file name of the new network
          configuration file to be created. This name
          is required; there is no default value.
```

Figure 3-6. NFILE Statement Format

The network configuration file identified by the NFILE statement is opened without rewinding. The NOS local file name specified in the NFILE statement should not be the same as that of an existing NOS local file. The network configuration file is described in more detail in section 5.

HOST SECTION

The host section is the first section of any network definition. In the host section you define the relationship of the host to the physical and logical elements of the network. The host section identifies the host's couplers, the NPUs to which the couplers are connected, and the message paths used by the host.

The host's relationship to the physical network is specified with two types of statements. The HOST statement provides information regarding the host processor, and the COUPLER statements (one for each coupler) describe the couplers connecting the host to the network. You must give names to both the host and its couplers; the network operator identifies these elements by these names.

In the host section you must also name and define the logical links used by the host. The LOGLINK statement is provided for this purpose.

HOST Statement

The HOST statement has the format shown in figure 3-7. You must specify one HOST statement for the host computer in the network (the network software currently supports only one host computer); all statements between the first HOST statement and the next HOST statement, or the end of the host section, must apply to the same host computer. For example, the HOST statement

```
HOST1: HOST NODE=1.
```

identifies a computer with the name of HOST1 as a host computer within the network. This host contains a copy of the Communications Supervisor that is addressable as node 1.

host:HOSTΔNODE=node.

host The name to assign to the host computer. This name is required; there is no default value.

node The node number, 1≤node≤MAXNN, and node ≤255, by which the NPU software can reference the copy of the Communications Supervisor resident in this host computer. This number is defined in the CCP release tape assembly. All values declared for NODE keywords must be unique within the current NDL program division. This number is required; there is no default value.

Figure 3-7. HOST Statement Format

COUPLER Statement

The COUPLER statement has the form shown in figure 3-8. One COUPLER statement is required for each host node coupler in the network. The maximum number of COUPLER statements permitted in an NDL program division is described in appendix F. The COUPLER statement identifies the coupler to both the NPU and host computer software. For example, the COUPLER statement

CPLR1: COUPLER NPUNAME=NPUA,NODE=2.

identifies coupler CPLR1 as a host node with a node number of 2 and connected to the NPU named NPUA. This coupler is physically located in NPUA.

coupler:COUPLERΔNPUNAME=npu,NODE=node.

coupler The name to assign to the front-end coupler being configured. This name is required; there is no default value.

npu The element-name value declared in the NPU statement that defines the NPU to which this coupler is connected. The value declared as npu must be a value declared in an NPU statement within the same network division of the NDL program. This name is required; there is no default value.

node The host node number, 1≤node≤MAXNN, and node≤31, by which the NPU software references this coupler. This number is defined in the CCP release tape assembly. All values declared for NODE keywords must be unique within the current NDL program division. This number is required; there is no default value.

Figure 3-8. COUPLER Statement Format

The value used for the NODE declaration must be the value assigned as the node number in the NPU's equipment status table (EST) entry. It must also be the same as the value assigned in the network definition table during assembly of the CCP modules loaded into the specified NPU. In the above example, a NODE value of 2 was used because the corresponding CCP variant definition was known to have assigned (via the CP parameter) a value of 2 for the user build parameters file (USERBPS). For a more detailed description of the interdependence of NDL and USERBPS, see the NOS 1 Installation Handbook.

The coupler's status at network initiation is controlled by its EST entry.

LOGLINK Statement

The LOGLINK statement has the format shown in figure 3-9. You must supply one LOGLINK statement for each logical link in your network, including those within a front-end NPU between the host node (coupler) and the terminal node (NPU). For example, in CDCnet (figure 3-1) NPUA (node 6) connects to a terminal and to NPUE (node 10). The host node in NPUA is node 2. The host node number is always the number associated with a coupler. The node number of the Communications Supervisor is not considered a host node number, and is never used in defining a logical link. The following LOGLINK statements would properly configure both logical links involving NPUA:

LINK1: LOGLINK HNODE=2, TNODE=6.

LINK2: LOGLINK HNODE=2, TNODE=10.

loglink:LOGLINKΔHNODE=hnode,TNODE=tnode[,DI].

loglink The name to assign to the logical link being configured. This name is required; there is no default value.

hnode The host node number, 1≤hnode≤MAXNN, and hnode≤31, of the coupler that is the host computer interface for the logical link. This node number must be a value declared for the NODE parameter in a COUPLER statement within the same network division of the NDL program. This number is required; there is no default value.

tnode The terminal node number, 1≤tnode≤MAXNN, and tnode≤255, of the NPU that is the terminal interface for the logical link. This node number must be a value declared for the NODE parameter in an NPU statement within the same network division of the NDL program. This number is required; there is no default value.

DI A reserved word that specifies the status assigned to this logical link at network initiation. This word is optional; if it is omitted, an initial status of enabled is assumed. When DI is specified, the logical link is disabled at network initiation and cannot be used until the network operator enables it.

Figure 3-9. LOGLINK Statement Format

These logical links are shown in figure 3-10.

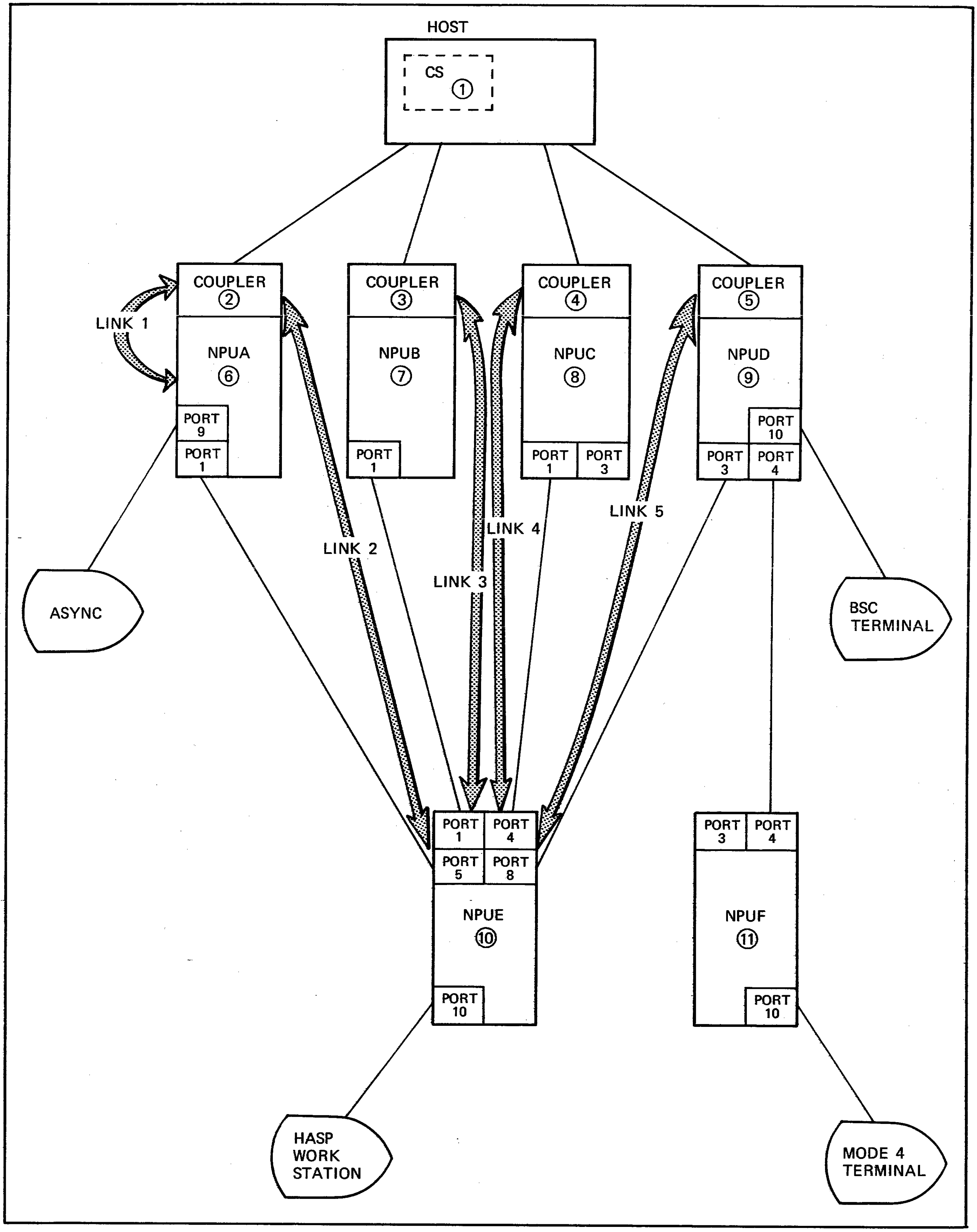


Figure 3-10. Logical Links

Only one logical link can be defined between any given pair of host and terminal node numbers. As shown in figure 3-1, it is possible to have up to four logical links between each remote terminal node NPU and the host computer. Each such logical link would use a separate trunk between the terminal node NPU and a different front-end NPU. NPUA, NPUB, NPUC, and NPUD are front-end NPUs containing the couplers with host node numbers of 2, 3, 4, and 5, respectively. The statements

```
LINK2: LOGLINK HNODE=2, TNODE=10.

LINK3: LOGLINK HNODE=3, TNODE=10.

LINK4: LOGLINK HNODE=4, TNODE=10.

LINK5: LOGLINK HNODE=5, TNODE=10.
```

would legally configure remote terminal node NPUE with four logical links, all enabled. The logical link actually used to address terminals at NPUE would depend on which of the physical connections between NPUE and the front-end NPUs were configured and enabled. If all the physical connections were configured and enabled, the logical link used would be chosen on the basis of load-leveling within the data communication network.

The maximum number of LOGLINK statements permitted in an NDL program division is described in appendix F.

NPU SECTION

The NPU section is the second section in a network division. In this section you identify each NPU in your network, specify the CCP software modules used in each NPU, and define physical linkages between the NPUs.

NPU Statement

You must define each NPU in your network with an NPU statement of the form shown in figure 3-11. The maximum number of NPU statements permitted in an NDL program division is described in appendix F.

Each front-end NPU must be referred to by a COUPLER statement in the same network division. Each NPU connected to another NPU must be referred to by a TRUNK statement (described below) in the same network division. These references are shown in figure 3-12. The coupler statement COUPLR2 refers to the NPU statement defining NPUA, the front-end NPU that contains the coupler with host node number 2. This reference is made with the value declaration NPUNAME=NPUA.

```
npu:NPUΔ[P1LID=p1lid,P2LID=p2lid,] NODE=node[,DUMP,SIZE=size,NPUTYPE=nputype,DI].
```

npu — The name of the NPU being configured. This name is required; there is no default value.

p1lid — The three-character name of the micromemory module (within the CCP load file) that should be used for the phase 1 load of this NPU. Currently, the only value acceptable to CCP is MIC. To use any other value successfully, you must alter build procedures CCPLOAD and CCPPH1. This parameter is required for a 255x NPU and is not permitted for an AUTO NPU.

p2lid — The three-character name of the macromemory module (within the CCP load file) that should be used for the phase 2 load of this NPU. The name declared for p2lid must identify the correct macromemory module for the node number specified in this NPU statement. This parameter is required for a 255x NPU and is not permitted for an AUTO NPU.

node — The network node number, 1≤node≤MAXNN, and node≤255, by which the network software references this NPU. This number is defined as NPUID in the CCP release tape assembly. All values declared for NODE keywords must be unique within the current network division. This number is required; there is no default value.

DUMP — A reserved word controlling the setting of the NPU dump flag in the network configuration file. If DUMP is specified, the NPU dump flag is set, enabling dumps of the NPU. If DUMP is not specified, the NPU dump flag is not set, disabling dumps of the NPU. The dump flag setting in the network configuration file determines the initial setting of the Network Supervisor dump flag, which controls NPU dumps. The network configuration file dump flag can affect only the first opportunity to dump the NPU. See the text.

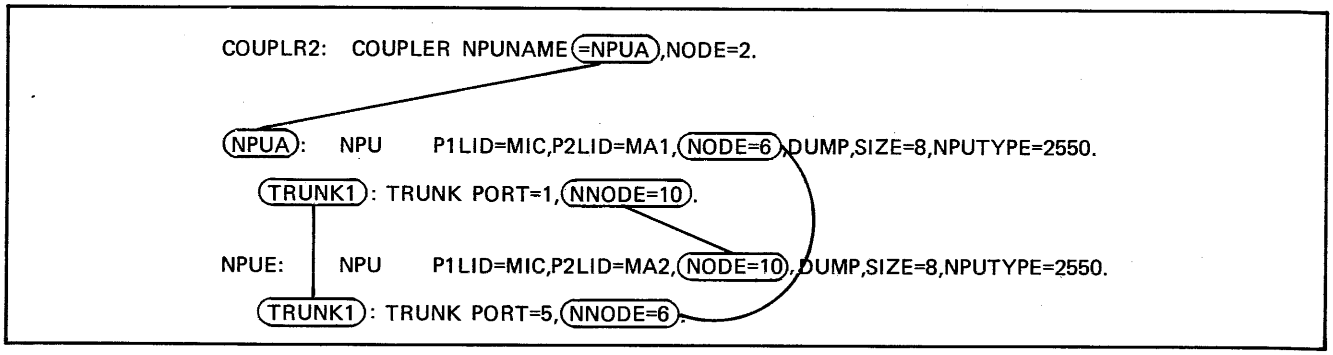
size — Indicates the size of the NPU's macromemory in multiples of 8192 words; 4≤size≤16, with a default value of 8 (65 536 words). SIZE is only used for dumping the NPU memory. The DUMP and SIZE parameters should only be used for 255x NPU s.

nputype — A reserved word that specifies the micromemory type of the NPU being configured. The types that can be specified are:

- 2550 — The NPU is a 2550 or 2551 that is loaded by the Network Supervisor; this is the default value.
- AUTO — The NPU is loaded off-line and joins the network without Network Supervisor loading.

DI — A reserved word that specifies the status assigned to this NPU at network initiation. This word is optional; if it is omitted, an initial status of enabled is assumed. When DI is specified, the NPU is disabled and cannot join the network until the network operator enables it.

Figure 3-11. NPU Statement Format



```
          COUPLR2:  COUPLER NPUNAME=NPUA,NODE=2.

          NPUA:   NPU    P1LID=MIC,P2LID=MA1,NODE=6,DUMP,SIZE=8,NPUTYPE=2550.
            TRUNK1: TRUNK PORT=1,NNODE=10.

          NPUE:   NPU    P1LID=MIC,P2LID=MA2,NODE=10,DUMP,SIZE=8,NPUTYPE=2550.
            TRUNK1: TRUNK PORT=5,NNODE=6.
```

Figure 3-12. NPU Statement Reference Requirements

A trunk must be described from both ends. Thus, the TRUNK statement associated with NPUA declares node 10 (NPUE) as its neighbor node. Likewise, the TRUNK statement associated with NPUE declares node 6 (NPUA) as its neighbor node. Both TRUNK statements define the same trunk; therefore, they are given the same element name (TRUNK1).

NDLP attempts to ensure a proper reference by using the NPUTYPE parameter for diagnostic processing of the network division. The NPUTYPE declaration identifies the micromemory type of the NPU, rather than the model type of the NPU. All CDC 255x network processing units have a micromemory type of 2550. Non-CDC equipment has an unknown micromemory type of AUTO; because such NPUs can only be remote NPUs, an NPU statement with an NPUTYPE declaration of AUTO must be referred to by a TRUNK statement.

When the reference requirement described above is satisfied by a COUPLER statement, the element name declared on the NPU statement must be used as the NPUNAME declaration of the COUPLER statement. When the reference requirement is satisfied by a TRUNK statement, the NODE parameter declared on the NPU statement must be used as the NNODE declaration of the TRUNK statement.

The phase 1 and phase 2 load ID parameters specify, respectively, the micromemory and macromemory software modules loaded by the Network Supervisor into the 255x Series NPU being defined. These parameters connect the host software and 255x Series NPU software installation procedures and ensure that the correct software resides in each operating 255x Series NPU.

The phase 2 load ID specified depends on the macromemory copy preconfigured for the NPU. Each NPU requires a different macromemory module and requires declaration of a different load ID. The phase 1 and phase 2 load IDs declared must be memory resident module identifiers within the CCP load file used to initiate the version of the network being defined by this network division.

A non-CDC remote NPU can be loaded off-line and subsequently join the network; in this case, the Network Supervisor cannot load either the 255x Series micromemory or macromemory modules into the NPU. You must assign such NPUs an NPUTYPE of AUTO and omit the load ID parameters from the NPU statement.

For 255x Series NPUs, the value that you declare for the NODE parameter must be the value assigned in the USERBPS file for the Communications Control Program module loaded into the NPU. The correspondence between NDL and USERBPS is explained in detail in the NOS Version 1 Installation Handbook section on CCP installation.

The Network Supervisor can dump the memories of 255x Series NPUs prior to loading them. For the Network Supervisor to dump an NPU, the Network Supervisor dump flag for an individual NPU must be set. The Network Supervisor dump flag is set if:

- You specify the DUMP parameter in the NPU statement defining the NPU. If the DUMP parameter is specified, the Network Supervisor dump flag for the NPU is set when the Network Supervisor is first brought up.
- The network operator sets the dump flag with the DUMP command. See the NOS operator's guide for more on this command.
- The Network Supervisor automatically sets the dump flag; it does so after each successful load of the NPU, after it recovers the NPU following a NAM recovery, and at network initialization if the previous NAM run aborted.

If you do not specify the DUMP parameter in the NPU statement, the dump flag is not set at network initiation. The network operator can unset the dump flag with the DUMP command. The effect of the DUMP parameter and command, however, lasts only until the Network Supervisor automatically sets the dump flag or until the network is brought down.

Because AUTO type NPUs are not loaded by the Network Supervisor, specification of the DUMP parameter for an AUTO type NPU does not result in an initial dump of the NPU. Dumps of AUTO type NPUs can only be obtained by off-line mechanisms.

In the sample network CDCnet, if NPUE is a cassette-loaded NPU, the following statements correctly define NPUA and NPUE.

```
NPUA: NPU P1LID=MIC,P2LID=MAA,NODE=6,
    NPUTYPE=2550.
    .
    .
    .
NPUE: NPU NODE=10,NPUTYPE=AUTO.
```

Both NPUs are configured with the default macromemory size of 65536 words, and neither is initially dumped by the Network Supervisor. The 2550 is loaded by the Network Supervisor with micromemory module MIC from the CCP load file, and with macromemory module MAA from the same load file. Both NPUs are enabled by default when they join the network.

If NPUE is a 2551, it can be similarly configured by the statement

```
NPUE: NPU P1LID=MIC,P2LID=MAE,NODE=10,
    NPUTYPE=2550.
```

TRUNK Statement

You must use a pair of TRUNK statements to fully define a trunk, as shown in figure 3-12. The maximum number of TRUNK statements permitted in an NDL program division is described in appendix F. Each TRUNK statement has the format shown in figure 3-13.

trunk:TRUNKΔPORT=port,NNODE=nnode.

trunk — The name to be assigned to the trunk communication line being defined. This name must also be specified for a TRUNK statement defining a trunk line connected to the neighbor NPU with the node number specified by the NNODE parameter of this TRUNK statement. There must be exactly two TRUNK statements declaring the same element name in a single network division. This name is required; there is no default value.

port — The number 1≤port≤4 (remote NPUs) or 1≤port≤ 8 (front-end NPUs), of the port to which this trunk connects on the NPU currently being defined. All values declared for PORT parameters of enabled elements must be unique within a given network definition of each NPU. This number cannot be the same as the number declared for a port in any other TRUNK or LINE statement configuring an enabled element on this NPU. This number is required; there is no default value.

nnode — The network node number, 1≤nnode≤MAXNN, and nnode≤255, by which the network software identifies the NPU at the other end of the trunk from the end defined by the current TRUNK statement. This node number must be a number declared for the NODE parameter in an NPU statement within the current network division of the NDL program. This number is required; there is no default value.

Figure 3-13. TRUNK Statement Format

You define a trunk in terms of the two NPUs it connects; the TRUNK statement defining one end of the trunk follows the NPU statement defining the NPU on that end. The neighbor node number specified in that TRUNK statement is the number of the other NPU. The port number specified in each TRUNK statement is the port number at which the trunk is connected to the NPU currently being defined. Each end of the trunk is, therefore, defined by the port to which it is connected and the node number of the NPU connected at the other end of the trunk.

The number of trunks that you configure as connected to a 255x Series NPU must be the same as the number predefined in the network definition table for the CCP macromemory module loaded into that NPU. (The network definition table trunk entries are described in the NOS 1 Installation Handbook section on CCP installation.) Predefinition of the trunks permits the trunks to be automatically configured as enabled when the network is initiated. Because trunk communication lines are automatically configured as enabled, only one trunk can be defined between two NPUs at any time.

For example, if NPUD and NPUF are linked by one trunk communication line, TRUNK6, that trunk is correctly defined by the following statements.

```
NPUD: NPU P1LID=MIC,P2LID=MAA,NODE=9,
    NPUTYPE=2550.
TRUNK6: TRUNK PORT=4,NNODE=11.
TRUNK4: TRUNK PORT=3,NNODE=10.
      .
      .
      .
NPUF: NPU NODE=11,NPUTYPE=AUTO.
TRUNK6: TRUNK PORT=4,NNODE=9.
TRUNK5: TRUNK PORT=3,NNODE=8.
```

Note that the port number specified in each of the TRUNK statements identifies the port of the NPU being defined by the immediately preceding NPU statement. On the other hand, the neighbor node number specified in each TRUNK statement refers to the node number of the NPU linked to the NPU currently being defined.

In the example, trunks TRUNK4 and TRUNK5 are shown with only one end defined. Unless a second TRUNK statement was supplied for TRUNK4 and for TRUNK5, the NDL Processor would abort the NDL program with an error indication in the program's listing file.

As the TRUNK4 statement indicates, trunks can connect more than one remote NPU to a single front-end NPU. Trunks can also connect a single remote NPU to more than one front-end NPU, as shown by TRUNK5. The latter configuration allows separate physical connections for separate logical links between a host computer with more than one front-end NPU and a given remote NPU.

This manual presents the functional form and use of the TRUNK statement, rather than the legal form and use permitted by the NDL Processor. The NDL Processor imposes fewer restrictions on trunk configurations than are imposed by the operational requirements of the network software and 255x Series NPUs. This flexibility allows the NDL Processor to create valid network definition files for configurations of non-CDC NPUs and software.

For example, the NDL Processor permits you to configure trunks with any unique port number between 1 and FE$_{16}$, while the remote 255x Series NPU restricts the port numbers as shown in figure 3-13. The NDL Processor imposes a limit of 254 trunks per NPU, while the remote 255x Series port numbering restriction imposes a limit of 4 trunks per NPU. The front-end 255x Series NPU imposes no restriction on port numbering. However, functional considerations make use of the range shown in figure 3-13 advisable.

The NDL processor also permits you to configure multiple trunks between the same two NPUs; NDLP issues a diagnostic message in this case only if TRUNK statements declaring the same NNODE value are not grouped together after each NPU statement. The network software, however, does not support more than one trunk between any two NPUs. The NDL Processor does not prohibit you from configuring trunks connecting two remote NPUs or two front-end NPUs, but the network software does not support such trunks.

LOCAL DIVISION STATEMENTS

The local division describes those components of the network that one host (or one copy of the Communications Supervisor) services, such as application programs, lines, and terminals. These components are considered local to that host. The statements you use to describe these elements of your network are grouped into two sections:

- The host data section, which describes the applications available within the host
- The NPU data section, which describes the communication lines and terminal configurations connected to the terminal node NPUs with which the host has logical links

An NDL program can contain more than one local division, because configuration requirements can vary for the elements defined in this division. If no local division is present in your program, the NDL Processor does not create a local configuration file; otherwise, it creates one local configuration file for each local division it encounters.

LOCAL STATEMENT

You identify the beginning of each local division with a header statement of the form shown in figure 3-14. An example of a valid LOCAL header statement is

```
LOCL1: LOCAL.
```

```
local:LOCAL.

local     The name to assign to this local division.
          This name is required; there is no default
          value.
```

Figure 3-14. LOCAL Statement Format

The element name that you declare in the LOCAL statement appears in the division name field of NDL dayfile messages.

NFILE STATEMENT

The second statement in a local division can be either an NFILE statement or an LFILE statement (discussed later). If you include an NFILE statement in a local division, it must immediately follow the LOCAL statement. The NFILE statement has the format shown in figure 3-15. The NOS local file identified by the NFILE statement is used as the network configuration file to supply information for the local configuration file being created. If you omit the NFILE statement from the local division, NDLP finds the last new network configuration file created by your current NDL program. NDLP then uses that NCF to supply information for the local configuration file being created by this local division.

```
nfile:NFILE.

nfile     The local file name of the network configura-
          tion file to use when creating a local config-
          uration file from the current local division.
          This name is required; when the NFILE state-
          ment is used in a local division, there is no
          default value.
```

Figure 3-15. NFILE Statement Format (Local Division)

For example, the NDL program in figure 3-16 contains two network divisions to create NOS local files TSTNCF1 and TSTNCF3. Because the statement

```
TSTNCF1: NFILE.
```

is included immediately after the LOCAL statement, the local configuration file created from local division LOCAL1 will contain information drawn from TSTNCF1. If the NFILE statement was omitted, the local configuration file would contain information from TSTNCF3 by default because TSTNCF3 immediately precedes the local division.

```
CDCNET:  NETNAME.
    TSTNCF1:  NFILE.
           .
           .
           .
    TSTNCF3:  NFILE.
           .
           .
           .
    LOCAL1:  LOCAL.
        TSTNCF1:  NFILE.
```

Figure 3-16. Network and Local Divisions

You can also create a local configuration file from a local division without including a network division in the same program. To do this, you must specify an NFILE statement in your local division. That NFILE statement must give the NOS local file name of an extant network configuration file. You must ensure that the file specified is attached to your job via a NOS control statement. The NDL Processor then uses the local file as the network configuration file associated with the local configuration file being created.

The network configuration file identified by an NFILE statement within a local division is read as an old file. The local file name that you specify in the NFILE statement must be that of an existing NOS local file.

The network configuration file used during creation of a local configuration file must contain a validation record (see section 5). Only network definition files created without fatal errors contain validation records. If a network configuration file without a validation record is used during local configuration file creation, a fatal diagnostic message is issued. The local configuration file produced in this case does not contain a validation record either.

LFILE STATEMENT

If the NFILE statement appears in a local division, the LFILE statement must be the third statement in the division. Otherwise, the LFILE statement must be the second statement in the division. The LFILE statement has the format shown in figure 3-17 and names the NOS local file being created as the local configuration file from the current network division. For example, the following statement is valid.

```
LOCFIL1: LFILE.
```

```
lfile:LFILE.
```

lfile — The local file name of the local configuration file to be created. This name is required; there is no default value.

Figure 3-17. LFILE Statement Format

The local configuration file identified by the LFILE statement is opened without rewinding. The NOS local file name specified in the LFILE statement should not be the same as that of an existing NOS local file. The local configuration file is described in more detail in section 5.

HOST DATA SECTION

In the host data section you specify the application program resources available in the host computer. This specification is accomplished by two types of statements: HOSTDATA and APPL. The HOSTDATA statement identifies the host; one or more APPL statements identify the application programs that are available in the host computer software libraries or files.

HOSTDATA Statement

The HOSTDATA statement (figure 3-18) is the first statement in each host data section; it identifies the host computer to which the subsequent host data statements apply. Each local division in your program must have one host data section and, therefore, one HOSTDATA statement for the host computer. The element name that you use in this statement must be defined in the NCF. This name appears in the NAM dayfile when the network is initiated. The statement

```
HOST1: HOSTDATA.
```

would identify subsequent host data statements as applying to the local configuration file for host computer HOST1.

```
host:HOSTDATA.
```

host — The name of the host computer to which the subsequent host data section statements apply. This name must be the same as one used in a HOST statement within the last network division, if the NFILE statement was omitted from this division, or the same as one used in the network configuration file specified in the NFILE statement. The host computer name is required; there is no default value.

Figure 3-18. HOSTDATA Statement Format

APPL Statement

The APPL statement is shown in figure 3-19. You must provide one APPL statement for each application program available in the host computer for which the local configuration file is being created. The maximum number of APPL statements permitted in an NDL program division is described in appendix F.

The CDC-written application programs supported in the released version of the network software are

- Remote Batch Facility (RBF)
- Transaction Facility (TAF)
- Interactive Facility (IAF)
- Terminal Verification Facility (TVF)
- Message Control System (MCS)

```
service:APPL[ΔPRIV,UID,DI].
```

service — The letter and digit name of the network application program being configured as a network resource. This must be the name used by the program in its Network Access Method NETON statement and is the name terminal users must specify when logging in to it. This name should not be any of the reserved names described in the accompanying text. This name is required; there is no default value.

PRIV — A reserved word that indicates whether the privileged application program flag should be set for this program. This word is optional; if it is omitted, the flag is not set and the program is permitted to NETON regardless of how it entered execution under the operating system. When PRIV is specified, the flag is set and the program cannot successfully NETON unless it is of system origin type.

UID — A reserved word that indicates whether the unique identifier flag should be set for this program. This word is optional; if it is omitted, the flag is not set and more than one terminal at a time can log in to the program with a given user name and family name combination. When UID is specified, the flag is set and only one interactive terminal user at a time with a given user name and family name combination can log in to the program. This flag should be set for the Remote Batch Facility and the Transaction Facility.

DI — A reserved word that specifies the status of this application program at network initiation. This word is optional; if it is omitted, the application program is assumed to be enabled. When DI is specified, the program is disabled and cannot access the network until the local operator enables it.

Figure 3-19. APPL Statement Format

You can correctly describe these programs to the NDL Processor with the following respective statements.

```
RBF: APPL PRIV,UID.

TAF: APPL PRIV,UID.

IAF: APPL PRIV.

TVF: APPL PRIV.

MCS: APPL PRIV.
```

Terminal users log into any one of the five programs with the three-character application name shown. All five programs are required to be of system origin type. Any program attempting to access the network (NETON) as one of these five programs but not of system origin type is denied access.

To avoid possible problems with the integrity of the network and with validated users' ability to access application programs, the following names should not be used as names of site-written application programs.

ALL
BYE
CS
DOP
HELLO
HOP
IAF
LOGIN
LOGOUT
LOP
MCS
NOP
NOPLOP
NS
NUL
NVF
PFU
PTF
QTF
RBF
TAF
TCF
TVF

Of the above names, only the CDC-written application names IAF, MCS, RBF, TAF, and TVF should appear in an APPL statement in your NDL program.

You should specify the unique identifier flag (UID parameter) for RBF and TAF. These application programs route output files according to the combinations of user name and family name associated with the terminals they service. These files cannot be properly routed unless only one interactive terminal at a time with a given user name and family name combination is connected to the application program. If an application program written by your site can support only one terminal with a given user name and family name login combination at a time, you should specify the UID parameter for it as well. Terminals capable of using the program can then use the same user name and family name combination for access to it without simultaneous access ever occurring.

If any application program should be available to terminal users only at certain times of the day, you can specify that it be initially disabled; the local operator can then enable it at the appropriate time. Unless an application program is enabled, it cannot access the network and a terminal cannot successfully log in to it. For example, if a program TEST is configured by the statement

```
TEST: APPL DI.
```

it logically exists as a network resource within the host computer, but cannot successfully NETON or be accessed by a terminal user until the local operator enables it and it is initiated. You will probably want to configure your site-written application programs this way until they are thoroughly debugged. Because TVF is started up when the network is initialized, you must provide an APPL statement for it, and it cannot be disabled.

Any application program defined in the local configuration file must also be identified with the same name in the operating system common deck COMTNAP. COMTNAP is used by NVF to validate terminal user login sequences. Twelve bits in the access word for each user name correspond to installation-written application programs in COMTNAP, and other bits correspond to CDC-supplied application programs. When NVF is given a program name during the login procedure, it searches COMTNAP for that name. The entry for the name tells which bit in the access word must be set for access to the program to be permitted. More than one program in COMTNAP can specify the same access word bit, so there is no formal limit on the number of application programs that are network resources. However, you must take care that the correct access word bit is set for all user names that should have permission to access a given application program.

For example, a network definition with separately named copies of the same application program is legal. In this case, the COMTNAP entries for all of the copies can specify the same access word bit. Each access word bit would then identify a superset or class of application programs, rather than a separate program.

COMTNAP and access word use are described in further detail in the NOS Version 1 System Maintenance Reference Manual.

NPU DATA SECTION

One NPU data section is required for each terminal node NPU with which the host computer of the preceding host data section has logical links.

In the NPU data section, you specify all the lines and terminals on the terminal node NPUs. Each NPU is described by a combination of three statement types:

- The NPUDATA statement, which identifies the NPU being described

- The LINE statements that describe the lines connected to that NPU

- The TERMINAL statements that describe all terminals on the line

Every port on an NPU to which a terminal's communication line connects requires a LINE statement. Every LINE statement requires at least one TERMINAL statement.

If a LINE statement does not specify automatic recognition of terminals, all corresponding TERMINAL statements must completely describe their respective terminals. A TERMINAL statement is required for each logical terminal that can use a line (see section 4). If more than one physical terminal can use a line simultaneously, you must declare a separate TERMINAL statement for each physical terminal. You can define up to 15 terminals on a line with the same hardware address, if only one of them is enabled.

NPUDATA Statement

The NPUDATA statement identifies the beginning of each NPU data section. There should be one NPUDATA statement for each NPU data section and, therefore, for each terminal node NPU within the network configuration accessible from the host to which the preceding host data section relates. For example, an NDL program to configure the network CDCnet (figure 3-1) would have a separate NPU data section for each of the NPUs NPUA, NPUD, NPUE, and NPUF. All NPU data section statements following an NPUDATA statement are assumed to apply to the NPU identified in the NPUDATA statement until the end of the section, or until another properly placed NPUDATA statement is encountered. The format of this statement is shown in figure 3-20.

npu:NPUDATA.

npu — The name of the NPU whose terminal configuration is being described in the data section statements following this statement. This name must be the same as that used in an NPU statement within the last network division of the program, if the NFILE statement was omitted from this division, or the same as one used in the network configuration file specified in the NFILE statement of this local division. This name is required; there is no default value.

Figure 3-20. NPUDATA Statement Format

As an example of the statement's use,

```
NPUA: NPUDATA.
```

would begin the NPU data section for the terminal node network processing unit that you identified as NPUA in a network division.

LINE Statement

You must supply a LINE statement of the form shown in figure 3-21 for every communication line over which a terminal accesses its terminal node NPU. Fifteen lines can be defined for each NPU port, if only one is enabled. The maximum number of LINE statements permitted in an NDL program division is described in appendix F. Table 3-1 shows possible line types.

You can define a communication line for automatic recognition of terminals by declaring the AUTO parameter in the statement that defines the line. When a terminal becomes active on such a line, the network software identifies which terminal is connected. This is done by first determining as many of the terminal's characteristics as possible. These characteristics are then matched against the characteristics you have defined for each terminal configured for the line. For each terminal on an automatic recognition line, there are certain parameters that you must specify and certain parameters that you cannot specify. This aspect of automatic recognition depends on the type of automatic recognition line, as described in section 4.

Declaration of automatic recognition provides more flexibility in the actual terminals that can access the line. A switchable line configured for automatic recognition can be configured with fewer logical terminals than the physical terminals that can access it. A mode 4 line configured for automatic recognition can have its terminals configured without specification of cluster or terminal addresses on TERMINAL statements.

The statement

```
LN3F: LINE
    PORT=10,LTYPE=S1,TIPTYPE=MODE4,
    AUTO.
```

defines a synchronous line for automatic recognition of mode 4 terminals. The line is identified as LN3F, connects to the NPU at port 10, and is a switchable (dial-up) line enabled by default at network initiation. If the statement

```
LN3F: LINE
    PORT=10,LTYPE=S1,TIPTYPE=MODE4.
```

were used instead, all of the terminals capable of accessing the line would have to be explicitly configured on their TERMINAL statements. Note that the LSPEED parameter is not used for defining synchronous communication lines.

The statement

```
LN2E: LINE PORT=10,LTYPE=S3,TIPTYPE=HASP.
```

defines a dedicated, enabled, synchronous line for multileaving workstation terminals, accessing the NPU (NPUE in CDCnet) through port 10. The LTYPE values S1 through S3 are the only values valid with the TIPTYPE declarations of MODE4, HASP, and BSC. The LTYPE values of A1 and A2 require declaration of the TIPTYPE value ASYNC or X25.

The following two LINE statements are equivalent.

```
LN1A: LINE
    PORT=9,LTYPE=A1,TIPTYPE=ASYNC,
    LSPEED=110.

LN1A: LINE
    PORT=9,LTYPE=A1,TIPTYPE=ASYNC.
```

Each statement defines a switchable, enabled, asynchronous line accessing the NPU (NPUA in CDCnet) through port 9 and interfacing modems at 110 bps (the default rate). All terminals accessing this line must be explicitly configured because the line is not configured for automatic recognition. If AUTO had been specified, the LSPEED declaration could not have been legally included in the first statement.

TERMINAL Statement

Each local division in your program must contain a separate TERMINAL statement for every logical terminal serviced by the CS for that local division. Because the network software provides support for a wide variety of terminals, the TERMINAL statement can become complicated. For this reason, the TERMINAL statement is described in the following section, Terminal Support.

line: LINEΔPORT=port,LTYPE=ltype,TIPTYPE=tiptype[,AUTO,DI,LSPEED=lspeed,DFL=dfl,K=k,T1=t1,N2=n2,psn,LAP].

line — The name to be assigned to the communication line being defined. This name must be unique within the local division currently being described. This name is required; there is no default value.

port — The number, $1 \leq port \leq FE_{16}$, of the port to which this communication line connects on the NPU currently being defined. To the NDL processor, the port number is independent of the number of ports on the NPU; for example, a 128-port NPU can have a port numbered FE_{16} (254). However, it is highly recommended that you assign port numbers consecutively, starting with 1. All values declared for PORT parameters of enabled elements must be unique within a given network definition of each NPU. This number cannot be the same as the number declared for a port in any other LINE or TRUNK statement configuring an enabled element on this NPU, and cannot be lower than the higher port number used by a trunk. This number is required; there is no default value. The maximum port number used cannot be larger than the number of lines indicated in the CCP build plus one.

ltype — A reserved word that identifies the type of communication line adapter/modem/circuit combination that is used on this line. This word must be supplied; there is no default value. The legal words for this value declaration are described in table 3-1.

tiptype — A reserved word that specifies the type of Terminal Interface Program protocol required for the terminals on this line. This word must be supplied; there is no default value. The following words are legal types:

ASYNC	Asynchronous protocol is required
BSC	Binary synchronous protocol is required
HASP	Multileaving station protocol is required
MODE4	Control Data mode 4 protocol is required
X25	X.25 protocol is required
USER1	Reserved for site-defined TIP of type 12
USER2	Reserved for site-defined TIP of type 13
USER3	Reserved for site-defined TIP of type 14
USER4	Reserved for site-defined TIP of type 15

When one of the last four words is declared in a LINE statement, the NDL processor suspends all diagnostic processing for other parameters on the same statement.

AUTO — A reserved word that specifies automatic recognition of terminals should be performed for this line. This word is not legal for TIPTYPE=X25, but is optional for all other TIPTYPE declarations. If AUTO is omitted, automatic recognition is not attempted and each TERMINAL statement associated with this line must explicitly declare all addressing parameters legal for that terminal; only one enabled terminal is permitted at each address. When AUTO is specified, the network software attempts to determine as many addressing parameters as possible at the time the line becomes active; up to 15 enabled terminals are permitted at each address.

DI — A reserved word that specifies the status of this communication line at network initiation. This word is optional; if it is omitted, the communication line is assumed to be enabled. When DI is specified, the line is disabled and no terminal on the line can access the network until the line is enabled by the local operator.

lspeed — The baud rate used by the modems or devices accessing this line. This parameter can only be used to configure asynchronous lines (LTYPE of A1 or A2). When the line is configured for automatic recognition (AUTO is declared), this parameter cannot be used. This parameter is optional when an asynchronous line is not configured for automatic recognition of terminals; the following values are legal rate declarations:

110 default	300	2400
134	600	4800
150	1200	9600

When no baud rate is declared for an asynchronous line configured without automatic recognition, the default shown above is used.

dfl — The maximum size of data packet to be used. $32 \leq dfl < 2^{16}$, where dfl is the decimal number of octets or 8-bit characters per packet. This parameter is required for lines with TIPTYPE=X25; it is not legal for any other lines.

k — The frame window, $1 \leq k \leq 127$; the maximum number of outstanding unacknowledged frames allowed between the network processing unit and the terminal. This parameter is required for lines with TIPTYPE=X25; it is not legal for any other lines.

t1 — The retransmission timer, $1 \leq t1 \leq 25500$; the elapsed time, in milliseconds, before an unacknowledged frame is retransmitted. The NDL processor rounds the value of t1 up to the next multiple of 100.

This parameter is required for lines with TIPTYPE=X25; it is not legal for any other lines.

Figure 3-21. LINE Statement Format (Sheet 1 of 2)

n2 The retransmission counter, 1≤n2≤15, the number of times retransmission of an unacknowledged frame is to be attempted. This parameter is optional for lines with TIPTYPE=X25, it is not legal for any other lines.

psn The packet switching network to which the line is connected. This parameter is required for lines with TIPTYPE=X25, it is not legal for any other lines. This parameter can have the following values:

TELENET GTE Telenet Corporation's packet-switching network

USERPSN A user-defined X.25 protocol variant for which a user-written terminal interface program is supplied

LAP The link access protocol used by the network processor to access a packet switching network. Support of LAPB as the default protocol is being considered for future versions of the network software; the current version supports only the LAP protocol. This parameter is required for lines with TIPTYPE=X25; it is not legal for any other lines.

Figure 3-21. LINE Statement Format (Sheet 2 of 2)

TABLE 3-1. LINE TYPE DEFINITIONS

LTYPE Parameter	Transmission Facility Protocol	CLA Type	Modem Type	Line Type	Carrier Type	Transmission Mode
S1	Half-duplex	2560-1	RS232C-201A/208B compatible	Switched (Dialup)	Controlled	Synchronous
S2	Full-duplex†	2560-1	RS232C-201B/208A compatible	Dedicated (Hardwired)	Controlled	Synchronous
S3	Full-duplex	2560-1	RS232C-201B/208A compatible	Dedicated (Hardwired)	Constant	Synchronous
A1	Full-duplex	2561-1	RS232-103E/113 compatible	Switched (Dialup)	Constant	Asynchronous
A2	Full-duplex	2561-1	RS232-103E compatible	Dedicated (Hardwired)	Constant	Asynchronous
H1	Full-duplex	2563-1	RS232-201B compatible	Dedicated (Hardwired)	Constant	Bit-oriented Synchronous

†Operating with half-duplex protocol.

SPECIAL PURPOSE STATEMENTS

Three special statement types are supplied to document, simplify, and separate network definitions:

- COMMENT statement
- DEFINE statement
- END statement

These three statements cannot have labels as element name specifications.

COMMENT STATEMENT

The COMMENT statement permits you to insert text comments into the NDL input stream. The NDL Processor copies these comments, without editing or interpretation, into the output listing of your NDL source statements. The format of the COMMENT statement is shown in figure 3-22. Use of the COMMENT statement is optional.

COMMENTΔstring.

string The character string of text that the programmer wants to appear on the source listing output. This string can contain any letters, digits, or special characters except for a period (which terminates the COMMENT statement), and can contain NDL reserved words.

Figure 3-22. COMMENT Statement Format

DEFINE STATEMENT

The DEFINE statement allows a character string that is too cumbersome for easy manipulation to be labeled with an identifier. The identifier can subsequently be used in place of the character string in other NDL statements, and the NDL Processor automatically substitutes the full character string for the identifier when it interprets the NDL program.

A DEFINE statement must precede any other statement that uses the identifier provided in the DEFINE statement.

DEFINE statements can be placed anywhere within NDL program divisions after the NETNAME or LOCAL statements. DEFINE statements apply only to the division in which they occur. Use of DEFINE statements is optional.

Nested DEFINE statements are not permitted. The identifier of one DEFINE statement cannot be used in the character string of the same or another DEFINE statement, nor can the character string of one DEFINE statement contain another DEFINE statement.

Identifiers and their strings are stored in a variable length table during NDL Processor execution. Approximately 2000 DEFINE statements can be specified in a single NDL program without requiring the NDL Processor to be run with additional field length.

The format of the DEFINE statement is shown in figure 3-23. As examples of the functioning of this statement, consider the following portion of an NDL program:

```
     .
     .
     .
DEFINE LTYPE3=LTYPE=A2,TIPTYPE=ASYNC,AUTO.
DEFINE PORTA=PORT=1.
     .
     .
     .
LINE5: LINE PORTA,LTYPE3.
     .
     .
     .
```

The first DEFINE statement permits the identifier LTYPE3 to be substituted in subsequent NDL statements (such as the last one above) for the character string LTYPE=A2,TIPTYPE=ASYNC,AUTO. Note that the LINE statement is still terminated by a period because this character string is considered to end with the O, not with the period that terminates the DEFINE statement. The second DEFINE statement equates the identifier PORTA to the character string PORT=1, so that the LINE statement is actually interpreted by the NDL Processor as if it had been written

```
LINE5: LINE PORT=1,LTYPE=A2,TIPTYPE=ASYNC,
    AUTO.
```

The defname identifier can only be used in an NDL statement where the NDL Processor expects to find a keyword or would normally infer an omitted keyword for an order-dependent value; the identifier cannot be substituted for a value when a keyword is supplied. For example, if the first DEFINE in the above sample coding had been

```
DEFINE LTYPE3=A2.
```

the LINE statement would be correctly expanded because the NDL Processor would infer the omitted keyword LTYPE after the defname substitution was performed. However, if the LINE statement had been

```
LINE5: LINE PORTA,LTYPE=LTYPE3,
    TIPTYPE=ASYNC,AUTO.
```

and was used with this new version of the DEFINE, the characters LTYPE3 would not be replaced by A2 (that is, expansion would not occur) and would be interpreted as an invalid line type value.

DEFINE statements receive separate diagnostic treatment when the NDL program statements are processed. However, if a DEFINE statement contains an NDL coding error in its character string, the error might not be diagnosed until an attempt is made to expand the defname identifier. Appendix B contains examples of this diagnostic treatment.

The following combination of statements would not be recognized as expanding into a valid NDL statement, because the reserved statement name LINE cannot be found in the unexpanded version of LINE5.

```
DEFINE LINEA=LINE PORT=1.
DEFINE LTYPE3=LTYPE=A2.
     .
     .
     .
LINE5: LINEA,LTYPE3,TIPTYPE=ASYNC,AUTO.
```

END STATEMENT

The END statement is required to explicitly terminate each division in the NDL program. This statement has the format shown in figure 3-24.

DEFINEΔdefname=string.

defname — The identifier assigned to the character string. This is a name of one to seven letters or digits, the first of which must be a letter. This name is required; there is no default value.

When defname occurs in an NDL statement in place of a keyword, the entire character string equated to it in the DEFINE statement is substituted for defname in that statement. This DEFINE substitution will occur only if defname is bound on both sides by a separator (a comma, blank, or period) within that NDL statement.

The defname identifier should not be an NDL reserved word. If such a word is used, the DEFINE statement substitution will not occur when defname is used in an NDL statement; the identifier will be interpreted within the statement as a value supplied without a prefixed keyword, producing errors.

The defname identifier cannot be used again in a DEFINE statement to change the character string it represents.

string — The character string for which the defname identifier is substituted when writing other NDL statements. This character string can contain separators, equal signs, blanks, and any letters or digits; it cannot contain a period, because a period is interpreted as the end of the DEFINE statement. The string can contain NDL reserved words but cannot contain reserved NDL statement names.

Figure 3-23. DEFINE Statement Format

```
END.
```

Figure 3-24. END Statement Format

TERMINAL SUPPORT 4

The way you use the Network Definition Language to define a terminal differs from the way that you use it to define other elements of the network. For example, the network software supports one coupler in the same manner that it supports any other coupler; therefore, you must name the coupler and provide information about how it connects to the network, but you do not have to describe the coupler itself. The network software supports many different types of terminals, however. To be able to distinguish one terminal from another, the network software needs information about the physical characteristics of the terminals. You provide this type of information with a TERMINAL statement for each terminal in your network.

This section describes the way that you can use NDL statements to define your terminals. The terminals supported by the network software are categorized by the line protocol required to service them. Five standard protocols are supported:

- ASYNC, for asynchronous terminals
- X25, for asynchronous terminals connected to packet-switching networks
- MODE4, for synchronous terminals like the CDC 200 User Terminal
- HASP, for synchronous terminals that conform to the IBM HASP multileaving workstation protocol
- BSC, for bisynchronous terminals such as the IBM 2780 and 3780

NDL provides a variant of the TERMINAL statement for each protocol. This section contains a separate subsection for each protocol. These subsections describe the support provided for the protocol and the TERMINAL statement variant you can use to define terminals of that protocol.

You can provide up to four additional protocols of your own definition using the USER1, USER2, USER3, and USER4 TIPTYPE declarations. CDC cannot anticipate the nature of such protocols; therefore, the NDL Processor is limited in its ability to check errors in NDL programs containing USERx TIPTYPE declarations.

TERMINAL DEFINITION CONCEPTS

In each local division in your NDL program, you must provide a separate TERMINAL statement for every logical terminal serviced by the Communication Supervisor for that division. The maximum number of TERMINAL statements permitted in an NDL program division is described in appendix F.

TERMINAL CLASSES

The supported terminals are grouped into 17 terminal classes, according to their hardware characteristics. In NDL each class is recognized by the characteristics of a single archetype terminal, as shown in table 4-1.

TABLE 4-1. TERMINAL CLASSES

Archetype Terminal	Terminal Class Number	Terminal Class Mnemonic
Teletype Corporation Model 30 Series	1	M33
CDC 713	2	713
Reserved for CDC use	3	
IBM 2741	4	2741
Teletype Model 40-2	5	M40
Hazeltine 2000	6	H2000
CDC 751	7	751
Tektronix 4000 Series	8	T4014
None	9	HASP
CDC 200 User Terminal	10	200UT
CDC 714-30	11	714X
CDC 711-10	12	711
CDC 714-10/20	13	714
None	14	HPRE
CDC 734	15	734/731
IBM 2780	16	2780
IBM 3780	17	3780
Reserved for CDC use	18 thru 27	
Site-defined classes:	28 29 30 31	USER1 USER2 USER3 USER4

The mnemonics identifying the terminal classes are derived from the identification of the archetype terminals defining the classes, with six exceptions. HASP and HPRE each identify the communication protocol employed by a terminal as the archetype. You can use USER1, USER2, USER3, and USER4 to identify site-defined terminals of unknown characteristics, corresponding to reserved Network Access Method terminal class numbers. Because the characteristics of terminals in the latter four classes are unknown, no TERMINAL statement value ranges or default values can be given for them.

AUTOMATIC RECOGNITION

The Network Definition Language allows you to define a line so that the network software can automatically determine various addressing parameters of the terminals on that line. Terminals configured on such lines are said to be automatically recognized. The lines themselves are known as automatic recognition lines or simply AUTO lines. The parameters determined by automatic recognition vary according to the line type.

You can omit these addressing parameters from the TERMINAL statements for lines configured for automatic recognition. If you choose to supply these parameters anyway, the supplied parameters are used as additional points of matching between the terminal descriptions in the local configuration file and the characteristics found by the network software when the terminal becomes active. The logical terminal in the local configuration file that is the closest match in specified characteristics to the physical terminal is chosen as the active configured terminal. If you choose not to supply the parameters that are automatically recognized, there is more flexibility in the logical terminals that can be chosen as a match for the physical terminal.

For example, an NDL statement omitting all parameters that can be supplied through automatic recognition adequately configures a logical terminal that will match either the console of a mode 4A terminal or a mode 4C terminal using the ASCII code set.

NOTE

> If you configure a communication line for automatic recognition, you can define up to 15 enabled terminals for one set of addressing parameters. When a line is not configured for automatic recognition, up to 15 terminals can be defined for one set of addressing parameters, but only one terminal can be enabled for the set.

It is possible for you to configure a logical terminal without declaring any parameters on a TERMINAL statement. The Communication Supervisor then configures the terminal as if it were of the type defined as the archetype terminal for its class. However, automatic recognition imposes certain requirements and restrictions on parameter declarations, as do the device type and terminal class of the specific terminal. Some terminals must have certain parameters declared; others cannot have certain parameters declared. These restrictions are presented in detail in the subsection pertaining to the line protocol.

PARAMETERIZATION

Various physical aspects of a particular terminal influence the way the terminal can use network software features. These physical characteristics can be specified as parameters on the NDL statements defining each terminal in the local configuration file. By choosing the appropriate values for these parameters, you can accurately define any terminal compatible with an archetype terminal. If you do not specify these parameters, the network software assigns default values for each applicable parameter on the basis of the terminal class.

Because terminal definition through the Network Definition Language involves more factors than are covered by these parameters, the word parameterization is used throughout this manual to describe this aspect of terminal definition. Other manuals describing network software use the expression "terminal characteristic definition" (or TERMDEF) when discussing this subset of the definition process.

Parameterization involves any of the following that are applicable to a specific terminal:

- Parity
- Backspace, break, and other control characters
- Idle counts for line feed and carriage return
- Echoplex operation
- Transparent mode delimiting
- Page length and width
- Specification of character mode input and output devices for asynchronous terminals
- Page wait
- Special editing

The terminal user can redefine any or all of the applicable parameters (as well as the terminal class) by entering terminal definition commands through the terminal. User-entered commands override both NDL-specified parameter values and software-supplied defaults, but do not change the supplied values or defaults contained in the local configuration file after NDL program execution. Such parameterization changes remain in effect only until the terminal is logged out. The only way the parameterization of a given terminal can be changed is with a new NDL program job that reconstructs the local configuration file.

An application program can also change the parameterization of a connected terminal. Such changes also temporarily override supplied values or defaults, but do not change the contents of the local configuration file.

PARAMETERIZATION RESTRICTIONS

Note that terminals cannot be parameterized for initial transparent mode operation. Specification of a transparent mode input device can be done only by a terminal operator or an application program; NDL permits parameterization of character mode input devices only. Specification of transparent mode output to a device can be done only by an application program.

Take care when deciding the parameterization to use for a given terminal. NOS parameters (associated with the user names in the system validation file VALIDUz) that can be used at a given terminal must correspond to part of the parameterization of that terminal; if such correspondence is not made, login problems and terminal operator confusion can result. The user name NOS parameters and the corresponding TERMINAL statement parameters are

User Name	TERMINAL Statement
PA (parity)	PA (parity)
PX (echoplex)	EP (echoplex)
RO (rubouts)	CI (carriage return idle) LI (line feed idle)
TT (terminal type)	TC (terminal class) SE (special editing) CSET (character set)

The parameters AL, BS, B1, B2, CN, and CT have related restrictions imposed on their declarations for a given terminal. A character defined for any of these parameters except AL and CN must be unique among all of them; that is, the same character code cannot be declared (or used by default) for more than one of these parameters. For example, the character code defining the B1 parameter cannot be used as the value declaration for the B2 parameter.

You cannot define any of these six parameters by declaring the character code for an equal sign; the value $3D_{16}$ is therefore not a legal part of the range permitted for the declarations of these parameters. Some of these parameters should not be defined by declaring character code values between 00 and $1F_{16}$, because those codes are incompatible or illogical when combined with the functions these six parameters configure. When a code that already has meaning in the terminal's character set is used for one of these parameters, the resulting terminal operation is unpredictable. Terminals configured in this manner can be rendered inoperative by the parameter declarations used. In addition, none of the six parameters should be assigned the ASCII characters NUL (00_{16}), STX (02_{16}), EOT (04_{16}), LF ($0A_{16}$), CR ($0D_{16}$), or DEL ($7F_{16}$). When one of these values is declared, either of the following can occur:

- CCP rejects the terminal configuration when the Communications Supervisor loads that terminal configuration into the CCP.
- The TIP edits the character out of upline messages from the terminal without acting upon the character.

There is a functional restriction on the ALUN and ALFAM values you can declare for console terminals serviced by the Transaction Facility. The Transaction Facility provides data base access for only one terminal with a given user name at a time. Whenever you declare TAF as the initial application program (IAPPL parameter) for a terminal, you must declare values for the ALFAM and ALUN parameters for that terminal that allow for simultaneous access. Whenever an ALUN value is declared for a terminal intended to access TAF, the value used must correspond to the value used when identifying the terminal in the NCTFid file accessed by TAF. Additional requirements of terminals configured to access TAF are described in the NOS Version 1 System Maintenance Reference Manual.

The Remote Batch Facility (RBF) routes its output files on the basis of the family name and user name of each console. If UID is specified on the RBF APPL statement, a console terminal with a duplicate family name and user name combination is not allowed to log in to the network. If UID is not specified, consoles with duplicate family name and user name combinations can log in to the network, but the routing of output for the terminals is unpredictable. Because of this, each console using RBF at any one time should have a unique family name and user name. This requirement applies whether you enter the names using the ALFAM and ALUN parameters or if the terminal user enters them from the terminal. RBF uses neither the family names nor user names declared for passive terminals. Instead, it uses the names associated with the owning console. The family name and user name declared for each passive terminal need not be unique, nor need they match the family name and user name specified for the owning console.

ASYNC PROTOCOL

The ASYNC protocol provides support for asynchronous terminals. Asynchronous terminals operate on asynchronous lines and include such devices as teletypewriters, inquiry/retrieval stations, and IBM-compatible terminals. You can define up to fifteen such terminals on a single line (see Automatic Recognition later in this subsection).

The network software does not provide support for passive devices on asynchronous lines. You can configure your own passive devices using the USER terminal classes, or you can connect them as secondary devices to merely echo the input and output of the console device.

Figure 4-1 shows the format of the TERMINAL statement used to describe asynchronous terminals on lines using the ASYNC TIP. A detailed description of each of the parameters follows. Default values are underlined.

For AUTO Lines[†]

```
terminal-name:TERMINAL   [ΔSTIP=stip, CSET=cset,
    TSPEED=tspeed, DT=dt, TC=tc, PRI, NBL=nbl,
    IAPPL=service, AL=al††, ALFAM=alfam, ALUN=alun,
    B1=b1††, B2=b2††, PA=pa††, PL=pl††, PW=pw††,
    BS=bs††, CI=ci††, CN=cn††, CT=ct††, DLC=dlc††,
    DLTO=dlto††, DLX=dlx††, EP=ep††, IN=in,
    OP=op††, LI=li††, PG=pg, SE=se, ABL=abl,
    BSZ=bsz, DI].
```

For Non-AUTO Lines[†]

```
terminal-name:TERMINAL   Δ STIP=stip, [CSET=cset,
    DT=dt,] TC=tc[,PRI, NBL=nbl, IAPPL=service,
    AL=al††, ALFAM=alfam, ALUN=alun, B1=b1††,
    B2=b2††, PA=pa††, PL=pl††, PW=pw††, BS=bs††,
    CI=ci††, CN=cn††, CT=ct††, DLC=dlc††,
    DLTO=dlto††, DLX=dlx††, EP=ep††, IN=in, OP=op††,
    LI=li††, PG=pg, SE=se, ABL=abl, BSZ=bsz, DI].
```

[†] The hexadecimal value to declare for a given character can be determined using table A-9.

[††] Ranges and/or defaults for this parameter depend on the TC value used for the terminal. See table 4-2.

Figure 4-1. TERMINAL Statement Format for ASYNC TIP

NOTE

To use the order-dependent value declaration convention, you must supply TERMINAL statement parameters in the order shown in appendix E.

TERMINAL-NAME PARAMETER

Use the terminal-name parameter to assign a name to the terminal for reference within the NDL program and by the local operator. This name is required and should be unique within the current local division.

STIP PARAMETER

Use the STIP parameter to indicate the sub-TIP type of the terminal. This reserved word parameter is optional for AUTO lines; otherwise, it is required. If you specify STIP for one terminal on the line, you must specify it for all the terminals on the line. If STIP is supplied, the network software performs an additional match against the local configuration file while identifying the terminal.

Any asynchronous terminal except an IBM 2741 (or a terminal that emulates it) can be configured with a STIP value of N2741. The following alternate values are legal if the line is configured for automatic recognition of terminals and the CSET and TSPEED parameters are not specified.

A110 Any N2741 terminal using the ASCII character set on a 110-bps communication line

A150 Any N2741 terminal using the ASCII character set on a 150-bps communication line

A300 Any N2741 terminal using the ASCII character set on a 300-bps communication line

An IBM 2741 (or any asynchronous terminal that emulates it) can be configured with a value of 2741. The following alternate values are legal only if LSPEED=134 and the CSET and TSPEED parameters are not specified.

2741E Any 2741 terminal with an EBCD character set

2741C Any 2741 terminal with a correspondence code character set

If TIPTYPE = USER1, USER2, USER3, or USER4, you can assign any of the values above to STIP; in the LCF summary STIP will be represented as 1 for N2741 terminals or as 2 for 2741 terminals.

CSET PARAMETER

Use the CSET parameter to specify the code set of the terminal. The possible values for this parameter are

ASCII ASCII code set

APLBP ASCII code set with bit-pairing APL character set

APLTP ASCII code set with typewriter-pairing APL character set

CORAPL Correspondence code set with APL character set

CORRES Correspondence code set

EBCD Extended BCD code set

EBCDAPL Extended BCD code set with APL character set

For AUTO lines, this parameter is optional. If you specify CSET for one terminal on the line, however, you must specify it for all terminals on the line. If you specify CSET, the network software performs an additional match while identifying the terminal in the configuration file. If you omit CSET, no additional match is performed.

For non-AUTO lines with STIP = 2741, you must specify this parameter. If STIP = N2741, and you omit CSET, ASCII is assumed. The default is provided only for non-AUTO lines.

If you specify any CSET value other than ASCII, you must ensure that CCP has the extended asynchronous TIP option (described later in this section).

TSPEED PARAMETER

Use the TSPEED parameter to specify the line speed of the terminal. This optional parameter is valid only when automatic recognition has been specified. If you specify TSPEED for one terminal on the line, you must specify it for all terminals on the line. If you supply TSPEED, the network software performs an additional match while identifying the terminal in the configuration file. The following values are possible for this parameter.

110

134

150

300

600

1200

DT PARAMETER

Use the DT parameter to specify the device type of the terminal. The following reserved words are valid for this optional parameter.

CON Identifies a keyboard or CRT console terminal

USER Identifies a site-defined device of type 7

TC PARAMETER

Use the TC parameter to specify the terminal class of the terminal. The reserved word value that you specify for this parameter determines the parameterization ranges and defaults used for the terminal (see table 4-2). The legal values depend on the values you declare for the STIP parameter. If STIP = N2741, A110, A150, or A300, TC can be

M33

H2000

M40

751

713

T4014

If STIP = 2741, 2741C, or 2741E, TC can be

2741

For AUTO lines this parameter is optional; otherwise, it is required. If you specify this parameter for one terminal on a line, you must specify it for all terminals on the line.

TABLE 4-2. PARAMETERIZATION RANGES AND DEFAULTS BY TERMINAL CLASS - ASYNC AND X25 TIPS

Parameter	Terminal Class					
	M33		713		2741[†]	
	Range	Default	Range	Default	Range	Default
AL	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	28_{16} (
BS	0 thru $7F_{16}$	8 CTRL/H	0 thru $7F_{16}$	8 ←	0 thru $7F_{16}$	8 BS
B1	0 thru $7F_{16}$	10_{16} CTRL/P	0 thru $7F_{16}$	10_{16} CTRL/P	0 thru $7F_{16}$	$3A_{16}$:
B2	0 thru $7F_{16}$	14_{16} CTRL/T	0 thru $7F_{16}$	14_{16} CTRL/T	0 thru $7F_{16}$	29_{16})
CI	0 thru 99	2	0 thru 99	0	0 thru 99	Calculated by TIP
CN	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	28_{16} (
CT	0 thru $7F_{16}$	$1B_{16}$ ESC	0 thru $7F_{16}$	$1B_{16}$ ESCAPE	0 thru $7F_{16}$	25_{16} %
DLC[†]	1 thru 4096	2043	1 thru 4096	2043	None	None
DLTO[†]	YES, NO	NO	YES, NO	NO	None	None
DLX	0 thru FF_{16}	$0D_{16}$ carriage return	0 thru FF_{16}	$0D_{16}$ carriage return	$6D_{16}$	$6D_{16}$ carriage return
EP[†]	YES, NO	NO	YES, NO	NO	None	None
IN[†]	KB, PT	KB	KB, PT	KB	KB, PT	KB
LI	0 thru 99	1	0 thru 99	0	0 thru 99	1
OP[†]	DI, PR, PT	PR	DI, PR, PT	DI	DI, PR, PT	PR
PA	E, O, N, Z	E	E, O, N, Z	E	E, O, N, Z	O
PG	YES, NO	NO	YES, NO	NO	YES, NO	NO
PL	0 thru 255	0	0 thru 255	0	0 thru 255	0
PW (DT=CON)	0 thru 255	72	0 thru 255	80	0 thru 255	132
SE[†]	YES, NO	NO	YES, NO	NO	YES, NO	NO
W	1 thru 127	None	1 thru 127	None	1 thru 127	None

LEGEND:

Range — Hexadecimal values of 0 through $7F_{16}$ correspond to characters in the 128-character ASCII set of table A-9. The value $3D_{16}$ is not legal.

Default — Character representations beneath default values are the keyboard keys that cause entry of the default value for the archetype terminal of the terminal class.

[†] Not valid for X.25 lines (LTYPE=H1).

TABLE 4-2. PARAMETERIZATION RANGES AND DEFAULTS BY TERMINAL CLASS - ASYNC AND X25 TIPS (Contd)

Parameter	Terminal Class							
	M40		H2000		751		T4014	
	Range	Default	Range	Default	Range	Default	Range	Default
AL	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	18_{16} CTRL/X
BS	0 thru $7F_{16}$	None	0 thru $7F_{16}$	8 CTRL/H	0 thru $7F_{16}$	8 ←	0 thru $7F_{16}$	8 CTRL/H
B1	0 thru $7F_{16}$	10_{16} CTRL/F	0 thru $7F_{16}$	10_{16} CTRL/P	0 thru $7F_{16}$	10_{16} CTRL/P	0 thru $7F_{16}$	10_{16} CTRL/P
B2	0 thru $7F_{16}$	14_{16} CTRL/T	0 thru $7F_{16}$	14_{16} CTRL/T	0 thru $7F_{16}$	14_{16} CTRL/T	0 thru $7F_{16}$	14_{16} CTRL/T
CI	0 thru 99	1	0 thru 99	0	0 thru 99	0	0 thru 99	0
CN	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	18_{16} CTRL/X	0 thru $7F_{16}$	18_{16} CTRL/X
CT	0 thru $7F_{16}$	$1B_{16}$ CTRL/P	0 thru $7F_{16}$	$1B_{16}$ ESC	0 thru $7F_{16}$	$1B_{16}$ ESC	0 thru $7F_{16}$	$1B_{16}$ ESC
DLC†	1 thru 4096	2043	1 thru 4096	2043	1 thru 4096	2043	1 thru 4096	2043
DLTO†	YES, NO	NO	YES, NO	NO	YES, NO	NO	YES, NO	NO
DLX	0 thru FF_{16}	$0D_{16}$ carriage return	0 thru FF_{16}	$0D_{16}$ carriage return	0 thru FF_{16}	$0D_{16}$ carriage return	0 thru FF_{16}	$0D_{16}$ carriage return
EP†	YES, NO	NO	YES, NO	NO	YES, NO	NO	YES, NO	NO
IN†	KB, PT	KB	KB, PT	KB	KB, PT	KB	KB, PT	KB
LI	0 thru 99	3	0 thru 99	3	0 thru 99	0	0 thru 99	0
OP†	DI, PR, PT	DI	DI, PR, PT	DI	DI, PR, PT	DI	DI, PR, PT	DI
PA	E, O, N, Z	E	E, O, N, Z	E	E, O, N, Z	E	E, O, N, Z	E
PG	YES, NO	NO	YES, NO	NO	YES, NO	NO	YES, NO	NO
PL	0 thru 255	0	0 thru 255	0	0 thru 255	0	0 thru 255	0
PW (DT=CON)	0 thru 255	74	0 thru 255	74	0 thru 255	80	0 thru 255	74
SE†	YES, NO	NO	YES, NO	NO	YES, NO	NO	YES, NO	NO
W	1 thru 127	None	1 thru 127	None	1 thru 127	None	1 thru 127	None

LEGEND:

Range Hexadecimal values of 0 through $7F_{16}$ correspond to characters in the 128-character ASCII set of table A-9. The value $3D_{16}$ is not legal.

Default Character representations beneath default values are the keyboard keys that cause entry of the default value for the archetype terminal of the terminal class.

†Not valid for X.25 lines (LTYPE=H1).

PRI PARAMETER

Use the PRI parameter to specify that the terminal is to have data traffic priority. This optional reserved word indicates that data to or from the terminal is to travel through the network with a higher priority than data to or from terminals for which this priority has not been specified. Data to or from terminals without priority is suspended to relieve network traffic congestion before data to or from terminals with priority is suspended. PRI is usually specified for terminals with DT=CON when interactive traffic is to receive priority over batch traffic.

NBL PARAMETER

Use the NBL parameter to establish the allowable block limit for this terminal in the network, specifying the number of downline blocks that can be buffered between the host computer and the terminal. This optional parameter has the range $1 \le nbl \le 7$; the default value is 2. The value you specify is used in conjunction with the ABL parameter to determine the block queuing required of the Network Access Method and of CCP. The relationship between the NBL and ABL parameters is explained later in this section under Buffering of Data.

IAPPL PARAMETER

This parameter is used to declare an initial application program for this terminal. This name must be one specified on an APPL statement in the current local division of the NDL program. This parameter is optional. If you declare IAPPL, the terminal is connected directly to the specified application upon completion of family name and user name validation; the terminal is not prompted by NVF for an application name. If you omit IAPPL, the terminal is prompted by NVF; there is no default value.

AL PARAMETER

Use the AL parameter to specify the hexadecimal value of the abort output line character for this terminal. Entry of this character by the terminal operator causes the TIP to cancel transmission of the current logical line of upline data from the terminal. This parameter is optional.

ALFAM PARAMETER

Use the ALFAM parameter to specify the NOS family name to use during automatic login of this terminal. If you specify this optional parameter, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a family name during the initial login attempt, but can be prompted during subsequent login attempts, even though ALFAM is specified. The value you declare for ALFAM must be chosen with consideration of the UID parameter specification for all application programs this terminal can access (see the APPL statement description). The value you choose for this parameter can be

0 — The system default family name is to be used.

Any valid value — Refer to the NOS System Maintenance Reference Manual for determination of valid values.

ALUN PARAMETER

Use the ALUN parameter to declare a NOS user name to use during automatic login of this terminal. If you declare this optional parameter for a terminal, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a user name and a password during the initial login attempt. The terminal can be prompted for manual entry of the user name and password during subsequent login attempts. The value declared for this parameter can be any valid user name from the NOS VALIDUz file (see the NOS System Maintenance Reference Manual).

If the terminal is configured with TAF as its initial application program (IAPPL=TAF), the value used for the ALUN parameter must be the same as that used to identify the terminal in the NOS terminal file NCTFid. NCTFid is described in detail in the System Maintenance Reference Manual.

B1 PARAMETER

Use the B1 parameter to establish the hexadecimal value of the break-1 character for the terminal. When this character is entered from the terminal, the TIP sends a break-1 supervisory message to the application program currently associated with the terminal. This parameter is optional.

B2 PARAMETER

Use the B2 parameter to define the hexadecimal value of the break-2 character for the terminal. When the terminal user enters this character from the terminal, the TIP sends a break-2 supervisory message to the application program currently associated with the terminal. This parameter is optional.

PA PARAMETER

Use the PA parameter to indicate the parity to be used by the TIP for transmissions to and from this terminal. The reserved words legal for this optional parameter are

Z — Zero parity. The upper character bit is always zero.

O — Odd parity (default for TC=2741). The upper character bit varies.

E — Even parity (default for all terminals except TC=2741). The upper character bit varies.

N — No parity. The upper character bit is data.

PL PARAMETER

Use the PL parameter to specify the number ($0 \le pl \le 255$) of physical lines per page of input or output for the terminal. This parameter is optional.

PW PARAMETER

Use the PW parameter to establish the number ($0 \le pw \le 255$) of characters per physical line of input or output for this terminal. This parameter is optional.

BS PARAMETER

Use the BS parameter to specify the hexadecimal value of the backspace character for this terminal. This parameter is optional.

CI PARAMETER

Use the CI parameter to specify the number $(0 \leq ci \leq 99)$ of idle characters to insert in the upline data from this terminal after a carriage return. The default value used for this parameter depends on the TC parameter value used for the terminal. The number of idle characters inserted must be sufficient to provide the time needed by the terminal to physically return the carriage of the device to its left margin for the next line of output. This parameter is optional.

CN PARAMETER

Use the CN parameter to specify the hexadecimal value of the cancel-input-line character for this terminal. When the terminal user enters this character, the preceding logical input line is discarded by the TIP. This parameter is optional.

CT PARAMETER

Use the CT parameter to establish the hexadecimal value of the control character for this terminal. When the terminal user enters this character, the TIP interprets the rest of the current logical line of input as a parameterization command. This parameter is optional.

DLC PARAMETER

Use the DLC parameter to declare the number $(1 \leq dlc \leq 4096)$ of characters in each transparent mode upline message from this terminal. When the application program servicing this terminal initiates transparent mode operation, the TIP terminates transparent mode operation after dlc characters are transmitted. This optional parameter is not legal for terminals with TC=2741. The default value for this parameter is 2043.

DLTO PARAMETER

Use the DLTO parameter to specify whether transparent mode timeout should be expected for the terminal. This optional parameter is not legal for terminals with TC=2741. The following reserved word values are legal for this parameter.

YES Transparent mode timeout should be expected.

NO Transparent mode timeout should not be expected.

If YES is specified, the TIP interprets a 200- to 400-millisecond timeout as the end of upline transparent mode transmission between the terminal and an application program.

DLX PARAMETER

Use the DLX parameter to specify the hexadecimal value of the transparent mode delimiter character for the terminal. When the terminal user enters this character during transparent mode operation, the TIP ends transparent mode input. This optional parameter is not legal for terminals with TC=2741. The hexadecimal value to declare for a given character depends on the character code generated by the terminal. The default value used for DLX is $0D_{16}$ (a carriage return).

EP PARAMETER

Use the EP parameter to specify whether echoplexing (character echoing) should be performed for the terminal. The following reserved word values are legal for this optional parameter.

YES The TIP should perform echoplexing.

NO The TIP should not perform echoplexing.

IN PARAMETER

Use the IN parameter to specify the character mode input source for the terminal. The following values are legal for this optional parameter.

PT Character mode input occurs from paper tape.

KB Character mode input occurs from the keyboard.

OP PARAMETER

Use the OP parameter to specify the output mechanism for the terminal. The following reserved word values are legal for this optional parameter.

PR Output occurs on a printer

PT Output is punched on paper tape

DI Output is displayed on a console

LI PARAMETER

Use the LI parameter to specify the number $(0 \leq li \leq 99)$ of idle characters to insert in the output of the terminal after a line feed is initiated. The number of idle characters inserted determines the vertical line spacing and tabulation effects of the line feed operation. This parameter is optional.

PG PARAMETER

Use the PG parameter to specify whether the TIP should wait at each output page boundary for terminal user acknowledgement before displaying the next page of data. The following reserved word values are legal for this optional parameter.

YES Page wait should occur.

NO Page wait should not occur.

SE PARAMETER

Use the SE parameter to specify whether special input editing is to be performed for this terminal. If special editing is selected, any cancel-input-line, linefeed, or backspace character entered from the terminal is removed from the data sent upline, and the TIP performs the appropriate function for the character. This feature is normally used to enable entry of APL overstrike characters. The following values are legal for this optional parameter.

<u>NO</u>	The TIP should not perform editing of cancel, backspace, and linefeed.
YES	The TIP should perform special editing of upline data.

ABL PARAMETER

Use the ABL parameter to specify the application block limit, or the total number (1≤abl≤7) of downline blocks that can be outstanding (unacknowledged) between an application program and this terminal. This parameter is optional. The value chosen should be sufficient for the application program to keep the terminal busy by maintaining ABL outstanding blocks of bsz characters. The value you declare for ABL should be greater than or equal to the network block limit (NBL) of the terminal. Guidelines for selecting ABL and NBL parameter values are given in the Buffering of Data subsection later in this section and in table 4-3.

BSZ PARAMETER

Use the BSZ parameter to specify the block size, or number (1≤bsz≤2043) of characters, that form a block to or from this terminal. The value chosen is a function of device type, line speed, and terminal class, as shown in table 4-3. You should choose the value for this parameter so that ABL outstanding blocks of BSZ characters constitute a queue sufficient to keep the terminal continuously busy.

DI PARAMETER

Use the DI parameter to specify that the terminal is disabled at network initiation. If you specify this optional parameter, the terminal cannot access the network until the local operator enables it. If you omit DI, the terminal is assumed to be enabled.

This parameter cannot be declared for terminals on automatic recognition lines unless you have declared the STIP and TC parameters.

PARAMETERIZATION RESTRICTIONS

Table 4-4 shows the legal combinations of LINE and TERMINAL parameters for ASYNC lines.

Automatic Recognition

With the TERMINAL statement variant for the ASYNC TIP, values for the STIP, TSPEED, CSET, and DT parameters can be automatically determined.

The NDL processor permits up to 15 terminal definitions to share the same address on a line. (Two terminal definitions have different addresses if you specify different value declarations for the DT parameter in the two TERMINAL statements.) If the terminals are on AUTO lines, any or all of the definitions for an address can be enabled at any one time. Otherwise, no more than one definition can be enabled at any one time. Regardless of the type of definition, only one definition per address can be active at a time.

TABLE 4-3. BLOCK LIMITS AND SIZES (ASYNC AND X25 PROTOCOLS)

Line Speed	Terminal Class	ABL	BSZ	
			Recommended	Default
110, 134, 150	M33, 713, 2741, M40, H2000, 751, T4014	2	225	130
300	M33, 713, 2741, M40, H2000, 751, T4014	2	225	130
600	M33, 713, M40, H2000, 751, T4014	2	225	130
1200	M33, 713, M40, H2000, 751, T4014	2	225	130
2400	M33, 713, M40, H2000, 751, T4014	2	465	130
4800	M33, 713, M40, H2000, 751, T4014	2	885	130
9600	M33, 713, M40, H2000, 751, T4014	3†	885	130

†Recommended ABL; default is 2. Application programs that do not make provision for roll-out should use an ABL value of 2.

TABLE 4-4. LEGAL COMBINATIONS OF LINE AND TERMINAL STATEMENT PARAMETERS FOR ASYNC TIP (LTYPE A1, A2)

AUTO Declared or Used	LSPEED Declared or Used	STIP Declared or Used	TC Declared or Used	CSET Declared or Used	TSPEED Declared or Used
AUTO	None	N2741, None	M33, 713, M40, H2000, 751, T4014, None	ASCII, APLBP, APLTP, None	110, 150, 300, 600, 1200, None
AUTO	None	2741, None	2741, None	CORAPL, CORRES, EBCDAPL, EBCD, None	134, 300, None
AUTO	None	A110, A150, A300	M33, 713, M40, H2000, 751, T4014	None	None
AUTO	None	2741C, 2741E	2741	None	None
None	110, 150, 300, 600, 1200, 2400, 4800, 9600	N2741	M33, 713, M40, H2000, 751, T4014	APLBP, ASCII, APLTP, None†	None
		None	M33, 713, M40, H2000, 751, T4014	None†	None
None	134, 300	2741	2741	CORAPL, EBCD, EBCDAPL, CORRES	None
None	134	2741C, 2741E	2741	None	None
None	None††	N2741	M33, 713, M40, H2000, 751, T4014	APLBP, ASCII APLTP, None†	None
None	None	None	M33, 713, M40, H2000, 751, T4014	None†	None

†Default is ASCII.

††Default is 110.

Extended Asynchronous TIP

A CCP variant must have the extended asynchronous TIP option to support the following features.

- Any APL character set (CSET=APLBP, CSET=APLTP, CSET=CORAPL, or CSET=EBCDAPL)
- The extended binary-coded decimal character set (CSET=EBCD)
- The correspondence character set (CSET=CORRES)
- The special editing feature (SE=YES)
- The 2741 terminal class (TC=2741)
- A line speed or transmission rate of 134 bits per second (LSPEED=134 or TSPEED=134)

CCP rejects any attempt to use these features unless CCP has the extended asynchronous TIP option.

Output Format Specification

Although the range permitted for the output mechanism (OP) parameter includes both DI and PR for all terminal classes, the OP parameter is intended as a mechanism for switching between either DI or PR and PT. The network software formats output for a given terminal class based on the default value of the OP parameter for that terminal class. Declaration of the value DI for a terminal in a terminal class with a default value of PR for the OP parameter does not alter output formatting from that performed for a printer to that performed for a display. The only effect such a change has on output is to permit page wait functions for the terminal.

Transparent Mode Delimiter

You should select transparent mode input delimiters with care. If the only delimiter you select is defined by the DLX parameter, the character code you declare as the DLX value must be the code of a character that the physical terminal can input. If the only delimiter you declare is a character which cannot be input (either because of terminal hardware limitations or because of the PA parameter value declared for the terminal), then the Terminal Interface Program cannot terminate transparent mode input and the terminal will be trapped in that mode of operation once it has begun. When character code values above $7F_{16}$ are declared for the DLX parameter, a value of N must be declared for the PA parameter; otherwise, the eighth bit of each input character byte is set to zero and values above $7F_{16}$ cannot be detected.

X25 PROTOCOL

The X25 protocol supports asynchronous terminals connected to packet-switching networks, as described in section 1. This support is accomplished via a packet assembly/disassembly sub-TIP, which communicates with the network PAD. The format of the X25 TERMINAL statement variant is shown in figure 4-2.

```
terminal-name:TERMINAL  Δ[DT=dt,] TC=tc, [PRI,
    NBL=nbl, IAPPL=service, ALFAM=alfam, ALUN=alun,
    B1=b1††, B2=b2††, PA=pa††, PL=pl††, PW=pw††,
    BS=bs††, CI=ci††, CN=cn††, CT=ct††, DLX=dlx††,
    LI=li††, PG=pg, ABL=abl, BSZ=bsz,] W=w.
```

†The hexadecimal value to declare for a given character can be determined using table A-9.

††Ranges and/or defaults for this parameter depend on the TC value used for the terminal. See table 4-2.

Figure 4-2. TERMINAL Statement Format for X25 TIP

NOTE

To use the order-dependent value declaration convention, you must supply TERMINAL statement parameters in the order shown in appendix E.

TERMINAL-NAME PARAMETER

Use the terminal-name parameter to assign a name to the terminal for reference within the NDL program and by the local operator. This name should be unique within the current local division. This name is required.

DT PARAMETER

Use the DT parameter to specify the device type of the terminal. The following reserved words are valid for this parameter.

<u>CON</u>	Identifies a keyboard or CRT console terminal
USER	Identifies a site-defined device of type 7

TC PARAMETER

Use the TC parameter to specify the terminal class of the terminal. The reserved word value you specify for this parameter determines the parameterization ranges and defaults used for the terminal (see table 4-2). The legal values for TC are

M33

H2000

M40

751

713

T4014

PRI PARAMETER

Use the PRI parameter to specify that the terminal is to have data traffic priority. This optional reserved word indicates that data to or from the terminal is to travel through the network with a higher priority than data to or from terminals for which this priority has not been specified. Data to or from terminals without priority is suspended to relieve network traffic congestion before data to or from terminals with priority is suspended. PRI is usually specified for terminals with DT=CON when interactive traffic is to receive priority over batch traffic.

NBL PARAMETER

Use the NBL parameter to specify the allowable block limit for this terminal in the network, specifying the number of downline blocks that can be buffered between the host computer and the terminal. This optional parameter has the range $1 \leq nbl \leq 7$; the default value is 2. The value specified is used in conjunction with the ABL parameter to determine the block queuing required of the Network Access Method and of CCP. The relationship between the NBL and ABL parameters is explained later in this section under Buffering of Data and in table 4-3.

IAPPL PARAMETER

Use the IAPPL parameter to declare an initial application program for this terminal. This name must be one specified on an APPL statement in the current local division of the NDL program. The IAPPL parameter is optional. If you declare this parameter, the terminal is connected directly to the specified application upon completion of family name and user name validation; the terminal is not prompted by NVF for an application name. If you omit this parameter, the terminal is prompted by NVF; there is no default value.

ALFAM PARAMETER

Use the ALFAM parameter to specify the NOS family name to be used during automatic login of this terminal. This parameter is optional. If you declare ALFAM, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a family name during the initial login attempt, but can be prompted during subsequent login attempts, even though alfam is specified. You should choose the value for

ALFAM with consideration of the UID parameter specification for all application programs this terminal can access (see the APPL Statement description). The value you choose for this parameter can be

0	The system default family name is to be used.
Any valid value	Refer to the NOS System Maintenance Reference Manual for determination of valid values.

ALUN PARAMETER

Use the ALUN parameter to specify a NOS user name to use during automatic login of this terminal. This parameter is optional. If you declare the ALUN parameter for a terminal, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a user name and a password during the initial login attempt. The terminal can be prompted for manual entry of the user name and password during subsequent login attempts. The value you declare for this parameter can be any valid user name from the NOS VALIDUz file (see the NOS System Maintenance Reference Manual).

If you configure the terminal with TAF as its initial application program (IAPPL=TAF), the value used for the ALUN parameter must be the same as that used to identify the terminal in the NOS terminal file NCTFid. NCTFid is described in detail in the System Maintenance Reference Manual.

B1 PARAMETER

Use the B1 parameter to specify the hexadecimal value of the break-1 character for the terminal. When this character is entered from the terminal, the TIP sends a break-1 supervisory message to the application program currently associated with the terminal. This parameter is optional.

B2 PARAMETER

Use the B2 parameter to specify the hexadecimal value of the break-2 character for this terminal. When this character is entered from the terminal, the TIP sends a break-2 supervisory message to the application program currently associated with the terminal. This parameter is optional.

PA PARAMETER

Use the PA parameter to specify the parity used by the TIP for transmissions to and from this terminal. The reserved words legal for this optional parameter are

Z	Zero parity. The upper character bit is always zero.
O	Odd parity. The upper character bit varies.
E	Even parity. The upper character bit varies.
N	No parity. The upper character bit is data.

PL PARAMETER

Use the PL parameter to specify the number ($0 \leq pl \leq 255$) of physical lines per page of input or output for this terminal. This parameter is optional.

PW PARAMETER

Use the PW parameter to establish the number ($0 \leq pw \leq 255$) of characters per physical line of input or output for this terminal. This parameter is optional.

BS PARAMETER

Use the BS parameter to specify the hexadecimal value of the backspace character for this terminal. This parameter is optional.

CI PARAMETER

Use the CI parameter to specify the number ($0 \leq ci \leq 99$) of idle characters to insert in the upline data from this terminal after a carriage return. The default value used for this parameter depends on the TC parameter value used for the terminal. The number of idle characters inserted must be sufficient to provide the time needed by the terminal to physically return the carriage of the device to its left margin for the next line of output. This parameter is optional.

CN PARAMETER

Use the CN parameter to specify the hexadecimal value of the cancel-input-line character defined for this terminal. When the terminal user enters this character, the preceding logical input line is discarded by the TIP. This parameter is optional.

CT PARAMETER

Use the CT parameter to specify the hexadecimal value of the control character for this terminal. When the terminal user enters this character, the TIP interprets the rest of the current logical line of input as a parameterization command. This parameter is optional.

DLX PARAMETER

Use the DLX parameter to specify the hexadecimal value of the transparent mode delimiter character for the terminal. When the terminal user enters this character during transparent mode operation, the TIP ends transparent mode input. The hexadecimal value to declare for a given character depends on the character code generated by the terminal. The default value used for this optional parameter is $0D_{16}$ (a carriage return).

LI PARAMETER

Use the LI parameter to specify the number ($0 \leq li \leq 99$) of idle characters to insert in the output of this terminal after a line feed is initiated. The number of idle characters inserted determines the vertical line spacing and tabulation effects of the line feed operation. This parameter is optional.

PG PARAMETER

Use the PG parameter to specify whether the TIP should wait at each output page boundary for terminal user acknowledgement before displaying the next page of data. The following reserved word values are legal for this optional parameter.

YES	Page wait should occur.
NO	Page wait should not occur.

ABL PARAMETER

Use the ABL parameter to specify the application block limit, or the total number ($1 \le abl \le 7$) of downline blocks that can be outstanding (unacknowledged) between an application program and this terminal. This parameter is optional. The value you choose should be sufficient for the application program to keep the terminal busy by maintaining ABL outstanding blocks of BSZ characters. The value you declare for ABL should be greater than or equal to the network block limit (NBL) of the terminal. Guidelines for selecting ABL and NBL parameter values are given in the Buffering of Data subsection and in table 4-3.

BSZ PARAMETER

Use the BSZ parameter to specify the block size, or number ($1 \le bsz \le 2043$) of characters, that form a block to or from this terminal. The value chosen is a function of device type, line speed, and terminal class, as shown in table 4-3. You should choose the value for this optional parameter so that ABL outstanding blocks of BSZ characters constitute a queue sufficient to keep the terminal continuously busy.

W PARAMETER

Use the W parameter to specify the packet window, $1 \le w \le 127$, or the maximum number of outstanding unacknowledged upline packets allowed before the NPU suspends further input for the terminal. This parameter is required.

PARAMETERIZATION RESTRICTIONS

Table 4-5 shows the legal combinations of parameters for the X25 LINE statement and TERMINAL statement variant.

Switched Virtual Circuit Definition

On an X.25 line, the parameters for all TERMINAL statements used to declare switched virtual circuits (SVCs) must be identical. (An SVC call is not automatically recognized; each incoming SVC call is assigned to the next available TERMINAL statement.) Once the parameters have been declared for the first SVC TERMINAL statement, you must use the abbreviated format of the TERMINAL statement (figure 4-3) to copy them for subsequent switched virtual circuits.

terminal:TERMINAL†

terminal	The name to assign to this switched virtual circuit for reference within the NDL program and by the local operator. This name should be unique within the current local division. This name is required; there is no default value.

†Valid only for switched virtual circuits on lines with TIPTYPE=X25.

Figure 4-3. Abbreviated Format of TERMINAL Statement (X.25 Lines)

Transparent Mode Delimiter

You should select transparent mode input delimiters with care. If the only delimiter you select is defined by the DLX parameter, the character code you declare as the DLX value must be the code of a character that the physical terminal can input. If the only delimiter you declare is a character which cannot be input (either because of terminal hardware limitations or because of the PA parameter value declared for the terminal), then the Terminal Interface Program cannot terminate transparent mode input and the terminal will be trapped in that mode of operation once it has begun. When you declare character code values above $7F_{16}$ for the DLX parameter, you must declare a value of N for the PA parameter; otherwise, the eighth bit of each input character byte is set to zero and values above $7F_{16}$ cannot be detected.

MODE 4 PROTOCOL

Mode 4 terminals meet the requirements of the CDC Mode 4A or Mode 4C protocols. They operate on synchronous communication lines, organized into clusters. You can define up to 15 clusters on a single dedicated line.

TABLE 4-5. LEGAL COMBINATIONS OF LINE AND TERMINAL STATEMENT PARAMETERS FOR X25 TIP (LTYPE H1)

AUTO Declared or Used	LSPEED Declared or Used	STIP Declared or Used	TC Declared or Used	CSET Declared or Used
None	None	None	M33, 713, M40, H2000, 751, T4014	None†

†Default is ASCII.

The TERMINAL statement variant for Mode 4 terminals is shown in figure 4-4.

For AUTO Lines†

```
terminal-name:TERMINAL  [Δ STIP=stip, CSET=cset,
    CA=ca, TA=ta, DT=dt, TC=tc, PRI, NBL=nbl,
    IAPPL=service, ALFAM=alfam, ALUN=alun, B1=b1††,
    B2=b2††, OCTERM=octerm, PL=pl††, PW=pw††,
    CN=cn††, CT=ct††, PG=pg, DO=do††, ABL=abl,
    BSZ=bsz, DI, SDT=sdt, XBZ=xbz].
```

For non-AUTO Lines†

```
terminal-name:TERMINAL   Δ STIP=stip, [CSET=cset,]
    CA=ca, TA=ta, [DT=dt,] TC=tc, [PRI, NBL=nbl,
    IAPPL=service, ALFAM=alfam, ALUN=alun, B1=b1††,
    B2=b2††, OCTERM=octerm, PL=pl††, PW=pw††,
    CN=cn††, CT=ct††, PG=pg, DO=do††, ABL=abl,
    BSZ=bsz, DI, SDT=sdt, XBZ=xbz].
```

†The hexadecimal value to declare for a given character can be determined using table A-9.

††Ranges and/or defaults for this parameter depend on the TC value used for the terminal. See table 4-8.

Figure 4-4. TERMINAL Statement Format for Mode 4 TIP

NOTE

To use the order-dependent value declaration convention, you must supply TERMINAL statement parameters in the order shown in appendix E.

TERMINAL-NAME PARAMETER

Use the terminal-name parameter to assign a name to the terminal for reference within the NDL program and by the local operator. This name is required and should be unique within the current local division.

STIP PARAMETER

Use the STIP parameter to specify the sub-TIP type of the terminal. This parameter is optional for AUTO lines; otherwise, it is required. If you specify the parameter for one terminal on the line, you must specify it for all terminals on the line. If you supply this parameter, the network software performs an additional match against the local configuration file while identifying the terminal. Legal reserved word values for STIP are

M4A   CDC Mode 4A terminal (CDC 200 User Terminals models 217, 222, 224; models 731-12, 732-12, 734-101/151, and 734-201/251) or any terminal emulating a 200 User Terminal.

M4BCD   Alternate form: same as M4A with CSET=BCD (see below). M4BCD cannot be used if CSET is specified.

M4ASCII   Alternate form: same as M4A with CSET=ASCII (see below). M4ASCII cannot be used if CSET is specified.

M4C   Mode 4C terminal (CDC models 711-10, 714-10/20, and 714-30).

If TIPTYPE = USER1, USER2, USER3, or USER4, STIP can be any of the above values, and will be represented in the LCF summary as 1 for M4A terminals or 2 for M4C terminals.

CSET PARAMETER

Use the CSET parameter to specify the code set of the terminal. The possible values for this parameter are

BCD	BCD code set
ASCII	ASCII code set

For AUTO lines CSET is optional; if you specify it for one terminal on the line, you must specify it for all terminals on the line. If you supply CSET, the network software performs an additional match while identifying the terminal in the configuration file. If you omit CSET, no additional match is performed. The default value is not provided for AUTO lines.

For non-AUTO lines with STIP = M4A, you must specify CSET; if STIP = M4C, ASCII is assumed.

CA PARAMETER

Use the CA parameter to specify the cluster hardware address ($70_{16} \leq ca \leq 7F_{16}$) of a terminal on the line. When the line is configured for automatic recognition of terminals, this parameter is optional; otherwise it is required. If you specify CA for one terminal on a line, you must specify it for all terminals on the line. If you supply this parameter for an AUTO line, the network software performs an additional match while identifying the terminal in the configuration file.

The TERMINAL statements for all terminals on a line with the same cluster address must be grouped together in the NDL program.

TA PARAMETER

Use the TA parameter to specify a terminal's hardware address within its cluster on the line. For AUTO lines this parameter is optional; otherwise, it is required. If you specify TA for one terminal on a line, you must specify it for all terminals on the line. If you supply this parameter for an AUTO line, the network software performs an additional match while identifying the terminal in the configuration file. The value to use for this parameter depends on the sub-TIP type of the terminal:

60_{16}	Mode 4A terminals (STIP=M4A, M4BCD, or M4ASCII)
61_{16} thru $6F_{16}$	Mode 4C terminals (STIP=M4C)

Each enabled mode 4C terminal in a cluster must have a unique TA value. A 714 or 714X console that is declared to be an owning console of a line printer must have a TA value of 61_{16}.

DT PARAMETER

Use the DT parameter to specify the device type of the terminal. The following reserved words are valid for this parameter.

CON Identifies a keyboard or CRT console terminal

CR Identifies a card reader terminal

LP Identifies a line printer terminal

USER Identifies a site-defined device of type 7

This parameter is optional; however, the device type is part of the hardware address of the terminal, and the appropriate value should be declared to correctly configure passive devices.

TC PARAMETER

Use the TC parameter to specify the terminal class of the terminal. The reserved word value you specify for this parameter determines the parameterization ranges and defaults used for the terminal. The legal values depend on the values declared for the STIP parameter, as shown in table 4-6.

For AUTO lines, this parameter is optional; otherwise, it is required. If you specify TC for one terminal on the line, you must specify it for all terminals on the line. If you omit the parameter, the default values indicated in table 4-6 are used at the time the terminal is automatically recognized.

PRI PARAMETER

Use the PRI parameter to specify that the terminal is to have data traffic priority. This optional reserved word indicates that data to or from the terminal is to travel through the network with a higher priority than data to or from terminals for which this priority has not been specified. Data to or from terminals without priority is suspended to relieve network traffic congestion before data to or from terminals with priority is suspended. PRI is usually specified for terminals with DT=CON when interactive traffic is to receive priority over batch traffic.

NBL PARAMETER

Use the NBL parameter to specify the allowable block limit for this terminal in the network, specifying the number of output blocks that can be buffered between the host computer and the terminal. This optional parameter has the range $1 \leq nbl \leq 7$.

For interactive terminals (DT=CON), the value you specify for NBL is used in conjunction with the ABL parameter to determine the block queuing required of the Network Access Method and of CCP. For passive terminals NBL specifies the number of blocks of size BSZ allowed in the network.

The relationship between the NBL and other blocking and buffering parameters is described in the Buffering of Data subsection, later in this section.

IAPPL PARAMETER

Use this parameter to declare an initial application program for this terminal. This name must be one specified on an APPL statement in the current local division of the NDL program. This parameter is optional for terminals with DT=CON and required for passive terminals. If you declare IAPPL, the terminal is connected directly to the specified application upon completion of family name and user name validation; the terminal is not prompted by NVF for an application name. If you omit IAPPL, an interactive (DT=CON) terminal is prompted by NVF; there is no default value.

TABLE 4-6. LEGAL COMBINATIONS OF LINE AND TERMINAL STATEMENT PARAMETERS FOR MODE4 TIP (LTYPE S1, S2, S3)

AUTO Declared or Used	LSPEED Declared or Used	STIP Declared or Used	TC Declared or Used	CSET Declared or Used	TSPEED Declared or Used
AUTO	None	M4A, None	200UT, 731, 734, None	ASCII, BCD, None	None
AUTO	None	M4C, None	711, 714, 714X, None	ASCII, None	None
AUTO	None	M4ASCII, M4BCD	200UT, 731, 734, None	None	None
None	None	M4A	200UT, 731, 734	ASCII, BCD	None
None	None	M4C	711, 714, 714X	ASCII, None[†]	None
None	None	M4ASCII, M4BCD	200UT, 731, 734	None	None

[†] Default is ASCII.

ALFAM PARAMETER

Use the ALFAM parameter to specify the NOS family name to be used during automatic login of this terminal. This parameter is optional for interactive terminals (DT=CON); it is required for passive terminals. If you declare ALFAM, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a family name during the initial login attempt, but can be prompted during subsequent login attempts, even though ALFAM is specified. You should choose the value for this parameter with consideration of the UID parameter specification for all application programs this terminal can access (see the APPL Statement description).

If you specify ALFAM=0, the system default family name will be used for automatic login. To determine other valid values, see the System Maintenance Reference Manual.

ALUN PARAMETER

Use the ALUN parameter to declare a NOS user name to use during automatic login of this terminal. This parameter is optional for interactive terminals, but required for passive terminals. If you declare ALUN for a terminal, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a user name and a password during the initial login attempt. Interactive terminals can be prompted for manual entry of the user name and password during subsequent login attempts. The value you declare for this parameter can be any valid user name from the NOS VALIDUz file (see the NOS System Maintenance Reference Manual).

If you configure the terminal with TAF as its initial application program (IAPPL=TAF), the value used for the ALUN parameter must be the same as that used to identify the terminal in the NOS terminal file NCTFid. NCTFid is described in detail in the System Maintenance Reference Manual.

B1 PARAMETER

Use the B1 parameter to specify the hexadecimal value of the break-1 character for this terminal. When this character is entered from the terminal, the TIP sends a break-1 supervisory message to the application program currently associated with the terminal. B1 is optional for interactive terminals (DT=CON); it is not legal for passive terminals.

B2 PARAMETER

Use the B2 parameter to specify the hexadecimal value of the break-2 character for this terminal. When this character is entered from the terminal, the TIP sends a break-2 supervisory message to the application program currently associated with the terminal. B2 is optional for interactive terminals (DT=CON); it is not legal for passive terminals.

OCTERM PARAMETER

Use the OCTERM parameter to specify the owning console for this terminal. This parameter is required for passive terminals; it is not legal for interactive terminals (DT=CON). The value you declare for this parameter must be a terminal-name identifier used in a TERMINAL statement in the same division of the NDL program. When declared for a 200UT passive terminal, the element name you specify must be that of a mode 4A interactive terminal on the same line, with the same CA and TA parameter values. When you specify OCTERM for a 714 or 714X passive terminal, the terminal name you specify must be that of a mode 4C interactive terminal on the same line, with a TA value of 61_{16} and the same CA parameter value as the passive terminal. If you specify OCTERM and the passive terminal is enabled, the owning console must be enabled.

PL PARAMETER

Use the PL parameter to specify the number ($0 \le pl \le 255$) of physical lines per page of input or output for this terminal. This parameter is optional for interactive terminals (DT=CON); it is not legal for passive terminals.

PW PARAMETER

Use the PW parameter to specify the number ($0 \le pw \le 255$) of characters per physical line of input or output for this terminal. This parameter is optional.

CN PARAMETER

Use the CN parameter to specify the hexadecimal value of the cancel-input-line character for this terminal. When the terminal user enters this character, the preceding logical input line is discarded by the TIP. This optional parameter is legal for interactive terminals (DT=CON); it is not legal for passive terminals.

CT PARAMETER

Use the CT parameter to specify the hexadecimal value of the control character for this terminal. When the terminal user enters this character, the TIP interprets the rest of the current logical line of input as a parameterization command. CT is optional for interactive terminals (DT=CON); it is not legal for any other terminals.

PG PARAMETER

Use the PG parameter to specify whether the TIP should pause at each output page boundary and wait for the terminal user to request the next page of data. This parameter is optional for interactive terminals (DT=CON); it is not legal for any other terminals. Legal reserved word values for PG are

YES	Page wait should occur.
NO	Page wait should not occur.

DO PARAMETER

Use the DO parameter to specify the device ordinal ($1 \le do \le 7$) of this terminal within its device type. This parameter is legal only for passive terminals of terminal classes 714 and 714X; it is not legal for any other terminals. The device ordinal must be unique within each cluster address. DO is required except if there is only one terminal of each device type within the cluster; in this case, DO defaults to 1.

ABL PARAMETER

Use the ABL parameter to specify the application block limit, or the total number ($1 \leq abl \leq 7$) of downline blocks that can be outstanding (unacknowledged) between an application program and an interactive terminal. If you specify a value for this optional parameter for a passive terminal, the NDL processor will ignore the value.

The value you choose should be sufficient for the application program to keep the terminal busy by maintaining ABL outstanding blocks of BSZ characters. The value you declare for ABL should be greater than or equal to the network block limit (NBL) of the terminal. Guidelines for selecting ABL and NBL parameter values are given in the Buffering of Data subsection.

Recommended values for ABL are given in table 4-7. The default value for this parameter is 2.

BSZ PARAMETER

Use the BSZ parameter to specify the block size, or number of characters that form a block to or from an interactive terminal. The value you choose or the default used is a function of device type, line speed, and terminal class, as shown in table 4-7. You should choose the value for this parameter so that ABL outstanding blocks of BSZ characters constitute a queue sufficient to keep the terminal continuously busy. The legal BSZ values for interactive devices are

$$256 \leq bsz \leq 2043$$

TABLE 4-7. BLOCK LIMITS AND SIZES - MODE4 PROTOCOL

Line Speed	Terminal Class	ABL	NBL		XBZ		BSZ	
			Recommended	Default	Recommended	Default	Recommended	Default
2400	200UT/734 (DT=CON)	2	2	2			1040	1040
	200UT (DT=CR)		7	7	1000	1000	1280	1280
	200UT (DT=LP)		1	1	1000	1000	1920	1920
	711/714/714X (DT=CON)	2	2	2			1280	1280
	714 (DT=LP)		1	1	240	240	640	640
	714X (DT=LP)		1	1	1280 1920	1280	1280 1920	1280
4800	200UT/734 (DT=CON)	2	2	2			1040	1040
	200UT (DT=CR)		7	7	1000	1000	1280	1280
	200UT (DT=LP)		1	1	1000	1000	1920	1920
	711/714/714X (DT=CON)	2	2	2			1280	1280
	714 (DT=LP)		1	1	240	240	640	640
	714X (DT=LP)		1	1	1280 1920	1280	1280 1920	1280
9600	200UT/734 (DT=CON)	2	2	2			1040	1040
	200UT (DT=CR)		7	7	1000	1000	1280	1280
	200UT (DT=LP)		1	1	1000	1000	1920	1920
	711/714/714X (DT=CON)	2	2	2			1280	1280
	714 (DT=LP)		1	1	240	240	640	640
	714X (DT=LP)		1	1	1280 1920	1280	1280 1920	1280
19.2K	200UT/734 (DT=CON)	2	2	2			1040	1040
	200UT (DT=CR)		7	7	1000	1000	1280	1280
	200UT (DT=LP)		2	1	1000	1000	1920	1920
	711/714/714X (DT=CON)	2	2	2			1280	1280
	714 (DT=LP)		1	1	240	240	640	640
	714X (DT=LP)		1	1	1280 1920	1280	1280 1920	1280

For passive terminals, the parameter specifies the size of blocks transferred between the host and an NPU. It is recommended that you assign one of the following values to BSZ.

640

1280

1920

If the value that you assign is not one of the recommended values but is within the range 1≤bsz≤2043, the NDL Processor will adjust the value as follows.

Value Specified	Adjusted Value
1≤bsz≤640	640
641≤bsz≤1280	1280
1281≤bsz≤2043	1920

The NDL Processor issues an informative message if it adjusts the BSZ value that you specify.

DI PARAMETER

Use the DI parameter to specify that the terminal is disabled at network initiation. If this optional parameter is specified, the terminal cannot access the network until the local operator enables it. If DI is omitted the terminal is assumed to be enabled.

You cannot declare DI for terminals on automatic recognition lines unless the STIP, CA, TA, and TC parameters have been declared.

If you specify DI for the owning console of a passive terminal, you must specify DI for that passive terminal.

SDT PARAMETER

Use the SDT parameter to specify the sub-device type of the terminal. This parameter is optional for printers (DT=LP); it is not legal for any other terminals. SDT can have the following values.

<u>A6</u>	ASCII 64-character train type
A9	ASCII 96-character train type
B6	BCD 64-character train type

XBZ PARAMETER

Use the XBZ parameter to specify the number (200≤xbz≤2047) of characters in a transmission block for the terminal. This parameter is optional for passive terminals; it is not legal for interactive terminals. The default for XBZ is as follows.

240	Default for 714 terminals
1000	Default for 200UT terminals
1200	Default for 714X terminals

The TIP reformats downline data if necessary, so as not to exceed XBZ for transmission to the terminal; thus, the value that you declare for this parameter should not exceed the actual buffer size of the terminal.

PARAMETERIZATION RESTRICTIONS

Table 4-8 defines the ranges and defaults for parameters in the Mode 4 TERMINAL statement variant.

Other parameterization restrictions for Mode 4 terminals are discussed below.

Automatic Recognition

On mode 4 synchronous lines, the sub-TIP type, cluster address, terminal console address, code set, and device type can be automatically determined.

You can define 15 enabled terminals at a single hardware address if the line is specified as an AUTO line. On a non-AUTO line only one terminal can be enabled at a hardware address. The hardware address comprises the NPU terminal node number, the port number of the communication line, a cluster controller address, the terminal console address, and the terminal device type. You can configure each component of the hardware address with a TERMINAL statement parameter.

Unique Device Ordinal Requirement

For an application program such as the Remote Batch Facility, the device ordinal for each passive terminal associated with an owning console must be unique within the device type of the passive terminal. Passive terminals with a terminal class of 200UT are implicitly assigned device ordinals of 1. For passive terminals with a TC value of 714 or 714X, the device ordinal represents the physical input or output stream number of the device within its device type, and thus must be unique within its device type. (A default value of one exists when there is only one device of each type owned by the same owning console.) The DO parameter is the only parameter with this uniqueness requirement.

HASP PROTOCOL

Multileaving terminals operate on synchronous communication lines and form multiple-device workstations. These workstations satisfy the requirements of the IBM Corporation HASP protocol. Within the workstation you can define up to 15 console terminals, 7 card readers, 7 line printers, and a combination of 7 card punches or plotters.

The format of the TERMINAL statement variant for the HASP protocol is shown in figure 4-5. Each parameter is described in detail below.

NOTE

> To use the order-dependent value declaration convention, you must supply TERMINAL statement parameters in the order shown in appendix E.

TERMINAL-NAME PARAMETER

Use the terminal-name parameter to assign a name to the terminal for reference within the NDL program and by the local operator. This name should be unique within the current local division. This name is required.

TABLE 4-8. PARAMETERIZATION RANGES AND DEFAULTS BY TERMINAL CLASS - MODE4 TIP

Parameter	Terminal Class											
	200UT		714X		711		714		731		734	
	Range	Default	Range	Default	Range	Default	Range	Default	Range	Default	Range	Default
B1	0 thru $7F_{16}$	$3A_{16}$:	0 thru $7F_{16}$	$3A_{16}$:	0 thru $7F_{16}$	$3A_{16}$:	0 thru $7F_{16}$	$3A_{16}$:	0 thru $7F_{16}$	$3A_{16}$:	0 thru $7F_{16}$	$3A_{16}$:
B2	0 thru $7F_{16}$	29_{16})	0 thru $7F_{16}$	29_{16})	0 thru $7F_{16}$	29_{16})	0 thru $7F_{16}$	29_{16})	0 thru $7F_{16}$	29_{16})	0 thru $7F_{16}$	29_{16})
CN	0 thru $7F_{16}$	28_{16} (	0 thru $7F_{16}$	28_{16} (	0 thru $7F_{16}$	28_{16} (	0 thru $7F_{16}$	28_{16} (	0 thru $7F_{16}$	28_{16} (	0 thru $7F_{16}$	28_{16} (
CT	0 thru $7F_{16}$	25_{16} %	0 thru $7F_{16}$	25_{16} %	0 thru $7F_{16}$	25_{16} %	0 thru $7F_{16}$	25_{16} %	0 thru $7F_{16}$	25_{16} %	0 thru $7F_{16}$	25_{16} %
DLX	None	None	None	ETX	None	ETX	None	ETX	None	None	None	None
DO	None	DT=CR,1 DT=LP,1	DT=LP 1 thru 7	None	None	None	DT=LP 1 thru 7	None	None	None	None	None
PG	YES, NO	YES	YES, NO	YES	YES, NO	YES	YES, NO	YES	YES, NO	YES	YES, NO	YES
PL	0 thru 255	13	0 thru 255	16	0 thru 255	16	0 thru 255	16	0 thru 255	13	0 thru 255	13
PW (DT=CON)	0 thru 255	80	0 thru 255	80	0 thru 255	80	0 thru 255	80	0 thru 255	80	0 thru 255	80
PW (DT=CR)	80	80	None	None	None	None	None	None	None	None	None	None
PW (DT=LP)	50 thru 150	136	50 thru 150	80	None	None	50 thru 150	80	None	None	None	None
SDT (DT=LP)	A6, A9 or B6	A6	A6, A9, or B6	A6	None	None	A6, A9, or B6	A6	None	None	None	None
XBZ	200 thru 2047	1000	200 thru 2047	1200	None	None	200 thru 2047	240	None	None	None	None

LEGEND:

Range — Hexadecimal values of 0 through $7F_{16}$ correspond to characters in the 128-character ASCII set of table A-9. The value $3D_{16}$ is not legal.

Default — Character representations beneath default values are the keyboard keys that cause entry of the default value for the archetype terminal of the terminal class.

For AUTO Lines†

terminal:TERMINAL [Δ STIP=stip, DT=dt, TC=tc, PRI, NBL=nbl, IAPPL=service, ALFAM=alfam, ALUN=alun, OCTERM=octerm, B1=b1††, B2=b2††, PL=pl††, PW=pw††, CN=cn††, CT=ct††, DO=do††, PDO=pdo††, ABL=abl, BSZ=bsz, DI, SDT=sdt, CO=co, XBZ=xbz].

For Non-AUTO Lines†

terminal:TERMINAL Δ [STIP=stip, DT=dt,] TC=tc, [PRI, NBL=nbl, IAPPL=service, ALFAM=alfam, ALUN=alun, OCTERM=octerm, B1=b1††, B2=b2††, PL=pl††, PW=pw††, CN=cn††, CT=ct††, DO=do††, PDO=pdo††, ABL=abl, BSZ=bsz, DI, SDT=sdt, XBZ=xbz].

†The hexadecimal value to declare for a given character can be determined using table A-9.

††Ranges and/or defaults for this parameter depend on the TC value used for the terminal. See table 4-9.

Figure 4-5. TERMINAL Statement Format for HASP TIP

STIP PARAMETER

Use the STIP parameter to specify the sub-TIP type of the terminal. This parameter is optional. If you specify STIP for one terminal on an AUTO line, you must specify it for all terminals on the line. For non-AUTO lines if you omit STIP, the default depends on the TC value used for the terminal. The following values are legal for STIP.

POST	HASP protocol post-print terminal
PRE	HASP protocol pre-print terminal

If TIPTYPE = USER1, USER2, USER3, USER4, you can assign either of the values above to STIP; in the LCF summary STIP will be represented as 1 for POST or as 2 for PRE.

DT PARAMETER

Use the DT parameter to specify the device type of the terminal. This parameter is optional; however, the device type is part of the hardware address of HASP terminals, and you should declare appropriate values to correctly configure passive devices. The following reserved words are valid for this parameter.

CON	Identifies a keyboard or CRT console terminal
CR	Identifies a card reader terminal
LP	Identifies a line printer terminal
CP	Identifies a card punch terminal
PL	Identifies a plotter terminal
USER	Identifies a site-defined device of type 7

TC PARAMETER

Use the TC parameter to specify the terminal class of the terminal. The value declared for this parameter determines the parameterization ranges and defaults used for the terminal (see table 4-9). This parameter is optional for AUTO lines; if you supply TC for one terminal on the line, you must supply it for all terminals on the line. For non-AUTO lines, this parameter is required. The legal values for this parameter are

HASP	A HASP protocol post-print terminal
HPRE	A HASP protocol pre-print terminal

PRI PARAMETER

Use the PRI parameter to specify that the terminal is to have data traffic priority. This optional reserved word indicates that data to or from the terminal is to travel through the network with a higher priority than data to or from terminals for which this priority has not been specified. Data to or from terminals without priority is suspended to relieve network traffic congestion before data to or from terminals with priority is suspended. PRI is usually specified for terminals with DT=CON when interactive traffic is to receive priority over batch traffic.

NBL PARAMETER

Use the NBL parameter to specify the allowable block limit for this terminal in the network, specifying the number of output blocks that can be buffered between the host computer and the terminal. This optional parameter has the range $1 \le nbl \le 7$.

For interactive terminals (DT=CON), the NBL value is used in conjunction with the ABL parameter to determine the block queuing required of the Network Access Method and of CCP. For passive terminals, NBL specifies the number of blocks of size BSZ allowed in the network.

The relationship between the NBL and other blocking and buffering parameters is described in the Buffering of Data subsection, later in this section.

IAPPL PARAMETER

Use the IAPPL parameter to specify an initial application program for this terminal. This name must be one that you specify on an APPL statement in the current local division of the NDL program. IAPPL is optional for interactive terminals (DT=CON) and required for passive terminals. If you declare IAPPL, the terminal is connected directly to the specified application upon completion of family name and user name validation; the terminal is not prompted by NVF for an application name. If you omit the parameter, an interactive (DT=CON) terminal is prompted by NVF; there is no default value.

ALFAM PARAMETER

Use the ALFAM parameter to specify the NOS family name to be used during automatic login of this terminal. This parameter is optional for interactive terminals (DT=CON); it is required for passive terminals. If you declare ALFAM, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a family name during the initial login

TABLE 4-9. PARAMETERIZATION RANGES AND DEFAULTS BY TERMINAL CLASS - HASP TIP

Parameter	Terminal Class			
	HASP		HPRE	
	Range	Default	Range	Default
B1	0 thru $7F_{16}$	$3A_{16}$:	0 thru $7F_{16}$	$3A_{16}$:
B2	0 thru $7F_{16}$	29_{16})	0 thru $7F_{16}$	29_{16})
CN	0 thru $7F_{16}$	28_{16} (	0 thru $7F_{16}$	28_{16} (
CO	1 thru 255	None	1 thru 255	None
CT	0 thru $7F_{16}$	25_{16} %	0 thru $7F_{16}$	25_{16} %
DO (DT=CP, DT=CR, DT=LP, DT=PL)	1 thru 7	None	1 thru 7	None
PDO (DT=CP, DT=CR, DT=LP, DT=PL)	1 thru 7	DO value used	1 thru 7	DO value used
PL	0	0	0	0
PW (DT=CON)	0 thru 255	80	0 thru 255	80
PW (DT=CP)	80	80	80	80
PW (DT=CR)	80	80	80	80
PW (DT=LP)	50 thru 150	120	50 thru 150	120
PW (DT=PL)	None	80	None	80
SDT (DT=LP)	A6, A9, or B6	A6	A6, A9, or B6	A6
SDT (DT=CR)	29 or 26	29	29 or 26	29
SDT (DT=PL)	6BIT or 8BIT	6BIT	6BIT or 8BIT	6BIT

LEGEND:

Range   Hexadecimal values of 0 thru $7F_{16}$ correspond to characters in the 128-character ASCII set of table A-9. The value $3D_{16}$ is not legal.

Default   Character representations beneath default values are the keyboard keys that cause entry of the default value for the archetype terminal of the terminal class.

attempt, but can be prompted during subsequent login attempts, even though ALFAM is specified. This parameter can be 0, indicating that the system default family name is to be used for automatic login, or it can be any previously established valid value (see the NOS 1 system maintenance reference manual).

You should choose the value for this parameter with consideration of the UID parameter specification for all application programs this terminal can access (see the APPL Statement description).

ALUN PARAMETER

Use the ALUN parameter to declare a NOS user name to be used during automatic login of this terminal. This parameter is optional for interactive terminals, but required for passive terminals. If you declare this parameter for a terminal, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a user name and a password during the initial login attempt. Interactive terminals can be prompted for manual entry of the user name and password during subsequent login attempts. The value that you declare for ALUN can be any valid user name from the NOS VALIDUz file (see the NOS System Maintenance Reference Manual).

If you configure the terminal with TAF as its initial application program (IAPPL=TAF), the value used for the ALUN parameter must be the same as that used to identify the terminal in the NOS terminal file NCTFid. NCTFid is described in detail in the System Maintenance Reference Manual.

B1 PARAMETER

Use the B1 parameter to specify the hexadecimal value of the break-1 character for this terminal. When this character is entered from the terminal, the TIP sends a break-1 supervisory message to the application program currently associated with the terminal. This parameter is optional for interactive terminals (DT=CON); it is not legal for passive terminals.

B2 PARAMETER

Use the B2 parameter to specify the hexadecimal value of the break-2 character for this terminal. When this character is entered from the terminal, the TIP sends a break-2 supervisory message to the application program currently associated with the terminal. This parameter is optional for interactive terminals (DT=CON); it is not legal for passive terminals.

OCTERM PARAMETER

Use the OCTERM parameter to identify the owning console for this terminal. This parameter is required for passive terminals; it is not legal for interactive terminals (DT=CON). The value that you declare for OCTERM must be the terminal-name identifier of a HASP interactive terminal on the same communication line. If this parameter is specified, and the passive terminal is enabled, the owning console must be enabled.

PL PARAMETER

Use the PL parameter to specify the number ($0 \leq pl \leq 255$) of physical lines per page of input or output for this terminal. This parameter is optional for interactive terminals (DT=CON); it is not legal for passive terminals.

PW PARAMETER

Use the PW parameter to specify the number ($0 \leq pw \leq 255$) of characters per physical line of input or output for this terminal. This optional parameter is legal for all terminals other than plotters (DT=PL).

CN PARAMETER

Use the CN parameter to specify the hexadecimal value of the cancel-input-line character for this terminal. When you enter this character, the preceding logical input line is discarded by the TIP.

CT PARAMETER

Use the CT parameter to specify the hexadecimal value of the control character defined for this terminal. When the terminal user enters this character, the TIP interprets the rest of the current logical line of input as a parameterization command. This parameter is optional for interactive terminals (DT=CON); it is not legal for passive terminals.

DO PARAMETER

Use the DO parameter to assign the device ordinal ($1 \leq do \leq 7$) of this terminal within its device type. This parameter is legal only for passive terminals. Within a multileaving workstation, the ordinal of each enabled terminal must be unique within its device type and must be the number of the physical input or output stream for the terminal (unless the PDO parameter is specified). This parameter is required except if there is only one device of each device type within the workstation; in this case the default value for DO is 1.

PDO PARAMETER

Use the PDO parameter to assign the pseudo device ordinal ($1 \leq pdo \leq 7$), or physical input or output stream number, of this terminal within its device type. This parameter is legal for passive terminals only. The stream number for each terminal in a workstation must be unique within its device type, whether specified with the DO parameter or the PDO parameter. For a plotter, this parameter is required and must not have the same value as the PDO parameter for a card punch within the same workstation. For all other passive devices this parameter is optional; the default value is the DO value for this terminal.

ABL PARAMETER

Use the ABL parameter to specify the application block limit, or the total number ($1 \leq abl \leq 7$) of downline blocks that can be outstanding (unacknowledged) between an application program and the interactive terminal. This parameter is optional for interactive terminals; if you specify ABL for a batch terminal, it is ignored by the NDL Processor.

The value that you choose should be sufficient for the application program to keep the terminal busy by maintaining ABL outstanding blocks of BSZ characters. The value declared for ABL should be greater than or equal to the network block limit (NBL) of the terminal. Guidelines for selecting ABL and NBL parameter values are given in the Buffering of Data subsection. Recommended values for this parameter are given in table 4-10. The default value for this parameter is 2.

BSZ PARAMETER

Use the BSZ parameter to specify the block size, or number of characters that form a block to or from an interactive terminal. The value that you choose is a function of device type, line speed, and terminal class, as shown in table 4-10. You should choose the value for BSZ so that ABL outstanding blocks of BSZ characters constitute a queue sufficient to keep the terminal continuously busy. The legal BSZ values for interactive devices are

$256 \leq bsz \leq 2043$

For passive terminals, the parameter specifies the size of blocks transferred between the host and an NPU. It is recommended that you assign one of the following values to BSZ.

640

1280

1920

TABLE 4-10. BLOCK LIMITS AND SIZES - HASP TIP

Line Speed	Terminal Class	ABL	NBL		XBZ		BSZ	
			Recommended	Default	Recommended	Default	Recommended	Default
2400	HASP/HPRE (DT=CON)	2	2	2	400 800	400	400	400
	HASP/HPRE (DT=CR)		7	7	400 800	400	1280	1280
	HASP/HPRE (DT=LP,CP,PL)		1	1	400 800	400	1280	1280
4800	HASP/HPRE (DT=CON)	2	2	2	400 800	400	400	400
	HASP/HPRE (DT=CR)		7	7	400 800	400	1280	1280
	HASP/HPRE (DT=LP,CP,PL)		1	1	400 800	400	1280	1280
9600	HASP/HPRE (DT=CON)	2	2	2	400 800	400	400	400
	HASP/HPRE (DT=CR)		7	7	400 800	400	1280	1280
	HASP/HPRE (DT=LP,CP,PL)		1	1	400 800	400	1920	1280
19.2K	HASP/HPRE (DT=CON)	2	2	2	800	400	800	400
	HASP/HPRE (DT=CR)		7	7	800	400	1280	1280
	HASP/HPRE (DT=LP,CP,PL)		2	1	800	400	1920	1280

If you do not assign one of the recommended values, but it is within the range 1≤bsz≤2043, the NDL Processor will adjust the value as follows.

Value Specified	Adjusted Value
1≤bsz≤640	640
641≤bsz≤1280	1280
1281≤bsz≤2043	1920

The NDL Processor issues an informative message if it adjusts the BSZ value that you specify.

DI PARAMETER

Use the DI parameter to specify that the terminal is disabled at network initiation. If you specify this optional reserved word parameter, the terminal cannot access the network until the local operator enables it. If you omit DI the terminal is assumed to be enabled.

You should not declare this parameter for terminals on AUTO lines unless you declare the STIP and TC parameters.

If the owning console of a passive terminal is initially disabled, you must specify DI for that passive terminal.

SDT PARAMETER

Use the SDT parameter to specify the sub-device type of the terminal. This parameter is optional. For printers (DT=LP), SDT can have the following values.

A6	ASCII 64-character train type
A9	ASCII 96-character train type
B6	BCD 64-character train type

For card readers (DT=CR), SDT represents the card reader default mode. The following values are legal.

29	029 mode
26	026 mode

For plotters (DT=PL), SDT indicates the plotter type. The following values are legal.

6BIT	6-bit code
8BIT	8-bit code

CO PARAMETER

Use the CO parameter to specify the configuration ordinal of the terminal within the network. This parameter is optional for terminals on AUTO lines; it is not legal for any other terminals. If you specify CO for one terminal on the line, you must specify it for all terminals on the line. You can assign CO a value within the range 1≤CO≤255.

XBZ PARAMETER

Use the XBZ parameter to specify the number of characters in a transmission block for the terminal. You can assign a value to XBZ within the range 200≤xbz≤2047. This parameter is optional; the default value is 400.

The TIP reformats downline data if necessary, so as not to exceed XBZ for transmission to the terminal; thus, XBZ should not exceed the actual buffer size of the terminal.

PARAMETERIZATION RESTRICTIONS

Legal combinations of parameters for the LINE statement and TERMINAL statement variant for the HASP protocol are shown in table 4-11. Further limitations on the parameterization of terminals serviced by the HASP TIP are described below.

UNIQUE STREAM NUMBER REQUIREMENT

For an application program such as the Remote Batch Facility, the physical input or output stream number for each passive device associated with an owning console must be unique within the device type of that passive device. You specify this stream number with either the DO parameter or the PDO parameter. The stream number that is actually used must be unique for each device within its device type, regardless of whether the value used is the DO or the PDO parameter value. The DO and PDO parameters are the only parameters affected by this uniqueness requirement.

AUTOMATIC RECOGNITION

To effect automatic recognition of terminal characteristics, those characteristics must be input as parameters on a punched card. This card must contain the characters

/*CONFIG

in the first eight columns. This card must occur in the SIGNON block or be the first card received after the AUTO line is enabled.

For the HASP protocol, the STIP, CSET, CO, DT, and PDO parameters can be automatically recognized.

BSC PROTOCOL

The BSC protocol provides support for synchronous terminals that use the IBM Binary Synchronous Communications protocol, such as the IBM 2780 and 3780. The format of the TERMINAL statement variant for configuring such terminals is shown in figure 4-6.

For AUTO Lines†

```
terminal:TERMINAL   [Δ STIP=stip, TA=ta, DT=dt, TC=tc,
      PRI, NBL=nbl, APPL=service, ALFAM=alfam,
      ALUN=alun, OCTERM=octerm, PW=pw††, CT=ct††,
      ABL=abl, BSZ=bsz, DI, RIC=ric, BCF=bcf, MREC=mrec,
      SDT=sdt, CO=co, XBZ=xbz].
```

For non-AUTO Lines†

```
terminal:TERMINAL    Δ [STIP=stip, TA=ta, DT=dt,]
      TC=tc, [PRI, NBL=nbl, APPL=service, ALFAM=alfam,
      ALUN=alun, OCTERM=octerm, PW=pw††, CT=ct††,
      ABL=abl, BSZ=bsz, DI, RIC=ric, BCF=bcf, MREC=mrec,
      SDT=sdt, CO=co, XBZ=xbz].
```

†The hexadecimal value to declare for a given character can be determined using table A-9.

††Ranges and/or defaults for this parameter depend on the TC value used for the terminal. See table 4-12.

Figure 4-6. TERMINAL Statement Format for BSC TIP

TABLE 4-11. LEGAL COMBINATIONS OF LINE AND TERMINAL STATEMENT PARAMETERS FOR HASP TIP (LTYPE S1, S2, S3)

AUTO Declared or Used	LSPEED Declared or Used	STIP Declared or Used	TC Declared or Used	CSET Declared or Used
AUTO	None	POST, None	HASP, None	None†
AUTO	None	PRE, None	HPRE, None	None†
None	None	POST, None	HASP	None†
None	None	PRE, None	HPRE	None†

†Default is EBCDIC.

NOTE

To use the order-dependent value declaration convention, you must supply TERMINAL statement parameters in the order shown in appendix E.

TERMINAL-NAME PARAMETER

Use the terminal-name parameter to assign a name to the terminal for reference within the NDL program and by the local operator. This name should be unique within the current local division. This name is required.

STIP PARAMETER

Use the STIP parameter to specify the sub-TIP type of the terminal. This parameter is optional. If you specify this parameter for one terminal on an AUTO line, you must specify it for all terminals on the line; if the parameter is supplied, the network software performs an additional match against the local configuration file while identifying the terminal. For non-AUTO lines, the default value used for this parameter depends on the TC value used for the terminal. Legal reserved word values for this parameter are

2780	An IBM 2780 or terminal that emulates it
3780	An IBM 3780 or terminal that emulates it

TA PARAMETER

Use the TA parameter to specify the component selection character for a 3780 card punch. This parameter is optional for lines configured for automatic recognition of terminals; if you supply this parameter for one terminal on the line, you must supply it for all terminals on the line. If the line is not configured for automatic recognition, this parameter is required for every 3780 card punch.

DT PARAMETER

Use the DT parameter to specify the device type of the terminal. The following reserved words are valid for this parameter.

<u>CON</u>	Identifies a keyboard or CRT console terminal that emulates the IBM 2780 or 3780 protocol, or a virtual console consisting of a card reader and a line printer
CR	Identifies a card reader terminal
LP	Identifies a line printer terminal
CP	Identifies a card punch terminal
USER	Identifies a site-defined device of type 7

Regardless of whether a console device exists on the line, you must provide a TERMINAL statement defining a logical terminal with DT=CON to establish the owning console relationship for any 2780 or 3780 terminal on the line. The fact that a console does not exist should be communicated to RBF via the RIC parameter.

TC PARAMETER

Use the TC parameter to specify the terminal class of the terminal. The value that you declare for this parameter determines the parameterization ranges and defaults used for the terminal (see table 4-12). For AUTO lines this parameter is optional; if you specify TC for one terminal on a line, you must specify it for all terminals on the line. For non-AUTO lines this parameter is required. Two values are legal for TC:

2780	The terminal is an IBM 2780 or a terminal that emulates it.
3780	The terminal is an IBM 3780 or a terminal that emulates it.

TABLE 4-12. PARAMETERIZATION RANGES AND DEFAULTS BY TERMINAL CLASS - BSC TIP

Parameter	Terminal Class			
	2780		3780	
	Range	Default	Range	Default
CT	0 thru $7F_{16}$	25_{16} %	0 thru $7F_{16}$	25_{16} %
PW (DT=CON)	0 thru 255	80	0 thru 255	80
PW (DT=LP)	50 thru 150	80	50 thru 150	120

LEGEND:

Range	Hexadecimal values of 0 through $7F_{16}$ correspond to characters in the 128-character ASCII set of table A-9. The value $3D_{16}$ is not legal.
Default	Character representations beneath default values are the keyboard keys that cause entry of the default value for the archetype terminal of the terminal class.

PRI PARAMETER

Use the PRI parameter to specify that the terminal is to have data traffic priority. This optional reserved word indicates that data to or from the terminal is to travel through the network with a higher priority than data to or from terminals for which this priority has not been specified. Data to or from terminals without priority is suspended to relieve network traffic congestion before data to or from terminals with priority is suspended. PRI is usually specified for terminals with DT=CON when interactive traffic is to receive priority over batch traffic.

NBL PARAMETER

Use the NBL parameter to specify the allowable block limit in the network for a console device, or the number of downline blocks that can be buffered between the host computer and the terminal. This optional parameter has the range $1 \leq nbl \leq 7$; the default value is 2. The value specified is used in conjunction with the ABL parameter to determine the block queuing required of the Network Access Method and of CCP.

For passive devices, this parameter specifies the number of BSZ blocks allowed in the network. The relationship between the NBL parameter and the BSZ and ABL parameters is explained in the Terminal Buffering subsection (later in this section) and shown in table 4-13.

IAPPL PARAMETER

Use the IAPPL parameter to declare an initial application program for this terminal. This name must be one specified on an APPL statement in the current local division of the NDL program. This parameter is optional for terminals with DT=CON and required for terminals of all other device types. If you declare this parameter, the terminal is connected directly to the specified application upon completion of family name and user name validation; the terminal is not prompted by NVF for an application name. If you omit the parameter, an interactive (DT=CON) terminal is prompted by NVF; there is no default value.

ALFAM PARAMETER

Use the ALFAM parameter to specify the NOS family name to be used during automatic login of this terminal. This parameter is optional for interactive terminals (DT=CON); it is required for passive terminals. If you declare this parameter, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a family name during the initial login attempt, but can be prompted during subsequent login attempts, even though ALFAM is specified. You should choose the value for this parameter with consideration of the UID parameter specification for all application programs this terminal can access (see the APPL Statement description).

If you specify ALFAM=0, the system default family name will be used for automatic login. To determine other valid values, see the System Maintenance Reference Manual.

ALUN PARAMETER

Use the ALUN parameter to declare a NOS user name to use during automatic login of this terminal. This parameter is optional for interactive terminals, but required for passive terminals. If you declare this parameter for a terminal, its value is passed to NVF as soon as the terminal becomes active; the terminal is not prompted for manual entry of a user name and a password during the initial login attempt. Interactive terminals can be prompted for manual entry of the user name and password during subsequent login attempts. The value that you declare for this parameter can be any valid user name from the NOS VALIDUz file (see the NOS System Maintenance Reference Manual).

If you configure the terminal with TAF as its initial application program (IAPPL=TAF), the value used for the ALUN parameter must be the same as that used to identify the terminal in the NOS terminal file NCTFid. NCTFid is described in detail in the System Maintenance Reference Manual.

OCTERM PARAMETER

Use the OCTERM parameter to specify the owning console for this terminal. This parameter is required for passive terminals; it is not legal for interactive terminals (DT=CON). The value declared for this parameter must be the terminal name from the TERMINAL statement for a console device or virtual console device on the same communication line. If you specify OCTERM and DI for a passive terminal and the passive terminal is enabled, the owning console must be enabled.

Note that if no console device exists on the line, you must provide a TERMINAL statement to declare a logical interactive device to which the OCTERM parameter can refer (see Device Type Parameter).

PW PARAMETER

Use the PW parameter to specify the number $(0 \leq pw \leq 255)$ of characters per physical line of input or output for this terminal. This optional parameter is legal for interactive terminals (DT=CON) and line printers (DT=LP) only.

CT PARAMETER

Use the CT parameter to specify the hexadecimal value of the control character for this terminal. When the terminal user enters this character, the TIP interprets the rest of the current logical line of input as a parameterization command. This parameter is optional for interactive terminals; it is not legal for passive terminals.

DI PARAMETER

Use the DI parameter to specify that the terminal is disabled at network initiation. If you specify this optional parameter, the terminal cannot access the network until the local operator enables it. If you omit DI the terminal is assumed to be enabled.

You should not declare this parameter for terminals on AUTO lines unless you also declare the STIP, CO, and DT parameters. (For a 3780 card punch only, you must also declare TA.)

If the owning console of a passive terminal is initially disabled, you must specify DI for that passive terminal.

TABLE 4-13. BLOCK LIMITS AND SIZES - BSC PROTOCOL

Line Speed	Terminal Class	ABL	NBL		XBZ		BSZ	
			Recommended	Default	Recommended	Default	Recommended	Default
2400	2780 (DT=CON)	2	2	2	400	400	400	400
	2780 (DT=CR)		7	7	400	400	1280	1280
	2780 (DT=LP,CP)		1	1	400	400	1280	1280
	3780 (DT=CON)	2	2	2	512	512	512	512
	3780 (DT=CR)		7	7	512	512	1280	1280
	3780 (DT=LP,CP)		1	1	512	512	1280	1280
4800	2780 (DT=CON)	2	2	2	400	400	400	400
	2780 (DT=CR)		7	7	400	400	1280	1280
	2780 (DT=LP,CP)		1	1	400	400	1280	1280
	3780 (DT=CON)	2	2	2	512	512	512	512
	3780 (DT=CR)		7	7	512	512	1280	1280
	3780 (DT=LP,CP)		1	1	512	512	1280	1280
9600	2780 (DT=CON)	2	2	2	400	400	400	400
	2780 (DT=CR)		7	7	400	400	1280	1280
	2780 (DT=LP,CP)		1	1	400	400	1920	1280
	3780 (DT=CON)	2	2	2	512	512	512	512
	3780 (DT=CR)		7	7	512	512	1280	1280
	3780 (DT=LP,CP)		1	1	512	512	1920	1280
19.2K	2780 (DT=CON)	2	2	2	400	400	400	400
	2780 (DT=CR)		7	7	400	400	1280	1280
	2780 (DT=LP,CP)		2	1	400	400	1920	1280
	3780 (DT=CON)	2	2	2	512	512	512	512
	3780 (DT=CR)		7	7	512	512	1280	1280
	3780 (DT=LP,CP)		2	1	512	512	1920	1280

RIC PARAMETER

Use the RIC parameter to specify whether the terminal has restricted interactive capability. This parameter is optional for interactive terminals (DT=CON); it is not legal for any other terminals. The following values are legal for this parameter.

YES This terminal has restricted interactive capabilities. Application programs such as RBF will not output the READY prompt nor require a GO command to enable passive devices.

NO This terminal has normal interactive capabilities.

BCF PARAMETER

Use the BCF parameter to specify whether the terminal is to have the blank compression feature. This parameter is optional for interactive terminals with STIP = 2780; it is not legal for any other terminals. If you specify a value for this parameter in the TERMINAL statement for an owning console, the NDL processor will use that value for all passive terminals owned by the console. The following reserved words are legal for BCF.

YES Blank compression is selected.

NO Blank compression is not selected.

If you select blank compression for a terminal, blank compression is selected for all data to or from all devices on the line.

MREC PARAMETER

Use the MREC parameter to specify the maximum number ($1 \leq mrec \leq 7$) of records per transmission block that the terminal can transmit or receive. This parameter is optional for interactive devices with STIP = 2780; it is not legal for any other terminals. If you specify this parameter for an owning console, the value specified is also used for all passive devices owned by that console.

SDT PARAMETER

Use the SDT parameter to specify the sub-device type of the terminal. This parameter is optional. For printers (DT=LP), SDT can have the following values.

A6	ASCII 64-character train type
A9	ASCII 96-character train type
B6	BCD 64-character train type

For card readers (DT=CR), SDT represents the card reader default mode. The following values are legal.

29	029 mode
26	026 mode

CO PARAMETER

The CO parameter specifies the configuration ordinal of the terminal within the network. This parameter is optional for terminals on AUTO lines; it is not legal for any other terminals. If you specify CO for one terminal on the line, you must specify it for all terminals on the line. You can assign CO a value within the range $1 \leq CO \leq 255$.

XBZ PARAMETER

Use the XBZ parameter to specify the number ($200 \leq xbz \leq 2047$) of characters in a transmission block for the terminal. This parameter is optional. The default for XBZ depends on the TC value declared for the terminal, as follows.

400	Default for 2780 terminals
512	Default for 3780 terminals

The TIP reformats downline data if necessary, so as not to exceed XBZ for transmission to the terminal; thus, this parameter should not exceed the actual buffer size of the terminal.

PARAMETERIZATION RESTRICTIONS

Table 4-14 shows legal combinations of parameters for the LINE statement and the TERMINAL statement variant used for the BSC protocol.

Further limitations on your ability to parameterize characteristics of terminals using the BSC protocol are described below.

TABLE 4-14. LEGAL COMBINATIONS OF LINE AND TERMINAL STATEMENT PARAMETERS FOR BSC TIP (LTYPE S1, S2, S3)

AUTO Declared or Used	LSPEED Declared or Used	STIP Declared or Used	TC Declared or Used	CSET Declared or Used
AUTO	None	2780, None	2780, None	None†
AUTO	None	3780, None	3780, None	None†
None	None	2780, None	2780	None†
None	None	3780, None	3780	None†

†Default is EBCDIC.

Interactive Capabilities

The archetype terminals supported by the BSC protocol are passive devices; they do not have an interactive console. For this reason, no interactive operator communication is required for terminal operation. However, a method is provided for the terminal operator to use a card reader and a line printer to emulate an interactive terminal with limited capabilities. This is achieved by using the RIC parameter in addition to the DT parameter, so that application programs such as RBF do not output a READY prompt nor require a GO command to enable passive devices.

Automatic Recognition

Automatic recognition of terminals is supported by the BSC protocol to allow terminals of different types to use the same line. However, because no interactive console exists, the terminal user must provide a configuration ordinal parameter on the first card in the card reader. This first card must contain /*CONFIG in the first eight columns. The Communication Supervisor uses the configuration ordinal to determine which terminal definition in the local configuration file is used to configure the terminal.

ADDITIONAL PARAMETERIZATION INFORMATION

Tables 4-15 and 4-16 are intended as guides to the configuration of terminals currently supported by various operations of Control Data Corporation; no implication of nonsupport of other compatible terminals is intended. The parameters listed in the tables are valid for the archetype terminals given. Some of the parameters could be invalid for terminals emulating these archetype terminals. You should use the parameter descriptions in the appropriate TERMINAL statement variant in conjunction with these two tables.

Table 4-17 shows the terminal class values you can specify, and the restrictions on device types that you can declare for those values.

BUFFERING OF DATA

Using block limit parameters, you can tailor your configuration of a terminal to keep it efficiently occupied. The value you should choose for each block limit depends on factors such as

- The number of application programs to be serviced
- The number of terminals transmitting and receiving data
- The frequency of terminal usage
- The size of the NPU macromemory available for buffering
- The roll-in/roll-out frequency desired for each application program

Buffering for Interactive Terminals

For each interactive terminal, you can declare the number of characters constituting a block of data , as well as the total number of downline (output) blocks the network is permitted to queue between the terminal and an application program (the application block limit, or ABL parameter). The ABL parameter value that you should choose for the terminal is dependent on the size of the CCP buffering available (the terminal's network block limit, or NBL parameter) for the terminal, and on a number of other factors in the network configuration.

CCP buffers NBL output (downline) blocks between a terminal and an application program. NAM buffers ABL minus NBL blocks. Together, the terminal block limit parameters allow you to empirically tune the network's throughput rates, NAM's central memory requirements, and system performance/resource usage.

For example, selection of a large ABL parameter implies less frequent roll-in/roll-out of an application program because each application program can send more data message blocks before its block limit is reached; a large value for ABL minus NBL also implies that a large field length for NAM is permissible, because the additional blocks are queued there. A small NPU macromemory size requires that you specify a small value for the NBL parameter to avoid swamping the available NPU buffer area and for the ABL parameter to avoid forcing a large NAM field length for queuing.

A large number of application programs need a small ABL parameter to facilitate frequent roll-in/roll-out and reduce NAM's central memory use; this condition also requires selection of a large NBL parameter (implying that the NPU servicing the terminal has a large macromemory) to prevent swamping NAM's buffers and increasing the percentage of memory resources NAM requires.

In general, you should assign a value to the ABL parameter that allows an application program to assume that ABL blocks of the indicated number of characters can keep the terminal continuously busy. The default values for the block limit parameters provide adequate tuning for the servicing of terminals by the Interactive Facility via NPUs with the default macromemory size.

The recommended values and the defaults used for the block size and limit parameters are shown in the Block Sizes and Limits table in the section for the appropriate TIP.

For interactive traffic, the following assumptions are made to derive the values in the table.

- Interactive traffic is from IAF. User written applications may require different parameter settings.
- Maximum line utilization of asynchronous interactive terminals is 90 percent, which is achieved with high volume applications such as graphics.
- IAF formats data into blocks of a maximum size less than or equal to BSZ, which contains an integral number of lines. This holds true if the user program is generating data in sufficient volume and is subject to IAF's maximum block size of 945 characters.

TABLE 4-15. TERMINALS ON ASYNCHRONOUS LINES

Valid TC	Valid DT	Optional Parameterization Declarations	Terminal Model	Manufacturer
M33	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	33, 35, 37, 38 ASR/KSR†	Teletype Corporation
M33	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	1030 Teleterm	Computer Devices
M33	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	Silent 700 Series	Texas Instruments
M33	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	Terminet 300	General Electric Corporation
M33	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	DCT 500	Univac Corporation
M33	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	Execuport 300	Computer Transceiver Systems
M33	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	AJ630	Anderson-Jacobson
M33	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	Sycor 340	Sycor and Sorbus, Inc.
M33	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	Datapoint 3300	Texas Computer Terminal Corporation
M40	CON	AL,B1,B2,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	40-2†	Teletype Corporation
T4014	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	40xx Series CRT†	Tektronix
H2000	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	2000 VDT†	Hazeltine
713	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	713-10†	Control Data Corporation
751	CON	AL,B1,B2,BS,CI,CN,CT,DLC,DLTO,DLX,EP,IN,LI,OP,PA,PG,PL,PW,SE	751-1	Control Data Corporation
2741	CON	AL,B1,B2,BS,CN,CT,DLC,DLTO,DLX,IN,LI,OP,PA,PG,PL,PW,SE	2741†	IBM Corporation
2741	CON	AL,B1,B2,BS,CN,CT,DLC,DLTO,DLX,IN,LI,OP,PA,PG,PL,PW,SE	Datel 30	Universal Computer Corporation
2741	CON	AL,B1,B2,BS,CN,CT,DLC,DLTO,DLX,IN,LI,OP,PA,PG,PL,PW,SE	Novar 5-41	General Telephone and Electronics

†Archetype terminals

TABLE 4-16. TERMINALS ON SYNCHRONOUS LINES

Valid TC	Valid DT	Valid Hardware Address Parameters	Optional Parameterization Declarations	Terminal Model	Manufacturer
714X	CON	CA,TA	B1,B2,CN,CT,PG,PL,PW	714-30†	Control Data Corporation
714X	LP	CA,TA,DO	PW,SDT	714-30†	Control Data Corporation
711	CON	CA,TA	B1,B2,CN,CT,PG,PL,PW	711-10†	Control Data Corporation

TABLE 4-16. TERMINALS ON SYNCHRONOUS LINES (Contd)

Valid TC	Valid DT	Valid Hardware Address Parameters	Optional Parameterization Declarations	Terminal Model	Manufacturer
714	CON	CA,TA	B1,B2,CN,CT,PG,PL,PW	714-10/20†	Control Data Corporation
714	LP	CA,TA,DO	PW,SDT	714-10/20	Control Data Corporation
731	CON	CA,TA	B1,B2,CN,CT,PG,PL,PW	731-12†	Control Data Corporation
731	CON	CA,TA	B1,B2,CN,CT,PG,PL,PW	732-12	Control Data Corporation
734	CON	CA,TA	B1,B2,CN,CT,PG,PL,PW	734-1†	Control Data Corporation
200UT	CR	CA,TA	PW	734-101/151	Control Data Corporation
200UT	LP	CA,TA	PW,SDT	734-201/251	Control Data Corporation
200UT	CON	CA,TA	B1,B2,CN,CT,PG,PL,PW	217†, 214	Control Data Corporation
200UT	CR	CA,TA	PW	224†	Control Data Corporation
200UT	LP	CA,TA	PW,SDT	222†	Control Data Corporation
200UT	CON	CA,TA	B1,B2,CN,CT,PG,PL,PW	CYBER 18 (COMM-18)	Control Data Corporation
200UT	CR	CA,TA	PW	CYBER 18 (COMM-18)	Control Data Corporation
200UT	LP	CA,TA	PW	CYBER 18 (COMM-18)	Control Data Corporation
HASP	CON	None	B1,B2,CN,CT,PW	CYBER 18 (COMM-18)	Control Data Corporation
HASP	CR	DO	SDT	CYBER 18 (COMM-18)	Control Data Corporation
HASP	LP	DO	PW,SDT	CYBER 18 (COMM-18)	Control Data Corporation
HASP	CP	DO	None	CYBER 18 (COMM-18)	Control Data Corporation
HASP	PL	PDO	None,SDT	CYBER 18 (COMM-18)	Control Data Corporation
HPRE	CON	None	B1,B2,CN,CT,PW	CYBER 18 (COMM-18)	Control Data Corporation
HPRE	CR	DO	None	CYBER 18 (COMM-18)	Control Data Corporation
HPRE	LP	DO	PW	CYBER 18 (COMM-18)	Control Data Corporation
HPRE	CP	DO	None	CYBER 18 (COMM-18)	Control Data Corporation
HPRE	PL	PDO	None	CYBER 18 (COMM-18)	Control Data Corporation
2780	CON	None	PW,CT,RIC,BCF,MREC	2780	IBM Corporation
2780	CR	None	SDT	2780	IBM Corporation
2780	LP	None	PW,SDT	2780	IBM Corporation
2780	CP	None	None	2780	IBM Corporation
3780	CON	None	PW,CT	3780	IBM Corporation
3780	CR	None	SDT	3780	IBM Corporation
3780	LP	None	PW,SDT	3780	IBM Corporation
3780	CP	TA	None	3780	IBM Corporation

†Archetype terminals.

TABLE 4-17. LEGAL TERMINAL CLASS VALUES

TIPTYPE Value Declared	STIP Value Declared	Terminal Class Value Used	Associated NAM Terminal Class Number[†]	Legal Device Type Values Declared or Used	Parameter Diagnostic Checking Performed
ASYNC	N2741, A110, A150, A300	M33	1	CON or USER	Not for USER-dependent ones
	N2741, A110, A150, A300	713[††]	2	CON or USER	Not for USER-dependent ones
	2741, 2741E, 2741C	2741	4	CON or USER	Not for USER-dependent ones
	N2741, A110, A150, A300	M40[††]	5	CON or USER	Not for USER-dependent ones
	N2741, A110, A150, A300	H2000[††]	6	CON or USER	Not for USER-dependent ones
	N2741, A110, A150, A300	751[††]	7	CON or USER	Not for USER-dependent ones
	N2741, A110, A150, A300	T4014[††]	8	CON or USER	Not for USER-dependent ones
	None	USER1	28	CON, CR, CP, LP, PL, or USER	Not for USER- or USER1-dependent ones
	None	USER2	29	CON, CR, CP, LP, PL, or USER	Not for USER- or USER2-dependent ones
	None	USER3	30	CON, CR, CP, LP, PL, or USER	Not for USER- or USER3-dependent ones
	None	USER4	31	CON, CR, CP, LP, PL, or USER	Not for USER- or USER4-dependent ones
X25	None	M33	1	CON or USER	Not for USER-dependent ones
	None	713	2	CON or USER	Not for USER-dependent ones
	None	M40	5	CON or USER	Not for USER-dependent ones
	None	H2000	6	CON or USER	Not for USER-dependent ones
	None	751	7	CON or USER	Not for USER-dependent ones
	None	T4014	8	CON or USER	Not for USER-dependent ones
	None	USER1	28	CON or USER	Not for USER- or USER1-dependent ones
	None	USER2	29	CON or USER	Not for USER- or USER2-dependent ones
	None	USER3	30	CON or USER	Not for USER- or USER3-dependent ones
	None	USER4	31	CON or USER	Not for USER- or USER4-dependent ones
MODE4	M4A, M4BCD, M4ASCII	200UT	10	CON, CR, LP, or USER	Not for USER-dependent ones
	M4C	714X[§]	11	CON, LP, or USER	Not for USER-dependent ones
	M4A, M4BCD, M4ASCII	731[†††]	15	CON or USER (associated LP or CR terminals are TC=200UT)	Not for USER-dependent ones

TABLE 4-17. LEGAL TERMINAL CLASS VALUES (Contd)

TIPTYPE Value Declared	STIP Value Declared	Terminal Class Value Used	Associated NAM Terminal Class Number	Legal Device Type Values Declared or Used	Parameter Diagnostic Checking Performed
	M4A, M4BCD, M4ASCII	734[†††]	15	CON or USER (associated LP or CR terminals are TC=200UT)	Not for USER-dependent ones
	M4C	714	13	CON, LP, or USER	Not for USER-dependent ones
	M4C	711[§]	12	CON or USER	Not for USER-dependent ones
	None	USER1	28	CON, CR, CP, LP, PL, or USER	Not for USER- or USER1-dependent ones
	None	USER2	29	CON, CR, CP, LP, PL, or USER	Not for USER- or USER2-dependent ones
	None	USER3	30	CON, CR, CP, LP, PL, or USER	Not for USER- or USER3-dependent ones
	None	USER4	31	CON, CR, CP, LP, PL, or USER	Not for USER- or USER4-dependent ones
HASP	POST	HASP	9	CON, CR, CP, LP, PL, or USER	Not for USER-dependent ones
	PRE	HPRE	14	CON, CR, CP, LP, PL, or USER	Not for USER-dependent ones
	None	USER1	28	CON, CR, CP, LP, PL, or USER	Not for USER- or USER1-dependent ones
	None	USER2	29	CON, CR, CP, LP, PL, or USER	Not for USER- or USER2-dependent ones
	None	USER3	30	CON, CR, CP, LP, PL, or USER	Not for USER- or USER3-dependent ones
	None	USER4	31	CON, CR, CP, LP, PL, or USER	Not for USER- or USER4-dependent ones
BSC	2780	2780	16	CON, CR, CP, LP, or USER	Not for USER-dependent ones
	3780	3780	17	CON, CR, CP, LP, or USER	Not for USER-dependent ones
	None	USER1	28	CON, CR, CP, LP, PL, or USER	Not for USER- or USER1-dependent ones
	None	USER2	29	CON, CR, CP, LP, PL, or USER	Not for USER- or USER2-dependent ones
	None	USER3	30	CON, CR, CP, LP, PL, or USER	Not for USER- or USER3-dependent ones
	None	USER4	31	CON, CR, CP, LP, PL, or USER	Not for USER- or USER4-dependent ones

TABLE 4-17. LEGAL TERMINAL CLASS VALUES (Contd)

TIPTYPE Value Declared	STIP Value Declared	Terminal Class Value Used	Associated NAM Terminal Class Number	Legal Device Type Values Declared or Used	Parameter Diagnostic Checking Performed
USER1, USER2, USER3, or USER4	None	M33	1	CON or USER	Not for USER-dependent ones
	None	713[††]	2	CON or USER	Not for USER-dependent ones
	None	2741	4	CON or USER	Not for USER-dependent ones
	None	M40[††]	5	CON or USER	Not for USER-dependent ones
	None	H2000[††]	6	CON or USER	Not for USER-dependent ones
	None	751[††]	7	CON or USER	Not for USER-dependent ones
	None	T4014[††]	8	CON or USER	Not for USER-dependent ones
	None	HASP	9	CON, CR, CP, LP, PL, or USER	Not for USER-dependent ones
	None	200UT	10	CON, CR, LP, or USER	Not for USER-dependent ones
	None	714X[§]	11	CON, LP, or USER	Not for USER-dependent ones
	None	711[§]	12	CON or USER	Not for USER-dependent ones
	None	714	13	CON, LP, or USER	Not for USER-dependent ones
	None	731[†††]	15	CON or USER (associated LP or CR terminals are TC=200UT)	Not for USER-dependent ones
	None	734[†††]	15	CON or USER (associated LP or CR terminals are TC=200UT)	Not for USER-dependent ones
	None	USER1	28	CON, CR, CP, LP, PL, or USER	Not for USER- or USER1-dependent ones
	None	USER2	29	CON, CR, CP, LP, PL, or USER	Not for USER- or USER2-dependent ones
	None	USER3	30	CON, CR, CP, LP, PL, or USER	Not for USER- or USER3-dependent ones
	None	USER4	31	CON, CR, CP, LP, PL, or USER	Not for USER- or USER4-dependent ones

[†] Not used in any NDL program statement.

[††] A default of M33 is used on lines configured for automatic recognition. The terminal user can enter the correct TC value from the terminal.

[†††] A default of 200UT is used on lines configured for automatic recognition. The terminal user can enter the correct TC value from the terminal.

[§] A default of 714 is used on lines configured for automatic recognition. The terminal user can enter the correct TC value from the terminal.

- Buffering can be optimized in IAF by setting BSZ to a value of 45 + 60 n, where n is in the range 3 to 15.

- IAF can supply no more than one block for each terminal in each t second period. IAF computes the value of t using the minimum value of 2/(ABL-1) for all terminals connected to IAF. Disk utilization in IAF is optimized if t is set to 2 seconds and the value of BSZ is changed. If the BSZ is changed to a value exceeding the maximum block size of IAF or the maximum size dictated by the terminal protocol (XBZ), disk utilization is not optimized.

- Line utilization can be maintained by setting BSZ to queue t seconds of data in NAM and CCP for each terminal connected to IAF.

- For maximum throughput, synchronous terminals must be set to a BSZ value no greater than XBZ for the terminal protocol minus any control bytes added by CCP. BSZ for asynchronous terminals is arbitrary, depending on the buffering capabilities (if any) of the terminal.

- Mode 4 terminals must be set to an XBZ value no greater than the screen size of the terminal. The screen size for Mode 4A terminals is 80 by 13 and for Mode 4C terminals is 80 by 16. Some emulators restrict the value of XBZ to 1000 for batch devices, which must be used for batch traffic. CCP must add one control byte for Mode 4 terminals.

- HASP terminals are normally set to an XBZ value of 400. This value is used for HASP terminals with a line speed no greater than 9600 bps. For 19.2 Kbps HASP terminals, XBZ is increased to 800 to maintain enough queuing. CCP must add four control bytes for HASP terminals.

- The minimum value of both ABL and NBL is 2. The maximum value of both ABL and NBL is 7.

- Line utilization can be maintained by setting NBL to queue 1 second of data in CCP for each terminal.

If NPU buffer space limitations cause frequent terminal input regulation by CCP, you can alter the ABL and NBL values to partially correct the problem. If the problem persists or cannot adequately be resolved by altering ABL and NBL, the PRI parameter for the terminals serviced by the NPU involved can be used in the following manner. All interactive terminals (terminals with a DT parameter value of CON) can be configured with the PRI parameter. Any terminals transmitting large blocks (any terminals configured with large BSZ parameter values) and any terminals on higher speed lines transmitting small blocks should not be configured with the PRI parameter. This algorithm will tend to cause regulation of terminals making frequent large block transmissions (such as batch terminals) before regulation of terminals making small or infrequent transmissions. The latter kinds are assumed to be the normal operational case for interactive terminals.

Buffering for Batch Terminals

Data blocking and buffering functions for batch terminals are performed in the NPUs by a module known as the PRU interface. Data is moved between the host and the NPUs in a format that can be processed directly by the disk drivers; once a data transfer is initiated by RBF, the data is moved between the NPU and the host disks without further CP processing.

The host must send downline (output) data to the NPU at a rate that will allow the NPU to keep the terminal busy. You can control that rate by choosing appropriate values for the network block limit (NBL) parameter and the network block size (BSZ) parameter. To minimize host processing, BSZ values are multiples of the PRU size. You should choose values for BSZ and NBL that will ensure that data is available in the NPU when required, minimize the required buffer space in the NPU, and minimize the number of disk accesses required.

CCP reblocks downline data into blocks acceptable to the terminal, as determined by the size of the terminal buffer.

The ABL parameter is not applicable to passive terminals. If it is specified, it is ignored by the NDL Processor.

TERMINAL MAINTENANCE MECHANISMS

In addition to tuning network throughput and customizing network software support of terminal devices other than the archetypes, you can use the TERMINAL statement to provide maintenance mechanisms. Such mechanisms include facilitation of Terminal Verification Facility use.

The Terminal Verification Facility (TVF) is provided so that a terminal operator can test for proper functioning of a console device. If a terminal is capable of successfully conducting a manual login to TVF, it is not likely that the terminal operator needs to access TVF. Conversely, if the operator needs TVF for testing, it might not be possible for the terminal to manually login to TVF. To avoid this access bottleneck, you can use the automatic login feature. A completely generalized logical terminal (configured only by defaults on an automatic recognition line) can be configured on each NPU; this generalized terminal would be on a dial-up communication line and would be configured for full, automatic login to TVF as an initial application program. If lines of more than one TIP type are supported by the NPU, a separate generalized terminal line is needed for each type of Terminal Interface Program. Terminal operators requiring the services of TVF could then simply dial into the generalized line and begin testing. When this approach is used, it probably would be necessary for the terminal operator to change the generalized terminal's parameterization during the course of testing. Not all defaults used for a generalized terminal are appropriate for a given physical terminal.

FILE STRUCTURE AND CONTENT 5

This section describes the logical structure and content of the major NOS local files handled by the NDL Processor. The NDL Processor can create one or more additional scratch files during processing (local file names ZZZZZU1 through ZZZZZU5). These scratch files are not significant to you as an NDL programmer and are not described. Files not directly handled by the NDL Processor, such as the CCP load file and VALIDUz, are described in the NOS Version 1 Installation Handbook and the NOS Version 1 System Maintenance Reference Manual.

The NDL Processor handles four major types of local files:

- The input file
- Network configuration files
- Local configuration files
- The job listing file

The NDL Processor also always produces a dayfile on file OUTPUT, indicating the error processing performed during execution. If no errors were encountered, a message indicating this is placed in the dayfile (see appendix B).

INPUT FILE

All file creation jobs (described in section 6) require an input file for the NDL Processor. The input file contains all NDL statements required for the creation of the network definition files being generated (see section 1). The input file name is either INPUT or a valid file name that you supply on the NDLP statement. The input file is a sequential file of 80-character records, terminated by an end-of-record indicator. This file is processed using the operating system input/output macros.

NETWORK CONFIGURATION FILE

One network configuration file is created from each network division in the input file during the first pass of the NDL Processor. This file contains information from the following NDL statements relating to the physical and logical configuration of the network elements.

- NETNAME
- HOST
- COUPLER
- NPU
- TRUNK
- LOGLINK

The network configuration file is constructed by the NDL Processor. It is accessed by the Network Supervisor as a mass storage, direct access, random permanent file.

When the NDL Processor finishes execution, any network configuration file it created remains attached to the job as a NOS local file. Postprocessing disposition of this file is your responsibility, using operating system control statements. If one or more files with verification records were produced, you should make them direct access permanent files. These permanent files must have either file names known to the network operator or the default file names used in the procedure files that initiate network operation. Section 6 outlines the mechanism for naming the network configuration files; the operating system installation handbook describes the initiation procedure files.

When a network defined by a network configuration file is operating, the file is dedicated to that network. Another network configuration file cannot be substituted for the one being used, nor can the network configuration file in use be modified. The NDL processor can create a new network configuration file at any time regardless of whether or not the network is operating. Once a network configuration file exists, it can be used the next time the network is initialized.

Each network configuration file can contain records of five types:

- A network record
- A host record
- NPU records
- A verification record
- An index record

The verification record is the next-to-last record of a properly constructed file. It consists of four 60-bit words. The first word contains the letters ZNCF, left-justified in display code with blank fill. The second word contains the hour, minute, and second the file was created; this value is in display code and is obtained from the system clock when creation of the file ends. The third word contains the month, day, and year the file was created; this value is in display code and is also obtained from the system when creation of the file ends. The fourth word contains the code level of NDLP when the file was created; this value is in display code, three characters long, left justified, and obtained from a common deck called NAMLEV. The time and date from the verification record are written as part of the page headers whenever the file is listed. If the NDL Processor detects any fatal errors during creation of the file, it does not write the verification record.

When the Network Supervisor accesses an old network configuration file at network initiation, it attempts to read the verification record. If a verification record cannot be found, the Network Supervisor requests another network configuration file from the network operator. The Network Supervisor also compares its code level with that found in the network configuration file verification record. If the code levels are not the same, an informative message is sent to the network operator.

The NDL Processor attempts to read the verification record when it accesses an old network configuration file during a file inspection job (see section 6) or during creation of a local configuration file. If a verification record cannot be found, the NDL Processor issues a dayfile message (see appendix B). The Processor attempts to read as much of an unverified network configuration file as possible during creation of a local configuration file. The Processor does not read unverified files during file inspection jobs.

The formatted content of the other records is not significant to you. These records are described in the internal documentation of the network software.

LOCAL CONFIGURATION FILE

One local configuration file is created from each local division in the input file during the second pass of the NDL Processor. This file also contains information from either the immediately preceding network division or another specified network configuration file. It contains information from the following statements relating to the physical and logical configuration of the communication elements.

- HOST
- COUPLER
- APPL
- LINE
- TERMINAL
- NPU
- NPUDATA

The local configuration file is constructed by the NDL Processor. It is accessed by the Communications Supervisor as a mass storage, direct access, random permanent file.

When the NDL Processor finishes execution, any local configuration file it created remains attached to the job as a NOS local file. Postprocessing disposition of this file is the NDL programmer's responsibility, using operating system control statements. If one or more files with verification records were produced, you should make them direct access permanent files. These permanent files must have file names known to the local operator or the default file names used in the procedure files that initiate network operation. Section 6 outlines the mechanism for naming the local configuration files; the operating system installation handbook describes the initiation procedure files.

When a network defined by a local configuration file is operating, the file is dedicated to that network. Another local configuration file cannot be substituted for the one being used, nor can the local configuration file in use be modified. The NDL processor can create a new local configuration file at any time regardless of whether or not the network is operating. Once a local configuration file exists, it can be used the next time the network is initialized.

Each local configuration file can contain records of six types:

- A host access record
- An NPU directory record
- Communication line records
- Terminal records
- A verification record
- An index record

The verification record is the next-to-last record of a properly constructed file. It consists of four 60-bit words. The first word contains the letters ZLCF, left-justified in display code with blank fill. The second word contains the hour, minute, and second the file was created; this value is in display code and is obtained from the system clock when creation of the file ends. The third word contains the month, day, and year the file was created; this value is in display code and is also obtained from the system when creation of the file ends. The fourth word contains the code level of NDLP when the file was created; this value is in display code, three characters long, left-justified, and obtained from a common deck called NAMLEV. The time and date from the verification record are written as part of the page headers whenever the file is listed. If the NDL Processor detects any fatal errors during creation of the file, it does not write the verification record.

When the Communications Supervisor accesses an old local configuration file at network initiation, it attempts to read the verification record. If a verification record cannot be found, the Communications Supervisor requests another local configuration file from the local operator. The Communications Supervisor also compares its code level with that found in the local configuration file verification record. If the code levels are not the same, an informative message is sent to the local operator.

When the NDL Processor accesses an old local configuration file during a file inspection job, the Processor attempts to read the verification record. If a verification record cannot be found, the NDL Processor issues a dayfile message and aborts the job. (See section 6 and appendix B.)

The formatted content of the other records is not significant to you. These records are described in the internal documentation of the network software.

JOB LISTING FILE

The NDL Processor creates the job listing file during its last pass by reading the other local files created during a previous pass. The job listing file contains any or all of the listing options described in section 6 and illustrated in section 7 and appendix B. The job listing file name is either OUTPUT or a NOS local file name that you have supplied on the NDLP statement. If a job listing file name of zero is supplied, no job listing file is created.

This file is a sequential file of 137-character records, intended for output on a line printer. The file is created using the operating system input/output macros.

JOB STRUCTURE

This section describes the logical structure of jobs using the NDL Processor. The form such jobs have when they are input to the operating system for execution depends on whether the jobs are NDL programs creating new network definition files, or are control statements intended to inspect existing network definition files.

All jobs using the NDL Processor contain a control statement portion. File creation programs can contain an additional program portion as the input file (see section 5), and can contain data portions not related to NDL program processing. File inspection jobs do not contain a program portion, but can contain data portions unrelated to NDL Processor use.

CONTROL STATEMENT PORTION

The control statement portion of any job using the NDL Processor begins with job name and NOS USER statements, and can contain any control statement valid for the operating system. The control statements required in this portion depend upon whether file creation or file inspection is performed.

FILE CREATION

The control statement portion of an NDL program job must contain the program execution statement shown in figure 6-1. This program execution statement causes the loading of the NDL Processor from the system library.

The parameters in the NDLP program execution control statement are order-independent. If an option is desired, the keyword character and equal sign must be specified explicitly; a statement of the following form is not valid.

```
NDLP(TAPE1,TAPE2,SF)
```

The significance of parentheses and embedded blanks depends on the operating system conventions used. Commas are not used to positionally identify omitted parameters; a statement of the following form, therefore, is also invalid.

```
NDLP(,,LO=SF)
```

The following sample program execution control statements are all valid.

- NDLP.

  Produces the defname and normal source listings by default.

- NDLP(LO=SF)

  Does not produce either default listing, but produces both the expanded source listing and the file content summary listing.

- NDLP(L=0)

  Creates the network definition files, but produces no listings.

```
NDLP[(I=ifn,L=ofn,LO=list)]
```

ifn The name of the file on which the input source statements are located. This file name is subject to the same constraints as all other file names processed by the operating system. If this parameter is omitted, the characters I= can also be omitted and a default file name of INPUT is assumed.

ofn The name of the file on which all of the NDL program's listable output should be written. This file name is subject to the same constraints as all other file names processed by the operating system. If this parameter is omitted, the characters L= can also be omitted and a default file name of OUTPUT is assumed. If ofn is zero, no listings are produced.

list A list of alphabetic characters specifying the type of listing output that should be produced for the NDL program. If no list parameter is specified, the characters LO= can also be omitted and the default values indicated below are assumed. The list parameters can be

N Indicates a normal source listing; N is a default value. This consists of a listing of the source statements as they were read from the input file. Each such statement has an NDL Processor-generated sequence number associated with it, each statement containing an error is prefixed with the letter E, each statement associated with a warning diagnostic is prefixed with the letter W, and each statement using a defname identifier (see section 3, Special Purpose Statements) is prefixed with the letter D.

D Indicates a defname listing; this is a default value. This list contains each defname used in the NDL program and the character string associated with it by its DEFINE statement.

S Indicates an expanded source listing. This is identical to the normal source listing except that, wherever a defname appears in the normal listing, the associated character string is substituted in the expanded listing.

F Indicates a file content summary listing. This consists of the name, type, and summarized content information for each of the validated network definition files produced by the NDL program. Unvalidated files are not listed.

These parameters can be specified in any order. If list is blank, no listings are produced. If one of the defaults is specified, the other default is suppressed.

Figure 6-1. NDLP File Creation Control Statement Format

- NDLP(LO=NDSF)

  Produces all four listings.

- NDLP(LO=N)

  Produces only the normal source listing; specification of only one of the listing output default values always suppresses the other default.

The control statement portion must also contain a NOS DEFINE control statement to make the NOS local files created by the NDL Processor direct access permanent files. This statement must precede the NDLP statement. The Network Supervisor and Communications Supervisor attach the network definition files using the permanent file names specified in their startup procedure files. The DEFINE control statement should have the first form shown in figure 6-2 if the files specified are to be used as the default network definition files in the network initialization commands. If the default permanent file names are not used in the commands, any unique names can be specified as permanent file names for lfn_1 and lfn_2. When more than one network or local configuration file is created by the NDL program, only one set can be given the default permanent file names.

DEFINE, $lfn_1 = \left\{ \begin{matrix} \text{NCFFILE} \\ pfn_1 \end{matrix} \right\}$,$lfn_2 = \left\{ \begin{matrix} \text{LCFFILE} \\ pfn_2 \end{matrix} \right\}$

lfn_1 The local file name assigned to the network configuration file on the NFILE statement within the network division of the data portion of the job.

lfn_2 The local file name assigned to the local configuration file on the LFILE statement within the data portion of the job.

pfn_1 The permanent file name to assign to the network configuration file for use in the administrative operator network initiation procedure.

pfn_2 The permanent file name to assign to the local configuration file for use in the administrative operator network initiation procedure.

Figure 6-2. DEFINE Control Statement Format

Any number of local or network configuration files can be made permanent files with appropriate alternative forms of the DEFINE control statement. Refer to the NOS 1 Reference Manual, Volume 1, for details.

The control statement portion can also contain an EXIT statement and subsequent control statements, if error exit processing of the job is desired. When NDL Processor execution ends abnormally, job processing control transfers to any EXIT statement present. The control statement portion is terminated by an EOR indicator, such as a multipunched card with 7/8/9 in column 1, or with /*EORnn if submitted through a multileaving station.

FILE INSPECTION

NDL Processor file inspection is limited to producing file content summary listings for files containing verification records. File inspection jobs contain the form of the NDL Processor control statement shown in figure 6-3. If the files to be inspected are not already local, the NDLP statement must be preceded by a NOS ATTACH control statement valid for the two files concerned. The NDL Processor can only list one local configuration file and one network configuration file for each NDLP control statement used. A file inspection job can be combined with a file creation job, but each portion requires a separate NDLP statement of the appropriate type.

NDLP(L=ofn,LF=lfn_1,NF=lfn_2)

ofn The name of the file on which the file content summary listings should be written. This file name is subject to the same constraints as all other file names processed by the operating system. If this parameter is omitted, the characters L= can also be omitted and a default file name of OUTPUT is assumed.

lfn_1 The local file name of the local configuration file for which a content summary listing should be prepared. This must be the local file name assigned to the permanent file form of the local configuration file in a preceding ATTACH statement or the local file name assigned in an LFILE statement in a preceding file creation portion of the job. This name is required; there is no default value. If no local configuration file content summary is desired, you can omit the LF=lfn_1 specification.

lfn_2 The local file name of the network configuration file for which a content summary listing should be prepared. This must be the local file name assigned to the permanent file form of the network configuration file in a preceding ATTACH statement or the local file name assigned in an NFILE statement in a preceding file creation portion of the job. This name is required; there is no default value. If no network configuration file content summary is desired, you can omit the NF=lfn_2 specification.

Figure 6-3. NDLP File Inspection Control Statement Format

The parameters in the NDLP inspection control statement are order-independent. If an option is desired, the keyword character or characters must be specified explicitly; a statement of the following form is not valid.

```
NDLP(TAPE1,TSTLCF,TSTNCF)
```

The significance of parentheses and embedded blanks depends on the operating system conventions used. Commas are not used to positionally identify omitted parameters; a statement of the following form is, therefore, also invalid.

```
NDLP(,,NF=TSTNCF)
```

If you omit all parameters from the parameter list, a dayfile error diagnostic can occur. The following statement form is not valid for file inspection jobs.

```
NDLP.
```

because the NDL Processor assumes this statement indicates a file creation job.

The following inspection control statements are all valid.

- NDLP(LF=TSTLCF,NF=TSTNCF)

  Produces content summary listings of one local configuration file and one network configuration file.

- NDLP(LF=TSTLCF)

  Produces content summary listing of one local configuration file.

- NDLP(LF=TSTLCF,L=TAPE1)

  Writes one local configuration file contents summary listing on file TAPE1.

The control statement portion can contain an EXIT statement and subsequent control statements, if you desire error exit processing. When NDL Processor execution ends abnormally, job processing control transfers to any EXIT statement present.

The control statement portion is terminated by a card with 7/8/9 multipunched in column 1, /*EORnn if submitted through a multileaving station, or an EOR indicator.

PROGRAM AND DATA PORTIONS

This portion of an NDL program's job structure contains only the language statements described in sections 3 and 4. The NDL Processor makes no provision for subroutines, subprograms, or processes written in compiler or assembly languages and intended to be run as part of an NDL program.

Each program or data portion is terminated by an EOR indicator, such as a multipunched card with 7/8/9 in column 1, or with /*EORnn if submitted through a multileaving station. If the job contains control statements for postprocessing of the network definition files, subsequent data portions can be specified to provide directives.

The job is terminated by an end-of-information indicator, such as a card with 6/7/8/9 multipunched in column 1, or /*EOI if submitted through a multileaving station. If the job contains only one program or data portion, you can omit the EOR indicator terminating that portion.

SAMPLE PROGRAM 7

This section contains a debugged sample NDL file creation program and the listings produced when it is executed by the NDL Processor. The dayfile listing is not shown.

PROGRAM INPUT

The input for this sample NDL program consists of 80-column cards submitted as a batch job under NOS. The job is structured to contain the control statement portion shown in figure 7-1 and the program portion, containing all of the NDL program statements, shown in figure 7-2.

```
ND99.
USER (LORI,PASWRD)
CHARGE (0059,7346219)
DEFINE(PUBSNCF=NCFFILE,PUBSLCF=LCFFILE)
NDLP (LO=NDSF)
7/8/9
```

Figure 7-1. Sample Program Control Statements

PROGRAM OUTPUT

The program execution statement in the input control statement portion requests the NDL Processor to produce all four possible types of output listing. The normal source listing is shown in figures 7-2 and 7-6; the defname listing is shown in figures 7-3 and 7-7; the expanded source listing is given in figures 7-4 and 7-8; and the file content summary listing appears as figures 7-5 and 7-9. The time and date shown at the top of each listing page are the same as the time and date stored in the verification records of the network definition files produced by the program.

If errors are detected in the program, a multiple-page error summary is produced in the format shown in appendix B. Dayfile messages are produced in both cases; these are also described in appendix B. Because the sample program contains no fatal errors, no error summary listing is shown.

If the DEFINE statement summary (defname listing) option has been specified on the NDLP control statement but no DEFINE statements occur in the NDL program, a summary containing the following message is produced:

NO DEFINE COMMANDS ENCOUNTERED

A complete file creation job output listing with all options would be produced in the following order for each division:

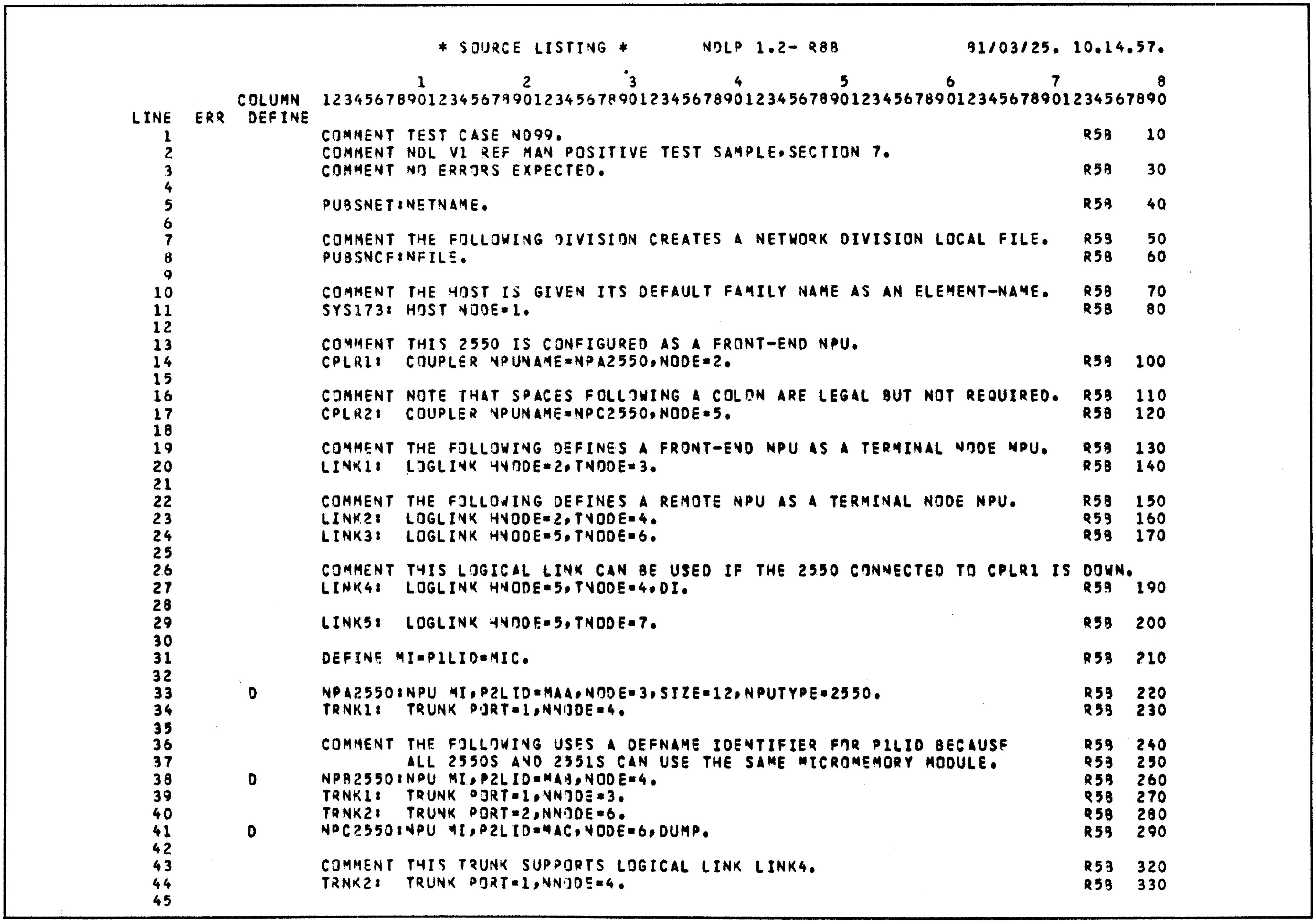
- Normal source listing
- Error summary listing with line, error code, and column index (only if errors occurred)
- Error summary listing, describing error codes (only if errors occurred)
- DEFINE statement summary (defname listing)
- Expanded source listing
- Network definition file summary (only if no fatal errors occurred)

NETWORK CONFIGURED

Figure 7-10 is a block diagram of the physical network configured by the NDL program shown in figures 7-1 through 7-9. The logical network configuration the program produces is shown in figure 7-11.

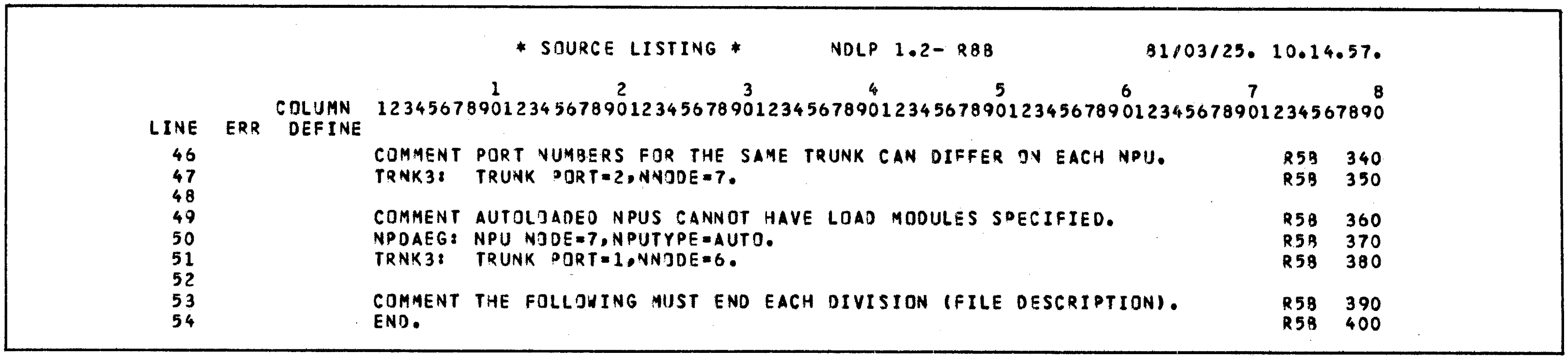
The NDL Processor does not access the operating system equipment status table (EST). The EST entries in figure 7-10 are shown as background for interpreting the listing. The CCP build entries appear in the figure for the same reason; the NDL Processor does not access or process the CCP file in any manner.

No attempt is made in figure 7-10 to show the full parameterization of each configured terminal. Parameterization values other than defaults are indicated. Note that trunk TRNK2 is enabled in figure 7-10, while the corresponding logical link, LINK4, is disabled in figure 7-11.



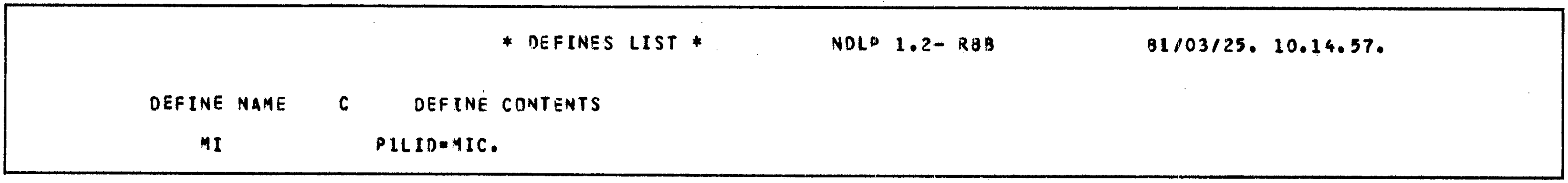
```
                                 * SOURCE LISTING *       NDLP 1.2- R8B           81/03/25. 10.14.57.

                                          1         2         3         4         5         6         7         8
                      COLUMN     12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE   ERR   DEFINE
   1                             COMMENT TEST CASE ND99.                                                R5B    10
   2                             COMMENT NDL V1 REF MAN POSITIVE TEST SAMPLE,SECTION 7.
   3                             COMMENT NO ERRORS EXPECTED.                                            R5B    30
   4
   5                             PUBSNET:NETNAME.                                                       R5B    40
   6
   7                             COMMENT THE FOLLOWING DIVISION CREATES A NETWORK DIVISION LOCAL FILE.  R5B    50
   8                             PUBSNCF:NFILE.                                                         R5B    60
   9
  10                             COMMENT THE HOST IS GIVEN ITS DEFAULT FAMILY NAME AS AN ELEMENT-NAME.  R5B    70
  11                             SYS173: HOST NODE=1.                                                   R5B    80
  12
  13                             COMMENT THIS 2550 IS CONFIGURED AS A FRONT-END NPU.
  14                             CPLR1:  COUPLER NPUNAME=NPA2550,NODE=2.                                R5B   100
  15
  16                             COMMENT NOTE THAT SPACES FOLLOWING A COLON ARE LEGAL BUT NOT REQUIRED. R5B   110
  17                             CPLR2:  COUPLER NPUNAME=NPC2550,NODE=5.                                R5B   120
  18
  19                             COMMENT THE FOLLOWING DEFINES A FRONT-END NPU AS A TERMINAL NODE NPU.  R5B   130
  20                             LINK1:  LOGLINK HNODE=2,TNODE=3.                                       R5B   140
  21
  22                             COMMENT THE FOLLOWING DEFINES A REMOTE NPU AS A TERMINAL NODE NPU.     R5B   150
  23                             LINK2:  LOGLINK HNODE=2,TNODE=4.                                       R5B   160
  24                             LINK3:  LOGLINK HNODE=5,TNODE=6.                                       R5B   170
  25
  26                             COMMENT THIS LOGICAL LINK CAN BE USED IF THE 2550 CONNECTED TO CPLR1 IS DOWN.
  27                             LINK4:  LOGLINK HNODE=5,TNODE=4,DI.                                    R5B   190
  28
  29                             LINK5:  LOGLINK HNODE=5,TNODE=7.                                       R5B   200
  30
  31                             DEFINE MI=P1LID=MIC.                                                   R5B   210
  32
  33             D               NPA2550:NPU MI,P2LID=MAA,NODE=3,SIZE=12,NPUTYPE=2550.                  R5B   220
  34                             TRNK1:  TRUNK PORT=1,NNODE=4.                                          R5B   230
  35
  36                             COMMENT THE FOLLOWING USES A DEFNAME IDENTIFIER FOR P1LID BECAUSE      R5B   240
  37                                     ALL 2550S AND 2551S CAN USE THE SAME MICROMEMORY MODULE.       R5B   250
  38             D               NPB2550:NPU MI,P2LID=MAB,NODE=4.                                       R5B   260
  39                             TRNK1:  TRUNK PORT=1,NNODE=3.                                          R5B   270
  40                             TRNK2:  TRUNK PORT=2,NNODE=6.                                          R5B   280
  41             D               NPC2550:NPU MI,P2LID=MAC,NODE=6,DUMP.                                  R5B   290
  42
  43                             COMMENT THIS TRUNK SUPPORTS LOGICAL LINK LINK4.                        R5B   320
  44                             TRNK2:  TRUNK PORT=1,NNODE=4.                                          R5B   330
  45
```

Figure 7-2. Normal Source Listing, Network Division (Sheet 1 of 2)



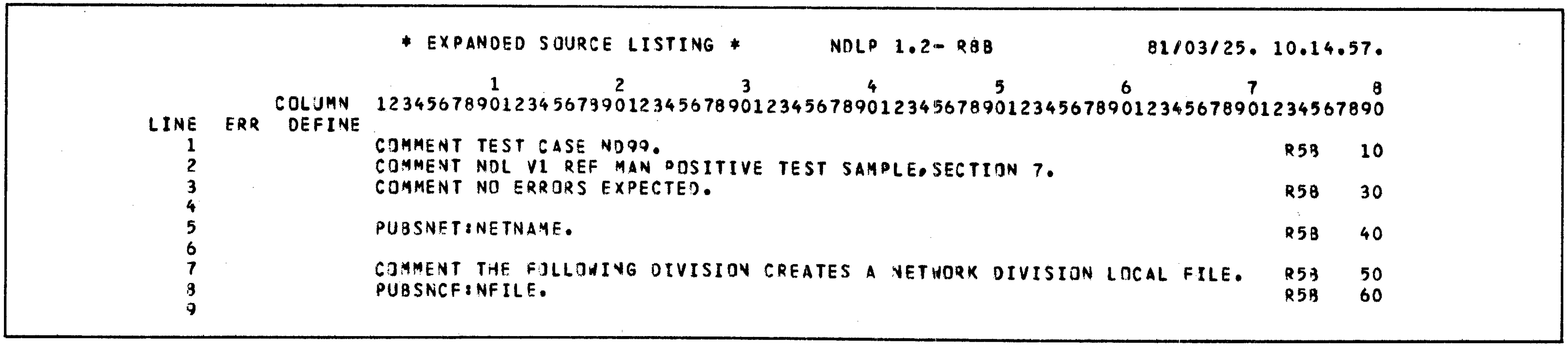
```
                              * SOURCE LISTING *        NDLP 1.2- R8B           81/03/25. 10.14.57.

                                     1         2         3         4         5         6         7         8
                  COLUMN    12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE    ERR    DEFINE
  46                        COMMENT PORT NUMBERS FOR THE SAME TRUNK CAN DIFFER ON EACH NPU.          R58   340
  47                        TRNK3:   TRUNK PORT=2,NNODE=7.                                           R58   350
  48
  49                        COMMENT AUTOLOADED NPUS CANNOT HAVE LOAD MODULES SPECIFIED.              R58   360
  50                        NPDAEG: NPU NODE=7,NPUTYPE=AUTO.                                         R58   370
  51                        TRNK3:   TRUNK PORT=1,NNODE=6.                                           R58   380
  52
  53                        COMMENT THE FOLLOWING MUST END EACH DIVISION (FILE DESCRIPTION).         R58   390
  54                        END.                                                                     R58   400
```

Figure 7-2. Normal Source Listing, Network Division (Sheet 2 of 2)



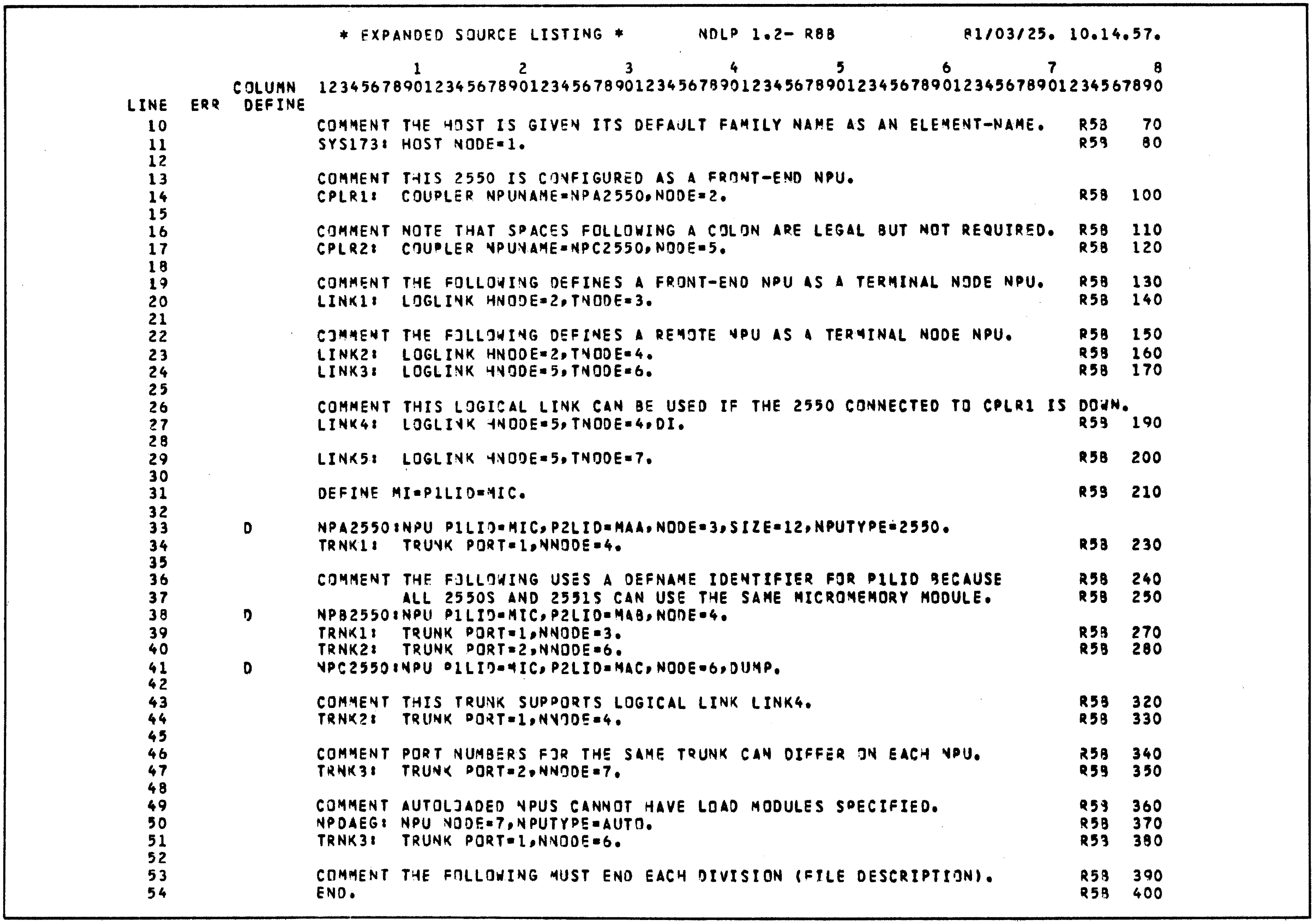
```
                              * DEFINES LIST *          NDLP 1.2- R8B           81/03/25. 10.14.57.

DEFINE NAME    C      DEFINE CONTENTS

    MI              P1LID=MIC.
```

Figure 7-3. DEFINE Statement Summary Listing, Network Division



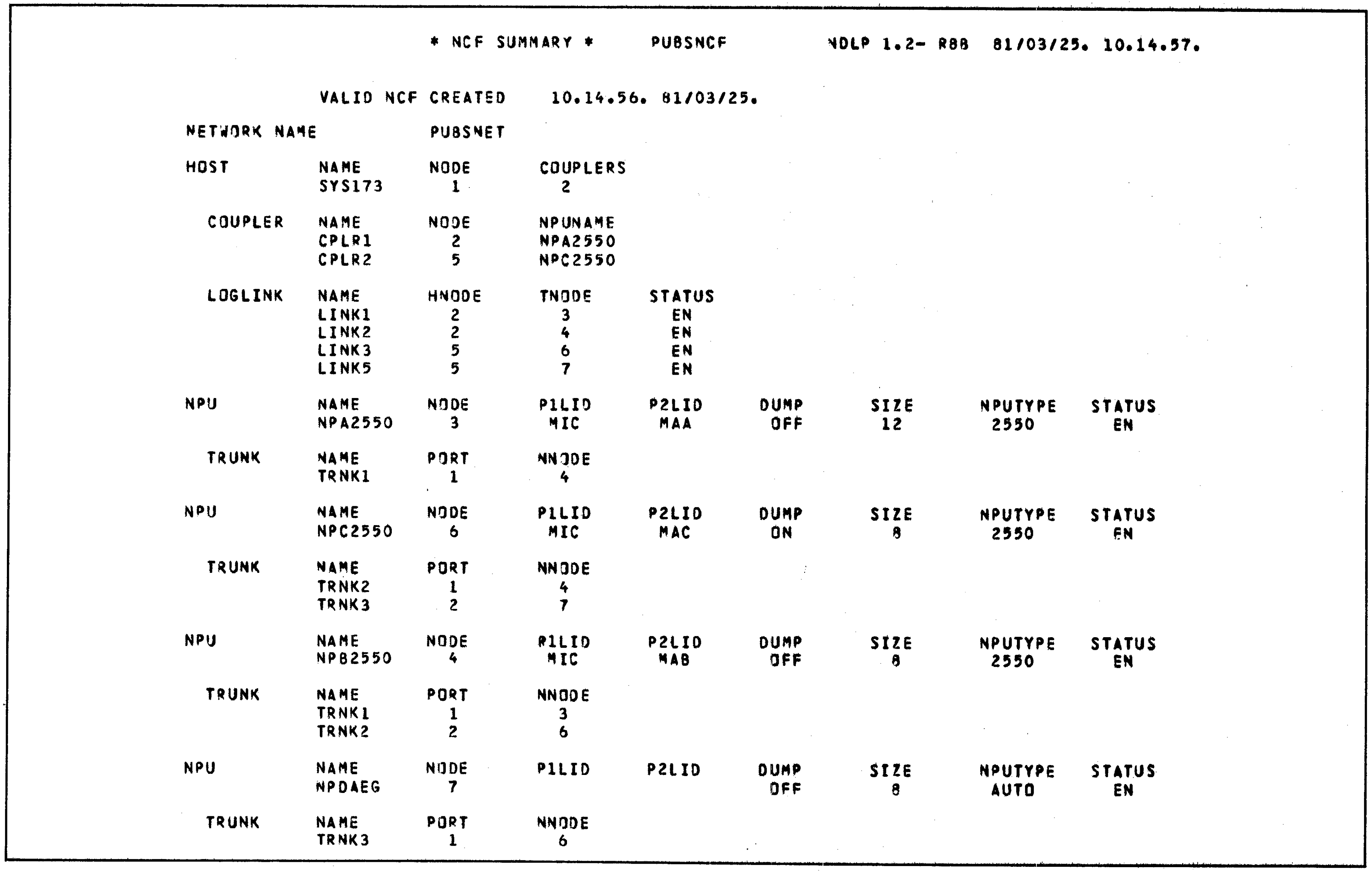
```
                         * EXPANDED SOURCE LISTING *    NDLP 1.2- R8B           81/03/25. 10.14.57.

                                     1         2         3         4         5         6         7         8
                  COLUMN    12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE    ERR    DEFINE
   1                        COMMENT TEST CASE ND99.                                                  R58    10
   2                        COMMENT NDL V1 REF MAN POSITIVE TEST SAMPLE,SECTION 7.
   3                        COMMENT NO ERRORS EXPECTED.                                              R58    30
   4
   5                        PUBSNET:NETNAME.                                                         R58    40
   6
   7                        COMMENT THE FOLLOWING DIVISION CREATES A NETWORK DIVISION LOCAL FILE.    R58    50
   8                        PUBSNCF:NFILE.                                                           R58    60
   9
```

Figure 7-4. Expanded Source Listing, Network Division (Sheet 1 of 2)



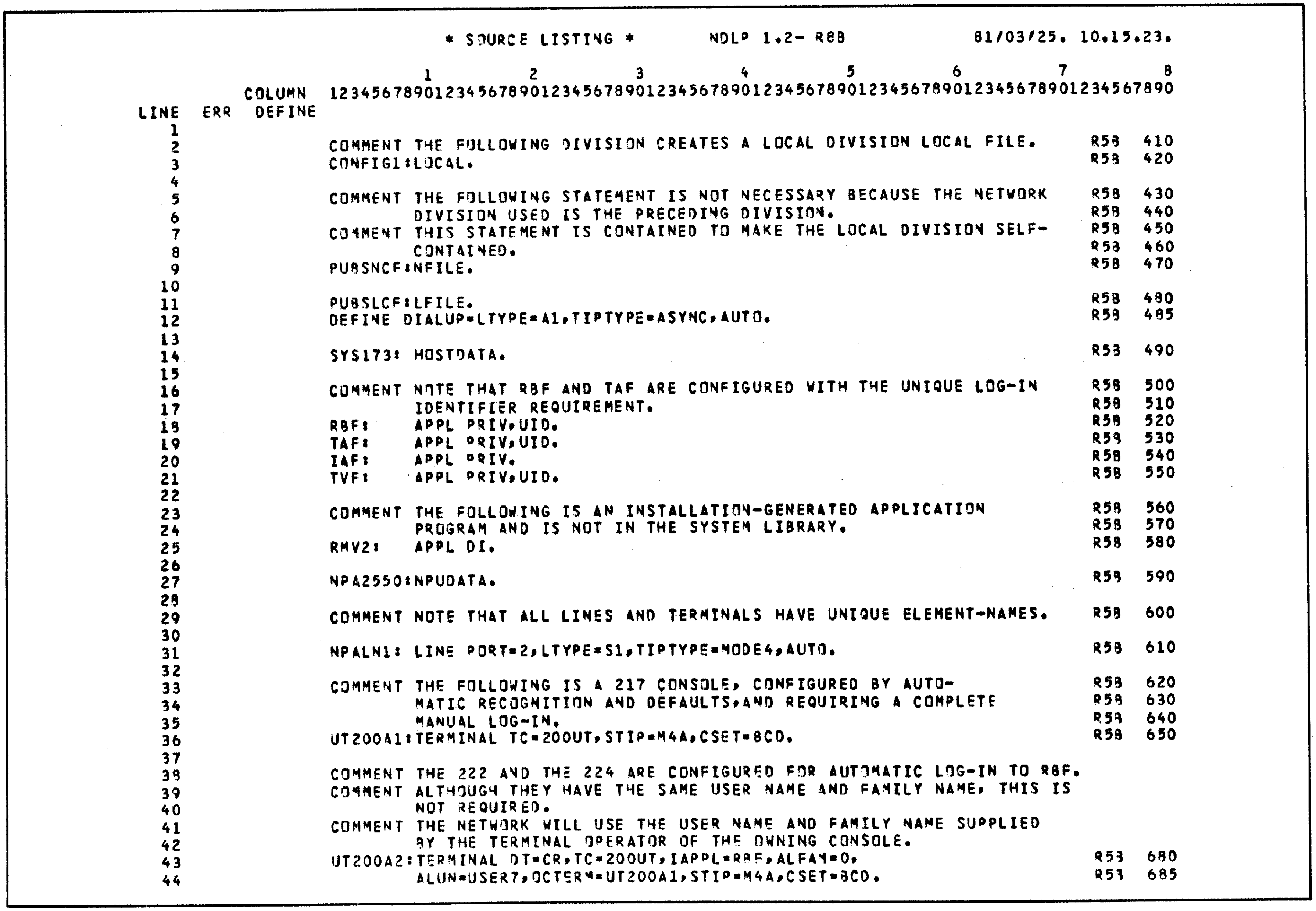
```
                    * EXPANDED SOURCE LISTING *       NDLP 1.2- R88          81/03/25. 10.14.57.

                           1         2         3         4         5         6         7         8
          COLUMN  12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE  ERR  DEFINE
  10              COMMENT THE HOST IS GIVEN ITS DEFAULT FAMILY NAME AS AN ELEMENT-NAME.   R58    70
  11              SYS173: HOST NODE=1.                                                    R58    80
  12
  13              COMMENT THIS 2550 IS CONFIGURED AS A FRONT-END NPU.
  14              CPLR1:  COUPLER NPUNAME=NPA2550,NODE=2.                                 R58   100
  15
  16              COMMENT NOTE THAT SPACES FOLLOWING A COLON ARE LEGAL BUT NOT REQUIRED.  R58   110
  17              CPLR2:  COUPLER NPUNAME=NPC2550,NODE=5.                                 R58   120
  18
  19              COMMENT THE FOLLOWING DEFINES A FRONT-END NPU AS A TERMINAL NODE NPU.   R58   130
  20              LINK1:  LOGLINK HNODE=2,TNODE=3.                                        R58   140
  21
  22              COMMENT THE FOLLOWING DEFINES A REMOTE NPU AS A TERMINAL NODE NPU.      R58   150
  23              LINK2:  LOGLINK HNODE=2,TNODE=4.                                        R58   160
  24              LINK3:  LOGLINK HNODE=5,TNODE=6.                                        R58   170
  25
  26              COMMENT THIS LOGICAL LINK CAN BE USED IF THE 2550 CONNECTED TO CPLR1 IS DOWN.
  27              LINK4:  LOGLINK HNODE=5,TNODE=4,DI.                                     R58   190
  28
  29              LINK5:  LOGLINK HNODE=5,TNODE=7.                                        R58   200
  30
  31              DEFINE MI=P1LID=MIC.                                                    R58   210
  32
  33       D      NPA2550:NPU P1LID=MIC,P2LID=MAA,NODE=3,SIZE=12,NPUTYPE=2550.
  34              TRNK1:  TRUNK PORT=1,NNODE=4.                                           R58   230
  35
  36              COMMENT THE FOLLOWING USES A DEFNAME IDENTIFIER FOR P1LID BECAUSE       R58   240
  37                      ALL 2550S AND 2551S CAN USE THE SAME MICROMEMORY MODULE.        R58   250
  38       D      NPB2550:NPU P1LID=MIC,P2LID=MAB,NODE=4.
  39              TRNK1:  TRUNK PORT=1,NNODE=3.                                           R58   270
  40              TRNK2:  TRUNK PORT=2,NNODE=6.                                           R58   280
  41       D      NPC2550:NPU P1LID=MIC,P2LID=MAC,NODE=6,DUMP.
  42
  43              COMMENT THIS TRUNK SUPPORTS LOGICAL LINK LINK4.                         R58   320
  44              TRNK2:  TRUNK PORT=1,NNODE=4.                                           R58   330
  45
  46              COMMENT PORT NUMBERS FOR THE SAME TRUNK CAN DIFFER ON EACH NPU.         R58   340
  47              TRNK3:  TRUNK PORT=2,NNODE=7.                                           R58   350
  48
  49              COMMENT AUTOLOADED NPUS CANNOT HAVE LOAD MODULES SPECIFIED.             R58   360
  50              NPDAEG: NPU NODE=7,NPUTYPE=AUTO.                                        R58   370
  51              TRNK3:  TRUNK PORT=1,NNODE=6.                                           R58   380
  52
  53              COMMENT THE FOLLOWING MUST END EACH DIVISION (FILE DESCRIPTION).        R58   390
  54              END.                                                                    R58   400
```

Figure 7-4. Expanded Source Listing, Network Division (Sheet 2 of 2)



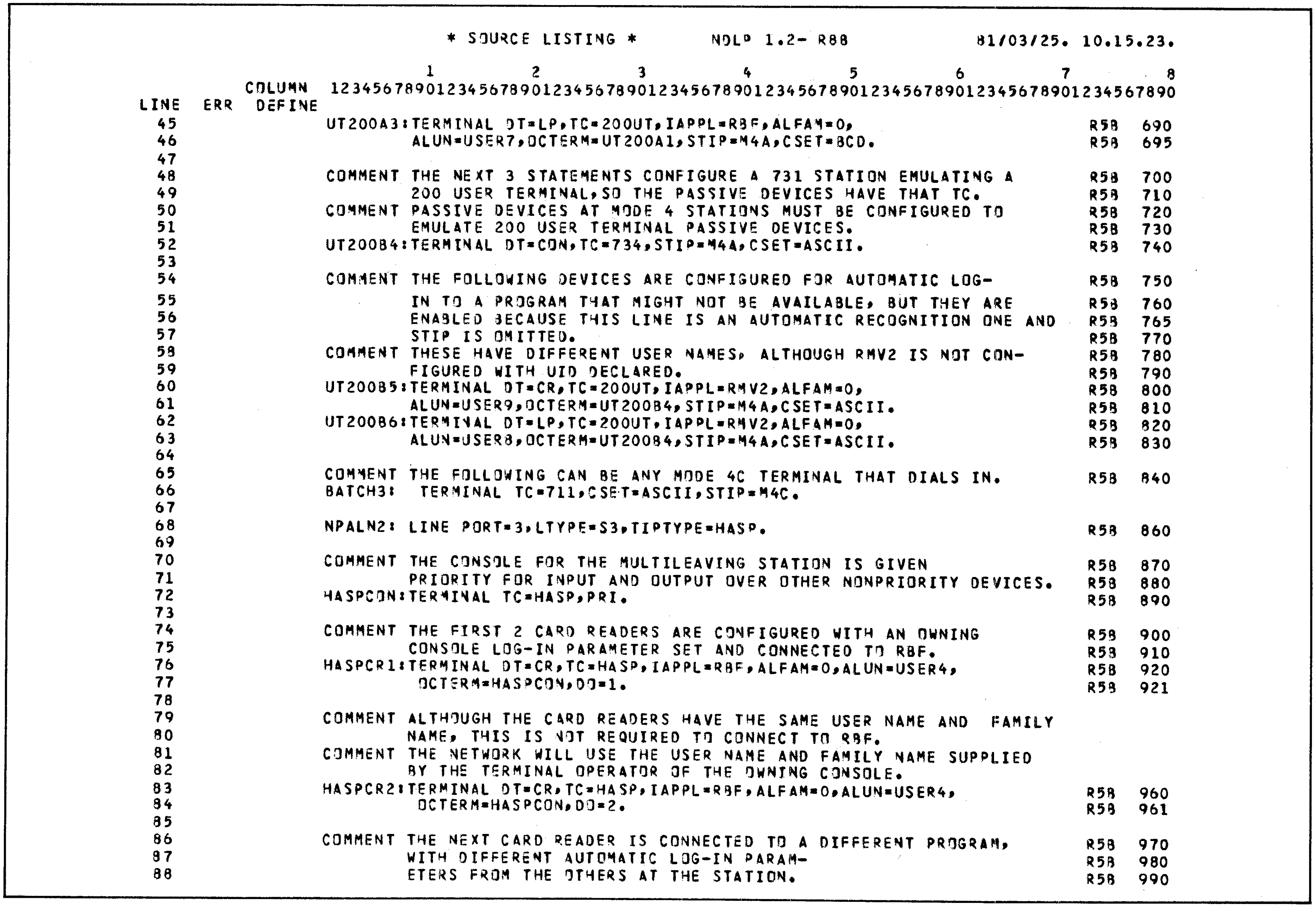
```
                  * NCF SUMMARY *     PUBSNCF        NDLP 1.2- R88  81/03/25. 10.14.57.

             VALID NCF CREATED    10.14.56. 81/03/25.

       NETWORK NAME          PUBSNET

       HOST          NAME      NODE      COUPLERS
                     SYS173     1          2

         COUPLER     NAME      NODE      NPUNAME
                     CPLR1      2        NPA2550
                     CPLR2      5        NPC2550

         LOGLINK     NAME      HNODE     TNODE     STATUS
                     LINK1      2          3         EN
                     LINK2      2          4         EN
                     LINK3      5          6         EN
                     LINK5      5          7         EN

       NPU           NAME      NODE      P1LID     P2LID     DUMP      SIZE      NPUTYPE   STATUS
                     NPA2550    3         MIC       MAA       OFF       12        2550      EN

         TRUNK       NAME      PORT      NNODE
                     TRNK1      1          4

       NPU           NAME      NODE      P1LID     P2LID     DUMP      SIZE      NPUTYPE   STATUS
                     NPC2550    6         MIC       MAC       ON         8        2550      EN

         TRUNK       NAME      PORT      NNODE
                     TRNK2      1          4
                     TRNK3      2          7

       NPU           NAME      NODE      P1LID     P2LID     DUMP      SIZE      NPUTYPE   STATUS
                     NPB2550    4         MIC       MAB       OFF        8        2550      EN

         TRUNK       NAME      PORT      NNODE
                     TRNK1      1          3
                     TRNK2      2          6

       NPU           NAME      NODE      P1LID     P2LID     DUMP      SIZE      NPUTYPE   STATUS
                     NPDAEG     7                            OFF        8        AUTO      EN

         TRUNK       NAME      PORT      NNODE
                     TRNK3      1          6
```

Figure 7-5. File Summary Listing, Network Division



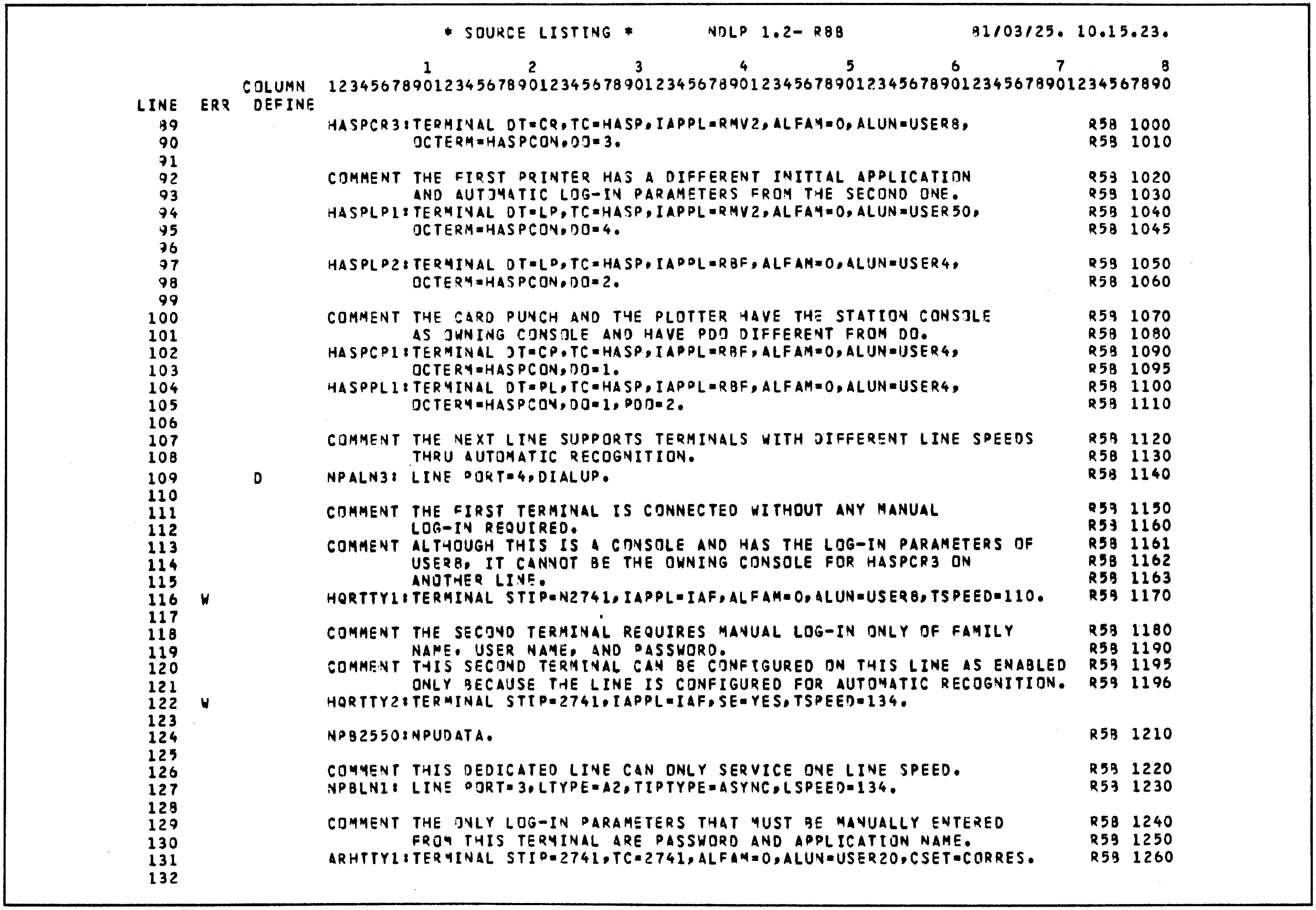
```
                            * SOURCE LISTING *        NDLP 1.2- R8B            81/03/25. 10.15.23.

                                      1         2         3         4         5         6         7         8
              COLUMN  12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE   ERR   DEFINE
  1
  2                   COMMENT THE FOLLOWING DIVISION CREATES A LOCAL DIVISION LOCAL FILE.       R58   410
  3                   CONFIG1:LOCAL.                                                            R58   420
  4
  5                   COMMENT THE FOLLOWING STATEMENT IS NOT NECESSARY BECAUSE THE NETWORK      R58   430
  6                           DIVISION USED IS THE PRECEDING DIVISION.                          R58   440
  7                   COMMENT THIS STATEMENT IS CONTAINED TO MAKE THE LOCAL DIVISION SELF-      R58   450
  8                           CONTAINED.                                                        R58   460
  9                   PUBSNCF:NFILE.                                                            R58   470
 10
 11                   PUBSLCF:LFILE.                                                            R58   480
 12                   DEFINE DIALUP=LTYPE=A1,TIPTYPE=ASYNC,AUTO.                                R58   485
 13
 14                   SYS173: HOSTDATA.                                                         R58   490
 15
 16                   COMMENT NOTE THAT RBF AND TAF ARE CONFIGURED WITH THE UNIQUE LOG-IN       R58   500
 17                           IDENTIFIER REQUIREMENT.                                           R58   510
 18                   RBF:    APPL PRIV,UID.                                                    R58   520
 19                   TAF:    APPL PRIV,UID.                                                    R58   530
 20                   IAF:    APPL PRIV.                                                        R58   540
 21                   TVF:    APPL PRIV,UID.                                                    R58   550
 22
 23                   COMMENT THE FOLLOWING IS AN INSTALLATION-GENERATED APPLICATION            R58   560
 24                           PROGRAM AND IS NOT IN THE SYSTEM LIBRARY.                         R58   570
 25                   RMV2:   APPL DI.                                                          R58   580
 26
 27                   NPA2550:NPUDATA.                                                          R58   590
 28
 29                   COMMENT NOTE THAT ALL LINES AND TERMINALS HAVE UNIQUE ELEMENT-NAMES.      R58   600
 30
 31                   NPALN1: LINE PORT=2,LTYPE=S1,TIPTYPE=MODE4,AUTO.                          R58   610
 32
 33                   COMMENT THE FOLLOWING IS A 217 CONSOLE, CONFIGURED BY AUTO-               R58   620
 34                           MATIC RECOGNITION AND DEFAULTS,AND REQUIRING A COMPLETE           R58   630
 35                           MANUAL LOG-IN.                                                    R58   640
 36                   UT200A1:TERMINAL TC=200UT,STIP=M4A,CSET=BCD.                              R58   650
 37
 38                   COMMENT THE 222 AND THE 224 ARE CONFIGURED FOR AUTOMATIC LOG-IN TO RBF.
 39                   COMMENT ALTHOUGH THEY HAVE THE SAME USER NAME AND FAMILY NAME, THIS IS
 40                           NOT REQUIRED.
 41                   COMMENT THE NETWORK WILL USE THE USER NAME AND FAMILY NAME SUPPLIED
 42                           BY THE TERMINAL OPERATOR OF THE OWNING CONSOLE.
 43                   UT200A2:TERMINAL DT=CR,TC=200UT,IAPPL=RBF,ALFAM=0,                        R58   680
 44                           ALUN=USER7,OCTERM=UT200A1,STIP=M4A,CSET=BCD.                      R58   685
```

Figure 7-6. Normal Source Listing, Local Division (Sheet 1 of 4)



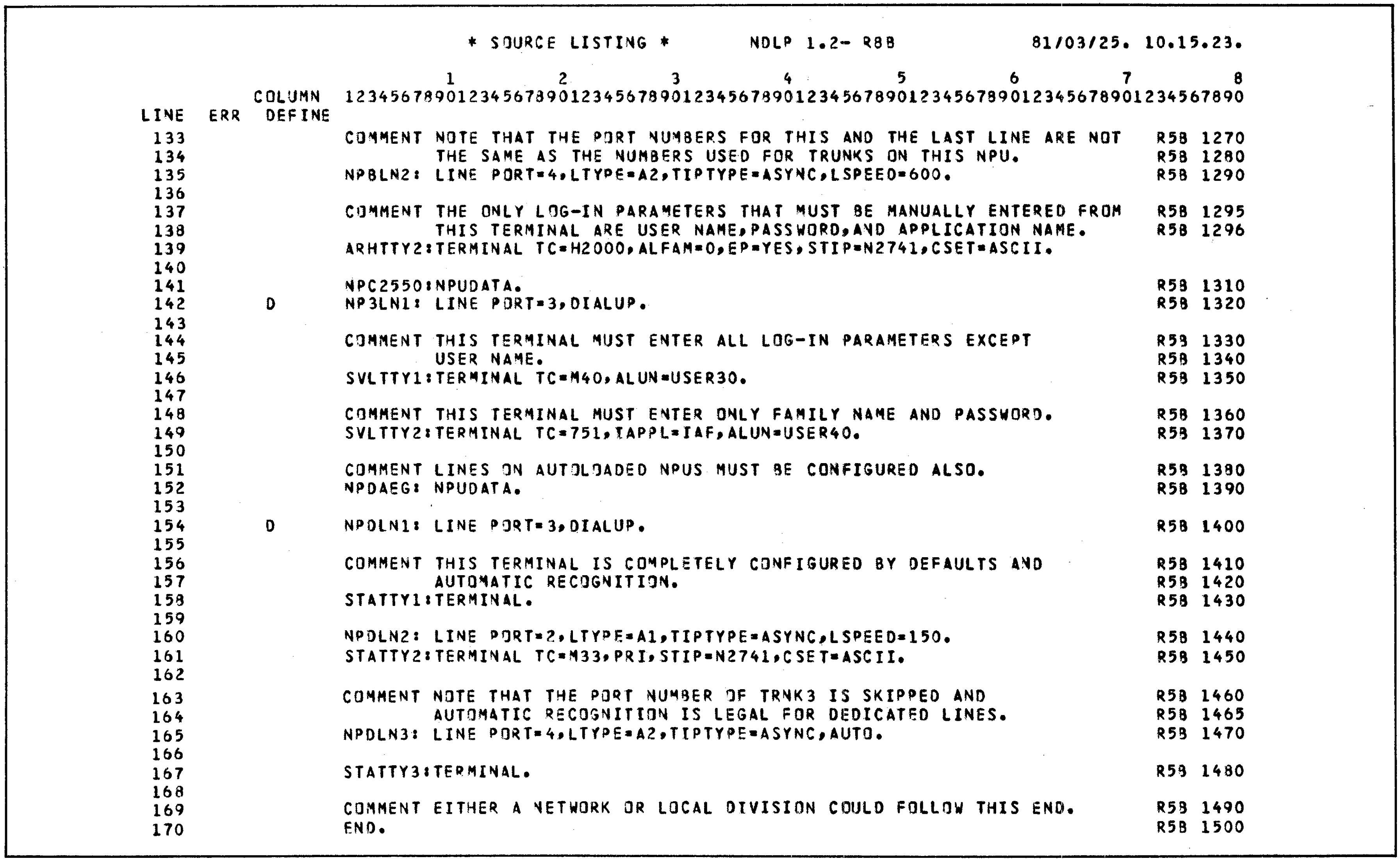
```
                               * SOURCE LISTING *       NDLP 1.2- R88           81/03/25. 10.15.23.

                                        1         2         3         4         5         6         7         8
                   COLUMN  12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE   ERR  DEFINE
  45                UT200A3:TERMINAL DT=LP,TC=200UT,IAPPL=RBF,ALFAM=0,                         R58   690
  46                        ALUN=USER7,OCTERM=UT200A1,STIP=M4A,CSET=BCD.                        R58   695
  47
  48                COMMENT THE NEXT 3 STATEMENTS CONFIGURE A 731 STATION EMULATING A          R58   700
  49                        200 USER TERMINAL,SO THE PASSIVE DEVICES HAVE THAT TC.             R58   710
  50                COMMENT PASSIVE DEVICES AT MODE 4 STATIONS MUST BE CONFIGURED TO           R58   720
  51                        EMULATE 200 USER TERMINAL PASSIVE DEVICES.                         R58   730
  52                UT200B4:TERMINAL DT=CON,TC=734,STIP=M4A,CSET=ASCII.                        R58   740
  53
  54                COMMENT THE FOLLOWING DEVICES ARE CONFIGURED FOR AUTOMATIC LOG-            R58   750
  55                        IN TO A PROGRAM THAT MIGHT NOT BE AVAILABLE, BUT THEY ARE          R58   760
  56                        ENABLED BECAUSE THIS LINE IS AN AUTOMATIC RECOGNITION ONE AND      R58   765
  57                        STIP IS OMITTED.                                                   R58   770
  58                COMMENT THESE HAVE DIFFERENT USER NAMES, ALTHOUGH RMV2 IS NOT CON-         R58   780
  59                        FIGURED WITH UID DECLARED.                                         R58   790
  60                UT200B5:TERMINAL DT=CR,TC=200UT,IAPPL=RMV2,ALFAM=0,                        R58   800
  61                        ALUN=USER9,OCTERM=UT200B4,STIP=M4A,CSET=ASCII.                     R58   810
  62                UT200B6:TERMINAL DT=LP,TC=200UT,IAPPL=RMV2,ALFAM=0,                        R58   820
  63                        ALUN=USER8,OCTERM=UT200B4,STIP=M4A,CSET=ASCII.                     R58   830
  64
  65                COMMENT THE FOLLOWING CAN BE ANY MODE 4C TERMINAL THAT DIALS IN.           R58   840
  66                BATCH3:  TERMINAL TC=711,CSET=ASCII,STIP=M4C.
  67
  68                NPALN2: LINE PORT=3,LTYPE=S3,TIPTYPE=HASP.                                 R58   860
  69
  70                COMMENT THE CONSOLE FOR THE MULTILEAVING STATION IS GIVEN                  R58   870
  71                        PRIORITY FOR INPUT AND OUTPUT OVER OTHER NONPRIORITY DEVICES.      R58   880
  72                HASPCON:TERMINAL TC=HASP,PRI.                                              R58   890
  73
  74                COMMENT THE FIRST 2 CARD READERS ARE CONFIGURED WITH AN OWNING             R58   900
  75                        CONSOLE LOG-IN PARAMETER SET AND CONNECTED TO RBF.                 R58   910
  76                HASPCR1:TERMINAL DT=CR,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,               R58   920
  77                         OCTERM=HASPCON,DO=1.                                              R58   921
  78
  79                COMMENT ALTHOUGH THE CARD READERS HAVE THE SAME USER NAME AND  FAMILY
  80                        NAME, THIS IS NOT REQUIRED TO CONNECT TO RBF.
  81                COMMENT THE NETWORK WILL USE THE USER NAME AND FAMILY NAME SUPPLIED
  82                        BY THE TERMINAL OPERATOR OF THE OWNING CONSOLE.
  83                HASPCR2:TERMINAL DT=CR,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,               R58   960
  84                         OCTERM=HASPCON,DO=2.                                              R58   961
  85
  86                COMMENT THE NEXT CARD READER IS CONNECTED TO A DIFFERENT PROGRAM,          R58   970
  87                        WITH DIFFERENT AUTOMATIC LOG-IN PARAM-                             R58   980
  88                        ETERS FROM THE OTHERS AT THE STATION.                              R58   990
```

Figure 7-6. Normal Source Listing, Local Division (Sheet 2 of 4)



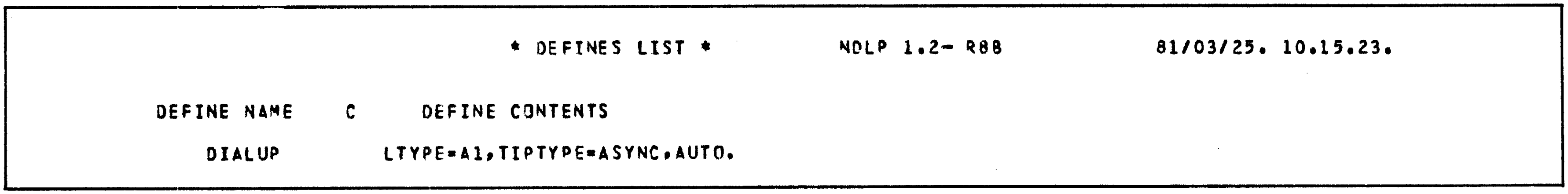
```
                                  * SOURCE LISTING *       NDLP 1.2- R58            81/03/25. 10.15.23.

                                      1         2         3         4         5         6         7         8
                      COLUMN  12345678901234567890123456789012345678901234567890123456789012345678901234567890
      LINE   ERR   DEFINE
       89                     HASPCR3:TERMINAL DT=CR,TC=HASP,IAPPL=RMV2,ALFAM=0,ALUN=USER8,            R58 1000
       90                             OCTERM=HASPCON,DO=3.                                            R58 1010
       91
       92                     COMMENT THE FIRST PRINTER HAS A DIFFERENT INITIAL APPLICATION           R58 1020
       93                             AND AUTOMATIC LOG-IN PARAMETERS FROM THE SECOND ONE.            R58 1030
       94                     HASPLP1:TERMINAL DT=LP,TC=HASP,IAPPL=RMV2,ALFAM=0,ALUN=USER50,          R58 1040
       95                             OCTERM=HASPCON,DO=4.                                            R58 1045
       96
       97                     HASPLP2:TERMINAL DT=LP,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,            R58 1050
       98                             OCTERM=HASPCON,DO=2.                                            R58 1060
       99
      100                     COMMENT THE CARD PUNCH AND THE PLOTTER HAVE THE STATION CONSOLE         R58 1070
      101                             AS OWNING CONSOLE AND HAVE PDO DIFFERENT FROM DO.               R58 1080
      102                     HASPCP1:TERMINAL DT=CP,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,            R58 1090
      103                             OCTERM=HASPCON,DO=1.                                            R58 1095
      104                     HASPPL1:TERMINAL DT=PL,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,            R58 1100
      105                             OCTERM=HASPCON,DO=1,PDO=2.                                      R58 1110
      106
      107                     COMMENT THE NEXT LINE SUPPORTS TERMINALS WITH DIFFERENT LINE SPEEDS     R58 1120
      108                             THRU AUTOMATIC RECOGNITION.                                     R58 1130
      109            D        NPALN3: LINE PORT=4,DIALUP.                                             R58 1140
      110
      111                     COMMENT THE FIRST TERMINAL IS CONNECTED WITHOUT ANY MANUAL              R58 1150
      112                             LOG-IN REQUIRED.                                                R58 1160
      113                     COMMENT ALTHOUGH THIS IS A CONSOLE AND HAS THE LOG-IN PARAMETERS OF     R58 1161
      114                             USER8, IT CANNOT BE THE OWNING CONSOLE FOR HASPCR3 ON           R58 1162
      115                             ANOTHER LINE.                                                   R58 1163
      116     W               HQRTTY1:TERMINAL STIP=N2741,IAPPL=IAF,ALFAM=0,ALUN=USER8,TSPEED=110.    R58 1170
      117
      118                     COMMENT THE SECOND TERMINAL REQUIRES MANUAL LOG-IN ONLY OF FAMILY       R58 1180
      119                             NAME, USER NAME, AND PASSWORD.                                  R58 1190
      120                     COMMENT THIS SECOND TERMINAL CAN BE CONFIGURED ON THIS LINE AS ENABLED  R58 1195
      121                             ONLY BECAUSE THE LINE IS CONFIGURED FOR AUTOMATIC RECOGNITION.  R58 1196
      122     W               HQRTTY2:TERMINAL STIP=2741,IAPPL=IAF,SE=YES,TSPEED=134.
      123
      124                     NPB2550:NPUDATA.                                                        R58 1210
      125
      126                     COMMENT THIS DEDICATED LINE CAN ONLY SERVICE ONE LINE SPEED.            R58 1220
      127                     NPBLN1: LINE PORT=3,LTYPE=A2,TIPTYPE=ASYNC,LSPEED=134.                  R58 1230
      128
      129                     COMMENT THE ONLY LOG-IN PARAMETERS THAT MUST BE MANUALLY ENTERED        R58 1240
      130                             FROM THIS TERMINAL ARE PASSWORD AND APPLICATION NAME.           R58 1250
      131                     ARHTTY1:TERMINAL STIP=2741,TC=2741,ALFAM=0,ALUN=USER20,CSET=CORRES.     R58 1260
      132
```

Figure 7-6. Normal Source Listing, Local Division (Sheet 3 of 4)



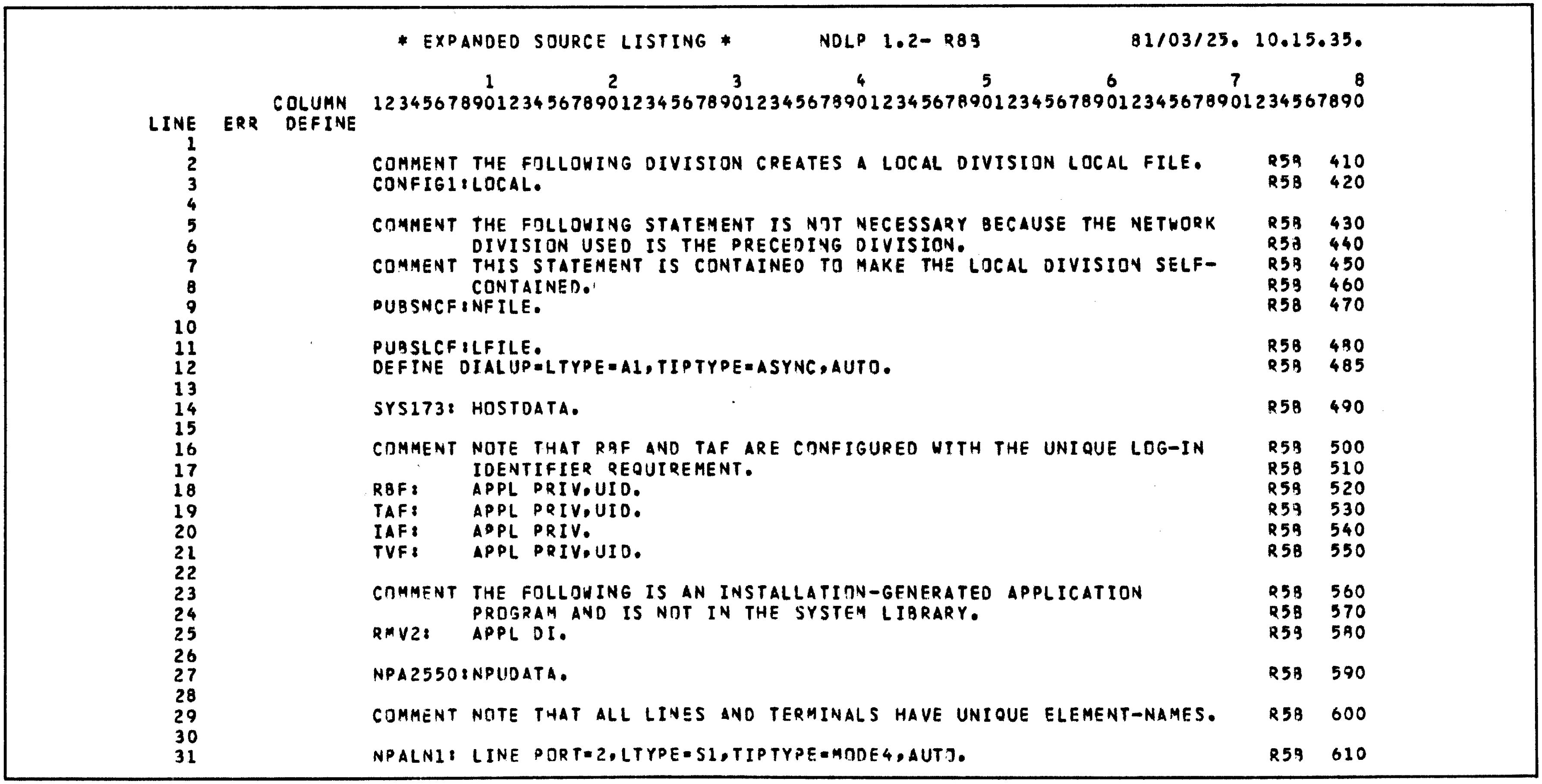
```
                              * SOURCE LISTING *       NDLP 1.2- R8B            81/03/25. 10.15.23.

                                    1         2         3         4         5         6         7         8
                  COLUMN  12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE   ERR   DEFINE
 133                      COMMENT NOTE THAT THE PORT NUMBERS FOR THIS AND THE LAST LINE ARE NOT   R5B 1270
 134                              THE SAME AS THE NUMBERS USED FOR TRUNKS ON THIS NPU.            R5B 1280
 135                      NPBLN2: LINE PORT=4,LTYPE=A2,TIPTYPE=ASYNC,LSPEED=600.                  R5B 1290
 136
 137                      COMMENT THE ONLY LOG-IN PARAMETERS THAT MUST BE MANUALLY ENTERED FROM   R5B 1295
 138                              THIS TERMINAL ARE USER NAME,PASSWORD,AND APPLICATION NAME.      R5B 1296
 139                      ARHTTY2:TERMINAL TC=H2000,ALFAM=O,EP=YES,STIP=N2741,CSET=ASCII.
 140
 141                      NPC2550:NPUDATA.                                                        R5B 1310
 142           D          NP3LN1: LINE PORT=3,DIALUP.                                             R5B 1320
 143
 144                      COMMENT THIS TERMINAL MUST ENTER ALL LOG-IN PARAMETERS EXCEPT           R5B 1330
 145                              USER NAME.                                                      R5B 1340
 146                      SVLTTY1:TERMINAL TC=M40,ALUN=USER30.                                    R5B 1350
 147
 148                      COMMENT THIS TERMINAL MUST ENTER ONLY FAMILY NAME AND PASSWORD.         R5B 1360
 149                      SVLTTY2:TERMINAL TC=751,IAPPL=IAF,ALUN=USER40.                          R5B 1370
 150
 151                      COMMENT LINES ON AUTOLOADED NPUS MUST BE CONFIGURED ALSO.               R5B 1380
 152                      NPDAEG: NPUDATA.                                                        R5B 1390
 153
 154           D          NPDLN1: LINE PORT=3,DIALUP.                                             R5B 1400
 155
 156                      COMMENT THIS TERMINAL IS COMPLETELY CONFIGURED BY DEFAULTS AND          R5B 1410
 157                              AUTOMATIC RECOGNITION.                                          R5B 1420
 158                      STATTY1:TERMINAL.                                                       R5B 1430
 159
 160                      NPDLN2: LINE PORT=2,LTYPE=A1,TIPTYPE=ASYNC,LSPEED=150.                  R5B 1440
 161                      STATTY2:TERMINAL TC=M33,PRI,STIP=N2741,CSET=ASCII.                      R5B 1450
 162
 163                      COMMENT NOTE THAT THE PORT NUMBER OF TRNK3 IS SKIPPED AND               R5B 1460
 164                              AUTOMATIC RECOGNITION IS LEGAL FOR DEDICATED LINES.             R5B 1465
 165                      NPDLN3: LINE PORT=4,LTYPE=A2,TIPTYPE=ASYNC,AUTO.                        R5B 1470
 166
 167                      STATTY3:TERMINAL.                                                       R5B 1480
 168
 169                      COMMENT EITHER A NETWORK OR LOCAL DIVISION COULD FOLLOW THIS END.       R5B 1490
 170                      END.                                                                    R5B 1500
```

Figure 7-6. Normal Source Listing, Local Division (Sheet 4 of 4)



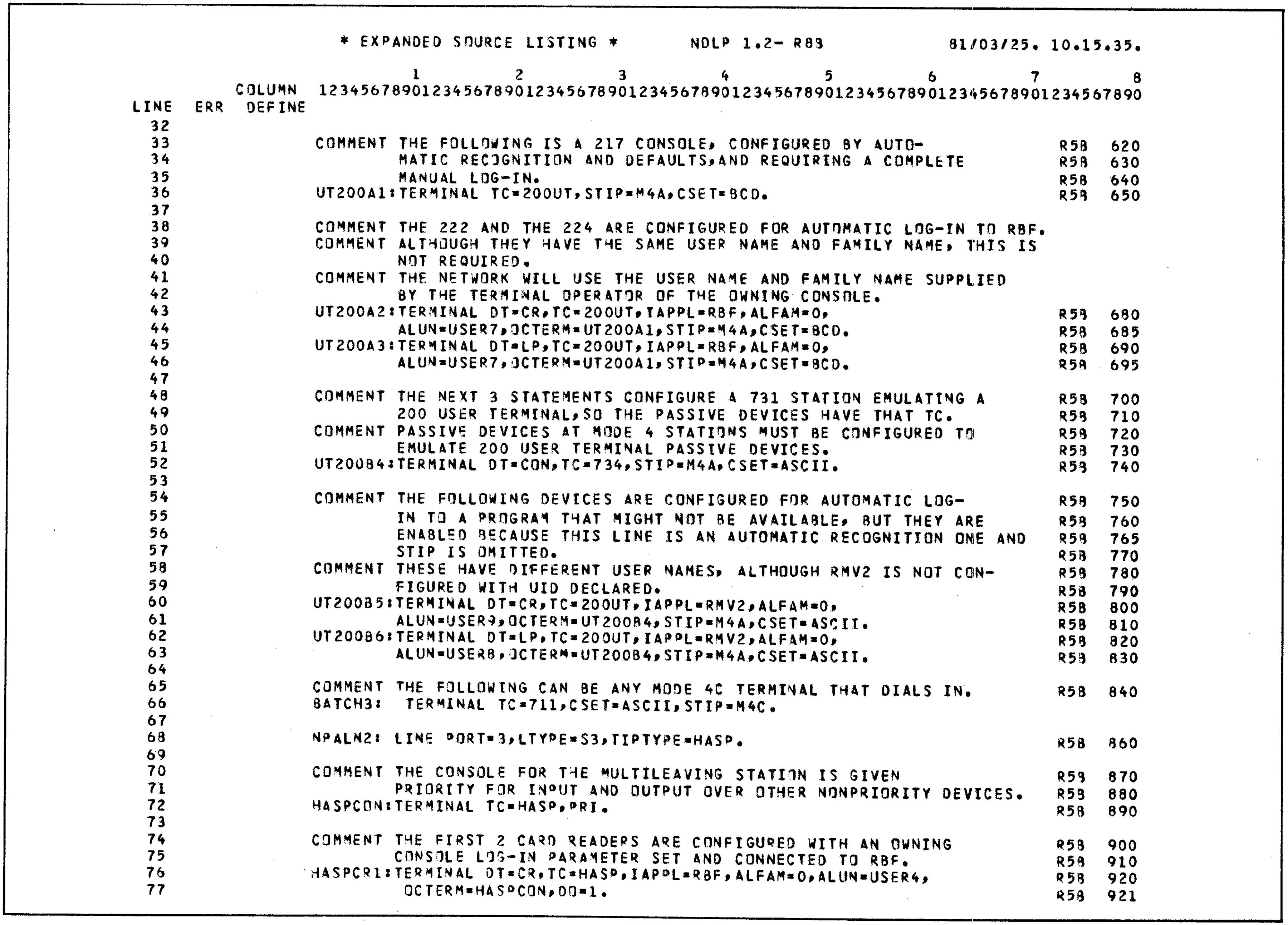
```
                          * DEFINES LIST *          NDLP 1.2- R88            81/03/25. 10.15.23.

     DEFINE NAME     C      DEFINE CONTENTS

        DIALUP         LTYPE=A1,TIPTYPE=ASYNC,AUTO.
```

Figure 7-7. DEFINE Statement Summary Listing, Local Division



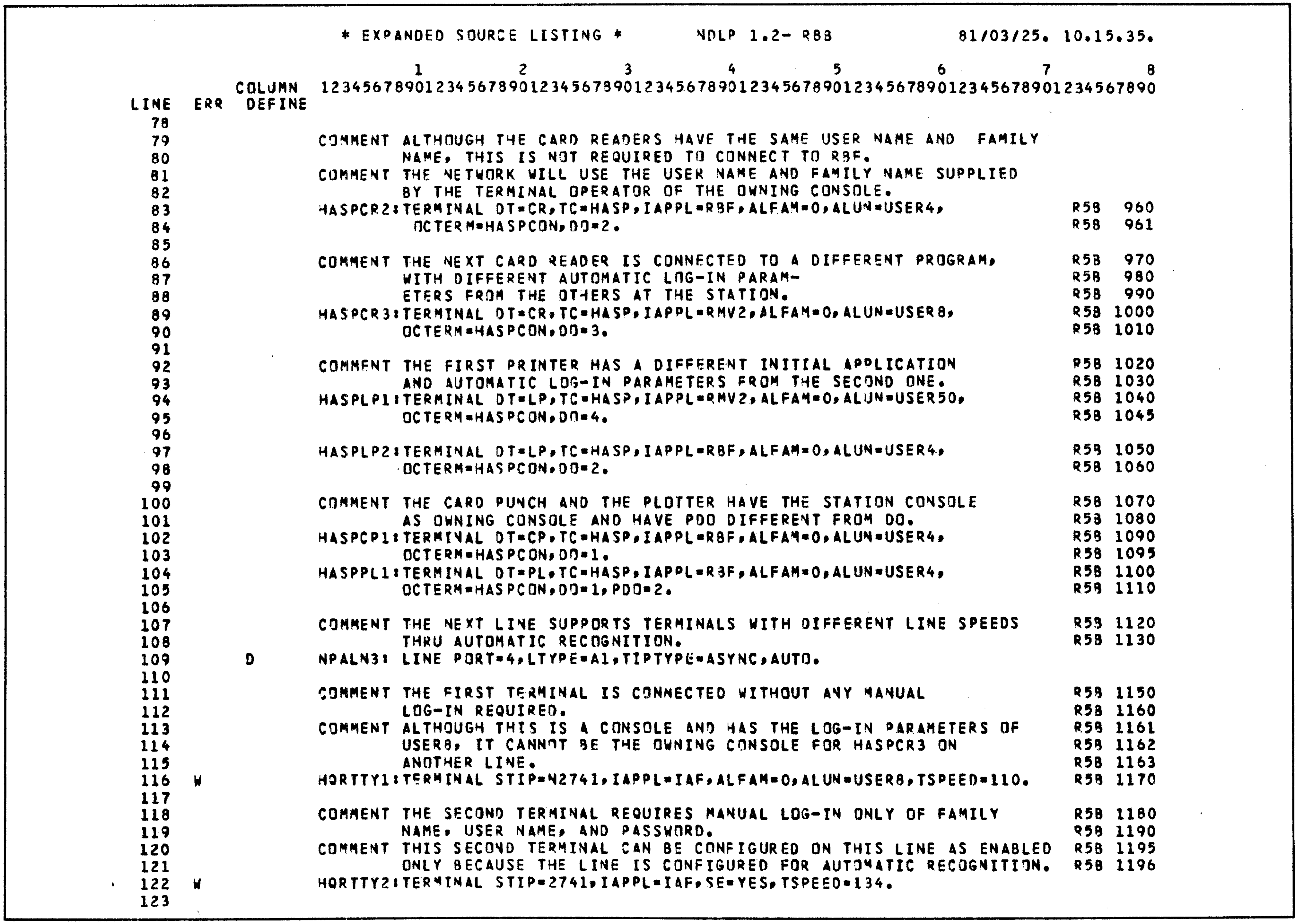
```
                     * EXPANDED SOURCE LISTING *       NDLP 1.2- R88            81/03/25. 10.15.35.

                                  1         2         3         4         5         6         7         8
                COLUMN   12345678901234567890123456789012345678901234567890123456789012345678901234567890
     LINE  ERR   DEFINE
        1
        2                COMMENT THE FOLLOWING DIVISION CREATES A LOCAL DIVISION LOCAL FILE.      R58   410
        3                CONFIG1:LOCAL.                                                           R58   420
        4
        5                COMMENT THE FOLLOWING STATEMENT IS NOT NECESSARY BECAUSE THE NETWORK     R58   430
        6                        DIVISION USED IS THE PRECEDING DIVISION.                         R58   440
        7                COMMENT THIS STATEMENT IS CONTAINED TO MAKE THE LOCAL DIVISION SELF-     R58   450
        8                        CONTAINED.                                                       R58   460
        9                PUBSNCF:NFILE.                                                           R58   470
       10
       11                PUBSLCF:LFILE.                                                           R58   480
       12                DEFINE DIALUP=LTYPE=A1,TIPTYPE=ASYNC,AUTO.                               R58   485
       13
       14                SYS173: HOSTDATA.                                                        R58   490
       15
       16                COMMENT NOTE THAT RBF AND TAF ARE CONFIGURED WITH THE UNIQUE LOG-IN      R58   500
       17                        IDENTIFIER REQUIREMENT.                                          R58   510
       18                RBF:    APPL PRIV,UID.                                                   R58   520
       19                TAF:    APPL PRIV,UID.                                                   R58   530
       20                IAF:    APPL PRIV.                                                       R58   540
       21                TVF:    APPL PRIV,UID.                                                   R58   550
       22
       23                COMMENT THE FOLLOWING IS AN INSTALLATION-GENERATED APPLICATION           R58   560
       24                        PROGRAM AND IS NOT IN THE SYSTEM LIBRARY.                        R58   570
       25                RMV2:   APPL DI.                                                         R58   580
       26
       27                NPA2550:NPUDATA.                                                         R58   590
       28
       29                COMMENT NOTE THAT ALL LINES AND TERMINALS HAVE UNIQUE ELEMENT-NAMES.     R58   600
       30
       31                NPALN1: LINE PORT=2,LTYPE=S1,TIPTYPE=MODE4,AUTO.                         R58   610
```

Figure 7-8. Expanded Source Listing, Local Division (Sheet 1 of 4)



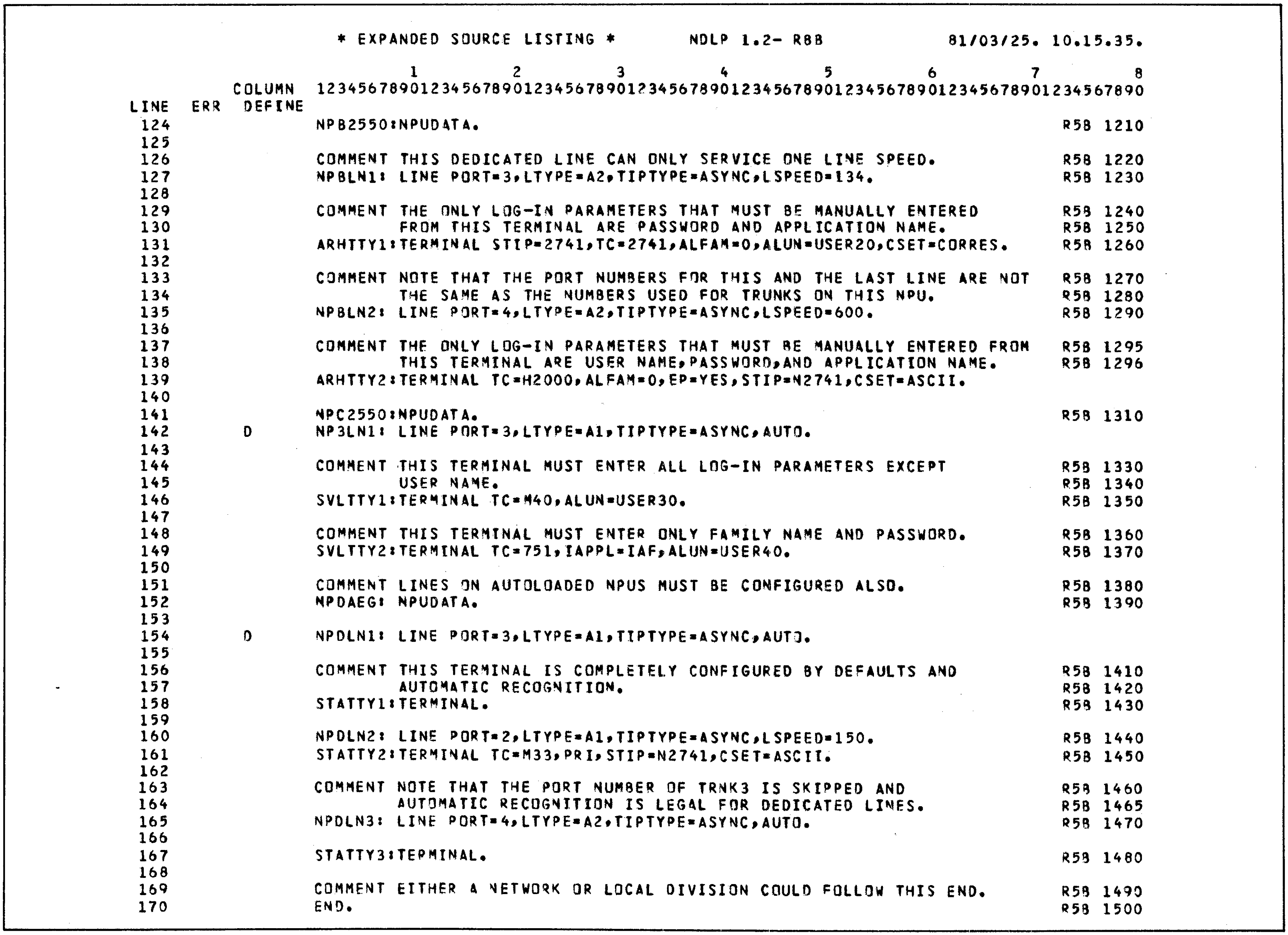
```
                        * EXPANDED SOURCE LISTING *       NDLP 1.2- R83            81/03/25. 10.15.35.

                                      1         2         3         4         5         6         7         8
                COLUMN  12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE   ERR   DEFINE
 32
 33                     COMMENT THE FOLLOWING IS A 217 CONSOLE, CONFIGURED BY AUTO-             R58   620
 34                             MATIC RECOGNITION AND DEFAULTS,AND REQUIRING A COMPLETE         R58   630
 35                             MANUAL LOG-IN.                                                  R58   640
 36                     UT200A1:TERMINAL TC=200UT,STIP=M4A,CSET=BCD.                            R58   650
 37
 38                     COMMENT THE 222 AND THE 224 ARE CONFIGURED FOR AUTOMATIC LOG-IN TO RBF.
 39                     COMMENT ALTHOUGH THEY HAVE THE SAME USER NAME AND FAMILY NAME, THIS IS
 40                             NOT REQUIRED.
 41                     COMMENT THE NETWORK WILL USE THE USER NAME AND FAMILY NAME SUPPLIED
 42                             BY THE TERMINAL OPERATOR OF THE OWNING CONSOLE.
 43                     UT200A2:TERMINAL DT=CR,TC=200UT,IAPPL=RBF,ALFAM=0,                      R58   680
 44                             ALUN=USER7,OCTERM=UT200A1,STIP=M4A,CSET=BCD.                    R58   685
 45                     UT200A3:TERMINAL DT=LP,TC=200UT,IAPPL=RBF,ALFAM=0,                      R58   690
 46                             ALUN=USER7,OCTERM=UT200A1,STIP=M4A,CSET=BCD.                    R58   695
 47
 48                     COMMENT THE NEXT 3 STATEMENTS CONFIGURE A 731 STATION EMULATING A       R58   700
 49                             200 USER TERMINAL,SO THE PASSIVE DEVICES HAVE THAT TC.          R58   710
 50                     COMMENT PASSIVE DEVICES AT MODE 4 STATIONS MUST BE CONFIGURED TO        R58   720
 51                             EMULATE 200 USER TERMINAL PASSIVE DEVICES.                      R58   730
 52                     UT200B4:TERMINAL DT=CON,TC=734,STIP=M4A,CSET=ASCII.                     R58   740
 53
 54                     COMMENT THE FOLLOWING DEVICES ARE CONFIGURED FOR AUTOMATIC LOG-         R58   750
 55                             IN TO A PROGRAM THAT MIGHT NOT BE AVAILABLE, BUT THEY ARE       R58   760
 56                             ENABLED BECAUSE THIS LINE IS AN AUTOMATIC RECOGNITION ONE AND   R58   765
 57                             STIP IS OMITTED.                                                R58   770
 58                     COMMENT THESE HAVE DIFFERENT USER NAMES, ALTHOUGH RMV2 IS NOT CON-      R58   780
 59                             FIGURED WITH UID DECLARED.                                      R58   790
 60                     UT200B5:TERMINAL DT=CR,TC=200UT,IAPPL=RMV2,ALFAM=0,                     R58   800
 61                             ALUN=USER9,OCTERM=UT200B4,STIP=M4A,CSET=ASCII.                  R58   810
 62                     UT200B6:TERMINAL DT=LP,TC=200UT,IAPPL=RMV2,ALFAM=0,                     R58   820
 63                             ALUN=USER8,OCTERM=UT200B4,STIP=M4A,CSET=ASCII.                  R58   830
 64
 65                     COMMENT THE FOLLOWING CAN BE ANY MODE 4C TERMINAL THAT DIALS IN.        R58   840
 66                     BATCH3:  TERMINAL TC=711,CSET=ASCII,STIP=M4C.
 67
 68                     NPALN2: LINE PORT=3,LTYPE=S3,TIPTYPE=HASP.                              R58   860
 69
 70                     COMMENT THE CONSOLE FOR THE MULTILEAVING STATION IS GIVEN               R58   870
 71                             PRIORITY FOR INPUT AND OUTPUT OVER OTHER NONPRIORITY DEVICES.   R58   880
 72                     HASPCON:TERMINAL TC=HASP,PRI.                                           R58   890
 73
 74                     COMMENT THE FIRST 2 CARD READERS ARE CONFIGURED WITH AN OWNING          R58   900
 75                             CONSOLE LOG-IN PARAMETER SET AND CONNECTED TO RBF.              R58   910
 76                     HASPCR1:TERMINAL DT=CR,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,            R58   920
 77                              OCTERM=HASPCON,DO=1.                                           R58   921
```

Figure 7-8. Expanded Source Listing, Local Division (Sheet 2 of 4)



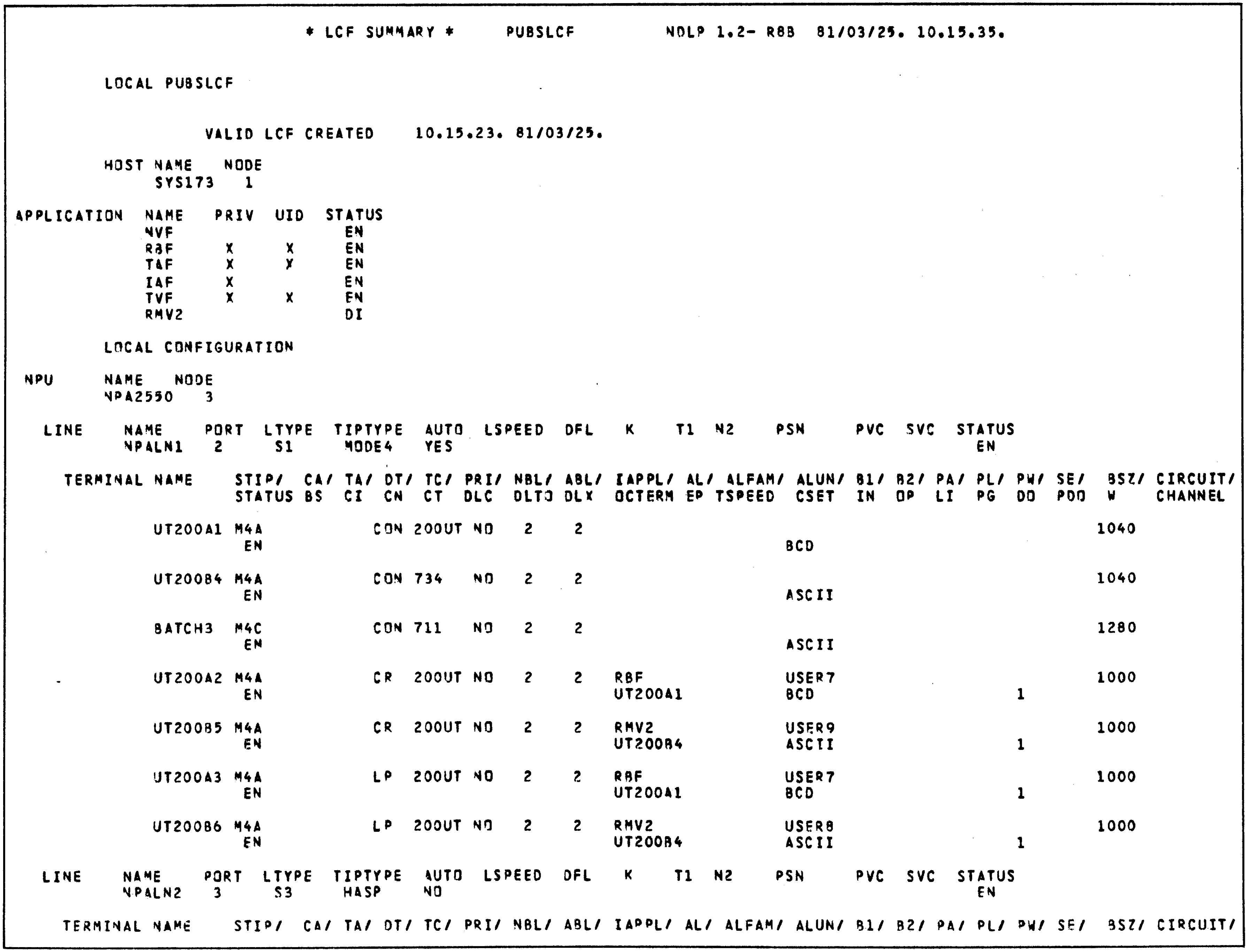
```
                      * EXPANDED SOURCE LISTING *       NDLP 1.2- R8B            81/03/25. 10.15.35.

                             1         2         3         4         5         6         7         8
           COLUMN   12345678901234567890123456789012345678901234567890123456789012345678901234567890
 LINE  ERR  DEFINE
   78
   79               COMMENT ALTHOUGH THE CARD READERS HAVE THE SAME USER NAME AND  FAMILY
   80                       NAME, THIS IS NOT REQUIRED TO CONNECT TO RBF.
   81               COMMENT THE NETWORK WILL USE THE USER NAME AND FAMILY NAME SUPPLIED
   82                       BY THE TERMINAL OPERATOR OF THE OWNING CONSOLE.
   83               HASPCR2:TERMINAL DT=CR,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,            R5B  960
   84                        OCTERM=HASPCON,DO=2.                                           R5B  961
   85
   86               COMMENT THE NEXT CARD READER IS CONNECTED TO A DIFFERENT PROGRAM,       R5B  970
   87                       WITH DIFFERENT AUTOMATIC LOG-IN PARAM-                          R5B  980
   88                       ETERS FROM THE OTHERS AT THE STATION.                           R5B  990
   89               HASPCR3:TERMINAL DT=CR,TC=HASP,IAPPL=RMV2,ALFAM=0,ALUN=USER8,           R5B 1000
   90                       OCTERM=HASPCON,DO=3.                                            R5B 1010
   91
   92               COMMENT THE FIRST PRINTER HAS A DIFFERENT INITIAL APPLICATION           R5B 1020
   93                       AND AUTOMATIC LOG-IN PARAMETERS FROM THE SECOND ONE.            R5B 1030
   94               HASPLP1:TERMINAL DT=LP,TC=HASP,IAPPL=RMV2,ALFAM=0,ALUN=USER50,          R5B 1040
   95                       OCTERM=HASPCON,DO=4.                                            R5B 1045
   96
   97               HASPLP2:TERMINAL DT=LP,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,            R5B 1050
   98                       OCTERM=HASPCON,DO=2.                                            R5B 1060
   99
  100               COMMENT THE CARD PUNCH AND THE PLOTTER HAVE THE STATION CONSOLE         R5B 1070
  101                       AS OWNING CONSOLE AND HAVE PDO DIFFERENT FROM DO.               R5B 1080
  102               HASPCP1:TERMINAL DT=CP,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,            R5B 1090
  103                       OCTERM=HASPCON,DO=1.                                            R5B 1095
  104               HASPPL1:TERMINAL DT=PL,TC=HASP,IAPPL=RBF,ALFAM=0,ALUN=USER4,            R5B 1100
  105                       OCTERM=HASPCON,DO=1,PDO=2.                                      R5B 1110
  106
  107               COMMENT THE NEXT LINE SUPPORTS TERMINALS WITH DIFFERENT LINE SPEEDS     R5B 1120
  108                       THRU AUTOMATIC RECOGNITION.                                     R5B 1130
  109       D       NPALN3: LINE PORT=4,LTYPE=A1,TIPTYPE=ASYNC,AUTO.
  110
  111               COMMENT THE FIRST TERMINAL IS CONNECTED WITHOUT ANY MANUAL              R5B 1150
  112                       LOG-IN REQUIRED.                                                R5B 1160
  113               COMMENT ALTHOUGH THIS IS A CONSOLE AND HAS THE LOG-IN PARAMETERS OF     R5B 1161
  114                       USER8, IT CANNOT BE THE OWNING CONSOLE FOR HASPCR3 ON           R5B 1162
  115                       ANOTHER LINE.                                                   R5B 1163
  116  W            HORTTY1:TERMINAL STIP=N2741,IAPPL=IAF,ALFAM=0,ALUN=USER8,TSPEED=110.    R5B 1170
  117
  118               COMMENT THE SECOND TERMINAL REQUIRES MANUAL LOG-IN ONLY OF FAMILY       R5B 1180
  119                       NAME, USER NAME, AND PASSWORD.                                  R5B 1190
  120               COMMENT THIS SECOND TERMINAL CAN BE CONFIGURED ON THIS LINE AS ENABLED  R5B 1195
  121                       ONLY BECAUSE THE LINE IS CONFIGURED FOR AUTOMATIC RECOGNITION.  R5B 1196
  122  W            HORTTY2:TERMINAL STIP=2741,IAPPL=IAF,SE=YES,TSPEED=134.
  123
```

Figure 7-8. Expanded Source Listing, Local Division (Sheet 3 of 4)



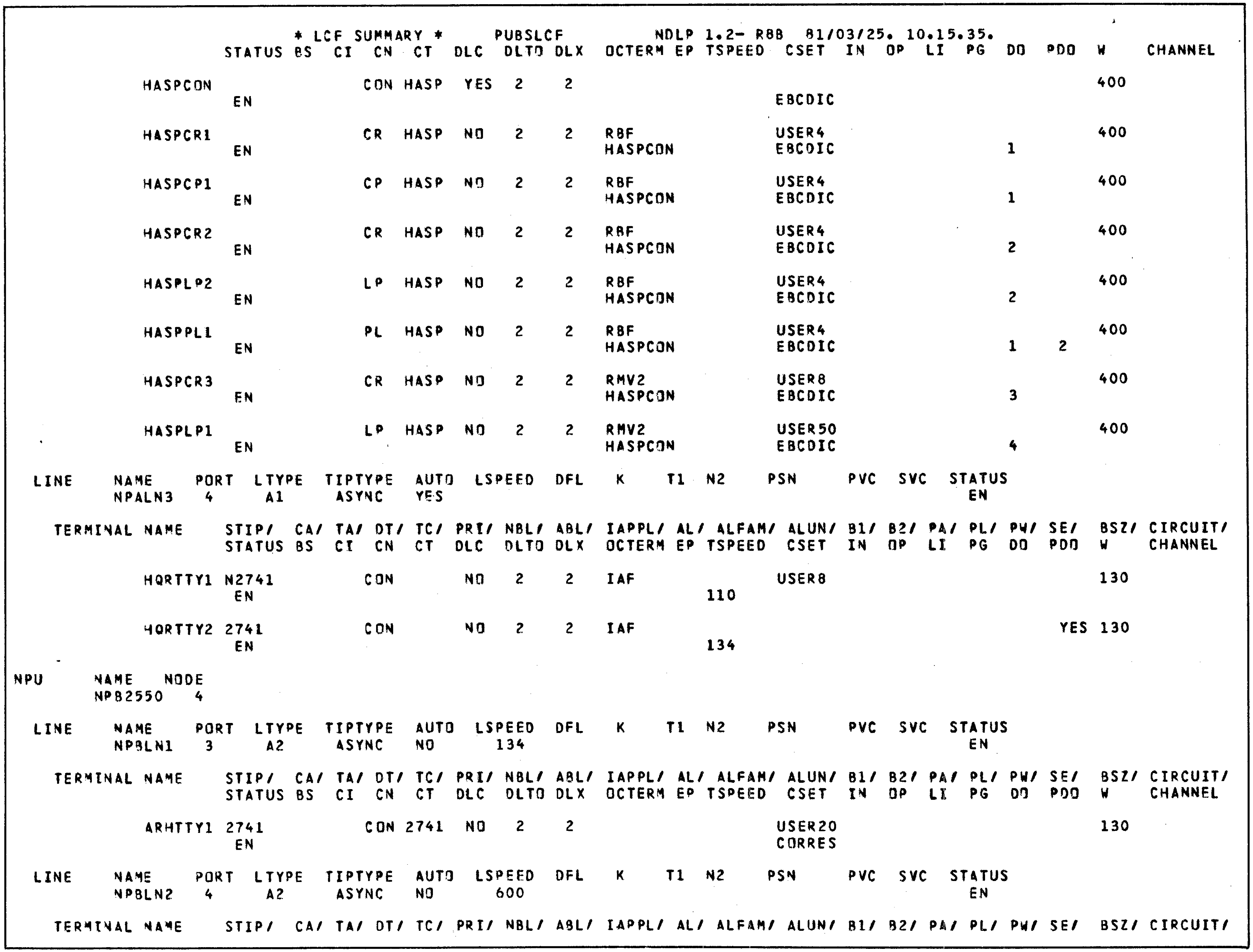
```
                      * EXPANDED SOURCE LISTING *       NDLP 1.2- R8B          81/03/25. 10.15.35.

                                   1         2         3         4         5         6         7         8
                COLUMN   12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE   ERR   DEFINE
 124                     NPB2550:NPUDATA.                                                        R5B 1210
 125
 126                     COMMENT THIS DEDICATED LINE CAN ONLY SERVICE ONE LINE SPEED.            R5B 1220
 127                     NPBLN1: LINE PORT=3,LTYPE=A2,TIPTYPE=ASYNC,LSPEED=134.                  R5B 1230
 128
 129                     COMMENT THE ONLY LOG-IN PARAMETERS THAT MUST BE MANUALLY ENTERED        R5B 1240
 130                             FROM THIS TERMINAL ARE PASSWORD AND APPLICATION NAME.           R5B 1250
 131                     ARHTTY1:TERMINAL STIP=2741,TC=2741,ALFAM=0,ALUN=USER20,CSET=CORRES.     R5B 1260
 132
 133                     COMMENT NOTE THAT THE PORT NUMBERS FOR THIS AND THE LAST LINE ARE NOT   R5B 1270
 134                             THE SAME AS THE NUMBERS USED FOR TRUNKS ON THIS NPU.            R5B 1280
 135                     NPBLN2: LINE PORT=4,LTYPE=A2,TIPTYPE=ASYNC,LSPEED=600.                  R5B 1290
 136
 137                     COMMENT THE ONLY LOG-IN PARAMETERS THAT MUST BE MANUALLY ENTERED FROM   R5B 1295
 138                             THIS TERMINAL ARE USER NAME,PASSWORD,AND APPLICATION NAME.      R5B 1296
 139                     ARHTTY2:TERMINAL TC=H2000,ALFAM=0,EP=YES,STIP=N2741,CSET=ASCII.
 140
 141                     NPC2550:NPUDATA.                                                        R5B 1310
 142          D          NP3LN1: LINE PORT=3,LTYPE=A1,TIPTYPE=ASYNC,AUTO.
 143
 144                     COMMENT THIS TERMINAL MUST ENTER ALL LOG-IN PARAMETERS EXCEPT           R5B 1330
 145                             USER NAME.                                                      R5B 1340
 146                     SVLTTY1:TERMINAL TC=M40,ALUN=USER30.                                    R5B 1350
 147
 148                     COMMENT THIS TERMINAL MUST ENTER ONLY FAMILY NAME AND PASSWORD.         R5B 1360
 149                     SVLTTY2:TERMINAL TC=751,IAPPL=IAF,ALUN=USER40.                          R5B 1370
 150
 151                     COMMENT LINES ON AUTOLOADED NPUS MUST BE CONFIGURED ALSO.               R5B 1380
 152                     NPDAEG: NPUDATA.                                                        R5B 1390
 153
 154          D          NPDLN1: LINE PORT=3,LTYPE=A1,TIPTYPE=ASYNC,AUTO.
 155
 156                     COMMENT THIS TERMINAL IS COMPLETELY CONFIGURED BY DEFAULTS AND          R5B 1410
 157                             AUTOMATIC RECOGNITION.                                          R5B 1420
 158                     STATTY1:TERMINAL.                                                       R5B 1430
 159
 160                     NPDLN2: LINE PORT=2,LTYPE=A1,TIPTYPE=ASYNC,LSPEED=150.                  R5B 1440
 161                     STATTY2:TERMINAL TC=M33,PRI,STIP=N2741,CSET=ASCII.                      R5B 1450
 162
 163                     COMMENT NOTE THAT THE PORT NUMBER OF TRNK3 IS SKIPPED AND               R5B 1460
 164                             AUTOMATIC RECOGNITION IS LEGAL FOR DEDICATED LINES.             R5B 1465
 165                     NPDLN3: LINE PORT=4,LTYPE=A2,TIPTYPE=ASYNC,AUTO.                        R5B 1470
 166
 167                     STATTY3:TERMINAL.                                                       R5B 1480
 168
 169                     COMMENT EITHER A NETWORK OR LOCAL DIVISION COULD FOLLOW THIS END.       R5B 1490
 170                     END.                                                                    R5B 1500
```

Figure 7-8. Expanded Source Listing, Local Division (Sheet 4 of 4)



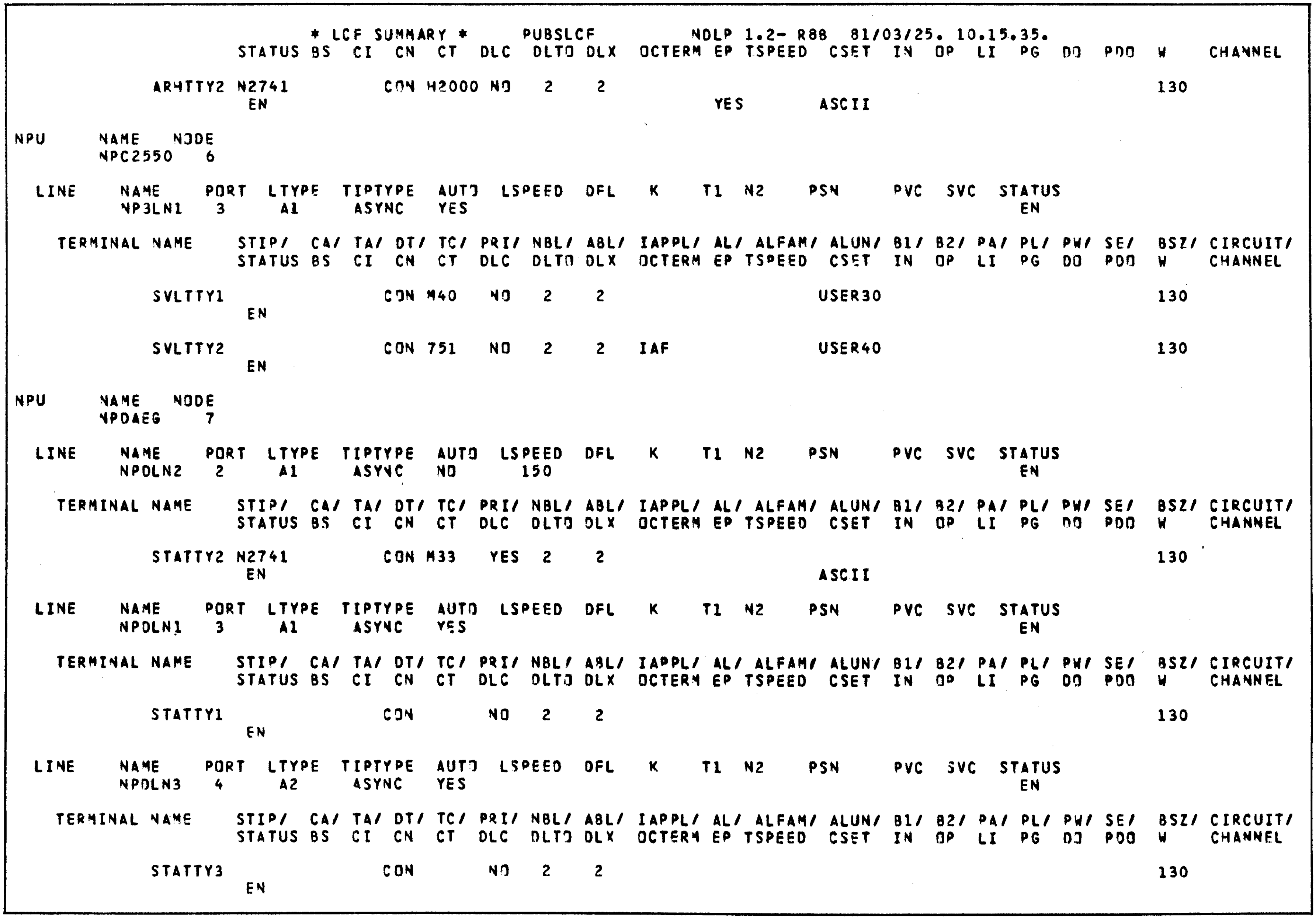
```
                             * LCF SUMMARY *     PUBSLCF         NDLP 1.2- R8B  81/03/25. 10.15.35.

         LOCAL PUBSLCF

                   VALID LCF CREATED    10.15.23. 81/03/25.

         HOST NAME   NODE
              SYS173   1

APPLICATION  NAME   PRIV  UID  STATUS
             NVF                 EN
             RBF     X     X     EN
             TAF     X     X     EN
             IAF     X           EN
             TVF     X     X     EN
             RMV2                DI

         LOCAL CONFIGURATION

 NPU     NAME   NODE
         NPA2550   3

   LINE    NAME    PORT  LTYPE  TIPTYPE  AUTO  LSPEED  DFL   K    T1  N2    PSN     PVC  SVC  STATUS
           NPALN1   2     S1     MODE4   YES                                                    EN

     TERMINAL NAME    STIP/  CA/ TA/ DT/ TC/ PRI/ NBL/ ABL/ IAPPL/ AL/ ALFAM/ ALUN/ B1/ B2/ PA/ PL/ PW/ SE/  BSZ/ CIRCUIT/
                      STATUS BS  CI  CN  CT  DLC  DLTO DLX  OCTERM EP TSPEED  CSET  IN  OP  LI  PG  DO  PDO  W    CHANNEL

              UT200A1 M4A           CON 200UT NO   2    2                                                    1040
                       EN                                                    BCD

              UT200B4 M4A           CON 734   NO   2    2                                                    1040
                       EN                                                    ASCII

              BATCH3  M4C           CON 711   NO   2    2                                                    1280
                       EN                                                    ASCII

              UT200A2 M4A           CR  200UT NO   2    2   RBF              USER7                           1000
                       EN                                   UT200A1          BCD                    1

              UT200B5 M4A           CR  200UT NO   2    2   RMV2             USER9                           1000
                       EN                                   UT200B4          ASCII                  1

              UT200A3 M4A           LP  200UT NO   2    2   RBF              USER7                           1000
                       EN                                   UT200A1          BCD                    1

              UT200B6 M4A           LP  200UT NO   2    2   RMV2             USER8                           1000
                       EN                                   UT200B4          ASCII                  1
   LINE    NAME    PORT  LTYPE  TIPTYPE  AUTO  LSPEED  DFL   K    T1  N2    PSN     PVC  SVC  STATUS
           NPALN2   3     S3     HASP    NO                                                     EN

     TERMINAL NAME    STIP/  CA/ TA/ DT/ TC/ PRI/ NBL/ ABL/ IAPPL/ AL/ ALFAM/ ALUN/ B1/ B2/ PA/ PL/ PW/ SE/  BSZ/ CIRCUIT/
```

Figure 7-9. File Summary Listing, Local Division (Sheet 1 of 3)



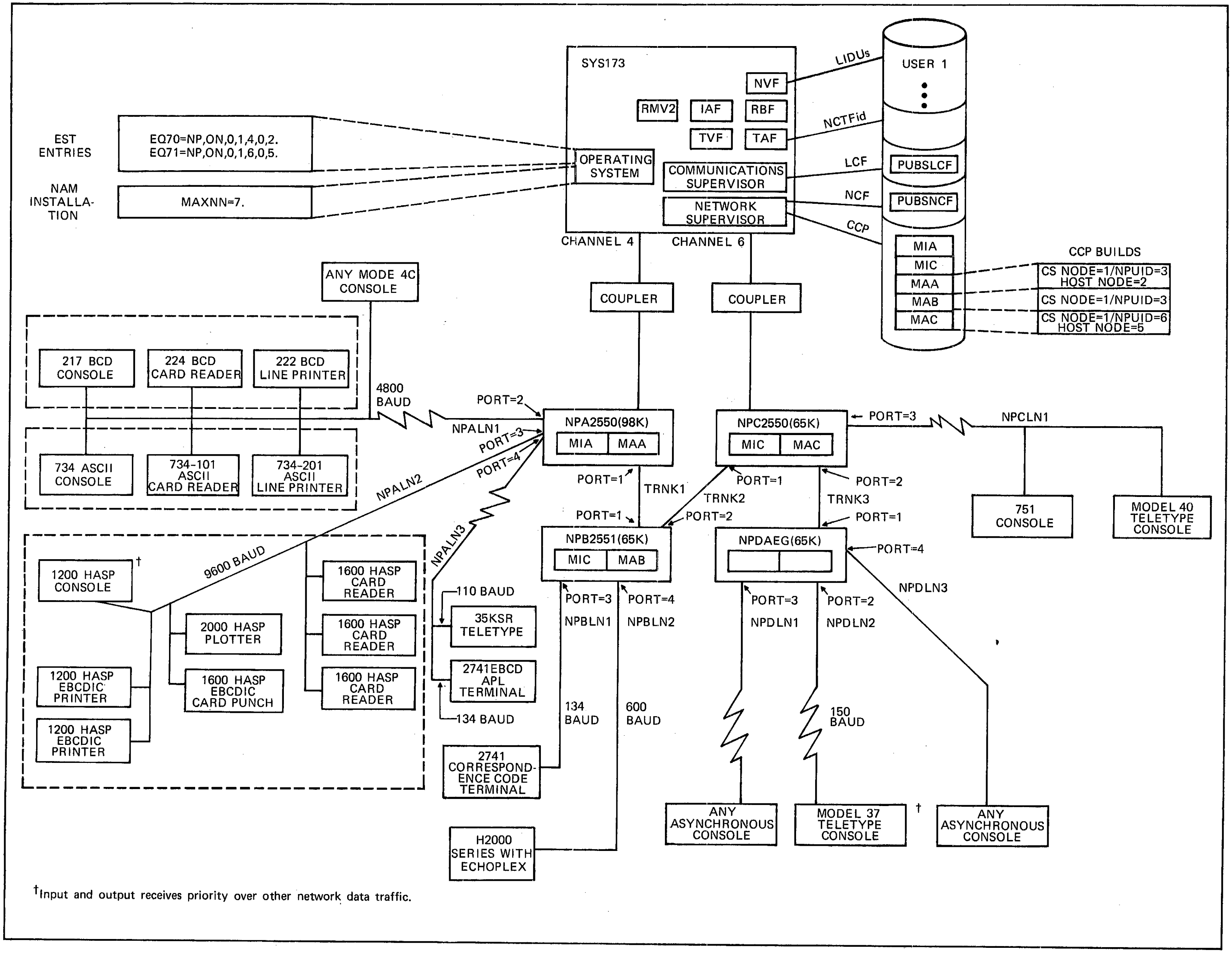
```
                             * LCF SUMMARY *     PUBSLCF         NDLP 1.2- R8B  81/03/25. 10.15.35.
                     STATUS BS   CI  CN  CT  DLC  DLTO DLX  OCTERM EP TSPEED   CSET  IN  OP  LI  PG  DO  PDO  W    CHANNEL

             HASPCON                CON HASP  YES  2    2                                                     400
                      EN                                                      EBCDIC

             HASPCR1                CR  HASP  NO   2    2   RBF               USER4                           400
                      EN                                    HASPCON           EBCDIC                 1

             HASPCP1                CP  HASP  NO   2    2   RBF               USER4                           400
                      EN                                    HASPCON           EBCDIC                 1

             HASPCR2                CR  HASP  NO   2    2   RBF               USER4                           400
                      EN                                    HASPCON           EBCDIC                 2

             HASPLP2                LP  HASP  NO   2    2   RBF               USER4                           400
                      EN                                    HASPCON           EBCDIC                 2

             HASPPL1                PL  HASP  NO   2    2   RBF               USER4                           400
                      EN                                    HASPCON           EBCDIC                 1     2

             HASPCR3                CR  HASP  NO   2    2   RMV2              USER8                           400
                      EN                                    HASPCON           EBCDIC                 3

             HASPLP1                LP  HASP  NO   2    2   RMV2              USER50                          400
                      EN                                    HASPCON           EBCDIC                 4

  LINE    NAME    PORT  LTYPE   TIPTYPE  AUTO  LSPEED  DFL   K    T1  N2     PSN     PVC  SVC  STATUS
          NPALN3  4     A1     ASYNC   YES                                                     EN

    TERMINAL NAME    STIP/  CA/  TA/ DT/ TC/ PRI/ NBL/ ABL/ IAPPL/ AL/ ALFAM/  ALUN/ B1/ B2/ PA/ PL/ PW/ SE/  BSZ/ CIRCUIT/
                     STATUS BS   CI  CN  CT  DLC  DLTO DLX  OCTERM EP TSPEED   CSET  IN  OP  LI  PG  DO  PDO  W    CHANNEL

             HQRTTY1 N2741          CON       NO   2    2   IAF               USER8                           130
                      EN                                              110

             HQRTTY2 2741           CON       NO   2    2   IAF                                           YES 130
                      EN                                              134

NPU     NAME   NODE
        NPB2550   4

  LINE    NAME    PORT  LTYPE   TIPTYPE  AUTO  LSPEED  DFL   K    T1  N2     PSN     PVC  SVC  STATUS
          NPBLN1  3     A2     ASYNC   NO      134                                             EN

    TERMINAL NAME    STIP/  CA/  TA/ DT/ TC/ PRI/ NBL/ ABL/ IAPPL/ AL/ ALFAM/  ALUN/ B1/ B2/ PA/ PL/ PW/ SE/  BSZ/ CIRCUIT/
                     STATUS BS   CI  CN  CT  DLC  DLTO DLX  OCTERM EP TSPEED   CSET  IN  OP  LI  PG  DO  PDO  W    CHANNEL

             ARHTTY1 2741           CON 2741  NO   2    2                     USER20                          130
                      EN                                                      CORRES

  LINE    NAME    PORT  LTYPE   TIPTYPE  AUTO  LSPEED  DFL   K    T1  N2     PSN     PVC  SVC  STATUS
          NPBLN2  4     A2     ASYNC   NO      600                                             EN

    TERMINAL NAME    STIP/  CA/  TA/ DT/ TC/ PRI/ NBL/ ABL/ IAPPL/ AL/ ALFAM/  ALUN/ B1/ B2/ PA/ PL/ PW/ SE/  BSZ/ CIRCUIT/
```

Figure 7-9. File Summary Listing, Local Division (Sheet 2 of 3)



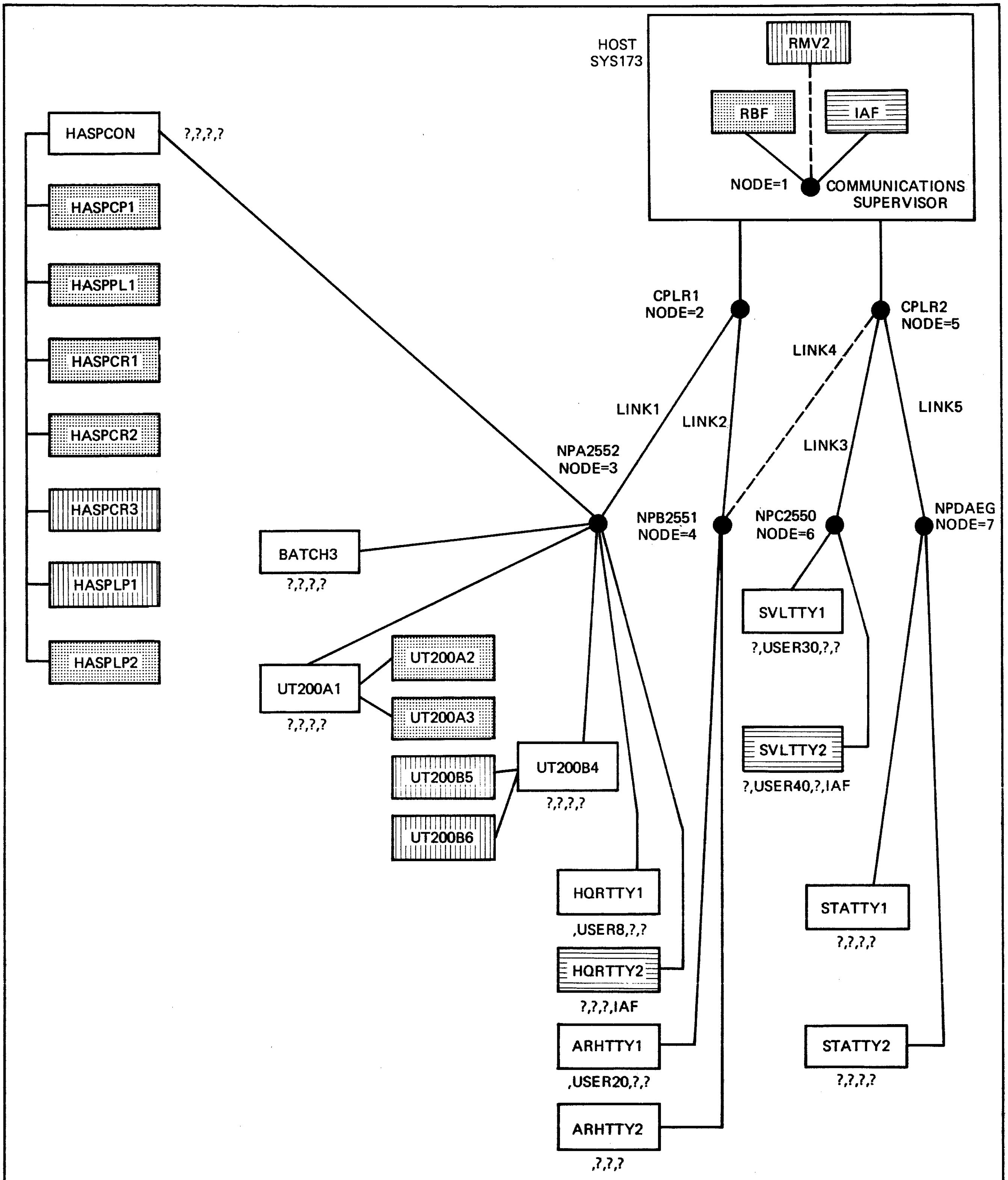
```
                            * LCF SUMMARY *     PUBSLCF         NDLP 1.2- R88  81/03/25. 10.15.35.
                     STATUS BS  CI  CN  CT  DLC  DLTO DLX  OCTERM EP TSPEED  CSET  IN  OP  LI  PG  DO  PDO  W    CHANNEL

             ARHTTY2 N2741         CON H2000 NO   2    2                                                    130
                      EN                                          YES       ASCII

NPU     NAME   NODE
        NPC2550   6

  LINE    NAME    PORT  LTYPE  TIPTYPE  AUTO  LSPEED  DFL   K    T1  N2    PSN     PVC  SVC  STATUS
          NP3LN1   3     A1     ASYNC   YES                                                    EN

    TERMINAL NAME    STIP/  CA/ TA/ DT/ TC/ PRI/ NBL/ ABL/ IAPPL/ AL/ ALFAM/ ALUN/ B1/ B2/ PA/ PL/ PW/ SE/  BSZ/ CIRCUIT/
                     STATUS BS  CI  CN  CT  DLC  DLTO DLX  OCTERM EP TSPEED  CSET  IN  OP  LI  PG  DO  PDO  W    CHANNEL

             SVLTTY1               CON M40   NO   2    2                    USER30                          130
                      EN

             SVLTTY2               CON 751   NO   2    2   IAF              USER40                          130
                      EN

NPU     NAME   NODE
        NPDAEG    7

  LINE    NAME    PORT  LTYPE  TIPTYPE  AUTO  LSPEED  DFL   K    T1  N2    PSN     PVC  SVC  STATUS
          NPDLN2   2     A1     ASYNC   NO      150                                            EN

    TERMINAL NAME    STIP/  CA/ TA/ DT/ TC/ PRI/ NBL/ ABL/ IAPPL/ AL/ ALFAM/ ALUN/ B1/ B2/ PA/ PL/ PW/ SE/  BSZ/ CIRCUIT/
                     STATUS BS  CI  CN  CT  DLC  DLTO DLX  OCTERM EP TSPEED  CSET  IN  OP  LI  PG  DO  PDO  W    CHANNEL

             STATTY2 N2741         CON M33   YES  2    2                                                    130
                      EN                                                    ASCII

  LINE    NAME    PORT  LTYPE  TIPTYPE  AUTO  LSPEED  DFL   K    T1  N2    PSN     PVC  SVC  STATUS
          NPDLN1   3     A1     ASYNC   YES                                                    EN

    TERMINAL NAME    STIP/  CA/ TA/ DT/ TC/ PRI/ NBL/ ABL/ IAPPL/ AL/ ALFAM/ ALUN/ B1/ B2/ PA/ PL/ PW/ SE/  BSZ/ CIRCUIT/
                     STATUS BS  CI  CN  CT  DLC  DLTO DLX  OCTERM EP TSPEED  CSET  IN  OP  LI  PG  DO  PDO  W    CHANNEL

             STATTY1               CON       NO   2    2                                                    130
                      EN

  LINE    NAME    PORT  LTYPE  TIPTYPE  AUTO  LSPEED  DFL   K    T1  N2    PSN     PVC  SVC  STATUS
          NPDLN3   4     A2     ASYNC   YES                                                    EN

    TERMINAL NAME    STIP/  CA/ TA/ DT/ TC/ PRI/ NBL/ ABL/ IAPPL/ AL/ ALFAM/ ALUN/ B1/ B2/ PA/ PL/ PW/ SE/  BSZ/ CIRCUIT/
                     STATUS BS  CI  CN  CT  DLC  DLTO DLX  OCTERM EP TSPEED  CSET  IN  OP  LI  PG  DO  PDO  W    CHANNEL

             STATTY3               CON       NO   2    2                                                    130
                      EN
```

Figure 7-9. File Summary Listing, Local Division (Sheet 3 of 3)



†Input and output receives priority over other network data traffic.

Figure 7-10. Physical Configuration of Network PUBSNET



?,?,?,? indicates log-in parameters required in an abbreviated sequence of family name, user name, password, application name

Figure 7-11. Logical Configuration of NETWORK PUBSNET

CODED CHARACTER DATA INPUT, OUTPUT, AND CENTRAL MEMORY REPRESENTATION

This appendix describes the code and character sets used by host computer operating system local batch device drivers, magnetic tape drivers, and terminal communication products. Some software products assume that certain graphic or control characters are associated with specific binary code values for collating or syntax processing purposes. This appendix does not describe those associations for all products.

All references within this manual to the ASCII character set or the ASCII code set refer to the character set and code set defined in the American National Standard Code for Information Interchange (ASCII, ANSI Standard X3.4-1977). References in this manual to the ASCII character set do not necessarily apply to the ASCII code set.

CHARACTER SETS AND CODE SETS

A character set differs from a code set. A character set is a set of graphic and/or control characters. A code set is a set of codes used to represent each character within a character set. Characters exist outside the computer system and communication network; codes are received, stored, retrieved, and transmitted within the computer system and network.

GRAPHIC AND CONTROL CHARACTERS

A graphic character can be displayed at a terminal or printed by a line printer. Examples of graphic characters are the characters A through Z, a blank, and the digits 0 through 9. A control character initiates, modifies, or stops a control operation. An example of a control character is the backspace character, which moves the terminal carriage or cursor back one space. Although a control character is not a graphic character, some terminals can produce a graphic representation when they receive a control character.

CODED AND BINARY CHARACTER DATA

Character codes can be interpreted as coded character data or as binary character data. Coded character data is converted from one code set representation to another as it enters or leaves the computer system; for example, data received from a terminal or sent to a magnetic tape unit is converted. Binary character data is not converted as it enters or leaves the system. Character codes are not converted when moved within the system; for example, data transferred to or from mass storage is not converted.

The distinction between coded character data and binary character data is important when reading or punching cards and when reading or writing magnetic tape. Only coded character data can be properly reproduced as characters on a line printer. Only binary character data can properly represent characters on a punched card when the data cannot be stored as display code.

The distinction between binary character data and characters represented by binary data (such as peripheral equipment instruction codes) is also important. Only such binary noncharacter data can properly reproduce characters on a plotter.

FORMATTED AND UNFORMATTED CHARACTER DATA

Character codes can be interpreted by a product as formatted character data or as unformatted character data. Formatted data can be stored or retrieved by a product in the form of the codes described for coded character data in the remainder of this appendix, or formatted data can be altered to another form during storage or retrieval; for example, 1 can be stored as a character code or as an integer value. Treatment of unformatted data by a product includes both coded character data and binary character data as described in this appendix.

NETWORK OPERATING SYSTEM

The Network Operating System (NOS) supports the following character sets:

- CDC graphic 64-character set
- CDC graphic 63-character set
- ASCII graphic 64-character set
- ASCII graphic 63-character set
- ASCII graphic 95-character set
- ASCII 128-character graphic and control set

Each installation must select either a 64-character set or a 63-character set. The differences between the codes of a 63-character set and the codes of a 64-character set are described under Character Set Anomalies. Any reference in this appendix to a 64-character set implies either a 63- or 64-character set unless otherwise stated.

To represent its six listed character sets in central memory, NOS supports the following code sets:

- 6-bit display code
- 12-bit ASCII code
- 6/12-bit display code

The 6-bit display code is a set of 6-bit codes from 00_8 to 77_8.

The 12-bit ASCII code is the ASCII 7-bit code (as defined by ANSI Standard X3.4-1977) right-justified in a 12-bit byte. Assuming that the bits are numbered from the right starting with 0, bits 0 through 6 contain the ASCII code, bits 7 through 10 contain zeros, and bit 11 distinguishes the 12-bit ASCII 0000_8 code from the end-of-line byte. The 12-bit codes are 0001_8 through 0177_8 and 4000_8.

The 6/12-bit display code is a combination of 6-bit codes and 12-bit codes. The 6-bit codes are 00_8 through 77_8, excluding 74_8 and 76_8. (The interpretation of the 00_8 and 63_8 codes is described under Character Set Anomalies later in this appendix.) The 12-bit codes begin with either 74_8 or 76_8 and are followed by a 6-bit code. Thus, 74_8 and 76_8 are considered escape codes and are never used as 6-bit codes within the 6/12-bit display code set. The 12-bit codes are 7401_8, 7402_8, 7404_8, 7407_8, and 7601_8 through 7677_8. All other 12-bit codes ($74xx_8$ and 7600_8) are undefined.

CHARACTER SET ANOMALIES

The operating system input/output software and some products interpret two codes differently when the installation selects a 63-character set rather than a 64-character set. If an installation uses a 63-character set, the colon graphic character is always represented by a 63_8 code, display code 00_8 is undefined (it has no associated graphic or punched card code), and the % graphic does not exist.

If the installation uses a 64-character set, output of a 7404_8 6/12-bit display code or a 00_8 display code produces a colon. A colon can be input only as a 7404_8 6/12-bit display code. The use of undefined 6/12-bit display codes in output files produces unpredictable results and should be avoided.

Two consecutive 00_8 codes can be confused with an end-of-line byte and should be avoided.

CHARACTER SET TABLES

The character set tables A-1 and A-2 are designed so that the user can find the character represented by a code (such as in a dump) or find the code that represents a character. To find the character represented by a code, the user looks up the code in the column listing the appropriate code set and then finds the character on that line in the column listing the appropriate character set. To find the code that represents a character, the user looks up the character and then finds the code on the same line in the appropriate column.

Conversational Terminal Users

Table A-1 shows the character sets and code sets available to an Interactive Facility (IAF) user at an ASCII code terminal using an ASCII character set. Table A-9 (later in this appendix) shows the octal and hexadecimal 7-bit ASCII code for each ASCII character, and can be used to convert codes from octal to hexadecimal. (Under NOS using network product software, certain Terminal Interface Program commands require specification of an ASCII code.)

IAF Usage

When in normal time-sharing mode (specified by the IAF NORMAL command), IAF assumes the ASCII graphic 64-character set is used and translates all input and output to or from display code. When in ASCII time-sharing mode (specified by the IAF ASCII command), IAF assumes the ASCII 128-character set is used and translates all input and output to or from 6/12-bit display code.

The IAF user can convert a 6/12-bit code file to a 12-bit ASCII code file using the NOS FCOPY control statement. The resulting 12-bit ASCII file can be routed to a line printer but cannot be output through IAF.

IAF supports both character mode and transparent mode transmissions through the network. These transmission modes are described under Network Access Method Terminal Transmission Code Sets in this appendix. IAF treats character mode transmissions as coded character data; IAF converts these transmissions to or from either 6-bit or 6/12-bit display code. IAF treats transparent mode transmissions as binary character data; transparent mode communication between IAF and ASCII terminals using any parity setting occurs in the 12-bit ASCII code shown in table A-1.

Local Batch Users

Table A-2 lists the CDC graphic 64-character set, the ASCII graphic 64-character set, and the ASCII graphic 95-character set. This table also lists the code sets and card keypunch codes (026 and 029) that represent the characters.

The 64-character sets use display code as their code set; the 95-character set uses 12-bit ASCII code. The 95-character set is composed of all the characters in the ASCII 128-character set that can be printed at a line printer (refer to Line Printer Output). Only 12-bit ASCII code files can be printed using the ASCII graphic 95-character set. To print a 6/12-bit display code file (usually created in IAF ASCII mode), the user must convert the file to 12-bit ASCII code. To do this, the NOS FCOPY control statement must be issued. The 95-character set is represented by the 12-bit ASCII codes 0040_8 through 0176_8.

Line Printer Output

The batch character set printed depends on the print train used on the line printer to which the file is sent. The following are the print trains corresponding to each of the batch character sets:

Character Set	Print Train
CDC graphic 64-character set	596-1
ASCII graphic 64-character set	596-5
ASCII graphic 95-character set	596-6

The characters of the default 596-1 print train are listed in the table A-2 column labeled CDC Graphic (64-Character); the 596-5 print train characters are listed in the table A-2 column labeled ASCII Graphic (64-Character); and the 596-6 print train characters are listed in the table A-2 column labeled ASCII Graphic (95-Character).

If a transmission error occurs during the printing of a line, NOS prints the line again. The CDC graphic print train prints a concatenation symbol (↱) in the first printable column of a line containing errors. The ASCII print trains print an underline instead of the concatenation symbol.

If an unprintable character exists in a line (that is, a 12-bit ASCII code outside of the range 0040_8 through 0176_8), the number sign (#) appears in the first printable column of a print line and a space replaces the unprintable character.

Punched Card Input and Output

Under NOS, coded character data is exchanged with local batch card readers or card punches according to the translations shown in table A-2. As indicated in the table, additional card keypunch codes are available for input of the ASCII and CDC characters] and [. The 95-character set cannot be read or punched as coded character data.

Depending on an installation or deadstart option, NOS assumes an input deck has been punched either in 026 or 029 keypunch code (regardless of the character set in use). The alternate keypunch codes can be specified by a 26 or 29 punched in columns 79 and 80 of any 6/7/9 card or 7/8/9 card. The specified code translation remains in effect throughout the job unless it is reset by specification of the alternate code translation on a subsequent 6/7/9 card or 7/8/9 card.

NOS keypunch code translation can also be changed by a card containing a 5/7/9 punch in column 1. A blank (no punch) in column 2 indicates 026 conversion mode; a 9 punch in column 2 indicates 029 conversion mode. The conversion change remains in effect until another change card is encountered or the job ends.

The 5/7/9 card also allows literal input when 4/5/6/7/8/9 is punched in column 2. Literal input can be used to read 80-column binary character data within a punched card deck of coded character data.

Literal cards are stored with each column in a 12-bit byte (a row 12 punch is represented by a 1 in bit 11, row 11 by bit 10, row 0 by bit 9, and rows 1 through 9 by bits 8 through 0 of the byte), 16 central memory words per card. Literal input cards are read until a card identical to the previous 5/7/9 card (4/5/6/7/8/9 in column 2) is read. The next card can specify a new conversion mode.

Remote Batch Users

When card decks are read from remote batch devices, the ability to select alternate keypunch code translations depends upon the remote terminal equipment.

Remote batch terminal line printer, punched card, and plotter character set support is described under Input Deck Structure in the Remote Batch Facility (RBF) reference manual. RBF supports only character mode transmission to and from consoles through the network. Character mode is described under Network Access Method Terminal Transmission Code Sets in this appendix.

Magnetic Tape Users

Coded character data to be copied from mass storage to magnetic tape is assumed to be represented in display code. NOS converts the data to external BCD code when writing a coded 7-track tape and to ASCII or EBCDIC code (as specified on the tape assignment statement) when writing a coded 9-track tape.

Because only 63 characters can be represented in 7-track even parity, one of the 64 display codes is lost in conversion to and from external BCD code. Figure A-1 shows the differences in conversion that depend on which character set (63 or 64) the system uses. The ASCII character for the specified character code is shown in parentheses. The output arrow shows how the display code changes when it is written on tape in external BCD. The input arrow shows how the external BCD code changes when the tape is read and converted to display code.

63-Character Set

Display Code		External BCD		Display Code
00		16(%)		00
33(0)	Output →	12(0)	Input →	33(0)
63(:)		12(0)		33(0)

64-Character Set

Display Code		External BCD		Display Code
00(:)		12(0)		33(0)
33(0)	Output →	12(0)	Input →	33(0)
63(%)		16(%)		63(%)

Figure A-1. Magnetic Tape Code Conversions

Tables A-3 and A-4 show the character set conversions for nine-track tapes. Table A-3 lists the conversions to and from 7-bit ASCII character code and 6-bit display code. Table A-4 lists the conversions between 8-bit EBCDIC character code and 6-bit display code. Table A-5 shows the character set conversions between 6-bit external BCD and 6-bit display code for seven-track tapes.

If a lowercase ASCII or EBCDIC code is read from a 9-track coded tape, it is converted to its uppercase 6-bit display code equivalent. To read and write lowercase ASCII or EBCDIC characters, the user must assign the tape in binary mode and then convert the binary character data.

During binary character data transfers to or from 9-track magnetic tape, the 7-bit ASCII codes shown in table A-3 are read or written unchanged; the 8-bit hexadecimal EBCDIC codes shown in table A-4 also can be read or written unchanged. ASCII and EBCDIC codes cannot be read or written to 7-track magnetic tape as binary character data.

Tables A-6 and A-7 list the magnetic tape codes and their punch code equivalents on IBM host computers.

Two CDC utility products, FORM and the 8-Bit Subroutines, can be used to convert to and from EBCDIC data. Table A-7 contains the octal values of each EBCDIC code right-justified in a 12-bit byte with zero fill. This 12-bit EBCDIC code can also be produced using FORM and the 8-Bit Subroutines.

TABLE A-1. CONVERSATIONAL TERMINAL CHARACTER SETS

ASCII Graphic (64-Character Set)	ASCII Character (128-Character Set)	Octal 6-Bit Display Code	Octal 6/12-Bit Display Code[†]	Octal 12-Bit ASCII Code
: colon[††]		00[††]		
A	A	01	01	0101
B	B	02	02	0102
C	C	03	03	0103
D	D	04	04	0104
E	E	05	05	0105
F	F	06	06	0106
G	G	07	07	0107
H	H	10	10	0110
I	I	11	11	0111
J	J	12	12	0112
K	K	13	13	0113
L	L	14	14	0114
M	M	15	15	0115
N	N	16	16	0116
O	O	17	17	0117
P	P	20	20	0120
Q	Q	21	21	0121
R	R	22	22	0122
S	S	23	23	0123
T	T	24	24	0124
U	U	25	25	0125
V	V	26	26	0126
W	W	27	27	0127
X	X	30	30	0130
Y	Y	31	31	0131
Z	Z	32	32	0132
0	0	33	33	0060
1	1	34	34	0061
2	2	35	35	0062
3	3	36	36	0063
4	4	37	37	0064
5	5	40	40	0065
6	6	41	41	0066
7	7	42	42	0067
8	8	43	43	0070
9	9	44	44	0071
+ plus	+ plus	45	45	0053
- minus	- minus	46	46	0055
* asterisk	* asterisk	47	47	0052
/ slash	/ slash	50	50	0057
(l. paren.	(l. paren.	51	51	0050
) r. paren.	) r. paren.	52	52	0051
$ dollar	$ dollar	53	53	0044
= equal to	= equal to	54	54	0075
space	space	55	55	0040
, comma	, comma	56	56	0054
. period	. period	57	57	0056
# number	# number	60	60	0043
[l. bracket	[l. bracket	61	61	0133
] r. bracket	] r. bracket	62	62	0135
% percent[††]	% percent[††]	63[††]	63[††]	0045
" quote	" quote	64	64	0042
_ underline	_ underline	65	65	0137
! exclam.	! exclam.	66	66	0041
& ampersand	& ampersand	67	67	0046
' apostrophe	' apostrophe	70	70	0047
? question	? question	71	71	0077
< less than	< less than	72	72	0074
> grtr. than	> grtr. than	73	73	0076
@ coml. at		74		
\ rev. slant	\ rev. slant	75	75	0134
^ circumflex		76		
; semicolon	; semicolon	77	77	0073
	@ coml. at		7401	0100

ASCII Graphic (64-Character Set)	ASCII Character (128-Character Set)	Octal 6-Bit Display Code	Octal 6/12-Bit Display Code[†]	Octal 12-Bit ASCII Code
	^ circumflex		7402	0136
	: colon[††]		7404[††]	0072
	` grave accent		7407	0140
	a		7601	0141
	b		7602	0142
	c		7603	0143
	d		7604	0144
	e		7605	0145
	f		7606	0146
	g		7607	1047
	h		7610	0150
	i		7611	0151
	j		7612	0152
	k		7613	0153
	l		7614	0154
	m		7615	0155
	n		7616	0156
	o		7617	0157
	p		7620	0160
	q		7621	0161
	r		7622	0162
	s		7623	0163
	t		7624	0164
	u		7625	0165
	v		7626	0166
	w		7627	0167
	x		7630	0170
	y		7631	0171
	z		7632	0172
	{ left brace		7633	0173
	| vert. line		7634	0174
	} right brace		7635	0175
	~ tilde		7636	0176
	NUL		7640	4000
	SOH		7641	0001
	STX		7642	0002
	ETX		7643	0003
	EOT		7644	0004
	ENQ		7645	0005
	ACK		7646	0006
	BEL		7647	0007
	BS		7650	0010
	HT		7651	0011
	LF		7652	0012
	VT		7653	0013
	FF		7654	0014
	CR		7655	0015
	SO		7656	0016
	SI		7657	0017
	DEL		7637	0177
	DLE		7660	0020
	DC1		7661	0021
	DC2		7662	0022
	DC3		7663	0023
	DC4		7664	0024
	NAK		7665	0025
	SYN		7666	0026
	ETB		7667	0027
	CAN		7670	0030
	EM		7671	0031
	SUB		7672	0032
	ESC		7673	0033
	FS		7674	0034
	GS		7675	0035
	RS		7676	0036
	US		7677	0037

[†] Generally available only on NOS, or through BASIC on NOS/BE.

[††] The interpretation of this character or code depends on its context. Refer to Character Set Anomalies in the text.

TABLE A-2. LOCAL BATCH DEVICE CHARACTER SETS

CDC Graphic (64-Character Set)	ASCII Graphic (64-Character Set)	ASCII Graphic (95-Character Set)	Octal 6-Bit Display Code	Octal 6/12-Bit Display Code†	Octal 12-Bit ASCII Code	Card Keypunch Code	
						026	029
: colon††	: colon††		00††			8-2	8-2
A	A	A	01	01	0101	12-1	12-1
B	B	B	02	02	0102	12-2	12-2
C	C	C	03	03	0103	12-3	12-3
D	D	D	04	04	0104	12-4	12-4
E	E	E	05	05	0105	12-5	12-5
F	F	F	06	06	0106	12-6	12-6
G	G	G	07	07	0107	12-7	12-7
H	H	H	10	10	0110	12-8	12-8
I	I	I	11	11	0111	12-9	12-9
J	J	J	12	12	0112	11-1	11-1
K	K	K	13	13	0113	11-2	11-2
L	L	L	14	14	0114	11-3	11-3
M	M	M	15	15	0115	11-4	11-4
N	N	N	16	16	0116	11-5	11-5
O	O	O	17	17	0117	11-6	11-6
P	P	P	20	20	0120	11-7	11-7
Q	Q	Q	21	21	0121	11-8	11-8
R	R	R	22	22	0122	11-9	11-9
S	S	S	23	23	0123	0-2	0-2
T	T	T	24	24	0124	0-3	0-3
U	U	U	25	25	0125	0-4	0-4
V	V	V	26	26	0126	0-5	0-5
W	W	W	27	27	0127	0-6	0-6
X	X	X	30	30	0130	0-7	0-7
Y	Y	Y	31	31	0131	0-8	0-8
Z	Z	Z	32	32	0132	0-9	0-9
0	0	0	33	33	0060	0	0
1	1	1	34	34	0061	1	1
2	2	2	35	35	0062	2	2
3	3	3	36	36	0063	3	3
4	4	4	37	37	0064	4	4
5	5	5	40	40	0065	5	5
6	6	6	41	41	0066	6	6
7	7	7	42	42	0067	7	7
8	8	8	43	43	0070	8	8
9	9	9	44	44	0071	9	9
+ plus	+ plus	+ plus	45	45	0053	12	12-8-6
- minus	- minus	- minus	46	46	0055	11	11
* asterisk	* asterisk	* asterisk	47	47	0052	11-8-4	11-8-4
/ slash	/ slash	/ slash	50	50	0057	0-1	0-1
(left paren.	(left paren.	(left paren.	51	51	0050	0-8-4	12-8-5
) right paren.	) right paren.	) right paren.	52	52	0051	12-8-4	11-8-5
$ dollar	$ dollar	$ dollar	53	53	0044	11-8-3	11-8-3
= equal to	= equal to	= equal to	54	54	0075	8-3	8-6
space	space	space	55	55	0040	no punch	no punch
, comma	, comma	, comma	56	56	0054	0-8-3	0-8-3
. period	. period	. period	57	57	0056	12-8-3	12-8-3
≡ equivalence	# number	# number	60	60	0043	0-8-6	8-3
[left bracket	[left bracket	[l. bracket	61	61	0133	8-7	12-8-2 or 12-0†††
] right bracket	] right bracket	] r. bracket	62	62	0135	0-8-2	11-8-2 or 11-0†††
% percent††	% percent††	% percent††	63	63	0045	8-6	0-8-4

TABLE A-2. LOCAL BATCH DEVICE CHARACTER SETS (Contd)

CDC Graphic (64-Character Set)	ASCII Graphic (64-Character Set)	ASCII Graphic (95-Character Set)	Octal 6-Bit Display Code	Octal 6/12-Bit Display Code[†]	Octal 12-Bit ASCII Code	Card Keypunch Code	
						026	029
≠ not equal	" quote	" quote	64	64	0042	8-4	8-7
↱ concat.	_ underline	_ underline	65	65	0137	0-8-5	0-8-5
∨ logical OR	! exclamation	! exclamation	66	66	0041	11-0	12-8-7
∧ logical AND	& ampersand	& ampersand	67	67	0046	0-8-7	12
↑ superscript	' apostrophe	' apostrophe	70	70	0047	11-8-5	8-5
↓ subscript	? question	? question	71	71	0077	11-8-6	0-8-7
< less than	< less than	< less than	72	72	0074	12-0	12-8-4
> greater than	> greater than	> greater than	73	73	0076	11-8-7	0-8-6
≤ less/equal	@ commercial at		74			8-5	8-4
≥ greater/equal	\ reverse slant	\ rev. slant	75	75	0134	12-8-5	0-8-2
¬ logical NOT	^ circumflex		76			12-8-6	11-8-7
; semicolon	; semicolon	; semicolon	77	77	0073	12-8-7	11-8-6
		@ coml. at		7401	0100		
		^ circumflex		7402	0136		
		: colon[††]		7404[††]	0072		
		` grave accent		7407	0140		
		a		7601	0141		
		b		7602	0142		
		c		7603	0143		
		d		7604	0144		
		e		7605	0145		
		f		7606	0146		
		g		7607	0147		
		h		7610	0150		
		i		7611	0151		
		j		7612	0152		
		k		7613	0153		
		l		7614	0154		
		m		7615	0155		
		n		7616	0156		
		o		7617	0157		
		p		7620	0160		
		q		7621	0161		
		r		7622	0162		
		s		7623	0163		
		t		7624	0164		
		u		7625	0165		
		v		7626	0166		
		w		7627	0167		
		x		7630	0170		
		y		7631	0171		
		z		7632	0172		
		{ left brace		7633	0173		
		| vert. line		7634	0174		
		} right brace		7635	0175		
		~ tilde		7636	0176		

[†] Generally available only on NOS, or through BASIC on NOS/BE.

[††] The interpretation of this character or code depends on its context. Refer to Character Set Anomalies in the text.

[†††] Available for input only, on NOS.

TABLE A-3. ASCII 9-TRACK CODED TAPE CONVERSION

ASCII				Display Code†††		ASCII				Display Code†††	
Code Conversion†		Character and Code Conversion††				Code Conversion†		Character and Code Conversion††			
Code (Hex)	Char	Code (Hex)	Char	ASCII Char	Code (Octal)	Code (Hex)	Char	Code (Hex)	Char	ASCII Char	Code (Octal)
20	space	00	NUL	space	55	40	@	60	`	@	74
21	!	7D	}	!	66	41	A	61	a	A	01
22	"	02	STX	"	64	42	B	62	b	B	02
23	#	03	ETX	#	60	43	C	63	c	C	03
24	$	04	EOT	$	53	44	D	64	d	D	04
25	%	05	ENQ	%	63	45	E	65	e	E	05
25	%	05	ENQ	space	55	46	F	66	f	F	06
26	&	06	ACK	&	67	47	G	67	g	G	07
27	'	07	BEL	'	70	48	H	68	h	H	10
28	(	08	BS	(	51	49	I	69	i	I	11
29	)	09	HT	)	52	4A	J	6A	j	J	12
2A	*	0A	LF	*	47	4B	K	6B	k	K	13
2B	+	0B	VT	+	45	4C	L	6C	l	L	14
2C	,	0C	FF	,	56	4D	M	6D	m	M	15
2D	-	0D	CR	-	46	4E	N	6E	n	N	16
2E	.	0E	SO	.	57	4F	O	6F	o	O	17
2F	/	0F	SI	/	50	50	P	70	p	P	20
30	0	10	DLE	0	33	51	Q	71	q	Q	21
31	1	11	DC1	1	34	52	R	72	r	R	22
32	2	12	DC2	2	35	53	S	73	s	S	23
33	3	13	DC3	3	36	54	T	74	t	T	24
34	4	14	DC4	4	37	55	U	75	u	U	25
35	5	15	NAK	5	40	56	V	76	v	V	26
36	6	16	SYN	6	41	57	W	77	w	W	27
37	7	17	ETB	7	42	58	X	78	x	X	30
38	8	18	CAN	8	43	59	Y	79	y	Y	31
39	9	19	EM	9	44	5A	Z	7A	z	Z	32
3A	:	1A	SUB	:	00	5B	[	1C	FS	[	61
Display code 00 is undefined at sites using the 63-character set.						5C	\	7C	|	\	75
						5D	[	01	SOH	]	62
3A	:	1A	SUB	:	63	5E	^	7E	~	^	76
3B	;	1B	ESC	;	77	5F	_	7F	DEL	_	65
3C	<	7B	{	<	72						
3D	=	1D	GS	=	54						
3E	>	1E	RS	>	73						
3F	?	1F	US	?	71						

†When these characters are copied from or to a tape, the characters remain the same and the code changes from/to ASCII to/from display code.

††These characters do not exist in display code. When the characters are copied from a tape, each ASCII character is changed to an alternate display code character. The corresponding codes are also changed. Example: When the system copies a lowercase a, 61_{16}, from tape, it writes an uppercase A, 01_8.

†††A display code space always translates to an ASCII space.

TABLE A-4. EBCDIC 9-TRACK CODED TAPE CONVERSION

EBCDIC				Display Code[†††]	
Code Conversion[†]		Character and Code Conversion[††]			
Code (Hex)	Char	Code (Hex)	Char	ASCII Char	Code (Octal)
40	space	00	NUL	space	55
4A	¢	1C	IFS	[	61
4B	.	0E	SO	.	57
4C	<	C0	{	<	72
4D	(	16	BS	(	51
4E	+	0B	VT	+	45
4F	|	D0	}	!	66
50	&	2E	ACK	&	67
5A	!	01	SOH	]	62
5B	$	37	EOT	$	53
5C	*	25	LF	*	47
5D	)	05	HT	)	52
5E	;	27	ESC	;	77
5F	¬	A1	~	/	76
60	-	0D	CR	-	46
61	'	0F	SI	'	50
6B	,	0C	FF	,	56
6C	%	2D	ENQ	%	63
6C	%	2D	ENQ	space	55
6D		07	DEL		65
6E	≯	1E	IRS	≯	73
6F	?	1F	IUS	?	71
7A	:	3F	SUB	:	00
Display code 00 is undefined at sites using the 63-character set.					
7A	:	3F	SUB	:	63
7B	#	03	ETX	#	60
7C	@	79	\	@	74
7D	'	2F	BEL	'	70
7E	=	1D	IGS	=	54
7F	"	02	STX	"	64
C1	A	81	a	A	01
C2	B	82	b	B	02
C3	C	83	c	C	03
C4	D	84	d	D	04
C5	E	85	e	E	05
C6	F	86	f	F	06
C7	G	87	g	G	07
C8	H	88	h	H	10
C9	I	89	i	I	11
D1	J	91	j	J	12
D2	K	92	k	K	13
D3	L	93	l	L	14
D4	M	94	m	M	15
D5	N	95	n	N	16
D6	O	96	o	O	17
D7	P	97	p	P	20
D8	Q	98	q	Q	21
D9	R	99	r	R	22
E0	\	6A	|	\	75
E2	S	A2	s	S	23
E3	T	A3	t	T	24
E4	U	A4	u	U	25
E5	V	A5	v	V	26
E6	W	A6	w	W	27
E7	X	A7	x	X	30
E8	Y	A8	y	Y	31
E9	Z	A9	z	Z	32
F0	0	10	DLE	0	33
F1	1	11	DC1	1	34
F2	2	12	DC2	2	35
F3	3	13	TM	3	36
F4	4	3C	DC4	4	37
F5	5	3D	NAK	5	40
F6	6	32	SYN	6	41
F7	7	26	ETB	7	42
F8	8	18	CAN	8	43
F9	9	19	EM	9	44

[†] All EBCDIC codes not listed translate to display code 55_8 (space). A display code space always translates to an EBCDIC space.

[††] These characters do not exist in display code. When the characters are copied from a tape, each EBCDIC character is changed to an alternate display code character. The corresponding codes are also changed. Example: When the system copies a lowercase a, 81_{16}, from tape, it writes an uppercase A, 01_8.

[†††] When these characters are copied from or to a tape, the characters remain the same (except EBCDIC codes $4A_{16}$, $4F_{16}$, $5A_{16}$, and $5F_{16}$) and the code changes from/to EBCDIC to/from display code.

TABLE A-5. 7-TRACK CODED TAPE CONVERSIONS

External BCD	ASCII Character	Octal Display Code	External BCD	ASCII Character	Octal Display Code
01	1	34	40	-	46
02	2	35	41	J	12
03	3	36	42	K	13
04	4	37	43	L	14
05	5	40	44	M	15
06	6	41	45	N	16
07	7	42	46	O	17
10	8	43	47	P	20
11	9	44	50	Q	21
12†	0	33	51	R	22
13	=	54	52	!	66
14	"	64	53	$	53
15	@	74	54	*	47
16†	%	63	55	'	70
17	[	61	56	?	71
20	space	55	57	>	73
21	/	50	60	+	45
22	S	23	61	A	01
23	T	24	62	B	02
24	U	25	63	C	03
25	V	26	64	D	04
26	W	27	65	E	05
27	X	30	66	F	06
30	Y	31	67	G	07
31	Z	32	70	H	10
32	]	62	71	I	11
33	,	56	72	<	72
34	(	51	73	.	57
35		65	74	)	52
36	#	60	75	\	75
37	&	67	76	^	76
			77	;	77

†As explained in the text of this appendix, conversion of these codes depends on whether the tape is being read or written.

TABLE A-6. AMERICAN NATIONAL STANDARD CODE FOR INFORMATION INTERCHANGE (ASCII) PUNCHED CARD CODES AND EBCDIC TRANSLATION

	b_8 b_7 b_6 b_5	0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
b_4 b_3 b_2 b_1	COL ROW	0	1	2	3	4	5	6	7	8	9	10 (A)	11 (B)	12 (C)	13 (D)	14 (E)	15 (F)
0 0 0 0	0	NUL 12-0-9-8-1 NUL 00	DLE 12-11-9-8-1 DLE 10	SP no-punch SP 40	0 0 0 F0	@ 8-4 @ 7C	P 11-7 P D7	` 8-1 ` 79	p 12-11-7 p 97	11-0-9-8-1 DS 20	12-11-0-9-8-1 30	12-0-9-1 41	12-11-9-8 58	12-11-0-9-6 76	12-11-8-7 9F	12-11-0-8 B8	12-11-9-8-4 DC
0 0 0 1	1	SOH 12-9-1 SOH 01	DC1 11-9-1 DC1 11	! 12-8-7 | 4F	1 1 1 F1	A 12-1 A C1	Q 11-8 Q D8	a 12-0-1 a 81	q 12-11-8 q 98	0-9-1 SOS 21	9-1 31	12-0-9-2 42	11-8-1 59	12-11-0-9-7 77	11-0-8-1 A0	12-11-0-9 B9	12-11-9-8-5 DD
0 0 1 0	2	STX 12-9-2 STX 02	DC2 11-9-2 DC2 12	" 8-7 " 7F	2 2 2 F2	B 12-2 B C2	R 11-9 R D9	b 12-0-2 b 82	r 12-11-9 r 99	0-9-2 FS 22	11-9-8-2 CC 1A	12-0-9-3 43	11-0-9-2 62	12-11-0-9-8 78	11-0-8-2 AA	12-11-0-8-2 BA	12-11-9-8-6 DE
0 0 1 1	3	ETX 12-9-3 ETX 03	DC3 11-9-3 TM 13	# 8-3 # 7B	3 3 3 F3	C 12-3 C C3	S 0-2 S E2	c 12-0-3 c 83	s 11-0-2 s A2	0-9-3 23	9-3 33	12-0-9-4 44	11-0-9-3 63	12-0-8-1 80	11-0-8-3 AB	12-11-0-8-3 BB	12-11-9-8-7 DF
0 1 0 0	4	EOT 9-7 EOT 37	DC4 9-8-4 DC4 3C	$ 11-8-3 $ 5B	4 4 4 F4	D 12-4 D C4	T 0-3 T E3	d 12-0-4 d 84	t 11-0-3 t A3	0-9-4 BYP 24	9-4 PN 34	12-0-9-5 45	11-0-9-4 64	12-0-8-2 8A	11-0-8-4 AC	12-11-0-8-4 BC	11-0-9-8-2 EA
0 1 0 1	5	ENQ 0-9-8-5 ENQ 2D	NAK 9-8-5 NAK 3D	% 0-8-4 % 6C	5 5 5 F5	E 12-5 E C5	U 0-4 U E4	e 12-0-5 e 85	u 11-0-4 u A4	11-9-5 NL 15	9-5 RS 35	12-0-9-6 46	11-0-9-5 65	12-0-8-3 8B	11-0-8-5 AD	12-11-0-8-5 BD	11-0-9-8-3 EB
0 1 1 0	6	ACK 0-9-8-6 ACK 2E	SYN 9-2 SYN 32	& 12 & 50	6 6 6 F6	F 12-6 F C6	V 0-5 V E5	f 12-0-6 f 86	v 11-0-5 v A5	12-9-6 LC 06	9-6 UC 36	12-0-9-7 47	11-0-9-6 66	12-0-8-4 8C	11-0-8-6 AE	12-11-0-8-6 BE	11-0-9-8-4 ⑁ EC
0 1 1 1	7	BEL 0-9-8-7 BEL 2F	ETB 0-9-6 ETB 26	' 8-5 ' 7D	7 7 7 F7	G 12-7 G C7	W 0-6 W E6	g 12-0-7 g 87	w 11-0-6 w A6	11-9-7 IL 17	12-9-8 GE 08	12-0-9-8 48	11-0-9-7 67	12-0-8-5 8D	11-0-8-7 AF	12-11-0-8-7 BF	11-0-9-8-5 ED
1 0 0 0	8	BS 11-9-6 BS 16	CAN 11-9-8 CAN 18	(12-8-5 (4D	8 8 8 F8	H 12-8 H C8	X 0-7 X E7	h 12-0-8 h 88	x 11-0-7 x A7	0-9-8 28	9-8 38	12-8-1 49	11-0-9-8 68	12-0-8-6 8E	12-11-0-8-1 B0	12-0-9-8-2 CA	11-0-9-8-6 EE
1 0 0 1	9	HT 12-9-5 HT 05	EM 11-9-8-1 EM 19	) 11-8-5) 5D	9 9 9 F9	I 12-9 I C9	Y 0-8 Y E8	i 12-0-9 i 89	y 11-0-8 y A8	0-9-8-1 29	9-8-1 39	12-11-9-1 51	0-8-1 69	12-0-8-7 8F	12-11-0-1 B1	12-0-9-8-3 CB	11-0-9-8-7 EF
1 0 1 0	10 (A)	LF 0-9-5 LF 25	SUB 9-8-7 SUB 3F	* 11-8-4 * 5C	: 8-2 : 7A	J 11-1 J D1	Z 0-9 Z E9	j 12-11-1 j 91	z 11-0-9 z A9	0-9-8-2 SM 2A	9-8-2 3A	12-11-9-2 52	12-11-0 70	12-11-8-1 90	12-11-0-2 B2	12-0-9-8-4 ⑀ CC	12-11-0-9-8-2 |(LVM) FA
1 0 1 1	11 (B)	VT 12-9-8-3 VT 0B	ESC 0-9-7 ESC 27	+ 12-8-6 + 4E	; 11-8-6 ; 5E	K 11-2 K D2	[12-8-2 ¢ 4A	k 12-11-2 k 92	{ 12-0 { C0	0-9-8-3 CU2 2B	9-8-3 CU3 3B	12-11-9-3 53	12-11-0-9-1 71	12-11-8-2 9A	12-11-0-3 B3	12-0-9-8-5 CD	12-11-0-9-8-3 FB
1 1 0 0	12 (C)	FF 12-9-8-4 FF 0C	FS 11-9-8-4 IFS 1C	, 0-8-3 , 6B	< 12-8-4 < 4C	L 11-3 L D3	\ 0-8-2 \ E0	l 12-11-3 l 93	¦ 12-11 ¦ 6A	0-9-8-4 2C	12-9-4 PF 04	12-11-9-4 54	12-11-0-9-2 72	12-11-8-3 9B	12-11-0-4 B4	12-0-9-8-6 ⑂ CE	12-11-0-9-8-4 FC
1 1 0 1	13 (D)	CR 12-9-8-5 CR 0D	GS 11-9-8-5 IGS 1D	– 11 – 60	= 8-6 = 7E	M 11-4 M D4	] 11-8-2 ! 5A	m 12-11-4 m 94	} 11-0 } D0	12-9-8-1 RLF 09	11-9-4 RES 14	12-11-9-5 55	12-11-0-9-3 73	12-11-8-4 9C	12-11-0-5 B5	12-0-9-8-7 CF	12-11-0-9-8-5 FD
1 1 1 0	14 (E)	SO 12-9-8-6 SO 0E	RS 11-9-8-6 IRS 1E	. 12-8-3 . 4B	> 0-8-6 > 6E	N 11-5 N D5	^ 11-8-7 ¬ 5F	n 12-11-5 n 95	~ 11-0-1 ~ A1	12-9-8-2 SMM 0A	9-8-6 3E	12-11-9-6 56	12-11-0-9-4 74	12-11-8-5 9D	12-11-0-6 B6	12-11-9-8-2 DA	12-11-0-9-8-6 FE
1 1 1 1	15 (F)	SI 12-9-8-7 SI 0F	US 11-9-8-7 IUS 1F	/ 0-1 / 61	? 0-8-7 ? 6F	O 11-6 O D6	0̄-8-5 _ 6D	o 12-11-6 o 96	DEL 12-9-7 DEL 07	11-9-8-3 CU1 1B	11-0-9-1 E1	12-11-9-7 57	12-11-0-9-5 75	12-11-8-6 9E	12-11-0-7 B7	12-11-9-8-3 DB	EO 12-11-0-9-8-7 FF

LEGEND

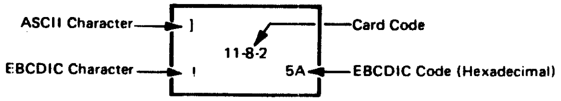


TABLE A-7. EXTENDED BINARY CODED DECIMAL INTERCHANGE CODE (EBCDIC) PUNCHED CARD CODES AND ASCII TRANSLATION

BITS 4 5 6 7	BITS 0 1 2 3 1ST HEX 2ND	0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
		0	1	2	3	4	5	6	7	8	9	A (10)	B (11)	C (12)	D (13)	E (14)	F (15)
0 0 0 0	0	NUL 12–0–9–8–1 NUL 00	DLE 12–11–9–8–1 DLE 10	DS 11–0–9–8–1 80	12–11–0–9–8–1 90	SP no punch SP 20	& 12 & 26	– 11 – 2D	12–11–0 BA	12–0–8–1 C3	12–11–8–1 CA	11–0–8–1 D1	12–11–0–8–1 D8	{ 12–0 { 7B	} 11–0 } 7D	\ 0–8–2 \ 5C	0 0 0 30
0 0 0 1	1	SOH 12–9–1 SOH 01	DC1 11–9–1 DC1 11	SOS 0–9–1 81	9–1 91	12–0–9–1 A0	12–11–9–1 A9	/ 0–1 / 2F	12–11–0–9–1 BB	a 12–0–1 a 61	j 12–11–1 j 6A	~ 11–0–1 ~ 7E	12–11–0–1 D9	A 12–1 A 41	J 11–1 J 4A	11–0–9–1 9F	1 1 1 31
0 0 1 0	2	STX 12–9–2 STX 02	DC2 11–9–2 DC2 12	FS 0–9–2 82	SYN 9–2 SYN 16	12–0–9–2 A1	12–11–9–2 AA	11–0–9–2 B2	12–11–0–9–2 BC	b 12–0–2 b 62	k 12–11–2 k 6B	s 11–0–2 s 73	12–11–0–2 DA	B 12–2 B 42	K 11–2 K 4B	S 0–2 S 53	2 2 2 32
0 0 1 1	3	ETX 12–9–3 ETX 03	TM 11–9–3 DC3 13	0–9–3 83	9–3 93	12–0–9–3 A2	12–11–9–3 AB	11–0–9–3 B3	12–11–0–9–3 BD	c 12–0–3 c 63	l 12–11–3 l 6C	t 11–0–3 t 74	12–11–0–3 DB	C 12–3 C 43	L 11–3 L 4C	T 0–3 T 54	3 3 3 33
0 1 0 0	4	PF 12–9–4 9C	RES 11–9–4 9D	BYP 0–9–4 84	PN 9–4 94	12–0–9–4 A3	12–11–9–4 AC	11–0–9–4 B4	12–11–0–9–4 BE	d 12–0–4 d 64	m 12–11–4 m 6D	u 11–0–4 u 75	12–11–0–4 DC	D 12–4 D 44	M 11–4 M 4D	U 0–4 U 55	4 4 4 34
0 1 0 1	5	HT 12–9–5 HT 09	NL 11–9–5 85	LF 0–9–5 LF 0A	RS 9–5 95	12–0–9–5 A4	12–11–9–5 AD	11–0–9–5 B5	12–11–0–9–5 BF	e 12–0–5 e 65	n 12–11–5 n 6E	v 11–0–5 v 76	12–11–0–5 DD	E 12–5 E 45	N 11–5 N 4E	V 0–5 V 56	5 5 5 35
0 1 1 0	6	LC 12–9–6 86	BS 11–9–6 BS 08	ETB 0–9–6 ETB 17	UC 9–6 96	12–0–9–6 A5	12–11–9–6 AE	11–0–9–6 B6	12–11–0–9–6 C0	f 12–0–6 f 66	o 12–11–6 o 6F	w 11–0–6 w 77	12–11–0–6 DE	F 12–6 F 46	O 11–6 O 4F	W 0–6 W 57	6 6 6 36
0 1 1 1	7	DEL 12–9–7 DEL 7F	IL 11–9–7 87	ESC 0–9–7 ESC 1B	EOT 9–7 EOT 04	12–0–9–7 A6	12–11–9–7 AF	11–0–9–7 B7	12–11–0–9–7 C1	g 12–0–7 g 67	p 12–11–7 p 70	x 11–0–7 x 78	12–11–0–7 DF	G 12–7 G 47	P 11–7 P 50	X 0–7 X 58	7 7 7 37
1 0 0 0	8	GE 12–9–8 97	CAN 11–9–8 CAN 18	0–9–8 88	9–8 98	12–0–9–8 A7	12–11–9–8 B0	11–0–9–8 B8	12–11–0–9–8 C2	h 12–0–8 h 68	q 12–11–8 q 71	y 11–0–8 y 79	12–11–0–8 E0	H 12–8 H 48	Q 11–8 Q 51	Y 0–8 Y 59	8 8 8 38
1 0 0 1	9	RLF 12–9–8–1 8D	EM 11–9–8–1 EM 19	0–9–8–1 89	9–8–1 99	12–8–1 A8	11–8–1 B1	0–8–1 B9	\` 8–1 \` 60	i 12–0–9 i 69	r 12–11–9 r 72	z 11–0–9 z 7A	12–11–0–9 E1	I 12–9 I 49	R 11–9 R 52	Z 0–9 Z 5A	9 9 9 39
1 0 1 0	A (10)	SMM 12–9–8–2 8E	CC 11–9–8–2 92	SM 0–9–8–2 8A	9–8–2 9A	¢ 12–8–2 [5B	! 11–8–2] 5D	¦ 12–11 ¦ 7C	: 8–2 : 3A	12–0–8–2 C4	12–11–8–2 CB	11–0–8–2 D2	12–11–0–8–2 E2	12–0–9–8–2 E8	12–11–9–8–2 EE	11–0–9–8–2 F4	| (LVM) 12–11–0–9–8–2 FA
1 0 1 1	B (11)	VT 12–9–8–3 VT 0B	CU1 11–9–8–3 8F	CU2 0–9–8–3 8B	CU3 9–8–3 9B	. 12–8–3 . 2E	$ 11–8–3 $ 24	, 0–8–3 , 2C	# 8–3 # 23	12–0–8–3 C5	12–11–8–3 CC	11–0–8–3 D3	12–11–0–8–3 E3	12–0–9–8–3 E9	12–11–9–8–3 EF	11–0–9–8–3 F5	12–11–0–9–8–3 FB
1 1 0 0	C (12)	FF 12–9–8–4 FF 0C	IFS 11–9–8–4 FS 1C	0–9–8–4 8C	DC4 9–8–4 DC4 14	< 12–8–4 < 3C	* 11–8–4 * 2A	% 0–8–4 % 25	@ 8–4 @ 40	12–0–8–4 C6	12–11–8–4 CD	11–0–8–4 D4	12–11–0–8–4 E4	⑀ 12–0–9–8–4 EA	12–11–9–8–4 F0	⑁ 11–0–9–8–4 F6	12–11–0–9–8–4 FC
1 1 0 1	D (13)	CR 12–9–8–5 CR 0D	IGS 11–9–8–5 GS 1D	ENQ 0–9–8–5 ENQ 05	NAK 9–8–5 NAK 15	(12–8–5 (28	) 11–8–5) 29	_ 0–8–5 _ 5F	' 8–5 ' 27	12–0–8–5 C7	12–11–8–5 CE	11–0–8–5 D5	12–11–0–8–5 E5	12–0–9–8–5 EB	12–11–9–8–5 F1	11–0–9–8–5 F7	12–11–0–9–8–5 FD
1 1 1 0	E (14)	SO 12–9–8–6 SO 0E	IRS 11–9–8–6 RS 1E	ACK 0–9–8–6 ACK 06	9–8–6 9E	+ 12–8–6 + 2B	; 11–8–6 ; 3B	> 0–8–6 > 3E	= 8–6 = 3D	12–0–8–6 C8	12–11–8–6 CF	11–0–8–6 D6	12–11–0–8–6 E6	⑂ 12–0–9–8–6 EC	12–11–9–8–6 F2	11–0–9–8–6 F8	12–11–0–9–8–6 FE
1 1 1 1	F (15)	SI 12–9–8–7 SI 0F	IUS 11–9–8–7 US 1F	BEL 0–9–8–7 BEL 07	SUB 9–8–7 SUB 1A	| 12–8–7 ! 21	¬ 11–8–7 ^ 5E	? 0–8–7 ? 3F	" 8–7 " 22	12–0–8–7 C9	12–11–8–7 D0	11–0–8–7 D7	12–11–0–8–7 E7	12–0–9–8–7 ED	12–11–9–8–7 F3	11–0–9–8–7 F9	12–11–0–9–8–7 EO FF

LEGEND

EBCDIC Character → !   11-8-2 ← Card Code

ASCII Character →]   5D ← ASCII Code (Hexadecimal)

TABLE A-8. FULL EBCDIC CHARACTER SET

Hexa-decimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic Character†	EBCDIC Control Character	Hexa-decimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic Character†	EBCDIC Control Character
00	0000		NUL	41	0101		undefined
01	0001		SOH	thru	thru		
02	0002		STX	49	0111		
03	0003		ETX	4A	0112	¢	
04	0004		PF	4B	0113	.	
05	0005		HT	4C	0114	<	
06	0006		LC	4D	0115	(	
07	0007		DEL	4E	0116	+	
08	0010		undefined	4F	0117	|	
09	0011		undefined	50	0120	&	
0A	0012		SMM	51	0121		undefined
0B	0013		VT	thru	thru		
0C	0014		FF	59	0131		
0D	0015		CR	5A	0132	!	
0E	0016		SO	5B	0133	$	
0F	0017		SI	5C	0134	*	
10	0020		DLE	5D	0135	)	
11	0021		DC1	5E	0136	;	
12	0022		DC2	5F	0137	¬	
13	0023		TM	60	0140	-	
14	0024		RES	61	0141	/	
15	0025		NL	62	0142		undefined
16	0026		BS	thru	thru		
17	0027		IL	69	0151		
18	0030		CAN	6A	0152	¦	
19	0031		EM	6B	0153	,	
1A	0032		CC	6C	0154	%	
1B	0033		CU1	6D	0155	_	
1C	0034		IFS	6E	0156	>	
1D	0035		IGS	6F	0157	?	
1E	0036		IRS	70	0160		undefined
1F	0037		IUS	thru	thru		
20	0040		DS	78	0170		
21	0041		SOS	79	0171	`	
22	0042		FS	7A	0172	:	
23	0043		undefined	7B	0173	#	
24	0044		BYP	7C	0174	@	
25	0045		LF	7D	0175	'	
26	0046		ETB or EOB	7E	0176	=	
27	0047		ESC or PRE	7F	0177	"	
28	0050		undefined	80	0200		undefined
29	0051		undefined	81	0201	a	
2A	0052		SM	82	0202	b	
2B	0053		CU2	83	0203	c	
2C	0054		undefined	84	0204	d	
2D	0055		ENQ	85	0205	e	
2E	0056		ACK	86	0206	f	
2F	0057		BEL	87	0207	g	
30	0060		undefined	88	0210	h	
31	0061		undefined	89	0211	i	
32	0062		SYN	8A	0212		undefined
33	0063		undefined	thru	thru		
34	0064		PN	90	0220		
35	0065		RS	91	0221	j	
36	0066		UC	92	0222	k	
37	0067		EOT	93	0223	l	
38	0070		undefined	94	0224	m	
39	0071		undefined	95	0225	n	
3A	0072		undefined	96	0226	o	
3B	0073		CU3	97	0227	p	
3C	0074		DC4	98	0230	q	
3D	0075		NAK	99	0231	r	
3E	0076		undefined	9A	0232		undefined
3F	0077		SUB	thru	thru		
40	0100	space		A0	0240		

TABLE A-8. FULL EBCDIC CHARACTER SET (Contd)

Hexadecimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic Character[†]	EBCDIC Control Character	Hexadecimal EBCDIC Code	Octal 12-Bit EBCDIC Code	EBCDIC Graphic Character[†]	EBCDIC Control Character
A1	0241	~		D7	0327	P	
A2	0242	s		D8	0330	Q	
A3	0243	t		D9	0331	R	
A4	0244	u		DA	0332		undefined
A5	0245	v		thru	thru		
A6	0246	w		DF	0337		
A7	0247	x		E0	0340	\	
A8	0250	y		E1	0341		undefined
A9	0251	z		E2	0342	S	
AA	0252		undefined	E3	0343	T	
thru	thru			E4	0344	U	
BF	0277			E5	0345	V	
C0	0300	{		E6	0346	W	
C1	0301	A		E7	0347	X	
C2	0302	B		E8	0350	Y	
C3	0303	C		E9	0351	Z	
C4	0304	D		EA	0352		undefined
C5	0305	E		EB	0353		undefined
C6	0306	F		EC	0354	⑁	
C7	0307	G		ED	0355		undefined
C8	0310	H		thru	thru		
C9	0311	I		EF	0357		
CA	0312		undefined	F0	0360	0	
CB	0313		undefined	F1	0361	1	
CC	0314	⑀		F2	0362	2	
CD	0315	⑂	undefined	F3	0363	3	
CE	0316			F4	0364	4	
CF	0317		undefined	F5	0365	5	
D0	0320	}		F6	0366	6	
D1	0321	J		F7	0367	7	
D2	0322	K		F8	0370	8	
D3	0323	L		F9	0371	9	
D4	0324	M		FA	0372	│	
D5	0325	N		FB	0373		undefined
D6	0326	O		thru	thru		
				FF	0377		

[†] Graphic characters shown are those used on the IBM System/370 standard (PN) print train. Other devices support subsets or variations of this character graphic set.

NETWORK ACCESS METHOD TERMINAL TRANSMISSION CODE SETS

There are two modes in which coded character data can be exchanged with a network terminal console. These two modes, character mode and transparent mode, correspond to the type of character code editing and translation performed by the network software during input and output operations. The transmission mode used by the network software for input can be selected by the terminal operator, using a Terminal Interface Program command (sometimes referred to as a terminal definition command). The transmission mode used by the network software for output can be selected by the application program providing the terminal facility service.

Character Mode Transmissions

Character mode is the initial and default mode used for both input and output transmissions. When the network software services the terminal in character mode, it translates input characters from the transmission code used by the terminal into the ASCII code shown in table A-9. The translation of a specific transmission code to a specific ASCII code depends on the terminal class the network software associates with the terminal. In character mode input, the parity of the terminal transmission code is not preserved in the corresponding ASCII code; the ASCII code received by the terminal-servicing facility program always has its eighth bit set to zero.

Character mode output is translated in a similar manner. The network software provides the parity bit setting appropriate for the terminal being serviced, even though translating from ASCII characters with zero parity bit settings.

Tables A-10 through A-21 show the character mode translations performed for each terminal class. The parity shown in the terminal transmission codes is the parity used as a default for the terminal class. The parity setting actually used by a terminal can be identified to the network software through a TIP command.

Tables A-10 through A-21 contain the graphic and control characters associated with the transmission codes used by the terminal because of the terminal class and code set in use. The network ASCII graphic and control characters shown are those of the standard ASCII character set. associated with the ASCII transmission codes of table A-9.

The general case for code translations of character mode data is summarized in the following paragraphs. This generalized description permits use of only table A-9 to explain all specific cases. The reader can logically extend this generalized description to allow use of tables A-1 through A-8 as descriptions of character set mapping for various functions initiated from a terminal. Tables A-1 through A-8 are provided for the reader's use while coding an application program to run under the operating system. They do not describe character transmissions between an application program and the network.

Table A-9 contains the ASCII 128-character set supported by the Network Access Method. A 96-character subset consists of the rightmost six columns and includes the 95-character graphic subset referenced previously in this appendix; the deletion character (DEL) is not a graphic character. A 64-character subset consists of the middle four columns. Note that 6-bit display code equivalents exist for the characters in this 64-character subset only.

Although the network supports the 128-character set, some terminals restrict output to a smaller subset. This restriction is supported by replacing the control characters in columns 0 and 1 of table A-9 with blanks to produce the 96-character subset, and, additionally, replacing the characters in columns 6 and 7 with the corresponding characters from columns 4 and 5, respectively, to produce the 64-character subset.

Similarly, input from a device may be limited to a smaller subset by the device itself because the device cannot produce the full 128-character set. A character input from a device using a character set other than ASCII is converted to an equivalent ASCII character; characters without ASCII character equivalents are replaced by the ASCII blank character.

An application can also cause character replacement (as described previously for output) as well as character conversion, by requesting display-coded input from the network.

The 7-bit hexadecimal code value for each character consists of the character's column number in the table, followed by its row number. For example, N is in row E of column 4, so its value is $4E_{16}$.

Transparent Mode Transmissions

Transparent mode is selected separately for input and output transmissions. During transparent mode input, the parity bit is stripped from each terminal transmission code (unless the N parity option has been selected by a Terminal Interface Program command), and the transmission code is placed in an 8-bit byte without translation to 7-bit ASCII code. Line transmission protocol characters are deleted from a mode 4C terminal input stream.

When the 8-bit bytes arrive in the host computer, a terminal servicing facility program such as the Interactive Facility can right-justify the bytes within a 12-bit byte. Upon transmission of 12-bit bytes from the host computer, the leftmost 4 bits (bits 11 through 8) are discarded.

During transparent mode output, processing similar to that performed for input occurs. The code in each 8-bit byte received by the network software from the terminal servicing facility program is not translated. The parity bit appropriate for the terminal class being used is altered as indicated by the parity option in effect for the terminal. The codes are then output in transmission bytes appropriate for the codes associated with the terminal class being used. Line transmission protocol characters are inserted into a mode 4C terminal output stream.

TABLE A-9. FULL ASCII CHARACTER SET

← 128-Character Set →

← 96-Character Subset →

← 64-Character Subset →

b_7					0	0	0	0	1	1	1	1
b_6					0	0	1	1	0	0	1	1
b_5					0	1	0	1	0	1	0	1
Bits b_4	b_3	b_2	b_1	Row / Column	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL 000	DLE 020	SP 040	0 060	@ 100	P 120	` 140	p 160
0	0	0	1	1	SOH 001	DC1 021	! 041	1 061	A 101	Q 121	a 141	q 161
0	0	1	0	2	STX 002	DC2 022	" 042	2 062	B 102	R 122	b 142	r 162
0	0	1	1	3	ETX 003	DC3 023	# 043	3 063	C 103	S 123	c 143	s 163
0	1	0	0	4	EOT 004	DC4 024	$ 044	4 064	D 104	T 124	d 144	t 164
0	1	0	1	5	ENQ 005	NAK 025	% 045	5 065	E 105	U 125	e 145	u 165
0	1	1	0	6	ACK 006	SYN 026	& 046	6 066	F 106	V 126	f 146	v 166
0	1	1	1	7	BEL 007	ETB 027	' 047	7 067	G 107	W 127	g 147	w 167
1	0	0	0	8	BS 010	CAN 030	(050	8 070	H 110	X 130	h 150	x 170
1	0	0	1	9	HT 011	EM 031	) 051	9 071	I 111	Y 131	i 151	y 171
1	0	1	0	A	LF 012	SUB 032	* 052	: 072	J 112	Z 132	j 152	z 172
1	0	1	1	B	VT 013	ESC 033	+ 053	; 073	K 113	[133	k 153	{ 173
1	1	0	0	C	FF 014	FS 034	, 054	< 074	L 114	\ 134	l 154	| 174
1	1	0	1	D	CR 015	GS 035	- 055	= 075	M 115	] 135	m 155	} 175
1	1	1	0	E	SO 016	RS 036	. 056	> 076	N 116	^ 136	n 156	~ 176
1	1	1	1	F	SI 017	US 037	/ 057	? 077	O 117	_ 137	o 157	DEL 177

LEGEND:

Numbers under characters are the octal values for the 7-bit character codes used within the network.

TABLE A-10. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 9, 14, 16, AND 17

Terminal EBCDIC			Network ASCII (Character Mode Use)		
Octal Code	Graphic†	Control Character	Octal Code††	Graphic	Control Character
000		NUL	000		null
001		SOH	001		start of header
002		STX	002		start of text
003		ETX	003		end of text
004		PF	040	space	
005		HT	011		horizontal tabulate
006		LC	040	space	
007		DEL	177		delete
010		undefined	040	space	
011		undefined	040	space	
012		SMM	040	space	
013		VT	013		vertical tabulate
014		FF	014		form feed
015		CR	015		carriage return
016		SO	016		shift out
017		SI	017		shift in
020		DLE	020		data link escape
021		DC1	021		device control 1
022		DC2	022		device control 2
023		TM	023		device control 3
024		RES	040	space	
025		NL	040	space	
026		BS	010		backspace
027		IL	040	space	
030		CAN	030		cancel
031		EM	031		end of medium
032		CC	040	space	
033		CU1	040	space	
034		IFS	034		file separator
035		IGS	035		group separator
036		IRS	036		record separator
037		IUS	037		unit separator
040		DS	040	space	
041		SOS	040	space	
042		FS	040	space	
043		undefined	040	space	
044		BYP	040	space	
045		LF	012		linefeed
046		ETB or EOB	027		end of transmission block
047		ESC or PRE	033		escape
050		undefined	040	space	
051		undefined	040	space	
052		SM	040	space	
053		CU2	040	space	
054		undefined	040	space	
055		ENQ	005		enquiry
056		ACK	006		positive acknowledgment
057		BEL	007		bell
060		undefined	040	space	
061		undefined	040	space	
062		SYN	026		synchronous idle
063		undefined	040	space	
064		PN	040	space	
065		RS	040	space	
066		UC	040	space	
067		EOT	004		end of transmission
070		undefined	040	space	
071		undefined	040	space	
072		undefined	040	space	
073		CU3	040	space	
074		DC4	024		device control 4
075		NAK	025		negative acknowledgement
076		undefined	040	space	
077		SUB	032		substitute
100	space		040	space	

TABLE A-10. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 9, 14, 16, AND 17 (Contd)

Terminal EBCDIC			Network ASCII (Character Mode Use)		
Octal Code	Graphic†	Control Character	Octal Code††	Graphic	Control Character
101 thru 111		undefined	040	space	
112	¢		133	[	
113	.		056	.	
114	<		074	<	
115	(		050	(	
116	+		053	+	
117	|		041	!	
120	&		046	&	
121 thru 131		undefined	040	space	
132	!		135	]	
133	$		044	$	
134	*		052	*	
135	)		051	)	
136	;		073	;	
137	¬		136	^	
140	-		055	-	
141	/		057	/	
142 thru 151		undefined	040	space	
152	¦		174	¦	
153	,		054	,	
154	%		045	%	
155	_		137	_	
156	>		076	>	
157	?		077	?	
160 thru 170		undefined	040	space	
171	`		140	`	
172	:		172	:	
173	#		043	#	
174	@		100	@	
175	'		047	'	
176	=		075	=	
177	"		042	"	
200		undefined	040	space	
201	a		141	a	
202	b		142	b	
203	c		143	c	
204	d		144	d	
205	e		145	e	
206	f		146	f	
207	g		147	g	
210	h		150	h	
211	i		151	i	
212 thru 220		undefined	040	space	
221	j		152	j	
222	k		153	k	
223	l		154	l	
224	m		155	m	
225	n		156	n	
226	o		157	o	
227	p		160	p	
230	q		161	q	
231	r		162	r	
232 thru 240		undefined	040	space	

TABLE A-10. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 9, 14, 16, AND 17 (Contd)

Terminal EBCDIC			Network ASCII (Character Mode Use)		
Octal Code	Graphic[†]	Control Character	Octal Code[††]	Graphic	Control Character
241	~		176	~	
242	s		163	s	
243	t		164	t	
244	u		165	u	
245	v		166	v	
246	w		167	w	
247	x		170	x	
250	y		171	y	
251	z		172	z	
252 thru 277		undefined	040	space	
300	{		173	{	
301	A		101	A	
302	B		102	B	
303	C		103	C	
304	D		104	D	
305	E		105	E	
306	F		106	F	
307	G		107	G	
310	H		110	H	
311	I		111	I	
312		undefined	040	space	
313		undefined	040	space	
314	⑀		040	space	
315		undefined	040	space	
316	⑂		040	space	
317		undefined	040	space	
320	}		175	}	
321	J		112	J	
322	K		113	K	
323	L		114	L	
324	M		115	M	
325	N		116	N	
326	O		117	O	
327	P		120	P	
330	Q		121	Q	
331	R		122	R	
332 thru 337		undefined	040	space	
340	\		134	\	
341		undefined	040	space	
342	S		123	S	
343	T		124	T	
344	U		125	U	
345	V		126	V	
346	W		127	W	
347	X		130	X	
350	Y		131	Y	
351	Z		132	Z	
352		undefined	040	space	
353		undefined	040	space	
354	⑁		040	space	
355 thru 357		undefined	040	space	
360	0		060	0	
361	1		061	1	
362	2		062	2	
363	3		063	3	
364	4		064	4	
365	5		065	5	
366	6		066	6	
367	7		067	7	

TABLE A-10. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 9, 14, 16, AND 17 (Contd)

Terminal EBCDIC			Network ASCII (Character Mode Use)		
Octal Code	Graphic†	Control Character	Octal Code††	Graphic	Control Character
370	8		070	8	
371	9		071	9	
372	|		040	space	
373 thru 377		undefined	040	space	

†Graphic characters shown are those used on the IBM System/370 standard (PN) print train. Other devices support subsets or variations of this character graphic set.

††Shown with zero parity (eighth or uppermost bit is always zero).

TABLE A-11. AMERICAN NATIONAL STANDARD CODE FOR INFORMATION INTERCHANGE (ASCII) WITH 029 PUNCHED CARD CODES AND EBCDIC TRANSLATION (BATCH OUTPUT DEVICES, TERMINAL CLASSES 9, 14, 16, AND 17)

ASCII Bit	b_8 b_7 b_6 b_5	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	ASCII Bit	b_8 b_7 b_6 b_5	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1
b_4 b_3 b_2 b_1	Col Row	2	3	4	5	b_4 b_3 b_2 b_1	Col Row	2	3	4	5
0 0 0 0	0	SP no punch SP 40	0 0 0 F0	@ 8-4 @ 7C	P 11-7 P D7	1 0 0 0	8	(12-8-5 (4D	8 8 8 F8	H 12-8 H C8	X 0-7 X E7
0 0 0 1	1	! 12-8-7 ! 4F	1 1 1 F1	A 12-1 A C1	Q 11-8 Q D8	1 0 0 1	9	) 11-8-5) 5D	9 9 9 F9	I 12-9 I C9	Y 0-8 Y E8
0 0 1 0	2	" 8-7 " 7F	2 2 2 F2	B 12-2 B C2	R 11-9 R D9	1 0 1 0	10 (A)	* 11-8-4 * 5C	: 8-2 : 7A	J 11-1 J D1	Z 0-9 Z E9
0 0 1 1	3	# 8-3 # 7B	3 3 3 F3	C 12-3 C C3	S 0-2 S E2	1 0 1 1	11 (B)	+ 12-8-6 + 4E	; 11-8-6 ; 5E	K 11-2 K D2	[12-8-2 ¢ 4A
0 1 0 0	4	$ 11-8-3 $ 5B	4 4 4 F4	D 12-4 D C4	T 0-3 T E3	1 1 0 0	12 (C)	, 0-8-3 , 6B	< 12-8-4 < 4C	L 11-3 L D3	\ 0-8-2 \ E0
0 1 0 1	5	% 0-8-4 % 6C	5 5 5 F5	E 12-5 E C5	U 0-4 U E4	1 1 0 1	13 (D)	- 11 - 60	= 8-6 = 7E	M 11-4 M D4	] 11-8-2 ! 5A
0 1 1 0	6	& 12 & 50	6 6 6 F6	F 12-6 F C6	V 0-5 V E5	1 1 1 0	14 (E)	. 12-8-3 . 4B	> 0-8-6 > 6E	N 11-5 N D5	∧ 11-8-7 ¬ 5F
0 1 1 1	7	' 8-5 ' 7D	7 7 7 F7	G 12-7 G C7	W 0-6 W E6	1 1 1 1	15 (F)	/ 0-1 / 61	? 0-8-7 ? 6F	O 11-6 O D6	_ 0-8-5 _ 6D

LEGEND:

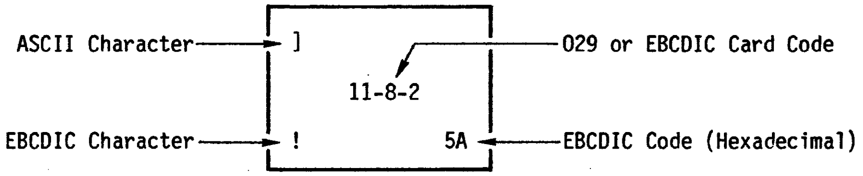
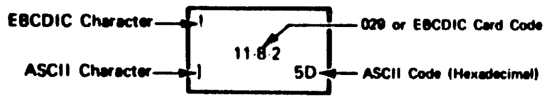


TABLE A-12. EXTENDED BINARY CODED DECIMAL INTERCHANGE CODE (EBCDIC) WITH 029 PUNCHED CARD CODES AND ASCII TRANSLATION (BATCH INPUT DEVICES, TERMINAL CLASSES 9, 14, 16, AND 17)

EBCDIC BITS	8 7 6 5	0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
4 3 2 1	1ST HEX 2ND	0	1	2	3	4	5	6	7	8	9	A (10)	B (11)	C (12)	D (13)	E (14)	F (15)
0 0 0 0	0	NUL 12-0-9-8-1 SP 20	DLE 12-11-9-8-1 SP 20	DS 11-0-9-8-1 SP 20	12-11-0-9-8-1 SP 20	SP no punch SP 20	& 12 & 26	- 11 - 2D	12-11-0 SP 20	12-0-8-1 SP 20	12-11-8-1 SP 20	11-0-8-1 SP 20	12-11-0-8-1 SP 20	{ 12-0 < 3C	} 11-0 ! 21	\\ 0-8-2 \\ 5C	0 0 0 30
0 0 0 1	1	SOH 12-9-1 SP 20	DC1 11-9-1 SP 20	SOS 0-9-1 SP 20	9-1 SP 20	12-0-9-1 SP 20	12-11-9-1 SP 20	/ 0-1 / 2F	12-11-0-9-1 SP 20	a 12-0-1 A 41	j 12-11-1 J 4A	~ 11-0-1 ^ 5E	12-11-0-1 SP 20	A 12-1 A 41	J 11-1 J 4A	11-0-9-1 9F	1 1 1 31
0 0 1 0	2	STX 12-9-2 SP 20	DC2 11-9-2 SP 20	FS 0-9-2 SP 20	SYN 9-2 SP 20	12-0-9-2 SP 20	12-11-9-2 SP 20	11-0-9-2 SP 20	12-11-0-9-2 SP 20	b 12-0-2 B 42	k 12-11-2 K 4B	s 11-0-2 S 53	12-11-0-2 SP 20	B 12-2 B 42	K 11-2 K 4B	S 0-2 S 53	2 2 2 32
0 0 1 1	3	ETX 12-9-3 SP 20	TM 11-9-3 SP 20	0-9-3 SP 20	9-3 SP 20	12-0-9-3 SP 20	12-11-9-3 SP 20	11-0-9-3 SP 20	12-11-0-9-3 SP 20	c 12-0-3 C 43	l 12-11-3 L 4C	t 11-0-3 T 54	12-11-0-3 SP 20	C 12-3 C 43	L 11-3 L 4C	T 0-3 T 54	3 3 3 33
0 1 0 0	4	PF 12-9-4 SP 20	RES 11-9-4 SP 20	BYP 0-9-4 SP 20	PN 9-4 SP 20	12-0-9-4 SP 20	12-11-9-4 SP 20	11-0-9-4 SP 20	12-11-0-9-4 SP 20	d 12-0-4 D 44	m 12-11-4 M 4D	u 11-0-4 U 55	12-11-0-4 SP 20	D 12-4 D 44	M 11-4 M 4D	U 0-4 U 55	4 4 4 34
0 1 0 1	5	HT 12-9-5 SP 20	NL 11-9-5 SP 20	LF 0-9-5 SP 20	RS 9-5 SP 20	12-0-9-5 SP 20	12-11-9-5 SP 20	11-0-9-5 SP 20	12-11-0-9-5 SP 20	e 12-0-5 E 45	n 12-11-5 N 4E	v 11-0-5 V 56	12-11-0-5 SP 20	E 12-5 E 45	N 11-5 N 4E	V 0-5 V 56	5 5 5 35
0 1 1 0	6	LC 12-9-6 SP 20	BS 11-9-6 SP 20	ETB 0-9-6 SP 20	UC 9-6 SP 20	12-0-9-6 SP 20	12-11-9-6 SP 20	11-0-9-6 SP 20	12-11-0-9-6 SP 20	f 12-0-6 F 46	o 12-11-6 O 4F	w 11-0-6 W 57	12-11-0-6 SP 20	F 12-6 F 46	O 11-6 O 4F	W 0-6 W 57	6 6 6 36
0 1 1 1	7	DEL 12-9-7 SP 20	IL 11-9-7 SP 20	ESC 0-9-7 SP 20	EOT 9-7 SP 20	12-0-9-7 SP 20	12-11-9-7 SP 20	11-0-9-7 SP 20	12-11-0-9-7 SP 20	g 12-0-7 G 47	p 12-11-7 P 50	x 11-0-7 X 58	12-11-0-7 SP 20	G 12-7 G 47	P 11-7 P 50	X 0-7 X 58	7 7 7 37
1 0 0 0	8	GE 12-9-8 SP 20	CAN 11-9-8 SP 20	0-9-8 SP 20	9-8 SP 20	12-0-9-8 SP 20	12-11-9-8 SP 20	11-0-9-8 SP 20	12-11-0-9-8 SP 20	h 12-0-8 H 48	q 12-11-8 Q 51	y 11-0-8 Y 59	12-11-0-8 SP 20	H 12-8 H 48	Q 11-8 Q 51	Y 0-8 Y 59	8 8 8 38
1 0 0 1	9	RLF 12-9-8-1 SP 20	EM 11-9-8-1 SP 20	0-9-8-1 SP 20	9-8-1 SP 20	12-8-1 SP 20	11-8-1 SP 20	0-8-1 SP 20	` 8-1 @ 40	i 12-0-9 I 49	r 12-11-9 R 52	z 11-0-9 Z 5A	12-11-0-9 SP 20	I 12-9 I 49	R 11-9 R 52	Z 0-9 Z 5A	9 9 9 39
1 0 1 0	A (10)	SMM 12-9-8-2 SP 20	CC 11-9-8-2 SP 20	SM 0-9-8-2 SP 20	9-8-2 SP 20	¢ 12-8-2 [5B	! 11-8-2] 5D	¦ 12-11 \\ 5C	: 8-2 : 3A	12-0-8-2 SP 20	12-11-8-2 SP 20	11-0-8-2 SP 20	12-11-0-8-2 SP 20	12-0-9-8-2 SP 20	12-11-9-8-2 SP 20	11-0-9-8-2 SP 20	I (LVM) 12-11-0-9-8-2 SP 20
1 0 1 1	B (11)	VT 12-9-8-3 SP 20	CU1 11-9-8-3 SP 20	CU2 0-9-8-3 SP 20	CU3 9-8-3 SP 20	. 12-8-3 . 2E	$ 11-8-3 $ 24	, 0-8-3 , 2C	# 8-3 # 23	12-0-8-3 SP 20	12-11-8-3 SP 20	11-0-8-3 SP 20	12-11-0-8-3 SP 20	12-0-9-8-3 SP 20	12-11-9-8-3 SP 20	11-0-9-8-3 SP 20	12-11-0-9-8-3 SP 20
1 1 0 0	C (12)	FF 12-9-8-4 SP 20	IFS 11-9-8-4 SP 20	0-9-8-4 SP 20	DC4 9-8-4 SP 20	< 12-8-4 < 3C	* 11-8-4 * 2A	% 0-8-4 % 25	@ 8-4 @ 40	12-0-8-4 SP 20	12-11-8-4 SP 20	11-0-8-4 SP 20	12-11-0-8-4 SP 20	⑁ 12-0-9-8-4 SP 20	12-11-9-8-4 SP 20	⑇ 11-0-9-8-4 SP 20	12-11-0-9-8-4 SP 20
1 1 0 1	D (13)	CR 12-9-8-5 SP 20	IGS 11-9-8-5 SP 20	ENQ 0-9-8-5 SP 20	NAK 9-8-5 SP 20	(12-8-5 (28	) 11-8-5) 29	0-8-5 _ 5F	' 8-5 ' 27	12-0-8-5 SP 20	12-11-8-5 SP 20	11-0-8-5 SP 20	12-11-0-8-5 SP 20	12-0-9-8-5 SP 20	12-11-9-8-5 SP 20	11-0-9-8-5 SP 20	12-11-0-9-8-5 SP 20
1 1 1 0	E (14)	SO 12-9-8-6 SP 20	IRS 11-9-8-6 SP 20	ACK 0-9-8-6 SP 20	9-8-6 SP 20	+ 12-8-6 + 2B	; 11-8-6 ; 3B	> 0-8-6 > 3E	= 8-6 = 3D	12-0-8-6 SP 20	12-11-8-6 SP 20	11-0-8-6 SP 20	12-11-0-8-6 SP 20	⑂ 12-0-9-8-6 SP 20	12-11-9-8-6 SP 20	11-0-9-8-6 SP 20	12-11-0-9-8-6 SP 20
1 1 1 1	F (15)	SI 12-9-8-7 SP 20	IUS 11-9-8-7 SP 20	BEL 0-9-8-7 SP 20	SUB 9-8-7 SP 20	| 12-8-7 ! 21	¬ 11-8-7 ^ 5E	? 0-8-7 ? 3F	" 8-7 " 22	12-0-8-7 SP 20	12-11-8-7 SP 20	11-0-8-7 SP 20	12-11-0-8-7 SP 20	12-0-9-8-7 SP 20	12-11-9-8-7 SP 20	11-0-9-8-7 SP 20	EO 12-11-0-9-8-7 SP 20

LEGEND



NOTE: A card with the character SUB punched in column 1 functions as a /*EOR separator card.

TABLE A-13. AMERICAN NATIONAL STANDARD CODE FOR INFORMATION INTERCHANGE (ASCII) WITH 026 PUNCHED CARD CODES AND EBCDIC TRANSLATION (BATCH OUTPUT DEVICES, TERMINAL CLASSES 9, 14, 16, AND 17)

ASCII Bit	b_8 b_7 b_6 b_5	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	ASCII Bit	b_8 b_7 b_6 b_5	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1
b_4 b_3 b_2 b_1	Col / Row	2	3	4	5	b_4 b_3 b_2 b_1	Col / Row	2	3	4	5
0 0 0 0	0	SP no punch SP 40	0 0 0 F0	≤ 8-5 ' 7D	P 11-7 P D7	1 0 0 0	8	(0-8-4 % 6C	8 8 8 F8	H 12-8 H C8	X 0-7 X E7
0 0 0 1	1	! 12-8-7 ! 4F	1 1 1 F1	A 12-1 A C1	Q 11-8 Q D8	1 0 0 1	9	) 12-8-4 < 4C	9 9 9 F9	I 12-9 I C9	Y 0-8 Y E8
0 0 1 0	2	≠ 8-4 @ 7C	2 2 2 F2	B 12-2 B C2	R 11-9 R D9	1 0 1 0	10 (A)	* 11-8-4 * 5C	: 8-2 : 7A	J 11-1 J D1	Z 0-9 Z E9
0 0 1 1	3	≡ 0-8-6 > 6E	3 3 3 F3	C 12-3 C C3	S 0-2 S E2	1 0 1 1	11 (B)	+ 12 & 50	; 12-8-7 ↓ 4F	K 11-2 K D2	[8-7 " 7F
0 1 0 0	4	$ 11-8-3 $ 5B	4 4 4 F4	D 12-4 D C4	T 0-3 T E3	1 1 0 0	12 (C)	, 0-8-3 , 6B	< 12-0 { C0	L 11-3 L D3	≥ 12-8-5 (4D
0 1 0 1	5	% 8-6 = 7E	5 5 5 F5	E 12-5 E C5	U 0-4 U E4	1 1 0 1	13 (D)	- 11 - 60	= 8-3 # 7B	M 11-4 M D4	] 0-8-2 \ E0
0 1 1 0	6	^ 0-8-7 ? 6F	6 6 6 F6	F 12-6 F C6	V 0-5 V E5	1 1 1 0	14 (E)	. 12-8-3 . 4B	> 11-8-7 ¬ 5F	N 11-5 N D5	¬ 12-8-6 + 4E
0 1 1 1	7	↑ 0-8-5) 5D	7 7 7 F7	G 12-7 G C7	W 0-6 W E6	1 1 1 1	15 (F)	/ 0-1 / 61	↓ 11-8-6 ; 5E	O 11-6 O D6	→ 0-8-5 _ 6D

LEGEND:

CDC Character → A ← 026 or EBCDIC Card Code
12-1
EBCDIC Character → A C1 ← EBCDIC Code (Hexadecimal)

TABLE A-14. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 1, 2, AND 5 THROUGH 8

Terminal ASCII (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	ASCII Graphic	Control Character[††]	Octal Code[†††]	ASCII Graphic	Control Character
000		NUL or @⃝	000		null
003	▲	ETX or Ⓒ	003		end of text
005		ENQ or WRU or Ⓔ	005		enquiry
006		ACK or RU or Ⓕ	006		positive acknowledgement
011		HT or Ⓘ	011		horizontal tabulate
012		LF or NL or ↓ or Ⓙ	012		linefeed
014		FF or FORM or Ⓛ	014		formfeed
017	≫	SI or Ⓞ	017		shift in

TABLE A-14. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 1, 2, AND 5 THROUGH 8 (Contd)

Terminal ASCII (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code	ASCII Graphic	Control Character††	Octal Code†††	ASCII Graphic	Control Character
021		DC1 or X-ON or Ⓠ	021		device control 1
022		DC2 or TAPE or Ⓡ	022		device control 2
024		DC4 or TAPE or Ⓣ	024		device control 4
027		ETB or Ⓦ	027		end transmission block
030		CAN or CLEAR or Ⓧ	030		cancel
033		ESC or ESCAPE or Ⓛ	033		escape
035		GS or]⃝	035		group separator
036		RS or ^⃝	036		record separator
041	!		041	!	
042	"		042	"	
044	$		044	$	
047	'		047	'	
050	(		050	(	
053	+		053	+	
055	-		055	-	
056	.		056	.	
060	0		060	0	
063	3		063	3	
065	5		065	5	
066	6		066	6	
071	9		071	9	
072	:		072	:	
074	<		074	<	
077	?		077	?	
101	A		101	A	
102	B		102	B	
104	D		104	D	
107	G		107	G	
110	H		110	H	
113	K		113	K	
115	M		115	M	
116	N		116	N	
120	P		120	P	
123	S		123	S	
125	U		125	U	
126	V		126	V	
131	Y		131	Y	
132	Z		132	Z	
134	\		134	\	
137	_ or ←		137	_	
140	`		140	`	
143	c		143	c	
145	e		145	e	
146	f		146	f	
151	i		151	i	
152	j		152	j	
154	l		154	l	
157	o		157	o	
161	q		161	q	
162	r		162	r	
164	t		164	t	
167	w		167	w	
170	x		170	x	
173	{		173	{	
174	¦ or ↑ or |		174	¦	
175	}		175	}	
176	~ or ¬		176	~	
201		SOH or Ⓐ	001		start of header
202		STX or Ⓑ	002		start of text
204		EOT or Ⓓ	004		end of transmission
207		BELL or Ⓖ	007		bell
210		BS or ← or Ⓗ	010		backspace
213		VT or Ⓚ	013		vertical tabulate
215		CR or RETURN or Ⓜ	015		carriage return

TABLE A-14. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 1, 2, AND 5 THROUGH 8 (Contd)

Terminal ASCII (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code	ASCII Graphic	Control Character††	Octal Code†††	ASCII Graphic	Control Character
216	≪	SO or Ⓝ	016		shift out
220		DLE or Ⓟ	020		data link escape
223		DC3 or X-OFF or Ⓢ	023		device control 3
225		NAK or→or Ⓤ	025		negative acknowledgement
226		SYN or LINE CLEAR or Ⓥ	026		synchronous idle
231		EM or RESET or Ⓨ	031		end of medium
232		SUB or↑or Ⓩ	032		substitute
234		FS or⦸	034		file separator
237		US or ⊖	037		unit separator
240	SPACE or blank		040	space	
243	#		043	#	
245	%		045	%	
246	&		046	&	
251	)		051	)	
252	*		052	*	
254	,		054	,	
257	/		057	/	
261	1		061	1	
262	2		062	2	
264	4		064	4	
267	7		067	7	
270	8		070	8	
273	;		073	;	
275	=		075	=	
276	>		076	>	
300	@		100	@	
303	C		103	C	
305	E		105	E	
306	F		106	F	
311	I		111	I	
312	J		112	J	
314	L		114	L	
317	O		117	O	
321	Q		121	Q	
322	R		122	R	
324	T		124	T	
327	W		127	W	
330	X		130	X	
333	[		133	[	
335	]		135	]	
336	∧or¬		136	∧	
341	a		141	a	
342	b		142	b	
344	d		144	d	
347	g		147	g	
350	h		150	h	
353	k		153	k	
355	m		155	m	
356	n		156	n	
360	p		160	p	
363	s		163	s	
365	u		165	u	
366	v		166	v	
371	y		171	y	
372	z		172	z	
377	■	DEL or RUBOUT	177		delete

†Shown with even parity, which is the default for these terminal classes (unless PA=N, an application program receives the same code as in character mode).

††A circle around a character indicates that the character key is pressed in conjunction with a CTL, CTRL, CNTRL, or CONTROL key to generate the code.

†††Shown with zero parity (eighth or uppermost bit is always zero).

TABLE A-15. CHARACTER CODE TRANSLATIONS, ASCII TERMINAL CLASSES 10 AND 15

Terminal ASCII†					Network ASCII (Character Mode Use)		
Octal Code††	Keyboard or Printer Graphic		029 Card Code	026 Card Code	Octal Code†††		Graphic
	ASCII	CDC			Input or Output	Console Output Only	
040	blank	Blank	no punch	no punch	040		space
043	#		8-3	0-8-6	043		#
045	%	%	0-8-4	8-6	045		%
046	&		12	0-8-7	046		&
051	)	)	11-8-5	12-8-4	051		)
052	*	*	11-8-4	11-8-4	052		*
054	,	,	0-8-3	0-8-3	054		,
057	/	/	0-1	0-1	057		/
061	1	1	1	1	061		1
062	2	2	2	2	062		2
064	4	4	4	4	064		4
067	7	7	7	7	067		7
070	8	8	8	8	070		8
073	;	;	11-8-6	12-8-7	073		;
075	=	=	8-6	8-3	075		=
076	>	>	0-8-6	11-8-7	076		>
100	@	≤	8-4	11-8-5	100	140	@
103	C	C	12-3	12-3	103	143	C
105	E	E	12-5	12-5	105	145	E
106	F	F	12-6	12-6	106	146	F
111	I	I	12-9	12-9	111	151	I
112	J	J	11-1	11-1	112	152	J
114	L	L	11-3	11-3	114	154	L
117	O	O	11-6	11-6	117	157	O
121	Q	Q	11-8	11-8	121	161	Q
122	R	R	11-9	11-9	122	162	R
124	T	T	0-3	0-3	124	164	T
127	W	W	0-6	0-6	127	167	W
130	X	X	0-7	0-7	130	170	X
133	[	[	12-0 or 12-8-2	12-0 or 12-8-3	133	173	[
135	]	]	11-0 or 11-8-2	11-0 or 11-8-2	135	175	]
136	^	¬	11-8-7	12-8-6	136	176	^
241	!		12-8-7	0-8-2	041		!
242	"	≠	8-7	8-4	042		"
244	$	$	11-8-3	11-8-3	044		$
247	'		8-5	8-7	047		'
250	(	(	12-8-5	0-8-4	050		(
253	+	+	12-8-6	12	053		+
255	-	-	11	11	055		-
256	.	.	12-8-3	12-8-3	056		.
260	0	0	0	0	060		0
263	3	3	3	3	063		3
265	5	5	5	5	065		5
266	6	6	6	6	066		6
271	9	9	9	9	071		9
272	:	:	8-2	8-2	072		:
274	<	<	12-8-4	8-5	074		<
277	?	↓	0-8-7	11-8-6	077		?
301	A	A	12-1	12-1	101	141	A
302	B	B	12-2	12-2	102	142	B
304	D	D	12-4	12-4	104	144	D
307	G	G	12-7	12-7	107	147	G
310	H	H	12-8	12-8	110	150	H
313	K	K	11-2	11-2	113	153	K
315	M	M	11-4	11-4	115	155	M
316	N	N	11-5	11-5	116	156	N
320	P	P	11-7	11-7	120	160	P
323	S	S	0-2	0-2	123	163	S
325	U	U	0-4	0-4	125	165	U

TABLE A-15. CHARACTER CODE TRANSLATIONS, ASCII TERMINAL CLASSES 10 AND 15 (Contd)

Terminal ASCII[†]					Network ASCII (Character Mode Use)		
Octal Code[††]	Keyboard or Printer Graphic		029 Card Code	026 Card Code	Octal Code[†††]		Graphic
	ASCII	CDC			Input or Output	Console Output Only	
326	V	V	0-5	0-5	126	166	V
331	Y	Y	0-8	0-8	131	171	Y
332	Z	Z	0-9	0-9	132	172	Z
334	\	≥	0-8-2	12-8-5	134	174	\
337	_	↱	0-8-5	0-8-5	135	177	_

[†] Escape codes and control codes are not listed. These are not treated as network data and have no equivalent character mode translations.

[††] Shown with odd parity, the only possible parity selection for these terminal classes.

[†††] Shown with zero parity (eighth or uppermost bit is always zero). During output, codes 000 through 037_8 are converted to code 040_8 (blank). Codes for lowercase ASCII characters sent to the console are converted to the codes for the equivalent uppercase characters supported by the terminal, as shown; codes for these lowercase characters cannot be sent to batch devices without causing errors.

TABLE A-16. CHARACTER CODE TRANSLATIONS, BCD TERMINAL CLASSES 10 AND 15

Terminal External BCD[†]					Network ASCII (Character Mode Use)		
Octal Code[††]	Keyboard or Printer Graphic		029 Card Code	026 Card Code	Octal Code[†††]		Graphic
	ASCII	CDC			Input or Output	Console Output Only	
020	:	:	8-2	8-2	072		:
040	-	-	11	11	055		-
043	L	L	11-3	11-3	114	154	L
045	N	N	11-5	11-5	116	156	N
046	O	O	11-6	11-6	117	157	O
051	R	R	11-9	11-9	122	162	R
052	!	∨	12-8-7	11-0	041		!
054	*	*	11-8-4	11-8-4	052		*
057	>	>	0-8-6	11-8-7	076		>
061	A	A	12-1	12-1	101	141	A
062	B	B	12-2	12-2	102	142	B
064	D	D	12-4	12-4	104	144	D
067	G	G	12-7	12-7	107	147	G
070	H	H	12-8	12-8	108	148	H
073	.	.	12-8-3	12-8-3	056		.
075			12-0	12-8-5	134	174	\
103	3	3	3	3	063		3
105	5	5	5	5	065		5
106	6	6	6	6	066		6
111	9	9	9	9	071		9
112	0	0	0	0	060		0
114	"	=	8-5 or 8-7	8-4	042		"
117	[	[	8-4	8-7	133	173	[
121	/	/	0-1	0-1	057		/
122	S	S	0-2	0-2	123	163	S
124	U	U	0-4	0-4	125	165	U
127	X	X	0-7	0-7	130	170	X
130	Y	Y	0-8	0-8	131	171	Y
133	,	,	0-8-3	0-8-3	054		,
135	_	↱	0-8-5	0-8-5	137	177	_

TABLE A-16. CHARACTER CODE TRANSLATIONS, BCD TERMINAL CLASSES 10 AND 15 (Contd)

	Terminal External BCD[†]				Network ASCII (Character Mode Use)		
Octal Code[††]	Keyboard or Printer Graphic		029 Card Code	026 Card Code	Octal Code[†††]		Graphic
	ASCII	CDC			Input or Output	Console Output Only	
136	#	≡	8-3	0-8-6	043		#
241	J	J	11-1	11-1	112	152	J
242	K	K	11-2	11-2	113	153	K
244	M	M	11-4	11-4	115	155	M
247	P	P	11-7	11-7	120	160	P
250	Q	Q	11-8	11-8	121	161	Q
253	$	$	11-8-3	11-8-3	044		$
255	'	↑	12	11-8-5	047		'
256	?	↓	12-8-7	11-8-6	077		?
263	C	C	12-3	12-3	103	143	C
265	E	E	12-5	12-5	105	145	E
266	F	F	12-6	12-6	106	146	F
271	I	I	12-9	12-9	111	151	I
272	<	<	12-8-4	12-0	074		<
274	)	)	11-8-5	12-8-4	051		)
277	;	;	11-8-6	12-8-7	073		;
301	1	1	1	1	061		1
302	2	2	2	2	062		2
304	4	4	4	4	064		4
307	7	7	7	7	067		7
310	8	8	8	8	070		8
313	=	=	8-6	8-3	075		=
315	@	<	11-8-7	8-5	100	140	@
316	%	%̄	0-8-4	8-6	045		%
320	blank	blank	no punch	no punch	040		space
323	T	T	0-3	0-3	124	164	T
325	V	V	0-5	0-5	126	166	V
326	W	W	0-6	0-6	127	167	W
331	Z	Z	0-9	0-9	132	172	Z
332	]	]	0-8-2	0-8-2	135	175	]
334	(	(	12-8-5	0-8-4	050		(
337	&	∧	0-8-7	0-8-7	046		&
320	∧ or blank	¬ or ■ or none	none	none		136, 176	^[§]

[†] Escape codes and control codes are not listed. These are not treated as network data and have no equivalent character mode translations.

[††] Shown with odd parity, the only possible parity selection for these terminal classes.

[†††] Shown with zero parity (eighth or uppermost bit is always zero). During output, codes 000 through 037_8 are converted to code 320_8 (blank). Codes for lowercase ASCII characters sent to the console are converted to the codes for the equivalent uppercase characters supported by the terminal, as shown; codes for these lowercase characters cannot be sent to batch devices without causing errors.

[§] Input and output of this symbol is not possible on some terminals. BCD transmission conventions support the rubout symbol ■ as an internal terminal memory parity error indicator instead. The ASCII codes 136_8 and 176_8 are output as a blank.

TABLE A-17. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 11, 12, AND 13

Terminal ASCII (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	ASCII Graphic	Control Character[††]	Octal Code[†††]	ASCII Graphic	Control Character
001		SOH or Ⓐ	001		start of header
002		STX or Ⓑ	002		start of text
004		EOT or Ⓓ	004		end of transmission
007		BELL or Ⓖ	007		bell
010		BS or ← or Ⓗ	010		backspace
013		VT or Ⓚ	013		vertical tabulate
015		CR or RETURN or Ⓜ	015		carriage return
016		SO or Ⓝ	016		shift out
020		DLE or Ⓟ	020		data link escape
025		NAK or → or Ⓤ	025		negative acknowledgement
026		SYN or LINE CLEAR or Ⓥ	026		synchronous idle
031		EM or RESET or Ⓨ	031		end of medium
032		SUB or ↑ or Ⓩ	032		substitute
034		FS or ⦸	034		file separator
037		US or ⊖	037		unit separator
040	SPACE or blank		040	space	
043	#		043	#	
045	%		045	%	
046	&		046	&	
051	)		051	)	
052	*		052	*	
054	,		054	,	
057	/		057	/	
061	1		061	1	
062	2		062	2	
064	4		064	4	
067	7		067	7	
070	8		070	8	
073	;		073	;	
075	=		075	=	
076	>		076	>	
100	@		100	@	
103	C		103	C	
105	E		105	E	
106	F		106	F	
111	I		111	I	
112	J		112	J	
114	L		114	L	
117	O		117	O	
121	Q		121	Q	
122	R		122	R	
124	T		124	T	
127	W		127	W	
130	X		130	X	
133	[		133	[	
135	]		135	]	
136	∧ or ¬		136	∧	
141	a		141	a	
142	b		142	b	
144	d		144	d	
147	g		147	g	
150	h		150	h	
153	k		153	k	
155	m		155	m	
156	n		156	n	
160	p		160	p	
163	s		163	s	
165	u		165	u	
166	v		166	v	
171	y		171	y	
172	z		172	z	
177	■	DEL or RUBOUT	177		delete

TABLE A-17. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 11, 12, AND 13 (Contd)

Terminal ASCII (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code†	ASCII Graphic	Control Character††	Octal Code†††	ASCII Graphic	Control Character
200		NUL or @⃝	000		null
203		ETX or Ⓒ or SEND	003		end of text
205		ENQ or WRU or Ⓔ	005		enquiry
206		ACK or RU or Ⓕ	006		positive acknowledgement
211		HT or Ⓘ	011		horizontal tabulate
212		LF or NL or ↓ or Ⓙ or NEW LINE	012		linefeed
214		FF or FORM or Ⓛ	014		formfeed
217		SI or Ⓞ	017		shift in
221		DC1 or X-ON or Ⓠ	021		device control 1
222		DC2 or TAPE or Ⓡ	022		device control 2
223		DC3 or X-OFF or Ⓢ	023		device control 3
224		DC4 or TAPE or Ⓣ	024		device control 4
227		ETB or Ⓦ	027		end transmission block
230		CAN or CLEAR or Ⓧ	030		cancel
233		ESC or ESCAPE or [⃝	033		escape
235		GS or]⃝	035		group separator
236		RS or ^⃝	036		record separator
241	!		041	!	
242	"		042	"	
244	$		044	$	
247	'		047	'	
250	(		050	(	
253	+		053	+	
255	-		055	-	
256	.		056	.	
260	0		060	0	
263	3		063	3	
265	5		065	5	
266	6		066	6	
271	9		071	9	
272	:		072	:	
274	<		074	<	
277	?		077	?	
301	A		101	A	
302	B		102	B	
304	D		104	D	
307	G		107	G	
310	H		110	H	
313	K		113	K	
315	M		115	M	
316	N		116	N	
320	P		120	P	
323	S		123	S	
325	U		125	U	
326	V		126	V	
331	Y		131	Y	
332	Z		132	Z	
334	\		134	\	
337	_ or ←		137	_	
340	`		140	`	
343	c		143	c	
345	e		145	e	
346	f		146	f	
351	i		151	i	
352	j		152	j	
354	l		154	l	
357	o		157	o	
361	q		161	q	
362	r		162	r	
364	t		164	t	
367	w		167	w	
370	x		170	x	

TABLE A-17. CHARACTER CODE TRANSLATIONS, CONSOLE TERMINAL CLASSES 11, 12, AND 13 (Contd)

Terminal ASCII (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code†	ASCII Graphic	Control Character††	Octal Code†††	ASCII Graphic	Control Character
373	{		173	{	
374	¦ or ↑ or |		174	¦	
375	}		175	}	
376	~ or ¬		176	~	

†Shown with odd parity, which is the default for these terminal classes (unless PA=N, an application program receives the same code as in character mode).

††A circle around a character indicates that the character key is pressed in conjunction with a CTL, CTRL, CNTRL, or CONTROL key to generate the code.

†††Shown with zero parity (eighth or uppermost bit is always zero).

TABLE A-18. ASCII CHARACTER CODE TRANSLATIONS, EBCD CONSOLE TERMINAL CLASS 4

Terminal EBCD (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code†	EBCD Graphic††	Control Character	Octal Code†††	ASCII Graphic	Control Character
000	space		040	space	
001	_ or -		137 or 055	_ or -	
002	¢ or @		140 or 100	` or @	
003	+ or &		053 or 046	+ or &	
004	* or 8		052 or 070	* or 8	
005	Q or q		121 or 161	Q or q	
006	Y or y		131 or 171	Y or y	
007	H or h		110 or 150	H or h	
010	: or 4		072 or 064	: or 4	
011	M or m		115 or 155	M or m	
012	U or u		125 or 165	U or u	
013	D or d		104 or 144	D or d	
014		PN or PUNCH ON	021		device control 1 (tape on)
015		RES or RESTORE	000		null
016		BY or BYPASS	000		null
017		PF or PUNCH OFF	023		device control 3 (tape off)
020	< or 2		074 or 062	< or 2	
021	K or k		113 or 153	K or k	
022	S or s		123 or 163	S or s	
023	B or b		102 or 142	B or b	
024	) or 0		051 or 060	) or 0	
025		undefined	000		null
026		undefined	000		null
027		undefined	000		null
030	' or 6		041 or 066	' or 6	
031	O or o		117 or 157	O or o	
032	W or w		127 or 167	W or w	
033	F or f		106 or 146	F or f	
034		UCS or UPPERCASE	017		shift in§
035		BS or BACKSPACE	010		backspace
036		EOB	027		end transmission block§
037		LCS or LOWERCASE	016		shift out§
040	= or 1		075 or 061	= or 1	
041	J or j		112 or 152	J or j	
042	? or /		077 or 057	? or /	
043	A or a		101 or 141	A or a	
044	(or 9		050 or 071	(or 9	
045	R or r		122 or 162	R or r	
046	Z or z		132 or 172	Z or z	
047	I or i		111 or 151	I or i	

TABLE A-18. ASCII CHARACTER CODE TRANSLATIONS, EBCD CONSOLE TERMINAL CLASS 4 (Contd)

Terminal EBCD (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	EBCD Graphic[††]	Control Character	Octal Code[†††]	ASCII Graphic	Control Character
050	% or 5		045 or 065	% or 5	
051	N or n		116 or 156	N or n	
052	V or v		126 or 166	V or v	
053	E or e		105 or 145	E or e	
054		RO or READER STOP	000		null
055		NL or CR or RETURN	015		carriage return
056		LF or LINE FEED	012		line feed
057		HT or TAB	006		horizontal tabulate
060	; or 3		073 or 063	; or 3	
061	L or l		114 or 154	L or l	
062	T or t		124 or 164	T or t	
063	C or c		103 or 143	C or c	
064	" or #		042 or 043	" or #	
065	! or $		041 or 044	! or $	
066	| or ,		174 or 054	¦ or ,	
067	¬ or .		136 or 056	^ or .	
070	> or 7		076 or 067	> or 7	
071	P or p		120 or 160	P or p	
072	X or x		130 or 170	X or x	
073	G or g		107 or 147	G or g	
074		EOT	004		end of transmission[§]
075		IL or IDLE or NULL	000		null
076		PRE or PREFIX	001		start of header[§]
077		DEL	177		delete
100	space		040	space	
101	_ or -		137 or 055	_ or -	
102	¢ or @		140 or 100	` or @	
103	+ or &		053 or 046	+ or &	
104	* or 8		052 or 070	* or 8	
105	Q or q		121 or 161	Q or q	
106	Y or y		131 or 171	Y or y	
107	H or h		110 or 150	H or h	
110	: or 4		072 or 064	: or 4	
111	M or m		115 or 155	M or m	
112	U or u		125 or 165	U or u	
113	D or d		104 or 144	D or d	
114		PN or PUNCH ON	021		device control 1 (tape on)
115		RES or RESTORE	000		null
116		BY or BYPASS	000		null
117		PF or PUNCH OFF	023		device control 3 (tape off)
120	< or 2		074 or 062	< or 2	
121	K or k		113 or 153	K or k	
122	S or s		123 or 163	S or s	
123	B or b		102 or 142	B or b	
124	) or 0		051 or 060	) or 0	
125		undefined	000		null
126		undefined	000		null
127		undefined	000		null
130	' or 6		041 or 066	' or 6	
131	O or o		117 or 157	O or o	
132	W or w		127 or 167	W or w	
133	F or f		106 or 146	F or f	
134		UCS or UPPERCASE	017		shift in[§]
135		BS or BACKSPACE	010		backspace
136		EOB	027		end transmission block[§]
137		LCS or LOWERCASE	016		shift out[§]
140	= or 1		075 or 061	= or 1	
141	J or j		112 or 152	J or j	
142	? or /		077 or 057	? or /	
143	A or a		101 or 141	A or a	
144	(or 9		050 or 071	(or 9	
145	R or r		122 or 162	R or r	
146	Z or z		132 or 172	Z or z	
147	I or i		111 or 151	I or i	

TABLE A-18. ASCII CHARACTER CODE TRANSLATIONS, EBCD CONSOLE TERMINAL CLASS 4 (Contd)

Terminal EBCD (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	EBCD Graphic[††]	Control Character	Octal Code[†††]	ASCII Graphic	Control Character
150	% or 5		045 or 065	% or 5	
151	N or n		116 or 156	N or n	
152	V or v		126 or 166	V or v	
153	E or e		105 or 145	E or e	
154		RO or READER STOP	000		null
155		NL or CR or RETURN	015		carriage return
156		LF or LINE FEED	012		line feed
157		HT or TAB	006		horizontal tabulate
160	; or 3		073 or 063	; or 3	
161	L or l		114 or 154	L or l	
162	T or t		124 or 164	T or t	
163	C or c		103 or 143	C or c	
164	" or #		042 or 043	" or #	
165	! or $		041 or 044	! or $	
166	| or ,		174 or 054	¦ or ,	
167	¬ or .		136 or 056	^ or .	
170	> or 7		076 or 067	> or 7	
171	P or p		120 or 160	P or p	
172	X or x		130 or 170	X or x	
173	G or g		107 or 147	G or g	
174		EOT	004		end of transmission[§]
175		IL or IDLE or NULL	000		null
176		PRE or PREFIX	001		start of header[§]
177		DEL	177		delete
000	space[§§]		133 thru 135	[or \ or]	
000	space[§§]		140	`	
000	space[§§]		173	{	
000	space[§§]		175 or 176	} or ~	
175		IL or IDLE or NULL[§§]	002		start of text
175		IL or IDLE or NULL[§§]	003		end of text
175		IL or IDLE or NULL[§§]	005		enquire
175		IL or IDLE or NULL[§§]	007		bell
175		IL or IDLE or NULL[§§]	013 or 014		vertical tabulate or form feed
175		IL or IDLE or NULL[§§]	020		data link escape
175		IL or IDLE or NULL[§§]	022		device control 2
175		IL or IDLE or NULL[§§]	024 thru 026		device control 4, negative acknowledge, or synchronize
175		IL or IDLE or NULL[§§]	030 thru 037		cancel, end of media, substitute, escape, file separator, group separator, record separator, or unit separator

[†] Shown with odd and even parity; odd parity is the default for this terminal class. (Unless PA=N, the application program receives the same code as in character mode.)

[††] Each input line is assumed to begin in lowercase. Input characters are translated to lowercase ASCII characters unless prefixed by the UCS code. Once a case shift occurs, it remains in effect until another case shift code is received, the page width is reached, or the line is transmitted to the host computer. During output, case is preserved by insertion of case shift codes where needed.

[†††] Shown with zero parity (eighth or uppermost bit is always zero).

[§] Not transmitted to the host computer after translation during input.

[§§] Output translation only.

TABLE A-19. APL CHARACTER CODE TRANSLATIONS, EBCD CONSOLE TERMINAL CLASS 4

Terminal EBCD-APL (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code†	EBCD-APL Graphic††	Control Character	Octal Code†††	ASCII-APL Graphic	Control Character
000	space		040	space	
001	_ or +		137 or 053	_ or +	
002	→ or ←		161 or 160	→ or ←	
003	÷ or X		045 or 146	÷ or X	
004	≠ or 8		042 or 070	≠ or 8	
005	? or Q		077 or 121	? or Q	
006	↑ or Y		171 or 131	↑ or Y	
007	Δ or H		150 or 110	Δ or H	
010	≤ or 4		100 or 064	≤ or 4	
011	| or M		174 or 115	| or M	
012	↓ or U		165 or 125	↓ or U	
013	⌊ or D		144 or 104	⌊ or D	
014		undefined	000		null
015		undefined	000		null
016		undefined	000		null
017		undefined	000		null
020	- or 2		055 or 062	- or 2	
021	⊣ or K		153 or 113	⊣ or K	
022	⌈ or S		163 or 123	⌈ or S	
023	⊥ or B		142 or 102	⊥ or B	
024	∧ or 0		046 or 060	∧ or 0	
025		undefined	000		null
026		undefined	000		null
027		undefined	000		null
030	≥ or 6		174 or 066	≥ or 6	
031	○ or O		157 or 117	○ or O	
032	ω or W		167 or 127	ω or W	
033	¯ or F		136 or 106	¯ or F	
034		UCS or UPPERCASE	017		shift in§
035		BS or BACKSPACE	010		backspace
036		EOB	027		end transmission block§
037		LCS or LOWERCASE	016		shift out§
040	" or 1		042 or 061	" or 1	
041	∘ or J		152 or 112	∘ or J	
042	\ or /		134 or 057	\ or /	
043	∝ or A		141 or 101	∝ or A	
044	∨ or 9		041 or 071	∨ or 9	
045	ρ or R		162 or 122	ρ or R	
046	⊂ or Z		172 or 132	⊂ or Z	
047	⍳ or I		151 or 111	⍳ or I	
050	= or 5		075 or 065	= or 5	
051	⊤ or N		156 or 116	⊤ or N	
052	∪ or V		166 or 126	∪ or V	
053	∊ or E		145 or 105	∊ or E	
054		undefined	000		null
055		NL or CR or RETURN	015		carriage return
056		LF or LINE FEED	012		line feed
057		HT or TAB	006		horizontal tabulate
060	< or 3		074 or 063	< or 3	
061	□ or L		154 or 114	□ or L	
062	~ or T		164 or 124	~ or T	
063	∩ or C		143 or 103	∩ or C	
064	) or]		051 or 135	) or]	
065	(or [		050 or 133	(or [	
066	; or ,		073 or 054	; or ,	
067	: or .		072 or 056	: or .	
070	> or 7		076 or 067	> or 7	
071	* or P		052 or 120	* or P	
072	⊃ or X		170 or 130	⊃ or X	
073	∇ or G		147 or 107	∇ or G	
074		EOT	004		end of transmission§
075		IL or IDLE or NULL	000		null

TABLE A-19. APL CHARACTER CODE TRANSLATIONS, EBCD CONSOLE TERMINAL CLASS 4 (Contd)

Terminal EBCD-APL (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code†	EBCD-APL Graphic††	Control Character	Octal Code†††	ASCII-APL Graphic	Control Character
076		PRE or PREFIX	001		start of header§
077		DEL	177		delete
100	space		040	space	
101	_ or +		137 or 053	_ or +	
102	→ or ←		161 or 160	→ or ←	
103	÷ or X		045 or 146	÷ or X	
104	≠ or 8		042 or 070	≠ or 8	
105	? or Q		077 or 121	? or Q	
106	↑ or Y		171 or 131	↑ or Y	
107	Δ or H		150 or 110	Δ or H	
110	≤ or 4		100 or 064	≤ or 4	
111	⌈ or M		174 or 115	⌈ or M	
112	↓ or U		165 or 125	↓ or U	
113	⌊ or D		144 or 104	⌊ or D	
114		undefined	000		null
115		undefined	000		null
116		undefined	000		null
117		undefined	000		null
120	- or 2		055 or 062	- or 2	
121	⊣ or K		153 or 113	⊣ or K	
122	⌐ or S		163 or 123	⌐ or S	
123	⊥ or B		142 or 102	⊥ or B	
124	∧ or 0		046 or 060	∧ or 0	
125		undefined	000		null
126		undefined	000		null
127		undefined	000		null
130	≥ or 6		174 or 066	≥ or 6	
131	○ or O		157 or 117	○ or O	
132	ω or W		167 or 127	ω or W	
133	¯ or F		136 or 106	¯ or F	
134		UCS or UPPERCASE	017		shift in§
135		BS or BACKSPACE	010		backspace
136		EOB	027		end transmission block§
137		LCS or LOWERCASE	016		shift out§
140	" or 1		042 or 061	" or 1	
141	° or J		152 or 112	° or J	
142	\ or /		134 or 057	\ or /	
143	∝ or A		141 or 101	∝ or A	
144	∨ or 9		041 or 071	∨ or 9	
145	ρ or R		162 or 122	ρ or R	
146	⊂ or Z		172 or 132	⊂ or Z	
147	ι or I		151 or 111	ι or I	
150	= or 5		075 or 065	= or 5	
151	τ or N		156 or 116	τ or N	
152	∪ or V		166 or 126	∪ or V	
153	∈ or E		145 or 105	∈ or E	
154		undefined	000		null
155		NL or CR or RETURN	015		carriage return
156		LF or LINE FEED	012		line feed
157		HT or TAB	006		horizontal tabulate
160	< or 3		074 or 063	< or 3	
161	□ or L		154 or 114	□ or L	
162	~ or T		164 or 124	~ or T	
163	∩ or C		143 or 103	∩ or C	
164	) or]		051 or 135	) or]	
165	(or [		050 or 133	(or [	
166	; or ,		073 or 054	; or ,	
167	: or .		072 or 056	: or .	
170	> or 7		076 or 067	> or 7	
171	* or P		052 or 120	* or P	
172	⊃ or X		170 or 130	⊃ or X	
173	∇ or G		147 or 107	∇ or G	
174		EOT	004		end of transmission§

TABLE A-19. APL CHARACTER CODE TRANSLATIONS, EBCD CONSOLE TERMINAL CLASS 4 (Contd)

Terminal EBCD-APL (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	EBCD-APL Graphic[††]	Control Character	Octal Code[†††]	ASCII-APL Graphic	Control Character
175		IL or IDLE or NULL	000		null
176		PRE or PREFIX	001		start of header[§]
177		DEL	177		delete
000	space[§§]		047	'	
000	space[§§]		140	◇	
000	space[§§]		173	{	
000	space[§§]		175	}	
175		IL or IDLE or NULL[§§]	002		start of text
175		IL or IDLE or NULL[§§]	003		end of text
175		IL or IDLE or NULL[§§]	005		enquire
175		IL or IDLE or NULL[§§]	007		bell
175		IL or IDLE or NULL[§§]	013 or 014		vertical tabulate or form feed
175		IL or IDLE or NULL[§§]	020 thru 026		data link escape, device control 1 thru device control 4, negative acknowledge, or synchronize
175		IL or IDLE or NULL[§§]	030 thru 037		cancel, end of media, substitute, escape, file separator, group separator, record separator, or unit separator

[†] Shown with odd and even parity; odd parity is the default for this terminal class. (Unless PA=N, the application program receives the same code as in character mode.)

[††] Each input line is assumed to begin in lowercase. Input characters are translated to lowercase ASCII characters unless prefixed by the UCS code. Once a case shift occurs, it remains in effect until another case shift code is received, the page width is reached, or the line is transmitted to the host computer. During output, case is preserved by insertion of case shift codes where needed.

[†††] Shown with zero parity (eighth or uppermost bit is always zero).

[§] Not transmitted to the host computer after translation during input.

[§§] Output translation only.

TABLE A-20. ASCII CHARACTER CODE TRANSLATIONS, CORRESPONDENCE CODE CONSOLE TERMINAL CLASS 4

Terminal Correspondence Code (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	Correspondence Code Graphic[††]	Control Character	Octal Code[†††]	ASCII Graphic	Control Character
000	space		040	space	
001	¼ or ½		137 or 135	[or]	
002	T or t		124 or 164	T or t	
003	J or j		112 or 152	J or j	
004	$ or 4		044 or 064	$ or 4	
005	O or o		117 or 157	O or o	
006	L or l		114 or 154	L or l	
007	? or /		077 or 057	? or /	
010	% or 5		045 or 065	% or 5	
011	" or '		042 or 041	" or '	
012	E or e		105 or 145	E or e	
013	P or p		120 or 160	P or p	

TABLE A-20. ASCII CHARACTER CODE TRANSLATIONS, CORRESPONDENCE CODE CONSOLE TERMINAL CLASS 4 (Contd)

Terminal Correspondence Code (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	Correspondence Code Graphic[††]	Control Character	Octal Code[†††]	ASCII Graphic	Control Character
014		PN or PUNCH ON	021		device control 1 (tape on)
015		RES or RESTORE	000		null
016		BY or BYPASS	000		null
017		PF or PUNCH OFF	023		device control 3 (tape off)
020	@ or 2		100 or 062	@ or 2	
021	.		056	.	
022	N or n		116 or 156	N or n	
023	+ or =		053 or 075	+ or =	
024	Z or z		132 or 172	Z or z	
025		undefined	000		null
026		undefined	000		null
027		undefined	000		null
030	¢ or 6		041 or 066	! or 6	
031	I or i		111 or 151	I or i	
032	K or k		113 or 153	K or k	
033	Q or q		121 or 161	Q or q	
034		UCS or UPPERCASE	017		shift in[§]
035		BS or BACKSPACE	010		backspace
036		EOB	027		end transmission block[§]
037		LCS or LOWERCASE	016		shift out[§]
040	± or 1		174 or 061	¦ or 1	
041	M or m		115 or 155	M or m	
042	X or x		130 or 170	X or x	
043	G or g		107 or 147	G or g	
044	) or 0		051 or 060	) or 0	
045	S or s		123 or 163	S or s	
046	H or h		110 or 150	H or h	
047	Y or y		131 or 171	Y or y	
050	& or 7		046 or 067	& or 7	
051	R or r		122 or 162	R or r	
052	D or d		104 or 144	D or d	
053	: or ;		072 or 073	: or ;	
054		RO or READER STOP	000		null
055		NL or CR or RETURN	015		carriage return
056		LF or LINE FEED	012		line feed
057		HT or TAB	006		horizontal tabulate
060	# or 3		043 or 063	# or 3	
061	V or v		126 or 166	V or v	
062	U or u		125 or 165	U or u	
063	F or f		106 or 146	F or f	
064	(or 9		050 or 071	(or 9	
065	W or w		127 or 167	W or w	
066	B or b		102 or 142	B or b	
067	_ or -		137 or 055	_ or -	
070	* or 8		052 or 070	* or 8	
071	A or a		101 or 141	A or a	
072	C or c		103 or 143	C or c	
073	,		054	,	
074		EOT	004		end of transmission[§]
075		IL or IDLE or NULL	000		null
076		PRE or PREFIX	033		escape
077		DEL	177		delete
100	space		040	space	
101	¼ or ½		133 or 135	[or]	
102	T or t		124 or 164	T or t	
103	J or j		112 or 152	J or j	
104	$ or 4		044 or 064	$ or 4	
105	0 or o		117 or 157	O or o	
106	L or l		114 or 154	L or l	
107	? or /		077 or 057	? or /	
110	% or 5		045 or 065	% or 5	
111	" or '		042 or 041	" or '	

TABLE A-20. ASCII CHARACTER CODE TRANSLATIONS, CORRESPONDENCE CODE CONSOLE TERMINAL CLASS 4 (Contd)

Terminal Correspondence Code (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	Correspondence Code Graphic[††]	Control Character	Octal Code[†††]	ASCII Graphic	Control Character
112	E or e		105 or 145	E or e	
113	P or p		120 or 160	P or p	
114		PN or PUNCH ON	021		device control 1 (tape on)
115		RES or RESTORE	000		null
116		BY or BYPASS	000		null
117		PF or PUNCH OFF	023		device control 3 (tape off)
120	@ or 2		100 or 062	@ or 2	
121	.		056	.	
122	N or n		116 or 156	N or n	
123	+ or =		053 or 075	+ or =	
124	Z or z		132 or 172	Z or z	
125		undefined	000		null
126		undefined	000		null
127		undefined	000		null
130	¢ or 6		041 or 066	! or 6	
131	I or i		111 or 151	I or i	
132	K or k		113 or 153	K or k	
133	Q or q		121 or 161	Q or q	
134		UCS or UPPERCASE	017		shift in[§]
135		BS or BACKSPACE	010		backspace
136		EOB	027		end transmission block[§]
137		LCS or LOWERCASE	016		shift out[§]
140	± or 1		174 or 061	± or 1	
141	M or m		115 or 155	M or m	
142	X or x		130 or 170	X or x	
143	G or g		107 or 147	G or g	
144	) or 0		051 or 060	) or 0	
145	S or s		123 or 163	S or s	
146	H or h		110 or 150	H or h	
147	Y or y		131 or 171	Y or y	
150	& or 7		046 or 067	& or 7	
151	R or r		122 or 162	R or r	
152	D or d		104 or 144	D or d	
153	: or ;		072 or 073	: or ;	
154		RO or READER STOP	000		null
155		NL or CR or RETURN	015		carriage return
156		LF or LINE FEED	012		line feed
157		HT or TAB	006		horizontal tabulate
160	# or 3		043 or 063	# or 3	
161	V or v		126 or 166	V or v	
162	U or u		125 or 165	U or u	
163	F or f		106 or 146	F or f	
164	(or 9		050 or 071	(or 9	
165	W or w		127 or 167	W or w	
166	B or b		102 or 142	B or b	
167	_ or -		137 or 055	_ or -	
170	* or 8		052 or 070	* or 8	
171	A or a		101 or 141	A or a	
172	C or c		103 or 143	C or c	
173	,		054	,	
174		EOT	004		end of transmission[§]
175		IL or IDLE or NULL	000		null
176		PRE or PREFIX	033		escape
177		DEL	177		delete
000	space[§§]		047	'	
000	space[§§]		134	\	
000	space[§§]		136	^	
000	space[§§]		140	`	
000	space[§§]		173	{	
000	space[§§]		175 or 176	} or ~	
175		IL or IDLE or NULL[§§]	001		start of header
175		IL or IDLE or NULL[§§]	002		start of text

TABLE A-20. ASCII CHARACTER CODE TRANSLATIONS, CORRESPONDENCE CODE CONSOLE TERMINAL CLASS 4 (Contd)

Terminal Correspondence Code (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	Correspondence Code Graphic[††]	Control Character	Octal Code[†††]	ASCII Graphic	Control Character
175		IL or IDLE or NULL[§§]	003		end of text
175		IL or IDLE or NULL[§§]	005		enquire
175		IL or IDLE or NULL[§§]	007		bell
175		IL or IDLE or NULL[§§]	013 or 014		vertical tabulate or form feed
175		IL or IDLE or NULL[§§]	020		data link escape
175		IL or IDLE or NULL[§§]	022		device control 2
175		IL or IDLE or NULL[§§]	024 thru 026		device control 4, negative acknowledge, or synchronize
175		IL or IDLE or NULL[§§]	030 thru 037		cancel, end of media, substitute, file separator, group separator, record separator, or unit separator

[†] Shown with odd and even parity; odd parity is the default for this terminal class. (Unless PA=N, the application program receives the same code as in character mode.)

[††] Each input line is assumed to begin in lowercase. Input characters are translated to lowercase ASCII characters unless prefixed by the UCS code. Once a case shift occurs, it remains in effect until another case shift code is received, the page width is reached, or the line is transmitted to the host computer. During output, case is preserved by insertion of case shift codes where needed.

[†††] Shown with zero parity (eighth or uppermost bit is always zero).

[§] Not transmitted to the host computer after translation during input.

[§§] Output translation only.

TABLE A-21. APL CHARACTER CODE TRANSLATIONS, CORRESPONDENCE CODE CONSOLE TERMINAL CLASS 4

Terminal Correspondence Code (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	Correspondence Code APL Graphic[††]	Control Character	Octal Code[†††]	ASCII-APL Graphic	Control Character
000	space		040	space	
001	→ or ←		161 or 160	→ or ←	
002	~ or T		164 or 124	~ or T	
003	. or J		056 or 112	. or J	
004	≤ or 4		100 or 064	≤ or 4	
005	○ or O		157 or 117	○ or O	
006	□ or L		154 or 114	□ or L	
007	\ or /		134 or 057	\ or /	
010	= or 5		075 or 065	= or 5	
011	) or]		051 or 035	) or]	
012	ϵ or E		145 or 105	ϵ or E	
013	* or P		052 or 120	* or P	
014		undefined	000		null
015		undefined	000		null
016		undefined	000		null
017		undefined	023		null
020	¯ or 2		136 or 062	¯ or 2	
021	: or .		072 or 056	: or .	
022	τ or N		156 or 116	τ or N	

TABLE A-21. APL CHARACTER CODE TRANSLATIONS, CORRESPONDENCE CODE CONSOLE TERMINAL CLASS 4 (Contd)

Terminal Correspondence Code (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code†	Correspondence Code APL Graphic††	Control Character	Octal Code†††	ASCII-APL Graphic	Control Character
023	÷ or X		045 or 146	÷ or X	
024	⊂ or Z		172 or 132	⊂ or Z	
025		undefined	000		null
026		undefined	000		null
027		undefined	000		null
030	≥ or 6		174 or 066	≥ or 6	
031	⍳ or I		151 or 111	⍳ or I	
032	⊣ or K		153 or 113	⊣ or K	
033	? or Q		077 or 121	? or Q	
034		UCS or UPPERCASE	017		shift in§
035		BS or BACKSPACE	010		backspace
036		EOB	027		end transmission block§
037		LCS or LOWERCASE	016		shift out§
040	" or 1		042 or 061	" or 1	
041	| or M		174 or 115	| or M	
042	⊃ or X		170 or 130	⊃ or X	
043	∇ or G		147 or 107	∇ or G	
044	∧ or 0		045 or 060	∧ or 0	
045	⌈ or S		163 or 123	⌈ or S	
046	∆ or H		150 or 110	∆ or H	
047	↑ or Y		171 or 131	↑ or Y	
050	> or 7		076 or 067	> or 7	
051	ρ or R		162 or 122	ρ or R	
052	⌊ or D		144 or 104	⌊ or D	
053	(or [		050 or 133	(or [	
054		undefined	000		null
055		NL or CR or RETURN	015		carriage return
056		LF or LINE FEED	012		line feed
057		HT or TAB	006		horizontal tabulate
060	< or 3		074 or 063	< or 3	
061	∪ or V		166 or 126	∪ or V	
062	↓ or U		165 or 125	↓ or U	
063	_ or F		137 or 106	_ or F	
064	∨ or 9		041 or 071	∨ or 9	
065	ω or W		167 or 127	ω or W	
066	⊥ or B		142 or 102	⊥ or B	
067	- or +		055 or 053	- or +	
070	≠ or 8		042 or 070	≠ or 8	
071	∝ or A		141 or 101	∝ or A	
072	∩ or C		143 or 103	∩ or C	
073	; or ,		073 or 054	; or ,	
074		EOT	004		end of transmission§
075		IL or IDLE or NULL	000		null
076		PRE or PREFIX	033		escape
077		DEL	177		delete
100	space		040	space	
101	→ or ←		161 or 160	→ or ←	
102	~ or T		164 or 124	~ or T	
103	. or J		056 or 112	. or J	
104	≤ or 4		100 or 064	≤ or 4	
105	○ or O		157 or 117	○ or O	
106	□ or L		154 or 114	□ or L	
107	\ or /		134 or 057	\ or /	
110	= or 5		075 or 065	= or 5	
111	) or]		051 or 035	) or]	
112	∊ or E		145 or 105	∊ or E	
113	* or P		052 or 120	* or P	
114		undefined	000		null
115		undefined	000		null
116		undefined	000		null
117		undefined	023		null
120	¯ or 2		136 or 062	¯ or 2	

TABLE A-21. APL CHARACTER CODE TRANSLATIONS, CORRESPONDENCE CODE CONSOLE TERMINAL CLASS 4 (Contd)

Terminal Correspondence Code (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	Correspondence Code APL Graphic[††]	Control Character	Octal Code[†††]	ASCII-APL Graphic	Control Character
121	: or .		072 or 056	: or .	
122	τ or N		156 or 116	τ or N	
123	÷ or X		045 or 146	÷ or X	
124	⊂ or Z		172 or 132	⊂ or Z	
125		undefined	000		null
126		undefined	000		null
127		undefined	000		null
130	≥ or 6		174 or 066	≥ or 6	
131	⍳ or I		151 or 111	⍳ or I	
132	⊣ or K		153 or 113	⊣ or K	
133	? or Q		077 or 121	? or Q	
134		UCS or UPPERCASE	017		shift in[§]
135		BS or BACKSPACE	010		backspace
136		EOB	027		end transmission block[§]
137		LCS or LOWERCASE	016		shift out[§]
140	" or 1		042 or 061	" or 1	
141	| or M		174 or 115	| or M	
142	⊃ or X		170 or 130	⊃ or X	
143	∇ or G		147 or 107	∇ or G	
144	∧ or O		045 or 060	∧ or O	
145	⌈ or S		163 or 123	⌈ or S	
146	Δ or H		150 or 110	Δ or H	
147	↑ or Y		171 or 131	↑ or Y	
150	> or 7		076 or 067	> or 7	
151	ρ or R		162 or 122	ρ or R	
152	⌊ or D		144 or 104	⌊ or D	
153	(or [		050 or 133	(or [	
154		undefined	000		null
155		NL or CR or RETURN	015		carriage return
156		LF or LINE FEED	012		line feed
157		HT or TAB	006		horizontal tabulate
160	< or 3		074 or 063	< or 3	
161	∪ or V		166 or 126	∪ or V	
162	↓ or U		165 or 125	↓ or U	
163	_ or F		137 or 106	_ or F	
164	∨ or 9		041 or 071	∨ or 9	
165	ω or W		167 or 127	ω or W	
166	⊥ or B		142 or 102	⊥ or B	
167	- or +		055 or 053	- or +	
170	≠ or 8		042 or 070	≠ or 8	
171	∝ or A		141 or 101	∝ or A	
172	∩ or C		143 or 103	∩ or C	
173	; or ,		073 or 054	; or ,	
174		EOT	004		end of transmission[§]
175		IL or IDLE or NULL	000		null
176		PRE or PREFIX	033		escape
177		DEL	177		delete
000	space[§§]		047	'	
000	space[§§]		140	◇	
000	space[§§]		173	{	
000	space[§§]		175 or 176	} or ⊢	
175		IL or IDLE or NULL[§§]	001		start of header
175		IL or IDLE or NULL[§§]	002		start of text
175		IL or IDLE or NULL[§§]	003		end of text
175		IL or IDLE or NULL[§§]	005		enquire
175		IL or IDLE or NULL[§§]	007		bell
175		IL or IDLE or NULL[§§]	013 or 014		vertical tabulate or form feed
175		IL or IDLE or NULL[§§]	020		data link escape
175		IL or IDLE or NULL[§§]	022		device control 2

TABLE A-21. APL CHARACTER CODE TRANSLATIONS, CORRESPONDENCE CODE CONSOLE TERMINAL CLASS 4 (Contd)

Terminal Correspondence Code (Transparent Mode Use)			Network ASCII (Character Mode Use)		
Octal Code[†]	Correspondence Code APL Graphic[††]	Control Character	Octal Code[†††]	ASCII-APL Graphic	Control Character
175		IL or IDLE or NULL[§§]	024 thru 026		device control 4, negative acknowledge, or synchronize
175		IL or IDLE or NULL[§§]	030 thru 037		cancel, end of media, substitute, file separator, group separator, record separator, or unit separator

[†] Shown with odd and even parity; odd parity is the default for this terminal class. (Unless PA=N, the application program receives the same code as in character mode.)

[††] Each input line is assumed to begin in lowercase. Input characters are translated to lowercase ASCII characters unless prefixed by the UCS code. Once a case shift occurs, it remains in effect until another case shift code is received, the page width is reached, or the line is transmitted to the host computer. During output, case is preserved by insertion of case shift codes where needed.

[†††] Shown with zero parity (eighth or uppermost bit is always zero).

[§] Not transmitted to the host computer after translation during input.

[§§] Output translation only.

ERROR PROCESSING

Two levels of error processing occur during an NDL Processor run:

- Compilation diagnostic processing
- File access and dayfile message processing

The diagnostic processing performed during the run is documented by the error summary listing produced as part of the listing output file. The NDL job's output file only contains an error summary list if errors were detected.

If any error is detected, the NDL Processor produces one of the diagnostic messages described in the following subsection. Statement processing is completed if feasible. If a fatal error is detected, no validated network definition files or file content summary listings are created and the NDL Processor aborts the job. After such an abort, control of the job transfers to the EXIT area in the control statement portion of the job image.

Informative messages are always placed in the job dayfile. In certain instances, a diagnostic-free job can be aborted because of file access problems; these cases also produce job dayfile messages. The dayfile messages are explained at the end of this appendix.

DIAGNOSTIC MESSAGES

The Network Definition Language source program's error summary lists all errors that occurred in the program with a cross-reference to the statement in the source listing where the error occurred. If the condition was a syntax error, the character position (column) within the statement is also given. Table B-1 explains the error codes used on the error summary. A flag character precedes each statement in the source listing on which the NDL Processor performed a diagnostic function. These characters are

E Indicates a fatal error. File construction continues, but no verification record is written.

W Indicates a warning message for a potential configuration error that does not significantly affect proper file construction. Such errors may actually be operationally fatal errors undetectable by the NDL Processor; for example, when a TC parameter is not declared, but a parameter dependent on the terminal class is declared.

Occurrence of fatal errors listed in table B-1 does not necessarily terminate statement scanning, and additional errors in the same statement can be detected. Occurrence of some errors does terminate statement scanning, so that additional undetected errors beyond the one flagged can exist in any given statement. This prevents a cascade effect in the errors detected in case of the inadvertent omission of a parameter, keyword, or separator.

When a cascade of errors occurs for a single statement, and many of the error diagnostics issued for the statement do not seem to apply to it, the number of characters permitted in a statement unit has probably been exceeded. If the diagnosed statement runs to more than 72 meaningful characters (or if the preceding statement exceeds 72 meaningful characters) not divided into valid statement units, the NDL Processor misinterprets the resulting character stream and can diagnose nonexistent errors.

Use of the three keywords TIPTYPE, DT, and TC can cause partial or complete suspension of diagnostic processing for their respective LINE and TERMINAL statements when a site-defined value is used as a parameter declaration. Occurrence of the reserved words USER1, USER2, USER3, USER4, or USER as valid declarations for these three parameters causes the NDL Processor to suspend diagnostic checking of all parameters on the same statement that have values dependent on the site-defined parameter. Parameters not dependent on the site-defined parameter receive normal diagnostic checking. For example:

- A line with a TIPTYPE of MODE4 can have a terminal with a TC of USER1 and a DT of USER, but still must have CA and TA values as part of its configuration.
- A line with a TIPTYPE of USER1 can have a terminal with a TC of HASP as long as all parameters useable for terminals of terminal class HASP are correctly configured.

The NDL Processor cannot perform all normal checks for completeness of definition in LINE and TERMINAL statements if site-defined parameter values are present.

All of the obtainable listings for an NDL program containing deliberate errors are shown in figure B-1. COMMENT statements in the source listings precede statements containing errors unique to those statements. In some instances, these errors have caused errors in other statements; these pyramided errors are explained in the next paragraph. The error summary listing gives line and column (character position) numbers wherever possible. When invalid values are detected, only line numbers are listed.

The sample diagnostic listing illustrates several kinds of pyramided errors. For example, use of an invalid value in a DEFINE cannot be diagnosed (network division line 21) until the DEFINE is used (network division line 40). Invalid use of a DEFINE can be diagnosed (local division line 13) and cause spurious errors (local division line 61) because the DEFINE cannot be correctly expanded. The NDL Processor is a multiple pass program, attempting in each pass to clarify errors diagnosed in a preceding pass; the first error listed for line 61 indicates a declaration is missing from the statement, while the second error indicates which declaration is missing. The multiple errors of local division lines 32 through 46 are caused by a single error in line 32, while errors in those statements that should be produced by the error in line 28 are hidden from the NDL Processor.

The DEFINE statement in local division line 28 does not expand correctly because the identifier used in the statement is not recognized as the identifier stored by the processing of the DEFINE statement. As the DEFINE statement listing shows, the overly long identifier has been stored in a wrapped form; that is, the end of the identifier has overwritten the beginning. The NDL Processor cannot expand MODE4DIALUP because it cannot equate this to PODE4DI.

JOB DAYFILE MESSAGES

All of the informative and file access error messages produced in the dayfile for the job are listed in table B-2.

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES

Error Code	Severity	Message	Significance	Action	Issued By
001	Fatal	INPUT INSUFFICIENT TO CREATE FILE.	Not even an invalid definition file can be created.	Review section 3.	NETWORK, LOCAL
002	Fatal	PUNCTUATION ERROR.	A comma, colon, blank, or period is missing or misplaced.	Review section 3.	CHKLABEL, GETNEXT
003	Fatal	KEYWORD NOT RECOGNIZED.	A keyword is misspelled or misplaced.	Check appendices D and E.	NDLPSS1
004	Fatal	DUPLICATE PARAMETER.	Self-explanatory.	Change and rerun the job.	NDLPSS1
005	Fatal	REQUIRED PARAMETER MISSING.	Self-explanatory.	Change and rerun the job.	NDLPSS1
006	Fatal	VALUE IS NOT TYPE REQUIRED.	Either a required keyword is missing or a value has the wrong radix.	Change and rerun the job.	All CHK routines
007	Fatal	VALUE IS TOO LONG.	Too many digits or characters in the value declaration.	Check the statement.	NDLPSS1
008	Fatal	SUPERFLUOUS DATA AT END OF COMMAND.	Too many value declarations in the statement.	Check the statement format in section 3.	NDLPSS1
009	Fatal	REQUIRED VALUE MISSING.	Not enough value declarations in the statement.	Check the statement format in section 3.	NDLPSS1
010	Fatal	EMPTY DEFINE.	Self-explanatory.	Change and rerun the job.	NDLPSS1
011	Fatal	DUPLICATE DEFINE.	An identifier has been used for more than one DEFINE statement.	Change identifiers and rerun the job.	NDLPSS1
012	Fatal	DEFINE NAME CANNOT BE A KEYWORD.	Self-explanatory.	Change and rerun the job.	NDLPSS1
013	Fatal	NESTED DEFINE.	An identifier has been used within the string of another DEFINE.	Change and rerun the job.	NDLPSS1

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES (Contd)

Error Code	Severity	Message	Significance	Action	Issued By
014	Fatal	STATEMENT OUT OF SEQUENCE.	The required hierarchy has been violated.	Check figure 3-4.	NDLPSS1
015	Fatal	DUPLICATE LABEL.	The element-name is not unique.	Change and rerun the job.	CHKLABEL
016	Fatal	REQUIRED LABEL MISSING.	The element-name field is empty.	Change and rerun the job.	NDLPSS1
017	Fatal	DUPLICATE NODE NUMBER.	Node numbers must be unique.	Change and rerun the job.	NODCHK
018	Fatal	DUPLICATE PORT NUMBER.	Port numbers must be unique.	Change and rerun the job.	CHKPORT
019	Fatal	NPUNAME NOT DEFINED IN AN NPU STATEMENT.	The referenced NPU cannot be identified.	Add an NPU statement and rerun the job.	NETTERM
020	Fatal	HNODE NOT DEFINED IN A COUPLER STATEMENT.	The host end of the logical link is not part of the communications network.	Add a COUPLER statement and rerun the job.	LLINKPR
021	Fatal	TNODE NOT DEFINED IN AN NPU STATEMENT.	The terminal end of the logical link is not part of the communications network.	Add an NPU statement and rerun	NETTERM
022	Fatal	NPU NOT REFERENCED BY A COUPLER OR TRUNK.	The NPU is not part of the communications network.	Add a COUPLER or TRUNK statement and rerun the job.	NETTERM
023	Fatal	DUPLICATE NPUNAME.	A coupler can only be connected to a single NPU.	Change and rerun the job.	COUPPR
024	Fatal	VALUE OUT OF RANGE.	The value is negative or too large.	Check the statement format in section 3.	All routines
025	Fatal	INVALID NPUTYPE.	Self-explanatory.	Change and rerun the job.	CHKNTYPE
026	Fatal	NNODE NOT DEFINED IN AN NPU STATEMENT.	The other end of the trunk cannot be found.	Add an NPU statement and rerun the job.	NETTERM
027	Fatal	TRUNK NAME NOT DEFINED ON NEIGHBOR NODE.	Only one end of the trunk is configured.	Add a TRUNK statement after the appropriate NPU statement.	NETTERM
028	Fatal	TOO MANY TRUNKS WITH THE SAME NAME.	There are more than two TRUNK statements with the same element-name or more than one on a single NPU.	Change and rerun the job.	TRUNKPR
029	Fatal	MISSING OR INVALID LCF.	Self-explanatory.	Add or change the LFILE statement.	LFILEPR

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES (Contd)

Error Code	Severity	Message	Significance	Action	Issued By
030	Fatal	B1 IS VALID FOR CONSOLES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
031	Fatal	CA IS VALID FOR MODE4 LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	CLUSADD
032	Fatal	TA IS VALID FOR MODE4 LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMADD
033	Fatal	INVALID LTYPE.	Self-explanatory.	Change and rerun the job.	CHKLINETYPE
034	Fatal	INVALID TIPTYPE.	The TIP declared cannot service the line type given, or the value is not recognized.	Change and rerun the job.	CHKTIPTYPE
035	Fatal	LSPEED IS VALID ON NONAUTO ASYNC LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	LINEPR
036	Fatal	TC NOT COMPATIBLE WITH TIPTYPE.	The TIP declared cannot service the terminal class given.	Change and rerun the job.	ENDCHKS
037	Fatal	TC IS REQUIRED ON NON-AUTO-RECOGNITION LINES.	Self-explanatory.	Change and rerun the job.	ENDCHKS
038	Fatal	STIP IS REQUIRED.	Self-explanatory.	Change and rerun the job.	ENDCHKS
039	Fatal	CA IS REQUIRED ON NON-AUTO-REC MODE 4 LINES.	Self-explanatory.	Change and rerun the job.	ENDCHKS
040	Fatal	TA IS REQUIRED ON NON-AUTO-REC MODE 4 LINES.	Self-explanatory.	Change and rerun the job.	ENDCHKS
041	Fatal	TERMINAL MUST BE ENABLED.	Terminal is on an automatic recognition line.	Change and rerun the job.	ENDCHKS
042	Fatal	PA IS VALID FOR ASYNCHRONOUS LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
043	Fatal	BS IS VALID FOR ASYNCHRONOUS LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
044	Fatal	CI IS VALID FOR ASYNCHRONOUS LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
045	Fatal	DLTO IS VALID FOR ASYNCHRONOUS LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
046	Fatal	EP IS VALID FOR ASYNCHRONOUS LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES (Contd)

Error Code	Severity	Message	Significance	Action	Issued By
047	Fatal	LI IS VALID FOR ASYNCHRONOUS LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
048	Fatal	DLX IS VALID FOR ASYNCHRONOUS LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
049	Fatal	DLC IS VALID FOR ASYNCHRONOUS LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
050	Fatal	AL IS VALID FOR CONSOLES ON ASYNC LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
051	Fatal	ALFAM IS REQUIRED FOR NON-CONSOLES.	Self-explanatory.	Add the parameter and rerun the job.	ENDCHKS
052	Fatal	ALUN IS REQUIRED FOR NON-CONSOLES.	Self-explanatory.	Add the parameter and rerun the job.	ENDCHKS
053	Fatal	B2 IS VALID FOR CONSOLES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
054	Fatal	VALUE IS NOT UNIQUE.	A value declared for BS, B1, B2, CN or CT is not unique for the terminal.	Change and rerun the job.	TERMPR
055	Fatal	CONSOLES CANNOT HAVE OWNING CONSOLES.	Self-explanatory.	Remove the owning console specification.	TERMPR, OCTERM
056	Fatal	CSET NOT ALLOWED FOR TERMINALS ON THIS LINE.	The first terminal on the line did not have a CSET parameter. This one does.	Remove the CSET parameter.	TERMPR
057	Fatal	CN IS VALID FOR CONSOLES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
058	Fatal	IN IS VALID FOR CONSOLES ON ASYNC LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
059	Fatal	OP IS VALID FOR CONSOLES ON ASYNC LINES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
060	Fatal	PG IS VALID FOR CONSOLES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
061	Fatal	CT IS VALID FOR CONSOLES ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
062	Fatal	DO IS REQUIRED.	All passive devices of TC=HASP require a device ordinal.	Change and rerun the job.	ENDCHKS

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES (Contd)

Error Code	Severity	Message	Significance	Action	Issued By
063	Fatal	DO IS INVALID FOR CONSOLES.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
064	Fatal	PDO IS VALID FOR PLOTTERS ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
065	Fatal	SE NOT VALID WITH TIPTYPE OR DT SPECIFIED.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
066	Warning	RANGE/DEFAULT DEPENDS ON TC - CANNOT BE CHECKED.	Self-explanatory.	None.	RCHECK, SUBTIP
067	Fatal	STIP NOT VALID FOR TIPTYPE SPECIFIED.	Self-explanatory.	Change the parameter and rerun the job.	SUBTIP
068	Fatal	STIP REQUIRED FOR TERMINALS ON THIS LINE.	Self-explanatory.	Add the parameter and rerun the job.	ENDCHKS
069	Fatal	STIP NOT ALLOWED FOR TERMINALS ON THIS LINE.	Self-explanatory.	Remove the parameter and rerun the job.	SUBTIP
070	Fatal	AUTO-LOAD NPUS MUST BE REMOTE.	AUTO was declared for a front-end NPU.	Change and rerun the job.	NPUPROC
071	Fatal	TABLE OVERFLOW.	One of the limits described in appendix F was exceeded.	Change and rerun the job.	NDLPSSI
072	Fatal	DT NOT COMPATIBLE WITH TC.	The device type given is not legal for the terminal class given.	Change and rerun the job.	TCLASS
073	Fatal	CA OUT OF SEQUENCE.	Mode 4 terminal definition must be grouped by hardware address.	Reorder the statements and rerun the job.	CLUSADD
074	Fatal	NNODE OUT OF SEQUENCE.	Trunk end definitions must be grouped by neighbor node number.	Reorder the statements and rerun the job.	TRUNKPR
075	Fatal	CA NOT ALLOWED FOR TERMINALS ON THIS LINE.	CA was not specified for the first terminal on the line.	Add the parameter and rerun the job.	CLUSADD
076	Fatal	CA REQUIRED FOR TERMINALS ON THIS LINE.	Self-explanatory.	Add the parameters and rerun the job.	ENDCHKS
077	Fatal	INVALID DEVICE TYPE.	The device type parameter is not recognized or not permitted for the terminal class given.	Change and rerun the job.	CHKDT
078	Fatal	OCTERM IS REQUIRED FOR NON-CONSOLES.	Self-explanatory.	Add the parameter and rerun the job.	ENDCHKS
079	Fatal	PDO IS REQUIRED FOR PLOTTERS.	Self-explanatory.	Add the parameter and rerun the job.	ENDCHKS

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES (Contd)

Error Code	Severity	Message	Significance	Action	Issued By
080	Warning	TA DEPENDENT ON STIP - CANNOT BE CHECKED.	Self-explanatory.	None.	ENDCHKS
081	Fatal	TA NOT ALLOWED FOR TERMINALS ON THIS LINE.	TA was not specified for the first terminal on the line.	Add the parameter and rerun the job.	TERMADD
082	Fatal	TA REQUIRED FOR TERMINALS ON THIS LINE.	Self-explanatory.	Add the parameter and rerun the job.	ENDCHKS
083	Fatal	MISSING END STATEMENT.	Divisions must be terminated by END statements.	Add the statement and rerun the job.	PASS1
084	Fatal	TC NOT ALLOWED FOR TERMINALS ON THIS LINE.	TC wasn't specified for the first terminal on the line.	Add the parameter and rerun the job.	TCLASS
085	Fatal	TC REQUIRED FOR TERMINALS ON THIS LINE.	Self-explanatory.	Add the parameters and rerun the job.	ENDCHKS
086	Fatal	INVALID OWNING CONSOLE NAME.	The owning console declared cannot be found, or a 200UT owning console is at the wrong hardware address.	Check for missing console definition.	OCTERM, LOCTERM
087	Fatal	CSET REQUIRED FOR TERMINALS ON THIS LINE.	The first terminal on the line had a CSET parameter. This one does not.	Add a CSET parameter to the TERMINAL statement.	ENDCHKS
088	Fatal	TSPEED NOT ALLOWED FOR TERMINALS ON THIS LINE.	The first terminal on the line did not have a TSPEED parameter. This one does.	Remove the TSPEED parameter.	TERMPR
089	Fatal	DUPLICATE DEVICE ORDINAL.	Device ordinals must be unique within a device type and for a given OCTERM declaration.	Change and rerun the job.	TERMPR
090	Fatal	INVALID PDO.	PDO cannot be the same as a card punch device ordinal.	Change and rerun the job.	TERMPR
091	Fatal	LOAD IDS ARE NOT VALID FOR AUTOLOAD NPUS.	Self-explanatory.	Remove the parameters and rerun the job.	NPUPROC
092	Fatal	MISSING OR INVALID NCF.	The local division must reference the local name of a validated network configuration file.	Add or change the NFILE statement, or correct the invalid file.	NFILEPR
093	Fatal	DUPLICATE LOAD ID.	Load-IDs must be unique for each phase.	Change and rerun the job.	NPUPROC
094	Fatal	LOAD ID MUST BE THREE CHARACTERS LONG.	Self-explanatory.	Change and rerun the job.	CHKLDID
095	Fatal	DUPLICATE LOGLINK DEFINITION.	Only one logical link is permitted between a host and terminal node pair.	Remove one statement and rerun the job.	LLINKPR

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES (Contd)

Error Code	Severity	Message	Significance	Action	Issued By
096	Fatal	INVALID HOSTDATA NAME.	The element-name field does not contain a valid value or the named host cannot be found in the NCF.	Change and rerun the job.	HDATAPR
097	Warning	BSZ VALUE IS NOT A MULTIPLE OF 640 CHARACTERS.	NDLP has changed a BSZ value to a multiple of 640 characters for a passive terminal.	None.	NDLMAIN
098	Fatal	INVALID NPUDATA NAME.	The element-name field does not contain a valid value or the named NPU cannot be found in the NCF.	Change and rerun the job.	NPUDTPR
099	Fatal	DUPLICATE LINE MUST BE DISABLED.	More than one enabled line has the given port number. Only 15 lines per port are permitted and only one can be enabled.	Change and rerun the job.	LINEPR
100	Fatal	TIPTYPE IS NOT COMPATIBLE WITH LTYPE.	The TIP declared cannot support the stated line type.	Change and rerun the job.	LINEPR
101	Fatal	P1LID IS REQUIRED.	The NPU is not of type AUTO.	Change and rerun the job.	NPUPROC
102	Fatal	P2LID IS REQUIRED.	The NPU is not of type AUTO.	Change and rerun the job.	NPUPROC
103	Fatal	RESERVED APPLICATION NAME.	The element-name field contains an NVF reserved application name. This is not permitted.	Rename the application and rerun the job.	APPLPR
104	Fatal	PL IS VALID FOR CONSOLES ONLY.	A page length is declared for a passive terminal. This is not permitted.	If the definition is for a passive terminal, remove the PL parameter and rerun the job. If the definition is for a console, correct the DT parameter to specify a console and rerun the job.	TERMPR
105	Fatal	INITIAL APPLICATION NOT DEFINED.	The initial application program declared for a passive terminal is not defined in an APPL statement.	Change and rerun the job.	TERMPR
106	Fatal	TOO MANY LINES ON THE SAME PORT.	More than one enabled terminal is defined for the same hardware address (STIP, TC, CA, TA, DT, DO). Only 15 terminals per address are permitted and only one can be enabled.	Change and rerun the job.	LINEPR
107	Fatal	DUPLICATE TERMINAL MUST BE DISABLED.	Self-explanatory.	Change and rerun the job.	TERMPR

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES (Contd)

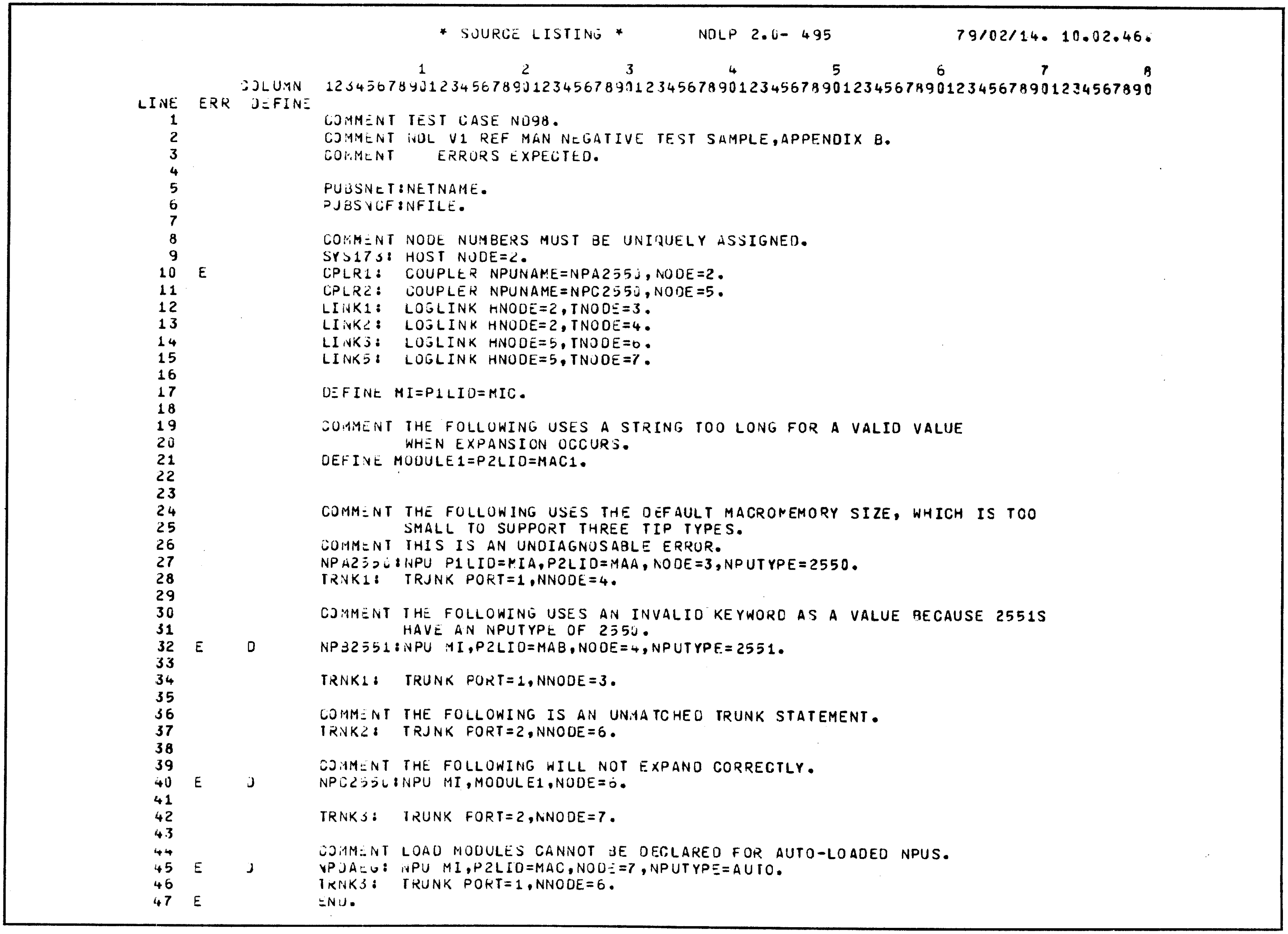
Error Code	Severity	Message	Significance	Action	Issued By
108	Fatal	STIP NOT COMPATIBLE WITH TC SPECIFIED.	The sub TIP-type declared does not support the terminal class declared.	Change the STIP or TC parameter and rerun the job.	TERMPR
109	Fatal	PARAMETER NOT ALLOWED FOR TC SPECIFIED.	One of the parameterization declarations is not permitted for the terminal class declared.	Change the parameterization or terminal class and rerun the job.	RCHECK
110	Fatal	OWNING CONSOLE NOT 200UT COMPATIBLE.	The terminal declared as owning console for passive terminals of TC=200UT must have a TC of 200UT, 731, or 734.	Change the parameter and rerun the job.	LOCTERM
111	Fatal	OWNING CONSOLE MUST BE ENABLED.	When passive terminals are enabled, their owning console cannot be disabled.	Change and rerun the job.	ENDCHKS
112	Fatal	UNRECOGNIZABLE COMMAND.	The statement name is not a valid reserved word.	Check appendixes D and E.	NDLPSS1
113	Fatal	LABEL ON COMMENT, DEFINE OR END.	An element-name field appears on the indicated special purpose statement. This is not permitted.	Delete the field and rerun the job.	NDLPSS1
114	Fatal	TOO MANY TERMINALS AT THE SAME ADDRESS.	More than 15 terminals have the same hardware address.	Remove the extra TERMINAL statements and rerun the job.	TERMPR
115	Fatal	DO IS VALID FOR HASP TERMINALS ONLY.	Self-explanatory.	Remove the parameter and rerun the job.	LINEPR
116	Fatal	IAPPL IS REQUIRED FOR NON-CONSOLES.	Self-explanatory.	Add the parameter and rerun the job.	ENDCHKS
117	Fatal	ILLEGAL USE OF RESERVED WORD.	Self-explanatory.	Review section 3.	STORNAME
118	Fatal	HASP TERMINALS ARE NOT AUTORECOGNIZED.	AUTO cannot be declared for lines supporting TC=HASP.	Remove the parameter and rerun the job.	LINEPR
119	Fatal	PW NOT ALLOWED FOR PLOTTERS.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
120	Fatal	TSPEED REQUIRED FOR TERMINALS ON THIS LINE.	The first terminal on the line had a TSPEED parameter. This one does not.	Add a TSPEED parameter to the TERMINAL statement.	ENDCHKS
121	Fatal	CSET NOT COMPATIBLE WITH TC.	Self-explanatory.	Change and rerun the job.	ENDCHKS
122	Fatal	CSET NOT COMPATIBLE WITH TIPTYPE.	Self-explanatory.	Change and rerun the job.	ENDCHKS
123	Fatal	CSET NOT COMPATIBLE WITH STIP.	Self-explanatory.	Change and rerun the job.	ENDCHKS

TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES (Contd)

Error Code	Severity	Message	Significance	Action	Issued By
124	Fatal	CSET/TSPEED NOT ALLOWED WITH STIP SPECIFIED.	Self-explanatory.	Remove the CSET or TSPEED parameter.	ENDCHKS
125	Fatal	TSPEED VALID FOR ASYNC AUTO-REC.	Self-explanatory.	Change and rerun the job.	TERMPR
126	Fatal	CSET IS REQUIRED.	Self-explanatory.	Change and rerun the job.	ENDCHKS
127	Fatal	CSET NOT ALLOWED FOR HASP TERMINALS.	Self-explanatory.	Remove the CSET parameter.	TERMPR
128	Fatal	NUMBER OF RESOURCE LINES EXCEED NDLP CAPACITY.	Self-explanatory.	Shorten or delete comments to reduce number of source lines.	NDLPSSI
129	Fatal	TSPEED NOT COMPATIBLE WITH TC.	Self-explanatory.	Change and rerun the job.	TERMPR
130	Fatal	DO NOT ALLOWED FOR TERMINALS ON X-25 LINES.	Self-explanatory.	Remove the parameter and rerun the job.	TERMPR
131	Fatal	AUTO-REC NOT ALLOWED ON X-25 LINES.	Self-explanatory.	Remove the AUTO parameter and rerun the job.	LINEPR
132	Fatal	ONLY 1ST SVC TERM STMT SHOULD CONTAIN PARAMETERS.	Self-explanatory.	Use the abbreviated format of the TERMINAL statement.	TERMPR
133	Fatal	SE NOT ALLOWED FOR TERMINALS ON X-25 LINES.	Self-explanatory.	Remove the SE parameter.	TERMPR
134	Fatal	PDO NOT ALLOWED FOR TERMINALS ON X-25 LINES.	Self-explanatory.	Remove the PDO parameter.	TERMPR
135	Fatal	DI NOT ALLOWED FOR TERMINALS ON X-25 LINES.	Self-explanatory.	Remove the DI parameter.	TERMPR
137	Fatal	W VALID FOR TERMINALS ON X-25 LINES ONLY.	Self-explanatory.	Remove the W parameter.	TERMPR
138	Fatal	OCTERM NOT ALLOWED FOR TERMINALS ON X-25 LINES.	Self-explanatory.	Remove the OCTERM parameter.	OCTERM
139	Fatal	DFL VALID FOR X-25 LINES ONLY.	Self-explanatory.	Remove the DFL parameter and rerun the job.	LINEPR
140	Fatal	K VALID FOR X-25 LINES ONLY.	Self-explanatory.	Remove the K parameter.	LINEPR
141	Fatal	T1 VALID FOR X-25 LINES ONLY.	Self-explanatory.	Remove the T1 parameter.	LINEPR
142	Fatal	N2 VALID FOR X-25 LINES ONLY.	Self-explanatory.	Remove the N2 parameter.	LINEPR

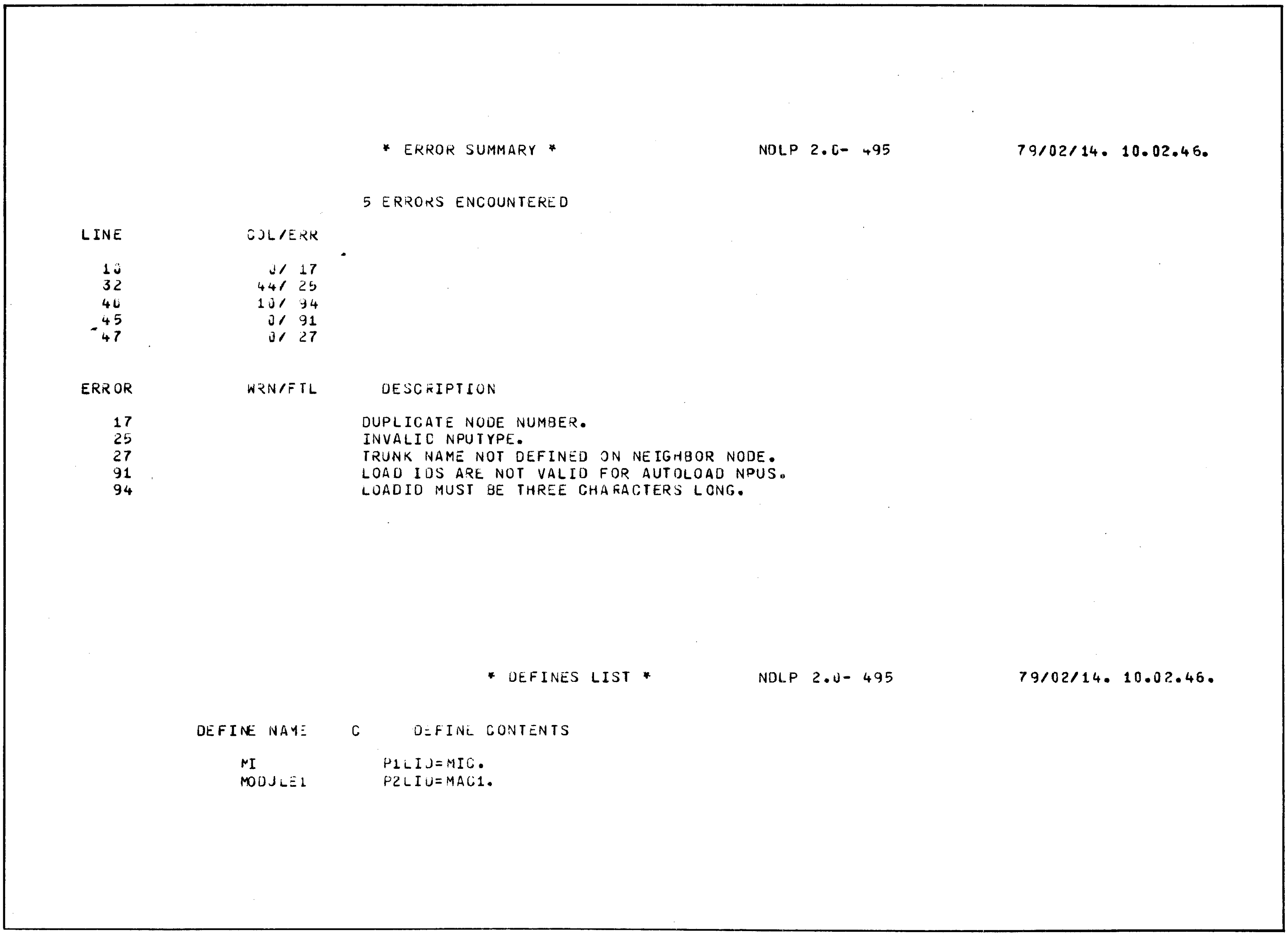
TABLE B-1. NDL PROCESSOR DIAGNOSTIC MESSAGES (Contd)

Error Code	Severity	Message	Significance	Action	Issued By
144	Fatal	TELENET VALID ON X-25 LINES ONLY.	Self-explanatory.	Remove TELENET (PSN parameter).	LINEPR
146	Fatal	USERPSN VALID ON X-25 LINES ONLY.	Self-explanatory.	Remove USERPSN (PSN parameter).	LINEPR
148	Fatal	LAP VALID FOR X-25 LINES ONLY.	Self-explanatory.	Remove the LAP parameter.	LINEPR
150	Fatal	RIC IS VALID FOR BSC CONSOLES ONLY.	Self-explanatory.	Remove the RIC parameter.	TERMPR
151	Fatal	BCF IS VALID FOR 2780 BSC CONSOLES ONLY.	Self-explanatory.	Remove the BCF parameter.	TERMPR
152	Fatal	MREC IS VALID FOR 2780 BSC CONSOLES ONLY.	Self-explanatory.	Remove the MREC parameter.	TERMPR
153	Fatal	SDT IS NOT ALLOWED FOR CONSOLES OR CARD PUNCH.	Self-explanatory.	Remove the SDT parameter.	TERMPR
154	Fatal	SDT IS NOT COMPATIBLE WITH DT.	Self-explanatory.	Change the SDT or DT parameter.	TERMPR
155	Fatal	SDT FOR CR IS VALID ON HASP OR BSC LINES ONLY.	Self-explanatory.	Remove the SDT parameter.	TERMPR
156	Fatal	SDT FOR PL IS VALID ON HASP LINES ONLY.	Self-explanatory.	Remove the SDT parameter.	TERMPR
157	Fatal	CO NOT ALLOWED FOR TERMINALS ON THIS LINE.	Self-explanatory.	Remove the CO parameter.	TERMPR
158	Fatal	CO VALID FOR HASP/BSC AUTO-REC TERMINALS ONLY.	The line is not an AUTO LINE, or TIPTYPE is not HASP or BSC.	Remove the CO parameter or specify AUTO in the LINE statement for the terminal.	TERMPR
159	Fatal	XBZ ILLEGAL FOR CONSOLES ON NON-HASP/BSC LINES.	Self-explanatory.	Change DT value or TIPTYPE.	TERMPR
160	Fatal	CO REQUIRED FOR TERMINALS ON THIS LINE.	Self-explanatory.	Add CO parameter to TERMINAL statement.	ENDCHKS
161	Fatal	PDO IS NOT VALID ON THIS TERMINAL.	Self-explanatory.	Remove the PDO parameter.	ENDCHKS
162	Fatal	DO REQUIRED ON PREVIOUS TERMINAL OF THIS DT.	More than one device of this type exists; DO must be unique.	Specify DO for each device of this DT with the same owning console.	ENDCHKS
163	Fatal	DUPLICATE PDO.	The PDO value is the same as the stream number of another device with the same owning console.	Choose a unique value for PDO.	ENDCHKS



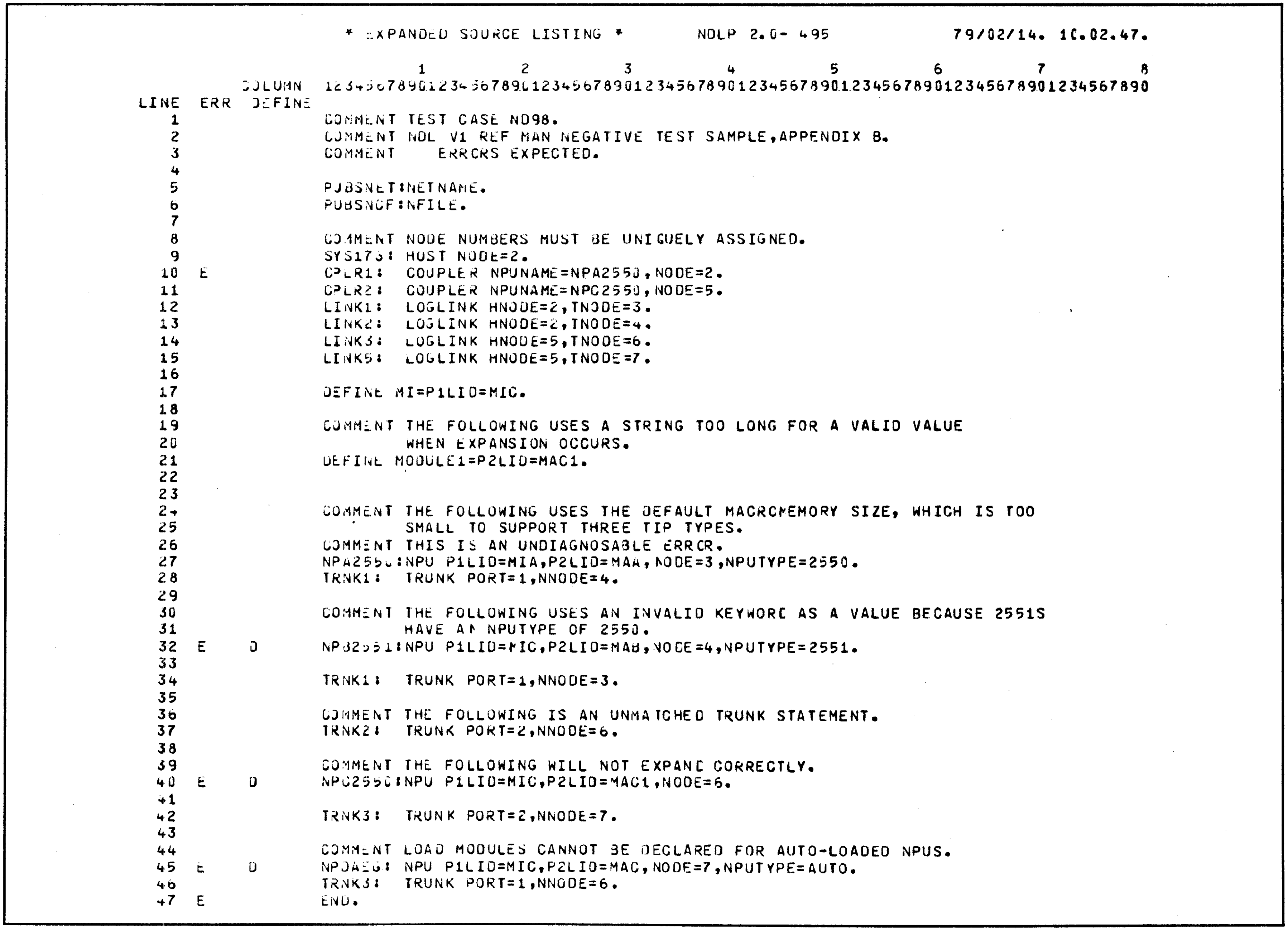
```
                                        * SOURCE LISTING *        NDLP 2.0- 495             79/02/14. 10.02.46.

                                  1         2         3         4         5         6         7         8
               COLUMN    12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE   ERR    DEFINE
   1                     COMMENT TEST CASE ND98.
   2                     COMMENT NDL V1 REF MAN NEGATIVE TEST SAMPLE,APPENDIX B.
   3                     COMMENT    ERRORS EXPECTED.
   4
   5                     PUBSNET:NETNAME.
   6                     PUBSNCF:NFILE.
   7
   8                     COMMENT NODE NUMBERS MUST BE UNIQUELY ASSIGNED.
   9                     SYS173: HOST NODE=2.
  10    E                CPLR1:  COUPLER NPUNAME=NPA2550,NODE=2.
  11                     CPLR2:  COUPLER NPUNAME=NPC2550,NODE=5.
  12                     LINK1:  LOGLINK HNODE=2,TNODE=3.
  13                     LINK2:  LOGLINK HNODE=2,TNODE=4.
  14                     LINK3:  LOGLINK HNODE=5,TNODE=6.
  15                     LINK5:  LOGLINK HNODE=5,TNODE=7.
  16
  17                     DEFINE MI=P1LID=MIC.
  18
  19                     COMMENT THE FOLLOWING USES A STRING TOO LONG FOR A VALID VALUE
  20                             WHEN EXPANSION OCCURS.
  21                     DEFINE MODULE1=P2LID=MAC1.
  22
  23
  24                     COMMENT THE FOLLOWING USES THE DEFAULT MACROMEMORY SIZE, WHICH IS TOO
  25                             SMALL TO SUPPORT THREE TIP TYPES.
  26                     COMMENT THIS IS AN UNDIAGNOSABLE ERROR.
  27                     NPA2550:NPU P1LID=MIA,P2LID=MAA,NODE=3,NPUTYPE=2550.
  28                     TRNK1:  TRUNK PORT=1,NNODE=4.
  29
  30                     COMMENT THE FOLLOWING USES AN INVALID KEYWORD AS A VALUE BECAUSE 2551S
  31                             HAVE AN NPUTYPE OF 2550.
  32    E      D         NPB2551:NPU MI,P2LID=MAB,NODE=4,NPUTYPE=2551.
  33
  34                     TRNK1:  TRUNK PORT=1,NNODE=3.
  35
  36                     COMMENT THE FOLLOWING IS AN UNMATCHED TRUNK STATEMENT.
  37                     TRNK2:  TRUNK PORT=2,NNODE=6.
  38
  39                     COMMENT THE FOLLOWING WILL NOT EXPAND CORRECTLY.
  40    E      D         NPC2550:NPU MI,MODULE1,NODE=6.
  41
  42                     TRNK3:  TRUNK FORT=2,NNODE=7.
  43
  44                     COMMENT LOAD MODULES CANNOT BE DECLARED FOR AUTO-LOADED NPUS.
  45    E      D         NPDAEG: NPU MI,P2LID=MAC,NODE=7,NPUTYPE=AUTO.
  46                     TRNK3:  TRUNK PORT=1,NNODE=6.
  47    E                END.
```

Figure B-1. Sample Diagnostic Listing (Sheet 1 of 11)



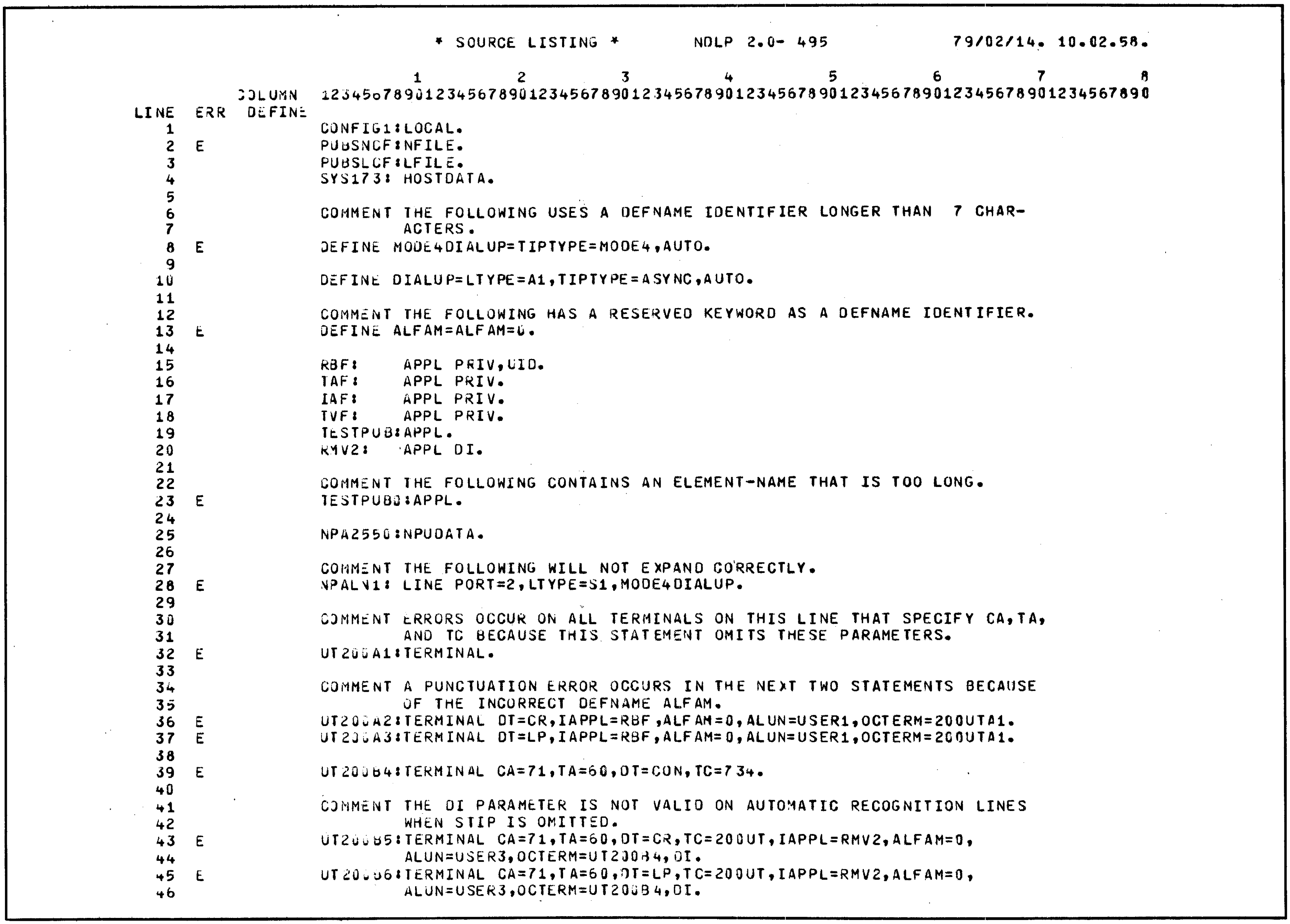
```
                    * ERROR SUMMARY *              NDLP 2.0- 495         79/02/14. 10.02.46.

                  5 ERRORS ENCOUNTERED

LINE            COL/ERR

  10              0/ 17
  32             44/ 25
  40             10/ 94
  45              0/ 91
  47              0/ 27

ERROR           WRN/FTL     DESCRIPTION

   17                     DUPLICATE NODE NUMBER.
   25                     INVALID NPUTYPE.
   27                     TRUNK NAME NOT DEFINED ON NEIGHBOR NODE.
   91                     LOAD IDS ARE NOT VALID FOR AUTOLOAD NPUS.
   94                     LOADID MUST BE THREE CHARACTERS LONG.




                              * DEFINES LIST *     NDLP 2.0- 495         79/02/14. 10.02.46.

          DEFINE NAME    C     DEFINE CONTENTS

              MI               P1LID=MIC.
              MODULE1          P2LID=MAC1.
```

Figure B-1. Sample Diagnostic Listing (Sheet 2 of 11)



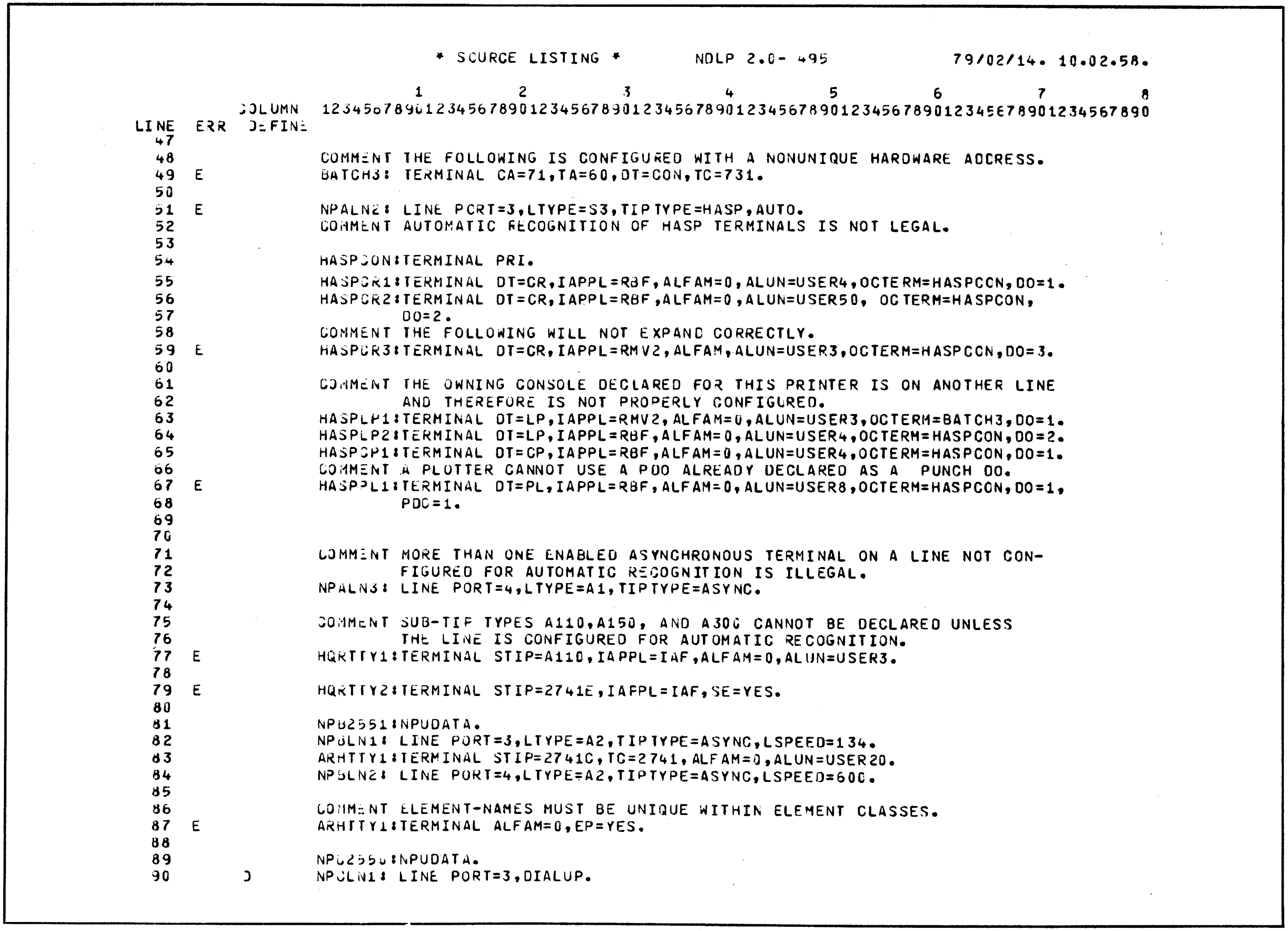
```
                        * EXPANDED SOURCE LISTING *       NDLP 2.0- 495           79/02/14. 10.02.47.

                                    1         2         3         4         5         6         7         8
              COLUMN   12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE  ERR  DEFINE
  1                    COMMENT TEST CASE ND98.
  2                    COMMENT NDL V1 REF MAN NEGATIVE TEST SAMPLE,APPENDIX B.
  3                    COMMENT    ERRORS EXPECTED.
  4
  5                    PUBSNET:NETNAME.
  6                    PUBSNCF:NFILE.
  7
  8                    COMMENT NODE NUMBERS MUST BE UNIQUELY ASSIGNED.
  9                    SYS173: HOST NODE=2.
 10   E                CPLR1:  COUPLER NPUNAME=NPA2550,NODE=2.
 11                    CPLR2:  COUPLER NPUNAME=NPC2550,NODE=5.
 12                    LINK1:  LOGLINK HNODE=2,TNODE=3.
 13                    LINK2:  LOGLINK HNODE=2,TNODE=4.
 14                    LINK3:  LOGLINK HNODE=5,TNODE=6.
 15                    LINK5:  LOGLINK HNODE=5,TNODE=7.
 16
 17                    DEFINE MI=P1LID=MIC.
 18
 19                    COMMENT THE FOLLOWING USES A STRING TOO LONG FOR A VALID VALUE
 20                            WHEN EXPANSION OCCURS.
 21                    DEFINE MODULE1=P2LID=MAC1.
 22
 23
 24                    COMMENT THE FOLLOWING USES THE DEFAULT MACROMEMORY SIZE, WHICH IS TOO
 25                            SMALL TO SUPPORT THREE TIP TYPES.
 26                    COMMENT THIS IS AN UNDIAGNOSABLE ERROR.
 27                    NPA2550:NPU P1LID=MIA,P2LID=MAA,NODE=3,NPUTYPE=2550.
 28                    TRNK1:  TRUNK PORT=1,NNODE=4.
 29
 30                    COMMENT THE FOLLOWING USES AN INVALID KEYWORD AS A VALUE BECAUSE 2551S
 31                            HAVE AN NPUTYPE OF 2550.
 32   E      D         NPB2551:NPU P1LID=MIC,P2LID=MAB,NODE=4,NPUTYPE=2551.
 33
 34                    TRNK1:  TRUNK PORT=1,NNODE=3.
 35
 36                    COMMENT THE FOLLOWING IS AN UNMATCHED TRUNK STATEMENT.
 37                    TRNK2:  TRUNK PORT=2,NNODE=6.
 38
 39                    COMMENT THE FOLLOWING WILL NOT EXPAND CORRECTLY.
 40   E      D         NPC2550:NPU P1LID=MIC,P2LID=MAC1,NODE=6.
 41
 42                    TRNK3:  TRUNK PORT=2,NNODE=7.
 43
 44                    COMMENT LOAD MODULES CANNOT BE DECLARED FOR AUTO-LOADED NPUS.
 45   E      D         NPOAEG: NPU P1LID=MIC,P2LID=MAC,NODE=7,NPUTYPE=AUTO.
 46                    TRNK3:  TRUNK PORT=1,NNODE=6.
 47   E                END.
```

Figure B-1. Sample Diagnostic Listing (Sheet 3 of 11)



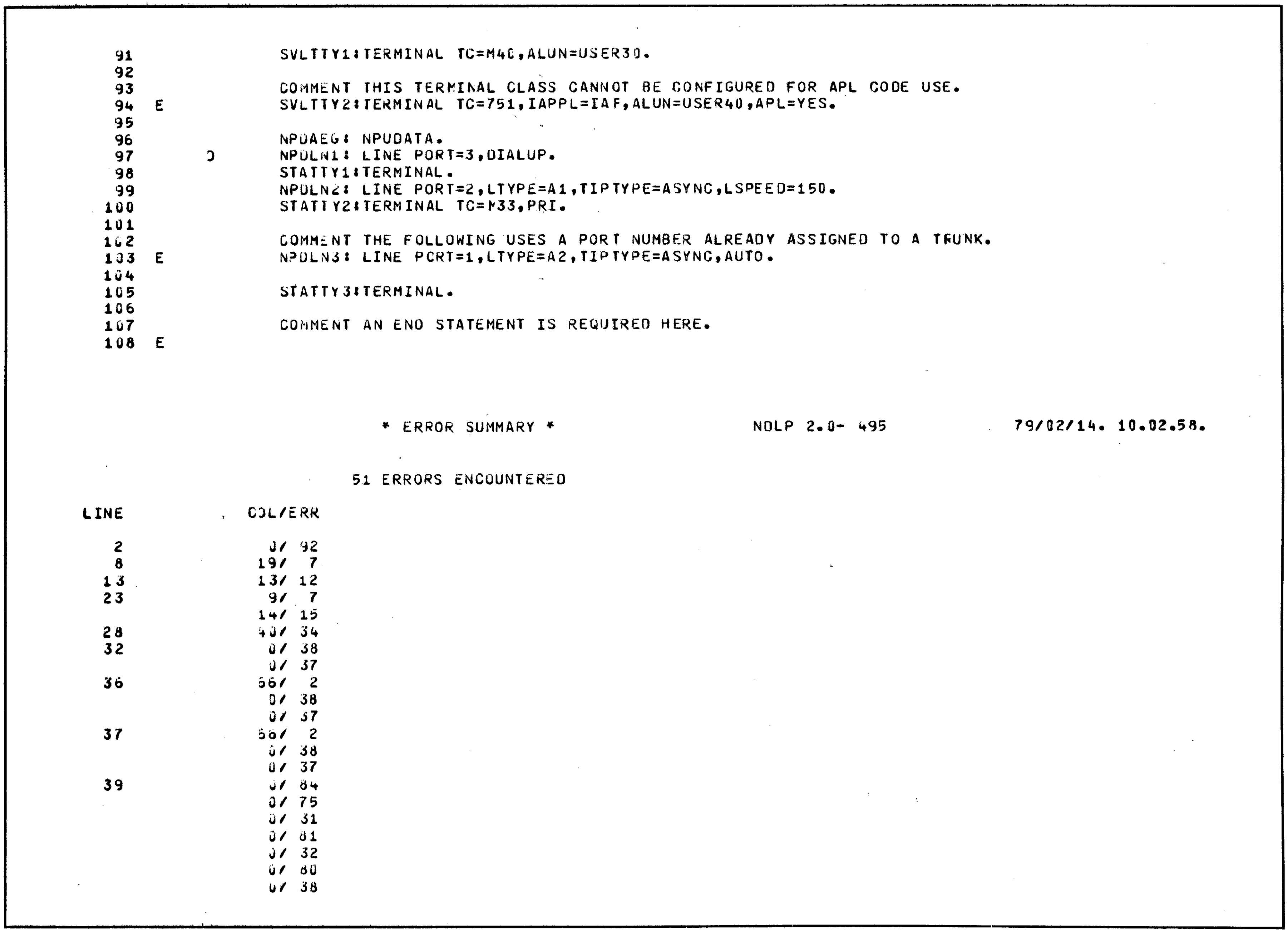
```
                              * SOURCE LISTING *        NDLP 2.0- 495            79/02/14. 10.02.58.

                                       1         2         3         4         5         6         7         8
             COLUMN   12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE  ERR  DEFINE
   1                  CONFIG1:LOCAL.
   2  E               PUBSNCF:NFILE.
   3                  PUBSLCF:LFILE.
   4                  SYS173: HOSTDATA.
   5
   6                  COMMENT THE FOLLOWING USES A DEFNAME IDENTIFIER LONGER THAN  7 CHAR-
   7                          ACTERS.
   8  E               DEFINE MODE4DIALUP=TIPTYPE=MODE4,AUTO.
   9
  10                  DEFINE DIALUP=LTYPE=A1,TIPTYPE=ASYNC,AUTO.
  11
  12                  COMMENT THE FOLLOWING HAS A RESERVED KEYWORD AS A DEFNAME IDENTIFIER.
  13  E               DEFINE ALFAM=ALFAM=0.
  14
  15                  RBF:    APPL PRIV,UID.
  16                  TAF:    APPL PRIV.
  17                  IAF:    APPL PRIV.
  18                  TVF:    APPL PRIV.
  19                  TESTPUB:APPL.
  20                  RMV2:   APPL DI.
  21
  22                  COMMENT THE FOLLOWING CONTAINS AN ELEMENT-NAME THAT IS TOO LONG.
  23  E               TESTPUB0:APPL.
  24
  25                  NPA2550:NPUDATA.
  26
  27                  COMMENT THE FOLLOWING WILL NOT EXPAND CORRECTLY.
  28  E               NPALN1: LINE PORT=2,LTYPE=S1,MODE4DIALUP.
  29
  30                  COMMENT ERRORS OCCUR ON ALL TERMINALS ON THIS LINE THAT SPECIFY CA,TA,
  31                          AND TC BECAUSE THIS STATEMENT OMITS THESE PARAMETERS.
  32  E               UT200A1:TERMINAL.
  33
  34                  COMMENT A PUNCTUATION ERROR OCCURS IN THE NEXT TWO STATEMENTS BECAUSE
  35                          OF THE INCORRECT DEFNAME ALFAM.
  36  E               UT200A2:TERMINAL DT=CR,IAPPL=RBF,ALFAM=0,ALUN=USER1,OCTERM=200UTA1.
  37  E               UT200A3:TERMINAL DT=LP,IAPPL=RBF,ALFAM=0,ALUN=USER1,OCTERM=200UTA1.
  38
  39  E               UT200B4:TERMINAL CA=71,TA=60,DT=CON,TC=734.
  40
  41                  COMMENT THE DI PARAMETER IS NOT VALID ON AUTOMATIC RECOGNITION LINES
  42                          WHEN STIP IS OMITTED.
  43  E               UT200B5:TERMINAL CA=71,TA=60,DT=CR,TC=200UT,IAPPL=RMV2,ALFAM=0,
  44                          ALUN=USER3,OCTERM=UT200B4,DI.
  45  E               UT200B6:TERMINAL CA=71,TA=60,DT=LP,TC=200UT,IAPPL=RMV2,ALFAM=0,
  46                          ALUN=USER3,OCTERM=UT200B4,DI.
```

Figure B-1. Sample Diagnqstic Listing (Sheet 4 of 11)



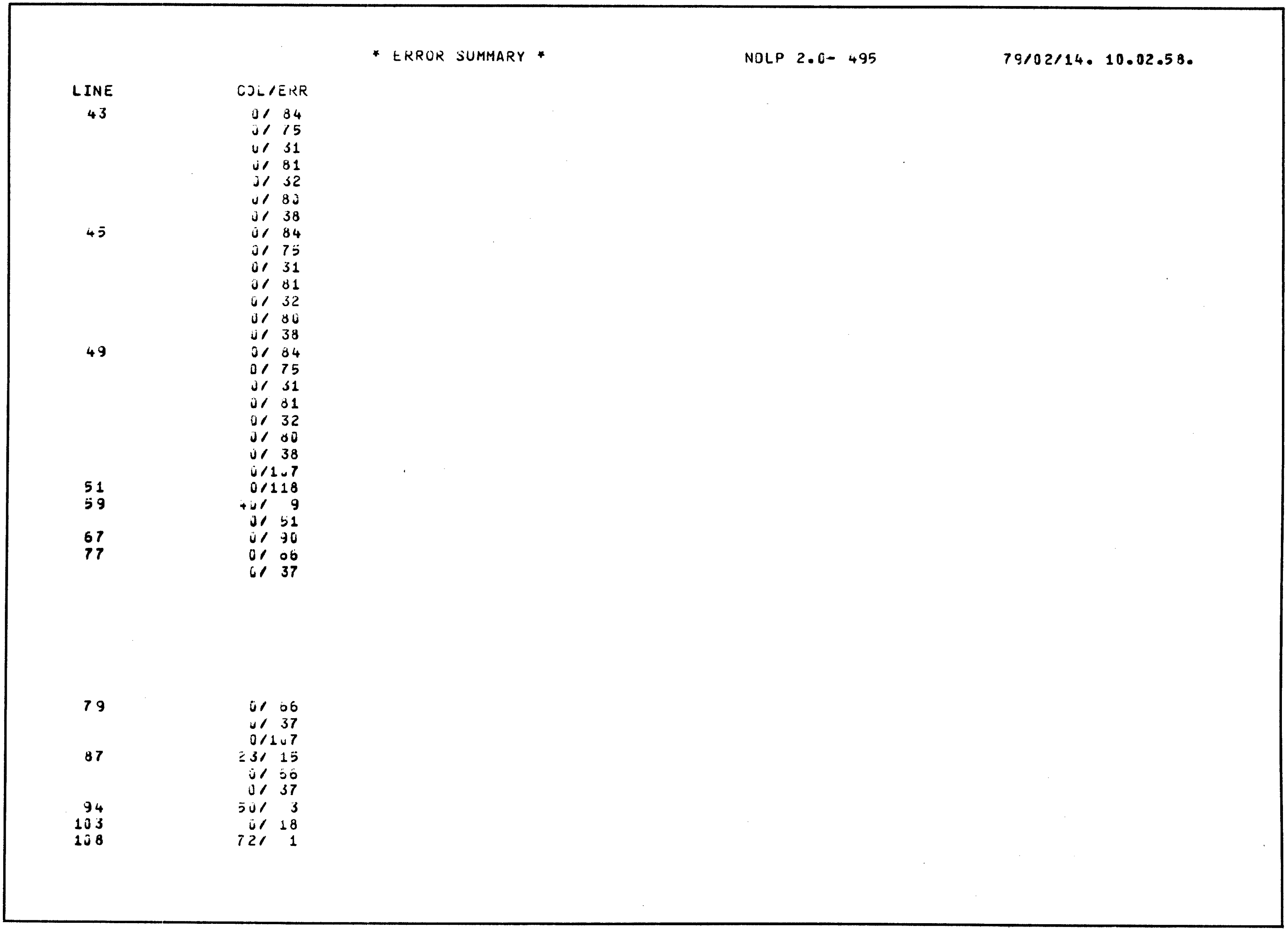
```
                              * SOURCE LISTING *       NDLP 2.0- 495            79/02/14. 10.02.58.

                                      1         2         3         4         5         6         7         8
                   COLUMN    12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE  ERR   DEFINE
 47
 48                          COMMENT THE FOLLOWING IS CONFIGURED WITH A NONUNIQUE HARDWARE ADDRESS.
 49    E                     BATCH3: TERMINAL CA=71,TA=60,DT=CON,TC=731.
 50
 51    E                     NPALN2: LINE PORT=3,LTYPE=S3,TIPTYPE=HASP,AUTO.
 52                          COMMENT AUTOMATIC RECOGNITION OF HASP TERMINALS IS NOT LEGAL.
 53
 54                          HASPCON:TERMINAL PRI.
 55                          HASPCR1:TERMINAL DT=CR,IAPPL=RBF,ALFAM=0,ALUN=USER4,OCTERM=HASPCON,DO=1.
 56                          HASPCR2:TERMINAL DT=CR,IAPPL=RBF,ALFAM=0,ALUN=USER50, OCTERM=HASPCON,
 57                                   DO=2.
 58                          COMMENT THE FOLLOWING WILL NOT EXPAND CORRECTLY.
 59    E                     HASPCR3:TERMINAL DT=CR,IAPPL=RMV2,ALFAM,ALUN=USER3,OCTERM=HASPCON,DO=3.
 60
 61                          COMMENT THE OWNING CONSOLE DECLARED FOR THIS PRINTER IS ON ANOTHER LINE
 62                                   AND THEREFORE IS NOT PROPERLY CONFIGURED.
 63                          HASPLP1:TERMINAL DT=LP,IAPPL=RMV2,ALFAM=0,ALUN=USER3,OCTERM=BATCH3,DO=1.
 64                          HASPLP2:TERMINAL DT=LP,IAPPL=RBF,ALFAM=0,ALUN=USER4,OCTERM=HASPCON,DO=2.
 65                          HASPCP1:TERMINAL DT=CP,IAPPL=RBF,ALFAM=0,ALUN=USER4,OCTERM=HASPCON,DO=1.
 66                          COMMENT A PLOTTER CANNOT USE A PDO ALREADY DECLARED AS A  PUNCH DO.
 67    E                     HASPPL1:TERMINAL DT=PL,IAPPL=RBF,ALFAM=0,ALUN=USER8,OCTERM=HASPCON,DO=1,
 68                                   PDO=1.
 69
 70
 71                          COMMENT MORE THAN ONE ENABLED ASYNCHRONOUS TERMINAL ON A LINE NOT CON-
 72                                   FIGURED FOR AUTOMATIC RECOGNITION IS ILLEGAL.
 73                          NPALN3: LINE PORT=4,LTYPE=A1,TIPTYPE=ASYNC.
 74
 75                          COMMENT SUB-TIP TYPES A110,A150, AND A300 CANNOT BE DECLARED UNLESS
 76                                   THE LINE IS CONFIGURED FOR AUTOMATIC RECOGNITION.
 77    E                     HQRTTY1:TERMINAL STIP=A110,IAPPL=IAF,ALFAM=0,ALUN=USER3.
 78
 79    E                     HQRTTY2:TERMINAL STIP=2741E,IAPPL=IAF,SE=YES.
 80
 81                          NPB2551:NPUDATA.
 82                          NPBLN1: LINE PORT=3,LTYPE=A2,TIPTYPE=ASYNC,LSPEED=134.
 83                          ARHTTY1:TERMINAL STIP=2741C,TC=2741,ALFAM=0,ALUN=USER20.
 84                          NPBLN2: LINE PORT=4,LTYPE=A2,TIPTYPE=ASYNC,LSPEED=600.
 85
 86                          COMMENT ELEMENT-NAMES MUST BE UNIQUE WITHIN ELEMENT CLASSES.
 87    E                     ARHTTY1:TERMINAL ALFAM=0,EP=YES.
 88
 89                          NPC2550:NPUDATA.
 90           D              NPCLN1: LINE PORT=3,DIALUP.
```

Figure B-1. Sample Diagnostic Listing (Sheet 5 of 11)



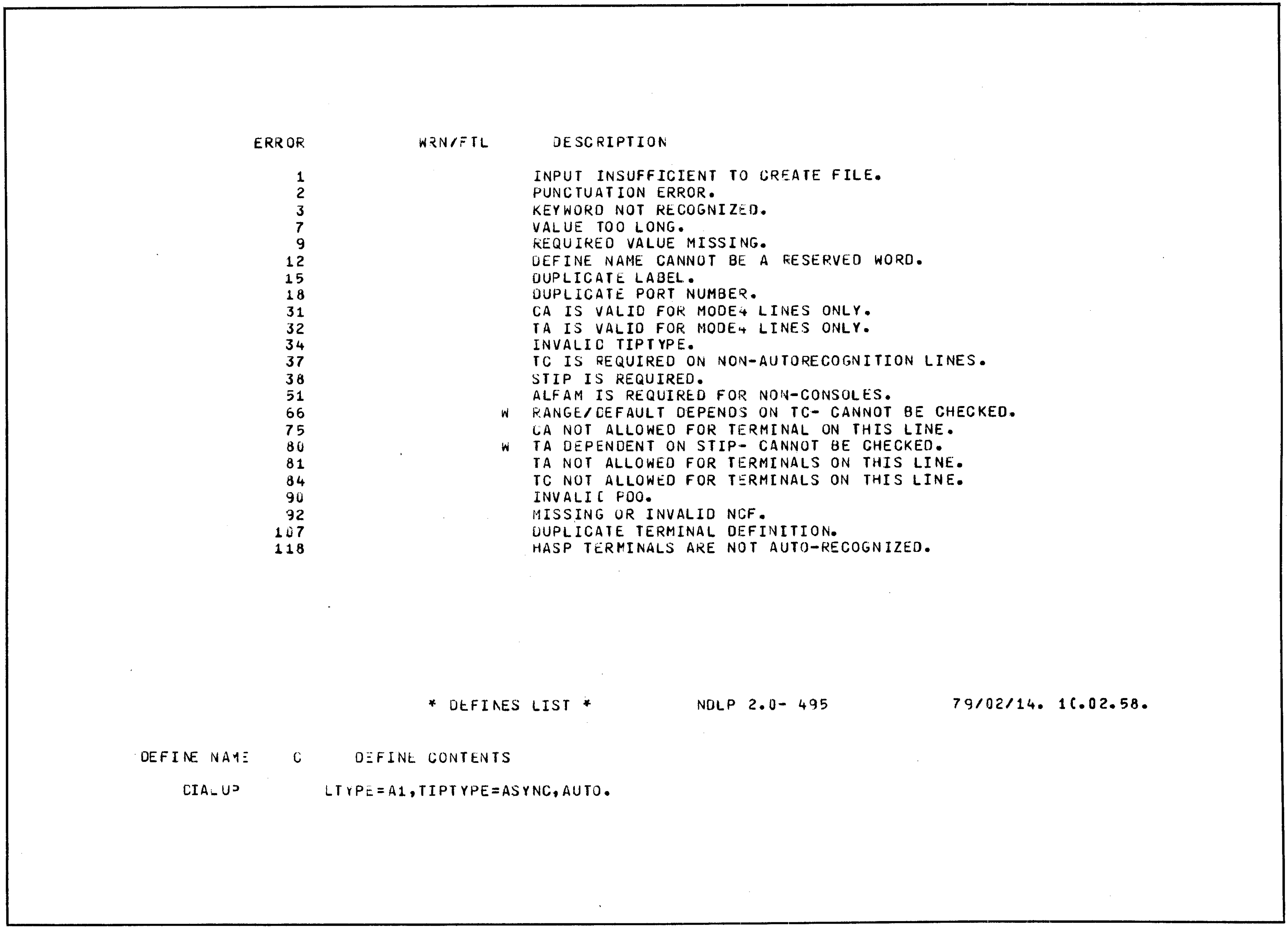
```
  91               SVLTTY1:TERMINAL TC=M40,ALUN=USER30.
  92
  93               COMMENT THIS TERMINAL CLASS CANNOT BE CONFIGURED FOR APL CODE USE.
  94  E            SVLTTY2:TERMINAL TC=751,IAPPL=IAF,ALUN=USER40,APL=YES.
  95
  96               NPUDAEG: NPUDATA.
  97        D      NPULN1: LINE PORT=3,DIALUP.
  98               STATTY1:TERMINAL.
  99               NPULN2: LINE PORT=2,LTYPE=A1,TIPTYPE=ASYNC,LSPEED=150.
 100               STATTY2:TERMINAL TC=M33,PRI.
 101
 102               COMMENT THE FOLLOWING USES A PORT NUMBER ALREADY ASSIGNED TO A TRUNK.
 103  E            NPULN3: LINE PORT=1,LTYPE=A2,TIPTYPE=ASYNC,AUTO.
 104
 105               STATTY3:TERMINAL.
 106
 107               COMMENT AN END STATEMENT IS REQUIRED HERE.
 108  E


                     * ERROR SUMMARY *            NDLP 2.0- 495          79/02/14. 10.02.58.


                    51 ERRORS ENCOUNTERED

LINE            COL/ERR

   2              0/ 92
   8             19/  7
  13             13/ 12
  23              9/  7
                 14/ 15
  28             40/ 34
  32              0/ 38
                  0/ 37
  36             56/  2
                  0/ 38
                  0/ 37
  37             56/  2
                  0/ 38
                  0/ 37
  39              0/ 84
                  0/ 75
                  0/ 31
                  0/ 81
                  0/ 32
                  0/ 80
                  0/ 38
```

Figure B-1. Sample Diagnostic Listing (Sheet 6 of 11)



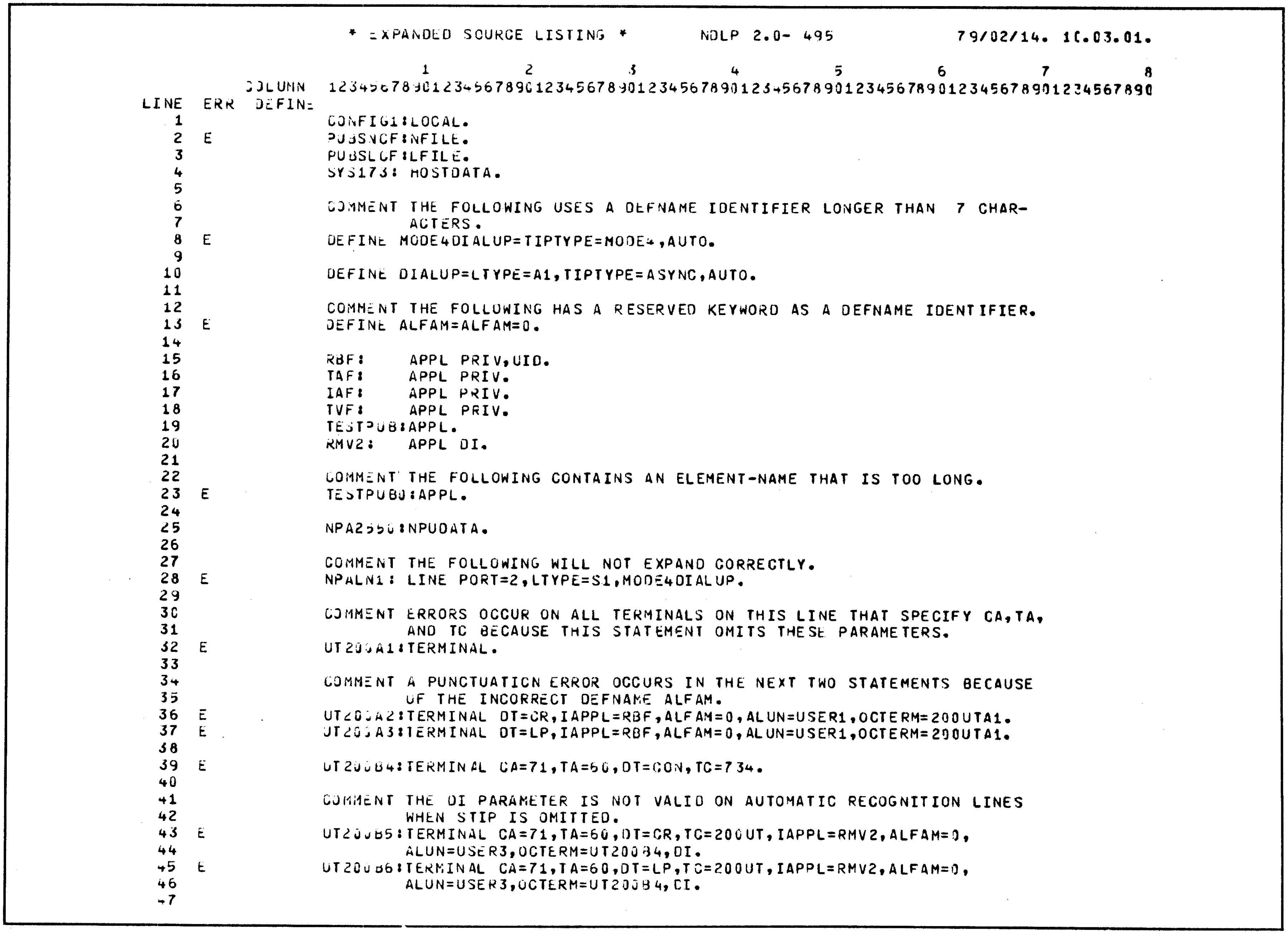
```
                          * ERROR SUMMARY *                    NDLP 2.0- 495            79/02/14. 10.02.58.
    LINE             COL/ERR
     43                0/ 84
                       0/ 75
                       0/ 31
                       0/ 81
                       0/ 32
                       0/ 80
                       0/ 38
     45                0/ 84
                       0/ 75
                       0/ 31
                       0/ 81
                       0/ 32
                       0/ 80
                       0/ 38
     49                0/ 84
                       0/ 75
                       0/ 31
                       0/ 81
                       0/ 32
                       0/ 80
                       0/ 38
                       0/107
     51                0/118
     59               40/  9
                       0/ 51
     67                0/ 90
     77                0/ 66
                       0/ 37




     79                0/ 66
                       0/ 37
                       0/107
     87               23/ 15
                       0/ 66
                       0/ 37
     94               50/  3
    103                0/ 18
    108               72/  1
```

Figure B-1. Sample Diagnostic Listing (Sheet 7 of 11)



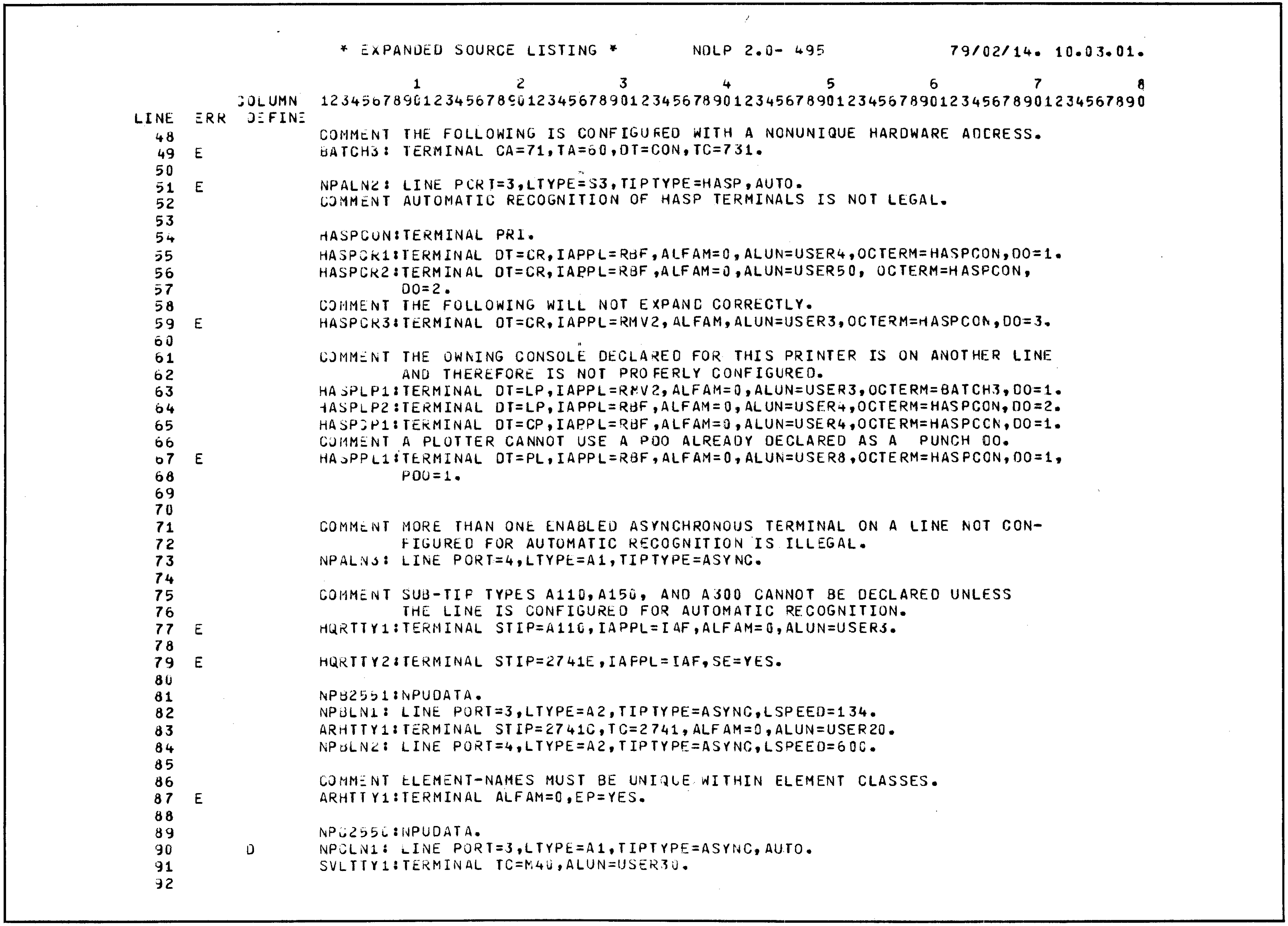
```
                ERROR           WRN/FTL      DESCRIPTION

                    1                       INPUT INSUFFICIENT TO CREATE FILE.
                    2                       PUNCTUATION ERROR.
                    3                       KEYWORD NOT RECOGNIZED.
                    7                       VALUE TOO LONG.
                    9                       REQUIRED VALUE MISSING.
                   12                       DEFINE NAME CANNOT BE A RESERVED WORD.
                   15                       DUPLICATE LABEL.
                   18                       DUPLICATE PORT NUMBER.
                   31                       CA IS VALID FOR MODE4 LINES ONLY.
                   32                       TA IS VALID FOR MODE4 LINES ONLY.
                   34                       INVALIC TIPTYPE.
                   37                       TC IS REQUIRED ON NON-AUTORECOGNITION LINES.
                   38                       STIP IS REQUIRED.
                   51                       ALFAM IS REQUIRED FOR NON-CONSOLES.
                   66                  W    RANGE/DEFAULT DEPENDS ON TC- CANNOT BE CHECKED.
                   75                       CA NOT ALLOWED FOR TERMINAL ON THIS LINE.
                   80                  W    TA DEPENDENT ON STIP- CANNOT BE CHECKED.
                   81                       TA NOT ALLOWED FOR TERMINALS ON THIS LINE.
                   84                       TC NOT ALLOWED FOR TERMINALS ON THIS LINE.
                   90                       INVALIC PDO.
                   92                       MISSING OR INVALID NCF.
                  107                       DUPLICATE TERMINAL DEFINITION.
                  118                       HASP TERMINALS ARE NOT AUTO-RECOGNIZED.




                          * DEFINES LIST *          NDLP 2.0- 495          79/02/14. 10.02.58.

      DEFINE NAME    C     DEFINE CONTENTS

         DIALUP          LTYPE=A1,TIPTYPE=ASYNC,AUTO.
```

Figure B-1. Sample Diagnostic Listing (Sheet 8 of 11)



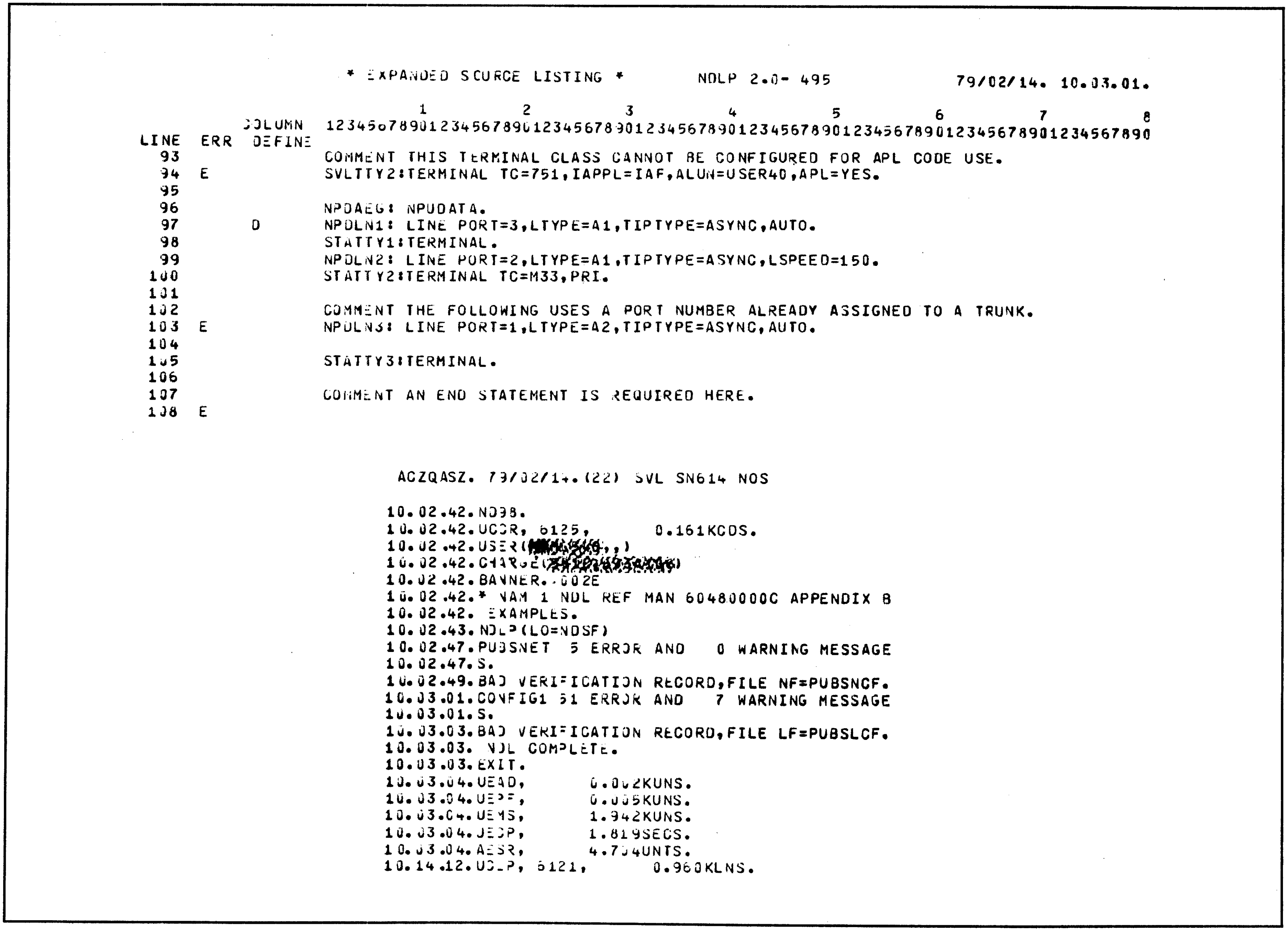
```
                          * EXPANDED SOURCE LISTING *      NDLP 2.0- 495           79/02/14. 10.03.01.

                                       1         2         3         4         5         6         7         8
                  COLUMN  12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE  ERR  DEFINE
   1                      CONFIG1:LOCAL.
   2   E                  PUBSNCF:NFILE.
   3                      PUBSLCF:LFILE.
   4                      SYS173: HOSTDATA.
   5
   6                      COMMENT THE FOLLOWING USES A DEFNAME IDENTIFIER LONGER THAN  7 CHAR-
   7                              ACTERS.
   8   E                  DEFINE MODE4DIALUP=TIPTYPE=MODE4,AUTO.
   9
  10                      DEFINE DIALUP=LTYPE=A1,TIPTYPE=ASYNC,AUTO.
  11
  12                      COMMENT THE FOLLOWING HAS A RESERVED KEYWORD AS A DEFNAME IDENTIFIER.
  13   E                  DEFINE ALFAM=ALFAM=0.
  14
  15                      RBF:    APPL PRIV,UID.
  16                      TAF:    APPL PRIV.
  17                      IAF:    APPL PRIV.
  18                      TVF:    APPL PRIV.
  19                      TESTPUB:APPL.
  20                      RMV2:   APPL DI.
  21
  22                      COMMENT THE FOLLOWING CONTAINS AN ELEMENT-NAME THAT IS TOO LONG.
  23   E                  TESTPUB0:APPL.
  24
  25                      NPA2550:NPUDATA.
  26
  27                      COMMENT THE FOLLOWING WILL NOT EXPAND CORRECTLY.
  28   E                  NPALN1: LINE PORT=2,LTYPE=S1,MODE4DIALUP.
  29
  30                      COMMENT ERRORS OCCUR ON ALL TERMINALS ON THIS LINE THAT SPECIFY CA,TA,
  31                              AND TC BECAUSE THIS STATEMENT OMITS THESE PARAMETERS.
  32   E                  UT200A1:TERMINAL.
  33
  34                      COMMENT A PUNCTUATION ERROR OCCURS IN THE NEXT TWO STATEMENTS BECAUSE
  35                              OF THE INCORRECT DEFNAME ALFAM.
  36   E                  UT200A2:TERMINAL DT=CR,IAPPL=RBF,ALFAM=0,ALUN=USER1,OCTERM=200UTA1.
  37   E                  UT200A3:TERMINAL DT=LP,IAPPL=RBF,ALFAM=0,ALUN=USER1,OCTERM=200UTA1.
  38
  39   E                  UT200B4:TERMINAL CA=71,TA=60,DT=CON,TC=734.
  40
  41                      COMMENT THE DI PARAMETER IS NOT VALID ON AUTOMATIC RECOGNITION LINES
  42                              WHEN STIP IS OMITTED.
  43   E                  UT200B5:TERMINAL CA=71,TA=60,DT=CR,TC=200UT,IAPPL=RMV2,ALFAM=0,
  44                              ALUN=USER3,OCTERM=UT200B4,DI.
  45   E                  UT200B6:TERMINAL CA=71,TA=60,DT=LP,TC=200UT,IAPPL=RMV2,ALFAM=0,
  46                              ALUN=USER3,OCTERM=UT200B4,DI.
  47
```

Figure B-1. Sample Diagnostic Listing (Sheet 9 of 11)



```
                   * EXPANDED SOURCE LISTING *       NDLP 2.0- 495            79/02/14. 10.03.01.

                                    1         2         3         4         5         6         7         8
               COLUMN   12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE  ERR  DEFINE
  48                    COMMENT THE FOLLOWING IS CONFIGURED WITH A NONUNIQUE HARDWARE ADDRESS.
  49   E                BATCH3: TERMINAL CA=71,TA=60,DT=CON,TC=731.
  50
  51   E                NPALN2: LINE PORT=3,LTYPE=S3,TIPTYPE=HASP,AUTO.
  52                    COMMENT AUTOMATIC RECOGNITION OF HASP TERMINALS IS NOT LEGAL.
  53
  54                    HASPCON:TERMINAL PR1.
  55                    HASPCR1:TERMINAL DT=CR,IAPPL=RBF,ALFAM=0,ALUN=USER4,OCTERM=HASPCON,DO=1.
  56                    HASPCR2:TERMINAL DT=CR,IAPPL=RBF,ALFAM=0,ALUN=USER50, OCTERM=HASPCON,
  57                            DO=2.
  58                    COMMENT THE FOLLOWING WILL NOT EXPAND CORRECTLY.
  59   E                HASPCR3:TERMINAL DT=CR,IAPPL=RMV2,ALFAM,ALUN=USER3,OCTERM=HASPCON,DO=3.
  60
  61                    COMMENT THE OWNING CONSOLE DECLARED FOR THIS PRINTER IS ON ANOTHER LINE
  62                            AND THEREFORE IS NOT PROPERLY CONFIGURED.
  63                    HASPLP1:TERMINAL DT=LP,IAPPL=RMV2,ALFAM=0,ALUN=USER3,OCTERM=BATCH3,DO=1.
  64                    HASPLP2:TERMINAL DT=LP,IAPPL=RBF,ALFAM=0,ALUN=USER4,OCTERM=HASPCON,DO=2.
  65                    HASPCP1:TERMINAL DT=CP,IAPPL=RBF,ALFAM=0,ALUN=USER4,OCTERM=HASPCON,DO=1.
  66                    COMMENT A PLOTTER CANNOT USE A PDO ALREADY DECLARED AS A  PUNCH DO.
  67   E                HASPPL1:TERMINAL DT=PL,IAPPL=RBF,ALFAM=0,ALUN=USER8,OCTERM=HASPCON,DO=1,
  68                            PDO=1.
  69
  70
  71                    COMMENT MORE THAN ONE ENABLED ASYNCHRONOUS TERMINAL ON A LINE NOT CON-
  72                            FIGURED FOR AUTOMATIC RECOGNITION IS ILLEGAL.
  73                    NPALN3: LINE PORT=4,LTYPE=A1,TIPTYPE=ASYNC.
  74
  75                    COMMENT SUB-TIP TYPES A110,A150, AND A300 CANNOT BE DECLARED UNLESS
  76                            THE LINE IS CONFIGURED FOR AUTOMATIC RECOGNITION.
  77   E                HQRTTY1:TERMINAL STIP=A110,IAPPL=IAF,ALFAM=0,ALUN=USER3.
  78
  79   E                HQRTTY2:TERMINAL STIP=2741E,IAPPL=IAF,SE=YES.
  80
  81                    NPB2551:NPUDATA.
  82                    NPBLN1: LINE PORT=3,LTYPE=A2,TIPTYPE=ASYNC,LSPEED=134.
  83                    ARHTTY1:TERMINAL STIP=2741C,TC=2741,ALFAM=0,ALUN=USER20.
  84                    NPBLN2: LINE PORT=4,LTYPE=A2,TIPTYPE=ASYNC,LSPEED=600.
  85
  86                    COMMENT ELEMENT-NAMES MUST BE UNIQUE WITHIN ELEMENT CLASSES.
  87   E                ARHTTY1:TERMINAL ALFAM=0,EP=YES.
  88
  89                    NPC2550:NPUDATA.
  90          D         NPCLN1: LINE PORT=3,LTYPE=A1,TIPTYPE=ASYNC,AUTO.
  91                    SVLTTY1:TERMINAL TC=M40,ALUN=USER30.
  92
```

Figure B-1. Sample Diagnostic Listing (Sheet 10 of 11)



```
                         * EXPANDED SOURCE LISTING *       NDLP 2.0- 495            79/02/14. 10.03.01.

                                  1         2         3         4         5         6         7         8
              COLUMN   12345678901234567890123456789012345678901234567890123456789012345678901234567890
LINE  ERR   DEFINE
  93                   COMMENT THIS TERMINAL CLASS CANNOT BE CONFIGURED FOR APL CODE USE.
  94   E               SVLTTY2:TERMINAL TC=751,IAPPL=IAF,ALUN=USER40,APL=YES.
  95
  96                   NPDAEG: NPUDATA.
  97           D       NPDLN1: LINE PORT=3,LTYPE=A1,TIPTYPE=ASYNC,AUTO.
  98                   STATTY1:TERMINAL.
  99                   NPDLN2: LINE PORT=2,LTYPE=A1,TIPTYPE=ASYNC,LSPEED=150.
 100                   STATTY2:TERMINAL TC=M33,PRI.
 101
 102                   COMMENT THE FOLLOWING USES A PORT NUMBER ALREADY ASSIGNED TO A TRUNK.
 103   E               NPDLN3: LINE PORT=1,LTYPE=A2,TIPTYPE=ASYNC,AUTO.
 104
 105                   STATTY3:TERMINAL.
 106
 107                   COMMENT AN END STATEMENT IS REQUIRED HERE.
 108   E


                            ACZQASZ. 79/02/14.(22) SVL SN614 NOS

                           10.02.42.NDJB.
                           10.02.42.UCCR, 6125,         0.161KCDS.
                           10.02.42.USER([redacted],,)
                           10.02.42.CHARGE([redacted])
                           10.02.42.BANNER. 002E
                           10.02.42.* NAM 1 NDL REF MAN 60480000C APPENDIX B
                           10.02.42. EXAMPLES.
                           10.02.43.NDLP(LO=NDSF)
                           10.02.47.PUBSNET  5 ERROR AND   0 WARNING MESSAGE
                           10.02.47.S.
                           10.02.49.BAD VERIFICATION RECORD,FILE NF=PUBSNCF.
                           10.03.01.CONFIG1 51 ERROR AND   7 WARNING MESSAGE
                           10.03.01.S.
                           10.03.03.BAD VERIFICATION RECORD,FILE LF=PUBSLCF.
                           10.03.03. NDL COMPLETE.
                           10.03.03.EXIT.
                           10.03.04.UEAD,       0.002KUNS.
                           10.03.04.UEPF,       0.005KUNS.
                           10.03.04.UEMS,       1.942KUNS.
                           10.03.04.UECP,       1.819SECS.
                           10.03.04.AESR,       4.704UNTS.
                           10.14.12.UCLP, 6121,         0.960KLNS.
```

Figure B-1. Sample Diagnostic Listing (Sheet 11 of 11)

TABLE B-2. NDL PROCESSOR DAYFILE MESSAGES

Message	Significance	Action	Issued By
BAD VERIFICATION RECORD FILE $\left\{\begin{matrix}LF\\NF\end{matrix}\right\}$ = lfn	The NDL Processor opened an existing local configuration file (LF) or network configuration file (NF) with the local file name indicated by lfn. This file did not contain a verification header record (see section 5). This message is also issued at the end of a file creation job when fatal errors were detected during processing.	The unverifiable file should be recreated or a different file used for NDL Processor input.	NDLMAIN
INPUT FILE EMPTY	The NDL Processor attempted to open the file specified for job input but could not find any information.	Check job structure or verify contents of the input file. Rerun the job.	NDLMAIN
INSUFFICIENT FIELD LENGTH	The NDL Processor requires additional central memory to completely process all input statements that cause table generation. Excessive use of the DEFINE statement can cause the Processor to need additional table space.	Remove as many NDL DEFINE statements as possible from the input file or add an RFL statement to the control statement portion of the job. Rerun the job.	STORDEF, STORNAME
INVALID CONTROL CARD OPTION	The NDLP control statement used by the job contains a format or syntax error.	Correct the statement and rerun the job.	NDLMAIN
NDL COMPLETE	The NDL Processor has finished all possible work on its input and ended its execution.	Informative only.	NDLMAIN
divname - NO ERRORS ENCOUNTERED	A properly verified network definition file has been created or listed by the NDL Processor from the indicated division.	Informative only.	
divname - mmm ERROR AND nn WARNING MESSAGES ISSUED	If mmm is not zero, the indicated number of fatal diagnostic message errors are described in the error summary listing produced by the NDL Processor as part of the listing output file. A non-zero value for mmm indicates that any network definition file created by the job from the named division does not contain a verification record. If nn is not zero, the indicated number of nonfatal diagnostic message errors are described in the error summary listing. A nonzero value for nn does not affect the verification record of any network definition file created by the job.	Correct the NDL statements input and rerun the job if mmm is not zero.	DAYYES, NDLLIST

GLOSSARY

This glossary defines terms unique to the description of the software presented in this manual. It also contains terms whose interpretation within this manual is intended to be more constrained or different from that commonly made. Some terms used in other manuals for the network software are included for the reader's convenience when reconciling terminology.

ACKNOWLEDGEMENT, BLOCK -
A message returned to the sender confirming the delivery of one or more messages or blocks.

APPLICATION CONNECTION NUMBER (ACN) -
A number assigned by the Communications Supervisor program to identify a particular logical connection within an application program.

APPLICATION INTERFACE PROGRAM (AIP) -
A group of routines that reside in the application program's field length to translate and buffer communication between the application program and the network.

APPLICATION NAME (ANAME) -
Up to seven 6-bit letters or digits (the first must be a letter) used to identify an application program. It is used by another application program or by a terminal operator when connection to the application is requested.

APPLICATION PROGRAM -
A program resident in a host computer that uses the Network Access Method and provides an information storage, retrieval, and/or processing service to a remote user via the data communication network.

ARCHETYPE TERMINAL -
The specific terminal equipment possessing all of the attributes used as defaults for the parameterization of one terminal class. Each terminal class has a corresponding archetype terminal.

AUTO-LOADING -
The process whereby a nonCDC network processing unit is loaded before joining the network. From the viewpoint of the Network Supervisor, the NPU loads itself automatically.

AUTOMATIC RECOGNITION -
The process whereby the Terminal Interface Program identifies characteristics of a terminal when the terminal's communication line becomes active. The Terminal Interface Program determines sub-TIP type and terminal class (and, for mode 4 terminals, the cluster and terminal addresses) by various methods for lines configured for automatic recognition. The Communications Supervisor then matches these parameters against the descriptions of specific terminals in the local configuration file; the terminal with the closest match to the empirically determined parameters is automatically recognized as the terminal on the communication line.

AUTOMATIC LOGIN -
The process whereby one or more of the Network Validation Facility login dialog parameters is supplied to NVF from the local configuration file. Parameters supplied through automatic login configuration of a terminal suppress prompting for the corresponding dialog entries and override any entries made from the terminal. Automatic login is required for terminals (passive devices) incapable of conducting dialog with NVF.

BATCH DEVICE -
see Passive Terminal.

BATCH TERMINAL -
see Passive Terminal.

BLOCK LIMIT -
The number of message blocks that can be awaiting delivery at any one time in either the host-to-NPU direction or the NPU-to-host direction for a single terminal or terminal device.

CHARACTER -
Unless otherwise specified, references to characters in this manual are to CDC 6-bit display-coded characters.

CLUSTER -
Mode 4 terminals grouped by a common cluster address.

CLUSTER ADDRESS -
The hardware address of a cluster. This term is used in several ways within mode 4 communications documentation, as shown in table C-1.

TABLE C-1. MODE 4A NOMENCLATURE EQUIVALENCE

Networks Nomenclature	Mode 4A Nomenclature	Mode 4C Nomenclature
Network processing unit	Data source	Control station
Cluster address	Site address	Station address
Cluster controller	Equipment controller	Station
Terminal address	Station address	Device address

COMMUNICATION ELEMENT -
Any entity that constitutes a point of input to, or output from, the data communication network. This includes terminal devices, terminals, communication lines, and application programs.

COMMUNICATION LINE -
A complete communication circuit between a terminal and its network processing unit.

COMMUNICATIONS CONTROL PROGRAM -
A portion of the network software that resides in a 255x Series network processing unit. This software can include such routines as the Terminal Interface Program.

COMMUNICATIONS SUPERVISOR -
A portion of the network software, written as an application program. The Communications Supervisor coordinates the network-oriented activities of its host computer and all of that host's communication elements.

CONSTANT CARRIER -
A communication line with a transmission carrier signal that remains on continuously; failure is reported if the carrier signal received remains off for a period of time that equals or exceeds a failure verification period.

CONSOLE -
see Interactive Terminal.

CONTENTION TERMINAL -
When a terminal can input at the discretion of the terminal user and has an input rate that cannot be controlled directly; contrast with controlled terminal. Asynchronous terminals are contention terminals.

CONTROLLED CARRIER -
A communication line with a transmission carrier signal that is raised and lowered with each block transmitted; failure is reported if the carrier signal received does not fluctuate in a similar fashion.

CONTROLLED TERMINAL -
When a terminal places data on a communication line only in response to a poll, the maximum input rate can be controlled by controlling the polling rate. Mode 4 terminals are controlled.

COUPLER -
A hardware module resident in a front-end network processing unit, that links the network processing unit to a host computer.

DATA -
Any portion of a message created by the source, exclusive of any information used to accomplish transmission of such a message.

DEDICATED LINE -
A communication line that is permanently connected between a terminal and a network processing unit. Contrast with switched line.

DEFINE -
An NDL statement that provides the macro-like capability of substituting an identifier in coding for a more complex entity. When the coding is processed, the identifier is interpreted as if it had been replaced by the complex entity. Also, a NOS 1 control statement that creates permanent files.

DESTINATION -
The terminal or application program designated to receive the message.

DESTINATION NODE -
The NPU node that directly interfaces to the destination of a data message block.

DEVICE -
A portion or all of a terminal. This term is used in various ways within mode 4 communications documentation, as shown in table C-1.

DIRECT ACCESS FILE -
In the context of NOS permanent files, a direct access file is a file that is accessed and modified directly.

DOWNLINE -
The direction of output flow, from host to terminal.

ECHOPLEX -
The process of returning received characters on a full-duplex line. Not all terminals on full-duplex communication lines are capable of echoplex operation.

FRONT-END NPU -
A network processing unit linked directly to a host computer through a coupler.

FULL DUPLEX -
Two-way simultaneous transmission on a communication line.

HALF DUPLEX -
Two-way alternating transmission on a communication line.

HOST -
A computer that executes application programs.

HOST NODE -
The node ID number of the NPU coupler that directly interfaces with a host computer.

INPUT -
Information flowing upline from terminal to host computer.

INTERACTIVE TERMINAL -
Any terminal capable of conducting both input and output, making it capable of dialog with the Network Validation Facility. Also known as a console-type device. Contrast with passive terminal.

LABEL -
The value declared as the element-name field of an NDL statement.

LEVEL -
For logical records, an octal number 0 through 17 in the system-supplied 48-bit marker that terminates a short or zero-length PRU. In input decks, an octal number specified on 7/8/9 or /*EOR cards.

LOCAL COMPUTER SYSTEM -
That portion of a network associated with one copy of the Communications Supervisor and controlled by a local operator.

LOCAL CONFIGURATION FILE -

A file in the host computer system, containing information on the physical and logical makeup of the communication elements in the system. The file contains a list of the application programs available for execution in the host computer, and the lines and terminals that can access it. This is a NOS direct access permanent file.

LOCAL OPERATOR -

The administrative operator who manages the communication elements of the network within the local computer system by communicating with the Communications Supervisor in the host computer. Contrast with network operator. The local operator is an administrative operator within the network and need not be the host computer's operating system operator.

LOGICAL CONNECTION -

A logical message path established between two application programs or between a network terminal and an application program. Until terminated, the logical connection allows messages to pass between the two entities.

LOGICAL LINK -

The portion of a logical connection defined by host node and terminal node ID numbers.

LOGICAL RECORD -

Under NOS, a data grouping that consists of one or more PRUs terminated by a short PRU or zero-length PRU. Equivalent to a system-logical-record under NOS/BE.

MACROMEMORY -

The portion of 255x Series network processing unit memory that contains code involved in data communication such as the Terminal Interface Program.

MESSAGE -

A logical unit of information, as processed by an application program. When transmitted over a network, a message can consist of one or more physical blocks.

MICROMEMORY -

The portion of 255x Series network processing unit memory that contains code defining the unit to itself.

MODE 4 -

A communication line transmission protocol which requires the polling of sources for input to the data communication network. Control Data supports two types of mode 4 equipment, mode 4A and mode 4C. Mode 4A equipment is polled through the hardware address of the console device, regardless of how many devices interface to the network. Mode 4C equipment is polled through separate hardware addresses, depending on the point each device uses to interface with the network.

NEIGHBOR NODE -

The node ID number associated with an NPU at one end of a trunk by the NPU at the other end of the trunk. The neighbor node can be the same as a terminal node.

NETWORK ACCESS METHOD (NAM) -

A software package that provides a generalized method of using a communication network for switching, buffering, queuing, and transmitting data.

NETWORK CONFIGURATION FILE -

A network definition file in the host computer, containing information on the network elements and permissible linkages between them. The status of the elements described in this file is modified by the network operator in the course of managing the network through the Network Supervisor. This is a NOS direct access permanent file.

NETWORK DEFINITION FILE -

Either of the two types of NDL program output files that determine the configuration of the network. This can be a network configuration file or a local configuration file.

NETWORK DEFINITION LANGUAGE (NDL) -

The compiler-level language used to define the network configuration file and local configuration file contents.

NETWORK DEFINITION LANGUAGE PROCESSOR (NDL Processor) -

The network software module that processes an NDL program as an off-line batch job to create the network definition files and other NDL program output.

NETWORK ELEMENT -

Any configurable entity supervised or loaded by the Network Supervisor. A network element consists of any entity in the total computer and terminal network that is not a communication element; this term is usually applied to the data communication network entities comprising the NPUs and their linkages.

NETWORK OPERATOR -

The administrative operator who manages the hardware, linkages, and other network elements of the data communication network by communicating with the Network Supervisor in a host computer. Contrast with local operator. The network operator can also be a local operator, but might not be the operating system operator for the host computer.

NETWORK PROCESSING UNIT (NPU) -

The collection of hardware and software that switches, buffers, and transmits data between terminals and host computers.

NETWORK SUPERVISOR -

A portion of the network software, written as a NAM application program. The Network Supervisor coordinates all of the NPUs in the communication network.

NODE -

A network element that creates, absorbs, switches, and/or buffers message blocks.

OUTPUT -

Information flowing downline from host to terminal.

OWNING CONSOLE -

The interactive terminal associated with a given passive terminal. The owning console enters commands that control the operation of the passive terminal.

PARAMETERIZATION -

The process whereby all of the configurable characteristics of a specific model of terminal are reconciled with the characteristics of that terminal's general terminal class. All characteristics not specifically declared for a given terminal are inferred from the terminal's assigned terminal class. Characteristics can be declared through the terminal's definition in the local configuration file, or by the terminal user through dialog with the Terminal Interface Program, or by an application program servicing the terminal.

PASSIVE TERMINAL -

Any terminal incapable of conducting both input and output and therefore incapable of dialog with the Network Validation Facility. Batch unit record peripherals are typical examples of passive terminals. Also known as a nonconsole device. Contrast with interactive terminal.

PERIPHERAL PROCESSOR UNIT (PPU) -

The hardware unit within the host computer that performs physical input and output through the computer's data channels.

PHASE 1 LOAD ID -

The identifier of the micromemory module portion of the CCP software loaded into a 255x Series Network Processing Unit during the first phase of downline loading.

PHASE 2 LOAD ID -

The identifier of the macromemory module unique to a given 255x Series Network Processing Unit. This module of the CCP software is loaded downline during the second phase of Network Supervisor loading.

PHYSICAL RECORD UNIT (PRU) -

Under NOS, the amount of information transmitted by a single physical operation of a specified device. The size of a PRU depends on the device:

Device	Size in Number of 60-Bit Words
Mass storage	64
Tape in SI format with binary data	512
Tape in I format	512
Tape in other format	Undefined

A PRU that is not full of user data is called a short PRU; a PRU that has a level terminator but no user data is called a zero-length PRU.

PORT NUMBER -

The hexadecimal number assigned to the communication line ports of an NPU.

PRIORITY -

The condition when traffic through the network is maintained preferentially for one or more terminals out of all terminals producing network traffic. Terminals with priority are the last terminals for which network traffic is suspended when traffic must be temporarily stopped because the network is operating at capacity. Terminals with priority do not otherwise receive any preferential treatment of their input or output.

PRU DEVICE -

Under NOS, a mass storage device or a tape in SI or I format, so called because records on these devices are written in PRUs.

RANDOM FILE -

In the context of the NOS operating system, a file with the random bit set in the file environment table in which individual records are accessed by their relative PRU numbers.

REMOTE NPU -

A network processing unit linked indirectly to a host computer through other network processing units.

SEQUENTIAL -

A file organization in which records are stored in the order in which they are generated.

SHORT PRU -

A PRU that does not contain as much user data as the PRU can hold, and that is terminated by a system terminator with a level number. Under NOS, a short PRU defines EOR.

SOURCE -

The terminal or host computer program that created the message.

SOURCE NODE -

The node that interfaces directly to the source of a data message block.

STATION -

A provider and/or recipient of data messages; usually synonymous with a grouping of terminals. This term is used in various ways within mode 4 communications documentation, as shown in table C-1.

SWITCHED LINE -

A communication line connected with one network processing unit but able to be connected to any one of several terminals via a switching mechanism, such as a dialed telephone line.

TERMINAL -

An entity, external to the communications network but connected to it via a communication line, that supplies input messages to, and/or accepts output messages from, an application program. A terminal can comprise only one device (batch terminals) or two devices (interactive terminals).

TERMINAL ADDRESS -

The hardware address of a mode 4 console. This term is used in various ways within mode 4 communications documentation, as shown in table C-1.

TERMINAL CLASS -

An NDL parameter describing the physical attributes of a group of similar terminals, in terms of an archetype terminal for the group.

TERMINAL INTERFACE PROGRAM (TIP) -

A portion of the Communications Control Program that provides an interface for terminals connected to a 255x Series Network Processing Unit. The TIP performs character conversion to and from 8-bit ASCII, limited editing of the input and output stream, parity checking, and so forth.

TERMINAL NAME -
A name of up to seven letters and digits known to the network and used to identify a terminal to the network's local operator.

TERMINAL NODE -
The node ID number associated with an NPU that interfaces with a terminal.

TERMINAL USER -
The person operating the controls of a terminal.

TRANSPARENT MODE -
A software feature provided by the Network Access Method and the network processing unit TIP. When transparent mode transmission occurs between an application program and a terminal, the Network Access Method does not convert data to or from display code, and the TIP does not completely edit the character stream to insert or remove terminal protocol control codes or convert the characters to or from 8-bit ASCII code. When no parity is in effect for the terminal and transparent mode transmission occurs, all eight bits of the character byte can be used to represent characters in 256-character sets (such as EBCDIC).

TRUNK -
The communication line connecting two network processing units.

UPLINE -
The direction of input flow from terminal to host computer.

USER NAME -
The NOS validation file parameter entered by the terminal operator during the Network Validation Facility login procedure.

VERIFICATION RECORD -
A record required in any network configuration file used by the Network Supervisor and in any local configuration file used by the Communications Supervisor.

ZERO-LENGTH PRU -
A PRU that contains system information, but no user data. Under NOS, a zero-length PRU defines EOF.

RESERVED WORDS D

The following words are used by NDL as statement identifiers or as keywords within statements. Their use is limited to statement identification and as values in declarations.

ABL
AL
ALFAM
ALUN
APLBP
APLTP
APPL
ASCII
ASYNC
AUTO
A1
A110
A150
A2
A300
A6
A9
BCD
BCF
BS
BSC
BSZ
B1
B2
B6
CA
CI
CN
CO
COMMENT
CON
CORAPL
CORRES
COUPLER
CP
CR
CSET
CT
DEFINE
DFL
DI
DLC
DLTO
DLX
DO
DT
DUMP
E
EBCD
EBCDAPL
END
EP
HASP
HNODE
HPRE
HOST
HOSTDATA
H2000
IAPPL
IN
K
KB
LAP
LFILE
LI
LINE
LOCAL
LOGLINK
LP
LSPEED
LTYPE
MODE4
MREC
M33
M4A
M40
M4ASCII
M4BCD
M4C
N
NBL
NETNAME
NFILE
NNODE
NO
NODE
NPU
NPUDATA
NPUNAME
NPUTYPE
N2
N2741
O
OCTERM
OP
PA
PDO
PG
PL
PORT
POST
PRE
PRI
PRIV
PT
PW
P1LID
P2LID
RIC
S
SDT
SE
SIZE
STIP
S1
S2
S3
TA
TC
TELENET
TERMINAL
TIPTYPE
TNODE
TRUNK

TSPEED
T1
T4014
UID
USER
USER1
USER2
USER3
USER4
USERPSN
W
YES
XBZ
Z
110
1200
134
150
200UT
2400
2550
2741
2741C
2741E
2780
300
3780
4800
600
711
713
714
714X
731
734
751
9600

FORMAL SYNTAX

The syntax of the Network Definition Language is formally described in Backus-Naur Form, as shown in figure E-1. This formal syntax is provided as background information only; the syntax used in coding an NDL program is described in sections 3 and 4.

```
The following list defines the terms of statement definition:

          <asterisk> : : = *
          <asterisks> : : = <asterisk> | <asterisks> <asterisk>
          <null> : : =
          <blank> : : = ƀ
          <letter> : : = A|B|C|. . .|Z
          <digit> : : = 0|1|2|. . .|9|A|. . .|F
          <alpha> : : = < letter> | <digit>
          <ID> : : = <letter> | <ID> <alpha>
          <number> : : = <digit> | <number> <digit>
          <blanks> : : = <blank> | <blanks> <blank>
          <nblanks> : : = <blanks> | <null>
          <equals> : : = =
          <comma> : : = <nblanks> , <nblanks>
          <period> : : = <nblanks> . <nblanks>
          <colon> : : = :
          <separator> : : = <comma> | <blanks>
          <value> : : = <number> | <ID> | <username>
          <username> : : = <ID> | <ID> <asterisks> |
                           <asterisks> <ID> |
                           <ID> <asterisks> <ID> |
                           <asterisks> <ID> <asterisks>

The following defines the general form of an NDL statement:

          <NDL-statement> : : = <element-name> <colon>
                                <nblanks> <statement-ID>
                                <separator> <declarations>
                                <statement-end>

where:

          <element-name> : : = <ID>
          <statement-ID> : : = <ID>
          <declarations> : : = <declaration> |<declarations>
                               <separator> <declaration> |

          <statement-end> : : = <period>

          <declaration> : : = <keyword> | <keyword> <equals>
                              <value> | <value> | <null>

          <keyword> : : = <ID>
```

Figure E-1. Formal Syntax of NDL in Backus-Naur Form

STATEMENT SUMMARIES

This appendix summarizes the formats of all recognized NDL statements. Formats are presented in a manner similar to that used in section 3, where the valid value declarations are described in detail. The following formats are listed alphabetically by statement name; default values are shown; they are underlined when no dependencies for the default value exist.

	Page
service:APPL [ΔPRIV,UID,DI].	3-12
COMMENTΔ string.	3-16
coupler:COUPLERΔ NPUNAME=npu,NODE=node.	3-6
DEFINEΔ defname=value declaration [,value declaration,...,value declaration].	3-17
END.	3-17
host:HOSTΔ NODE=node.	3-6
host:HOSTDATA.	3-12
lfile:LFILE.	3-12
line:LINEΔ PORT=port,LTYPE=ltype,TIPTYPE=tiptype [,AUTO,DI,LSPEED= $\begin{Bmatrix}\underline{110}\\ \text{lspeed}\end{Bmatrix}$,DFL=dfl,K=k, T1=t1,N2=n2,psn,LAP].	3-14
local:LOCAL.	3-11
loglink:LOGLINKΔ HNODE=hnode,TNODE=tnode [,DI].	3-6
netname:NETNAME.	3-5
nfile:NFILE.	3-5, 3-11
npu:NPUΔ [P1LID=p1lid,P2LID=p2lid,] NODE=node [,DUMP,SIZE= $\begin{Bmatrix}\underline{8}\\ \text{size}\end{Bmatrix}$,NPUTYPE= $\begin{Bmatrix}\underline{2550}\\ \text{AUTO}\end{Bmatrix}$,DI].	3-8
npu:NPUDATA.	3-14

terminal:TERMINAL [ΔSTIP=stip,CA=ca,TA=ta,DT= $\begin{Bmatrix}\underline{\text{CON}}\\ \text{dt}\end{Bmatrix}$,TC=tc,PRI,CSET=cset,TSPEED=tspeed,
NBL= $\begin{Bmatrix}\underline{2}\\ \text{nbl}\end{Bmatrix}$,IAPPL=service,AL=al,ALFAM=alfam,ALUN=alun,
B1=b1,B2=b2,OCTERM=octerm,PA=pa,
PL=pl,PW=pw,BS=bs,CI=ci,CN=cn,CT=ct,DLC=dlc,
DLTO= $\begin{Bmatrix}\text{YES}\\ \text{NO}\end{Bmatrix}$,DLX=dlx,EP= $\begin{Bmatrix}\text{YES}\\ \text{NO}\end{Bmatrix}$,IN= $\begin{Bmatrix}\text{KB}\\ \text{PT}\end{Bmatrix}$,OP= $\begin{Bmatrix}\text{PR}\\ \text{DI}\\ \text{PT}\end{Bmatrix}$,LI=li,PG= $\begin{Bmatrix}\text{YES}\\ \text{NO}\end{Bmatrix}$,SE= $\begin{Bmatrix}\text{YES}\\ \text{NO}\end{Bmatrix}$,
DO=do,PDO=pdo,ABL= $\begin{Bmatrix}\underline{2}\\ \text{abl}\end{Bmatrix}$,BSZ=bsz,DI,W=w,RIC=ric,BCF=bcf,MREC=mrec,SDT=sdt,CO=co,XBZ=xbz].
	4-3, 4-11, 4-13, 4-18, 4-24

trunk:TRUNKΔ PORT=port,NNODE=nnode.	3-10

LIMITATIONS ON CONFIGURATIONS

The NDL Processor permits configuration of more network elements and communication elements than the current version of the standard release of CDC network software supports. The Network Definition Language can therefore be used to define configurations supported by site-developed variants of the standard network software. Practical implementation of the standard network software requires imposition of upper limits on the number of elements configured. These limits are required to avoid excessive central memory usage by the network software. The limits consist of a maximum number of table entries for one or more of seven NDL statements, as described in table F-1.

An NDL program division containing more than the maximum number of any statement listed in table F-1 causes an NDL Processor table overflow condition and the error summary listing diagnostic message mentioned in appendix B. When this error occurs, the NDL Processor ignores all further occurrences of that statement type within the program division. Because there is some dependency among statements, exceeding one of the stated limits can cause the NDL Processor to also ignore statements of a different type that are related to that limit. For example, there is an internal table relationship between TRUNK and NPU statements. Exceeding the number of permitted TRUNK statements can also cause the NPU table to overflow, even when fewer than 22 NPU statements have been processed. Usually, the combination of NPU and TRUNK statements cannot exceed 62 within an error-free NDL program division.

TABLE F-1. MAXIMUM NUMBER OF NDL STATEMENTS

Statement	Maximum Number
APPL	500
COUPLER	10
LINE	430
LOGLINK	72
NPU	22
TERMINAL	500
TRUNK	40

INDEX

COMMENT SHEET

MANUAL TITLE: Network Access Method Version 1
Network Definition Language Reference Manual

PUBLICATION NO.: 60480000

REVISION: J

NAME:

COMPANY:

STREET ADDRESS:

CITY:

STATE:

ZIP CODE:

This form is not intended to be used as an order blank. Control Data Corporation welcomes your evaluation of this manual. Please indicate any errors, suggested additions or deletions, or general comments below (please include page number references).

___ Please reply ___ No reply necessary

NO POSTAGE STAMP NECESSARY IF MAILED IN U.S.A.

FOLD ON DOTTED LINES AND TAPE

TAPE

TAPE

FOLD

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 8241 MINNEAPOLIS, MINN.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION

Publications and Graphics Division

215 Moffett Park Drive
Sunnyvale, California 94086

CUT ALONG LINE

FOLD

FOLD

CORPORATE HEADQUARTERS, P.O. BOX 0, MINNEAPOLIS, MINN. 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.A.

CONTROL DATA CORPORATION