SC30-3142-0
File No. S370-30

ACF/NCP/VS

Network Control Program

System Support Programs

Program Product

Installation

Program Numbers: 5735-XX1
5735-XX3

Release 2

First Edition (May 1979)

This edition applies to Release 2 of the ACF/NCP/VS program product (program number 5735-XX1) and the System Support Programs for ACF/NCP/VS program product (program number 5735-XX3) and to all subsequent releases and modifications unless otherwise indicated in new editions or Technical Newsletters. Changes are periodically made to the information herein; before using this publication in connection with the operation of IBM systems, refer to *IBM System/370 Bibliography*, GC20-0001, for the editions that are applicable and current.

Publications are not stocked at the address given below; requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

This publication has been written by the IBM System Communications Division, Publications Development, Department E02, P. O. Box 12195, Research Triange Park, North Carolina 27709. A reader's comment form is provided at the back of this publication. If the form has been removed, comments may be sent to the above address. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Preface

This publication provides the information necessary to define and generate an Advanced Communications Function for Network Control Program/VS (ACF/NCP/VS) for the IBM 3705-I and 3705-II Communications Controllers.

The publication is directed to system analysts and system programmers responsible for preparing an ACF/NCP/VS to be used in communicating with an IBM System/370 in which one or more of the following access methods are being executed: ACF/TCAM, ACF/VTAM, TCAM, and VTAM. Also in this publication is information on the partitioned emulation programming (PEP) extension to ACF/NCP/VS which permits the 3705 to emulate the operation of an IBM 2701, 2702, or 2703 transmission control unit for specified communication lines. Stations on such lines communicate in emulation mode with application programs in the System/370 via BTAM, QTAM, TCAM, or equivalent access methods that can be used with the transmission control units mentioned.

Note: In this publication, generic terms are used for brevity, as follows. *Network control program* (or NCP) refers to ACF/NCP/VS. *Access method* refers to whichever access method is communicating with ACF/NCP/VS, when there is no need to distinguish between TCAM and VTAM or between their ACF and non-ACF versions. Where necessary, the text refers to *VTAM*—meaning ACF/VTAM or VTAM—and to *TCAM*—meaning ACF/TCAM or TCAM. *ACF/TCAM | iet and ACF/VTAM* refer specifically to the ACF versions of these access methods.

In the context of operations in emulation mode, *access method* refers to whichever access method (BTAM, QTAM, TCAM, etc.) is communicating with those lines operating in emulation mode. Some other terms used in this publication are explained in Chapter 1.

This publication also contains the information needed by users of the IBM Airlines Control Program to specify communication lines on which airlines line control is used. (RPQ numbers 858911 and 858912 must be installed in the 3705 to which such lines are attached.) See Appendix L for information on how to specify communication lines requiring airlines line control (ALC).

The term network has at least two meanings. A *public network* is a network established and operated by common carriers or telecommunication Administrations for the specific purpose of providing circuit-switched, packet-switched, and leased-circuit services to the public. A *user application network* is a configuration of data processing products, such as processors, controllers, and terminals, established and operated by users for the purpose of data processing or information exchange, which may use transport services offered by common carriers or telecommunication Adminstrations.

Network, as used in this publication, refers to a user application network.

Prerequisite and Related Manuals

Prerequisite to use of this publication is a basic understanding of data communications and related access methods. You should also have a general knowledge of the purposes of the IBM 3705 Communications Controllers; this knowledge may be obtained from the publications, *Introduction to the IBM 3704 and 3705 Communications Controllers*, GA27-3057 and *IBM 3704 and 3705 Communications Controllers Principles of Operation,* GC30-3004.

See the *ACF/VTAM Installation* and *ACF/TCAM Installation* manuals respectively, for VTAM and TCAM information.

Within this publication, mention is made of the following publications:

Control Panel Guide refers to:

Guide to Using the IBM 3705 Communications Controller Control Panel (GA27-3087)

Program Reference Handbook refers to:

IBM 3704 and 3705 Program Reference Handbook (GY30-3012)

NCP Program Logic Manual refers to:

IBM 3705 Advanced Communications Function for Network Control Program/VS Program Logic Manual (SY30-3013)

VTAM System Programmer's Guide refers to any of these publications, as appropriate:

DOS/VS ACF/VTAM System Programmer's Guide (SC38-0268)
OS/VS ACF/VTAM System Programmer's Guide (SC38-0258)
DOS/VS VTAM System Programmer's Guide (GC27-6957)
OS/VS1 VTAM System Programmer's Guide (GC27-6996)
OS/VS2 MVS System Programmer's Library; VTAM (GC20-0688)
OS/VS SVS VTAM System Programmer's Guide (GC27-0049)

NOSP Installation Manual refers to:

Network Operation Support Program Installation and Maintenance Manual (SC38-0279 [for DOS/VS]; SC38-0278 [for OS/VS])

TCAM Programmer's Guide refers to any of the following publications, as appropriate:

OS/VS1 TCAM Programmer's Guide (GC30-2054)
OS/VS2 TCAM Programmer's Guide (GC30-2041)
OS/VS TCAM System Programmer's Guide (GC30-2051)

Consult your IBM representatiave for the titles and order numbers of ACF/TCAM publications you may need. The *Teleprocessing Preinstallation Guide for IBM 3704 and 3705 Communications Controllers* (GC30-3020) and the *Teleprocessing Installation Record for IBM 3704 and 3705 Communications Controllers* (GC30-3021) may also be used as aids to defining you network control program. The *Preinstallation Guide* provides the appropriate values for many of the network configuration parameters that must be specified in the control program and gives the value(s) appropriate to each type of station and line set accommodated by the program. The *Installation Record* contains a set of formatted sheets representing the communications controller and the attached lines, with labeled spaces suitable for recording the parameter values appropriate for the network being documented. This record should be filled out soon as the equipment configuration of your network is known, and should subsequently be updated each time any changes are made to the configuration.

Please consult your IBM representative for the editions that are current and applicable.

IMPORTANT

This publication explains what functions the network control program can perform, what network configurations are supported, and how to define and generate a program that will perfrom the functions your installation requires. This book does not contain the restrictions and programming considerations imposed by specific types of stations (terminals) or by other program components (such as CICS) with which the network control program may communicate.

Successfully defining a network control program suitable for your installation requires that you consult the appropriate programming manuals for each of the program components and each of the types of data communication equipment (controllers, stations) that make up your installation. Consult your IBM representative to determine the applicable publications that are available.

CAUTION
NCP Source statements originally assembled under prior (non-ACF) releases of the network control program generation procedure can be reassembled using the ACF/NCP/VS generation procedure. However, all INNODE macros and all CLUSTER macros for SDLC stations that appear in the source statements must first be removed from the statements; appropriately coded PU macros may be used in place of the removed statements. The ACF/NCP/VS generation procedure recognizes only PU macros as representing SDLC physical units.

Contents

Figures

Chapter 1. Introduction

The IBM 3705 Communications Controller can be programmed to communicate with a large variety of remote terminals, transmission control units, and computers. This programming can accommodate many different data communication applications and operational requirements.

Control programs for the communications controller are first defined in the form of macro instruction source statements, then generated by a compilation process, and finally loaded into the communications controller.

Advanced Communications Function for Network Control Program/VS

The Advanced Communications Function for the Network Control Program/VS (ACF/NCP/VS) (referred to in this book as the network control program or NCP) controls the transmission of data between the host processor and the remote stations in the network connected to the communications controller.

An ACF/NCP/VS can communicate with up to eight access methods concurrently via channel connections and/or data links to other network control programs. In a 3705-I it can communicate with one or two access methods over channel connections to one or two host processors. In a 3705-II the NCP can communicate with up to four access methods over channel connections to between one and four host processors. The NCP may also communicate with access methods that are channel-connected to adjacent network control programs, via communication links. Some terminology pertaining to relationships between network control programs and access methods in a network is discussed in this chapter under the heading "Subareas and Network Resources."

The functions executable by ACF/NCP/VS are divided into two major categories: network control functions and emulation functions.

Network Control Functions

Network control functions include the wide range of capabilities for which the 3705-I and 3705-II controllers are principally designed, as follows.

The NCP recognizes and fulfills requests by the communication access method to transmit data to and receive data from the network. In so doing, the program performs whatever operations are needed to establish communications with stations; including polling, addressing, dialing, and answering, as appropriate for the type of station and type of communication line linking the station and the controller. Then the program receives message data into buffers, inserts and deletes transmission control characters as required, and translates message data from processing code (EBCDIC) into transmission code, and vice versa. Finally, the NCP transmits the data from the buffers to an access method or to the stations in the network.

The NCP also governs many aspects of communication between the network and the host processor, such as the amount of data to be accepted from a station once connection is established, the number of devices on a multipoint line with which the access method can communicate concurrently, and exchange of identification sequences with stations on switched lines.

Network control functions also include: (1) automatic error recovery and statistical recording, (2) diagnosis of controller, line, and station malfunctions, and (3) changes to operating parameters during program execution upon request from an access method.

Emulation Functions

Emulation functions comprise a more restricted range of functions, equivalent to those provided by the IBM 2701, 2702, and 2703 (collectively referred to in this book as *transmission control units).* These emulation functions permit most existing user application programs to operate unchanged when a 3705 replaces one or more transmission control units.

You may generate a program that performs only network control functions or both network control and emulation functions, according to the needs of your installation. A program capable of both categories of functions is called a network control program with the partitioned emulation programming (PEP) extension. When generating a program having the PEP extension, you specify, for each communication line in the network, whether that line is to operate in network control mode or emulation mode, or both (alternately). Operation in network control mode means that all of the network control functions apply to data transmission over that line. Operation in emulation mode means that only the emulation functions are performed for that line. These are equivalent to the functions performed by the 2701, 2702, or 2703 to which the line was formerly attached. Because the functions performed by the three types of transmission control units differ in some respects, you specify which of these units is to be emulated for each line.

If you specify operation in both modes, operation can be changed whenever desired from one to the other by command from the access method that communicates with the NCP.

The emulation mode of the NCP with the partitioned emulation programming (PEP) extension allows many programs written to be used with the IBM 2701, 2702, and 2703 transmission control units to operate with the IBM 3705 controllers with no modification. These programs include IBM Type I access methods for the 2701, 2702, and 2703, as well as IBM Type II and Type III programs and user-written programs that interface with the 2701, 2702, and 2703 in a manner equivalent to Type I access method programs. Programs that involve timing dependencies and support of certain special and custom features may, however, require modification.

The emulation mode of operation requires that a type 1 or type 4 channel adapter be installed in the controller for attachment to a byte-multiplexer channel of the System/370. All models of the 3705-I and 3705-II have enough storage to accommodate small networks operated in emulation mode, but larger networks require more than the minimum amount of storage.

Emulation functions, in conjunction with the type 1 or type 4 channel adapter, permit the use of the same control sequences and data transfers as do the 2701, 2702, and 2703. They also provide most of the standard functions of these control units. Not supported are the parallel data adapter, synchronous data adapter type 1 programmable two-processor switch, 230 400 bps synchronous speed, direct attachment of the IBM 1032 Digital Time Unit, the IBM 2712 Remote Multiplexer attachment features, and the reverse channel feature. ASCII transparency can be used only for a communication line serviced by a type 3 communication scanner. (*Exception:* Programming RPQs [PRPQ] are available that permit the type 1 and type 2 scanners to accommodate ASCII transparency and 6-bit transcode.)

Subareas and Network Resources

The term *network control program* (or NCP) appears frequently throughout this publication. When the network configuration includes several network control programs, confusion about which program is being referred to can be avoided by construing the unqualified term, network control program, as meaning that program which is presently under consideration (being planned, coded, generated, executed, etc.). For clarity, this program is sometimes referred to as "the network control program you are defining," or "the present network control program." Where necessary to refer to some other network control program in the network, the text uses an appropriate qualifying word or phrase such as "the distant NCP" or "an adjacent NCP." Similarly, access methods with which the present network control program communicates may be referred to with qualifiers such as "adjacent."

Figures 1-1 through 1-4 show elements of the network as they relate to the present network control program and indicate the terms by which these elements are referred to throughout this publication.

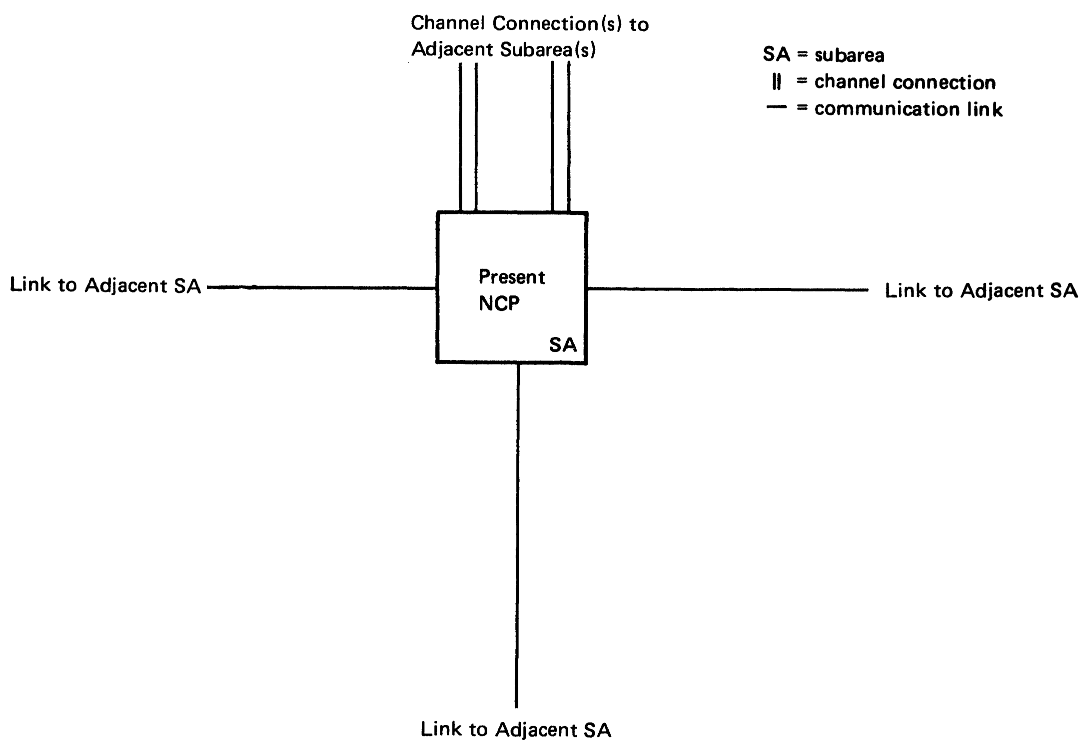


Figure 1-1. Network Control Program and Links to Adjacent Subareas

As shown in Figure 1-1, the present NCP is a subarea of the network and communicates with other subareas (adjacent subareas) over either a channel connection or a communication link. In this publication, the term *subarea* refers to either a network control program or an access method. (Groups of locally attached devices controlled directly by VTAM are also subareas; see the *ACF/VTAM Installation* manual for information on this subject.) Any subarea with which the present network control program communicates directly (that is, without passing through an intervening subarea) is an adjacent subarea.

A local NCP communicates with adjacent subareas through either a channel connection or a communication link. A remote NCP communicates with only a single adjacent subarea over a single communication link. That adjacent subarea is always a local network control program.

Figure 1-2 shows two adjacent local network control programs and one adjacent remote NCP to which the present network control program is attached. The link between any two local network control programs is called a local-local link; that between a local and a remote network control program is a local-remote link. These links always employ synchronous data link control (SDLC) procedures. Figure 1-2 also shows two access methods adjacent to the present NCP. (This figure and the two figures following are intended only to show relationships between the elements of a network as they relate to the present network control program and should not be interpreted as indicating how many of each kind of element may be associated with the NCP.)

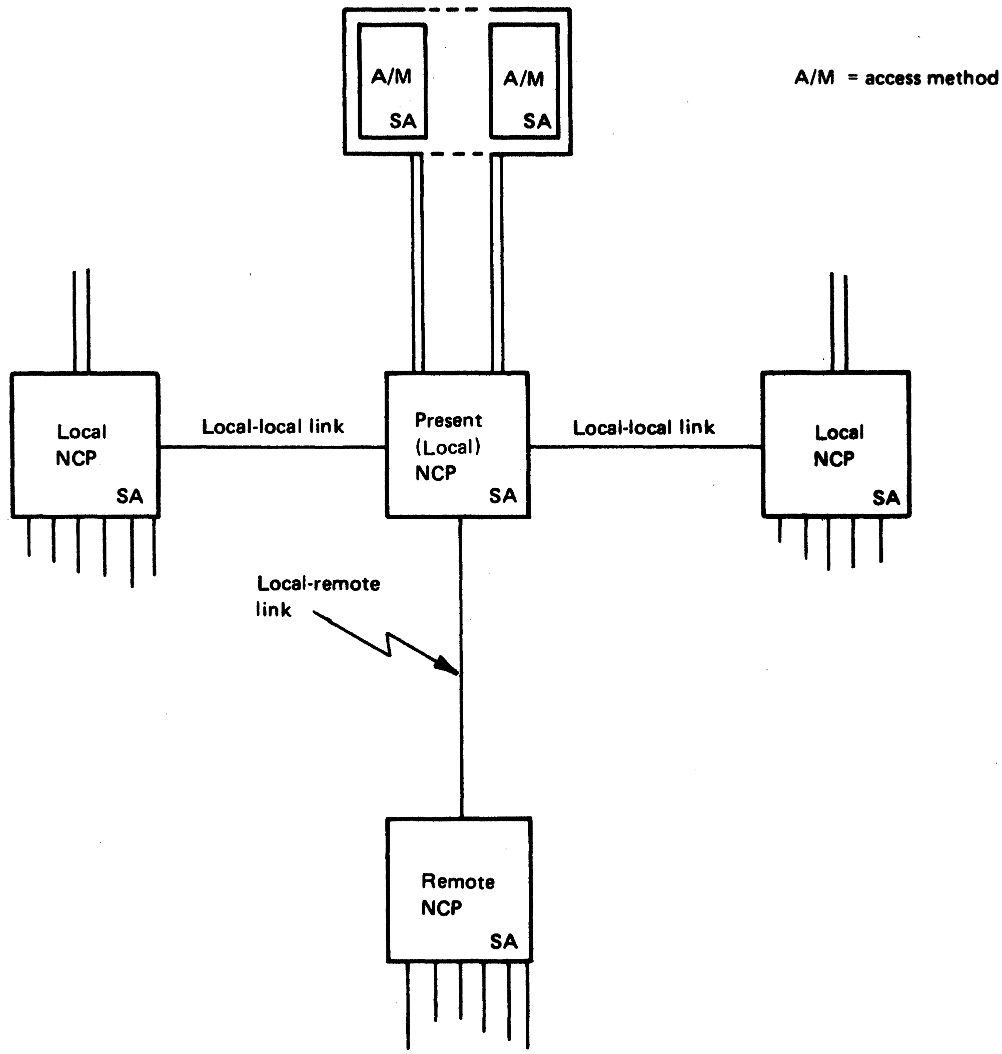


Figure 1-2. Relationship of Network Control Program to Adjacent Subareas

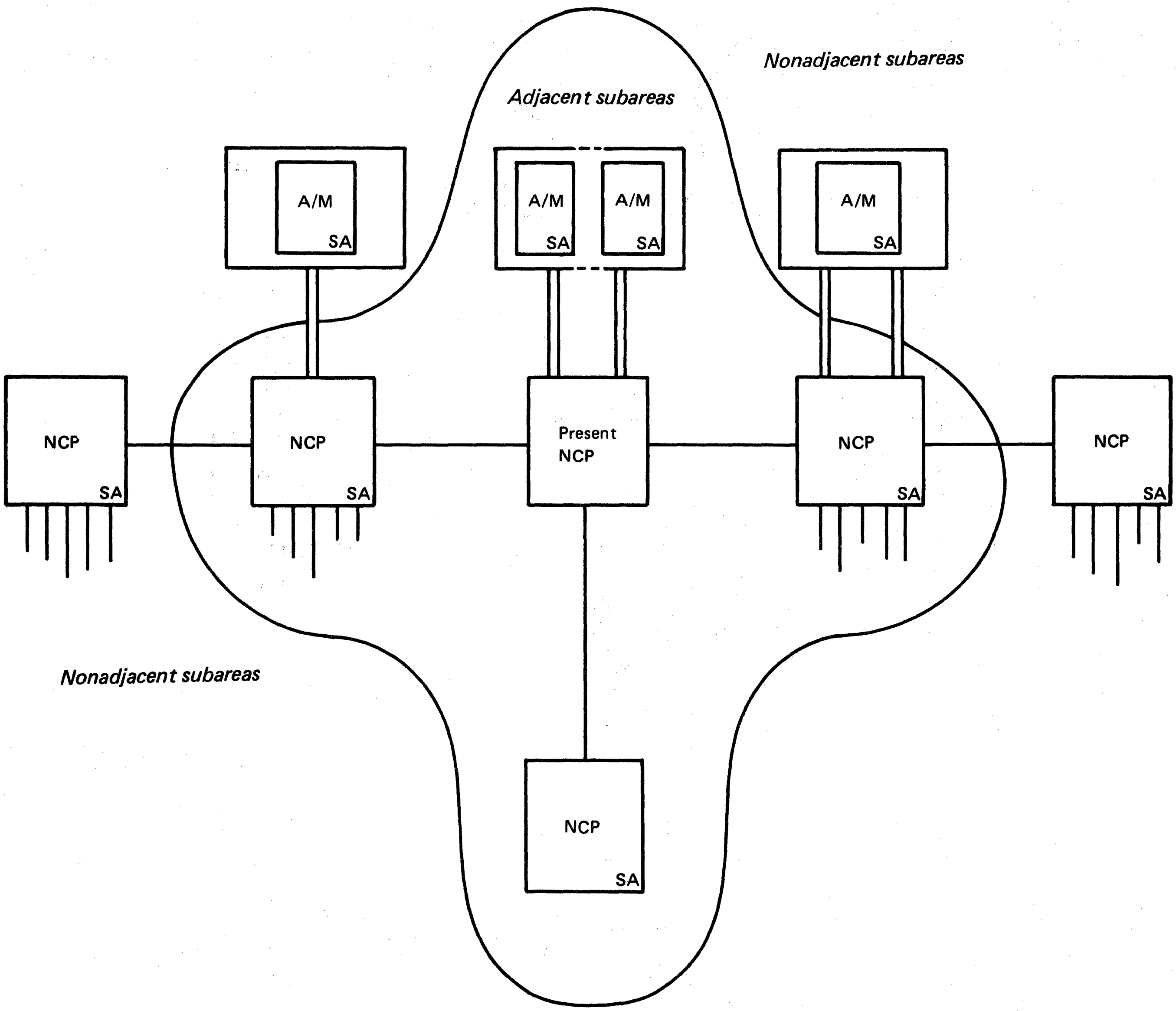


Figure 1-3. Relationship of Network Control Program to Nonadjacent Subareas

The present NCP can communicate with subareas that are separated from it by more than a single link or channel connection, as shown in Figure 1-3. These are referred to as nonadjacent subareas and, like adjacent subareas, may be access methods or other network control programs. Nonadjacent subareas may only be communicated with via adjacent subareas.

Each subarea in a network has a unique subarea address by which it is known to other subareas in the network for the purpose of routing communications between them. The source statements for each program specify that program's own subarea address and the subarea addresses of each other subarea in the network with which it is to communicate. (These addresses are specified in the SUBAREA operands of the BUILD, HOST, PATH, and PU macros, described later in this publication.) The maximum subarea address value that can appear anywhere in the network must also be specified in each NCP (in the BUILD macro) and in each access method in the network.

Figure 1-4 shows a simple configuration involving one access method and one NCP. In this case there are but two subareas, and the subarea addresses assigned might be 1 and 2, as shown. Figure 1-5 shows three access methods and three network control programs, with addresses 1 to 6 assigned to them. Figure 1-6 shows a larger network containing twelve subareas.

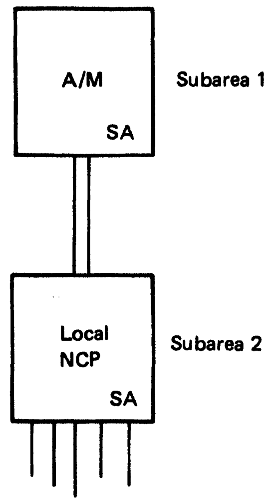


Figure 1-4. A Network with Two Subareas

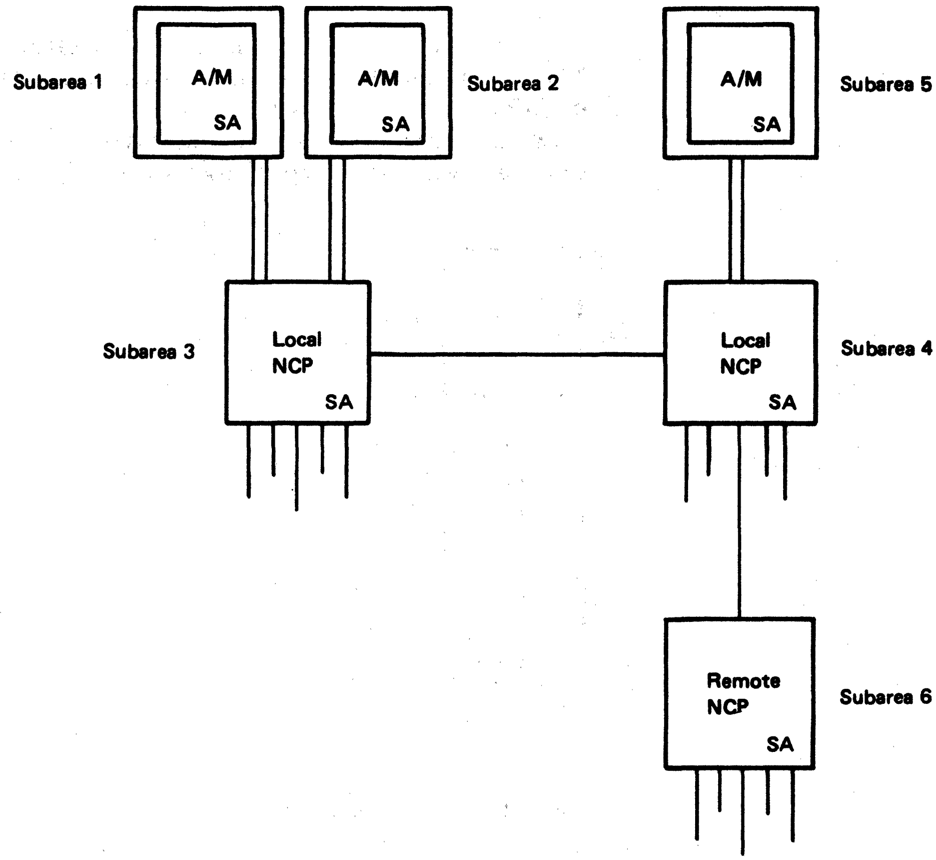


Figure 1-5. A Network with Six Subareas

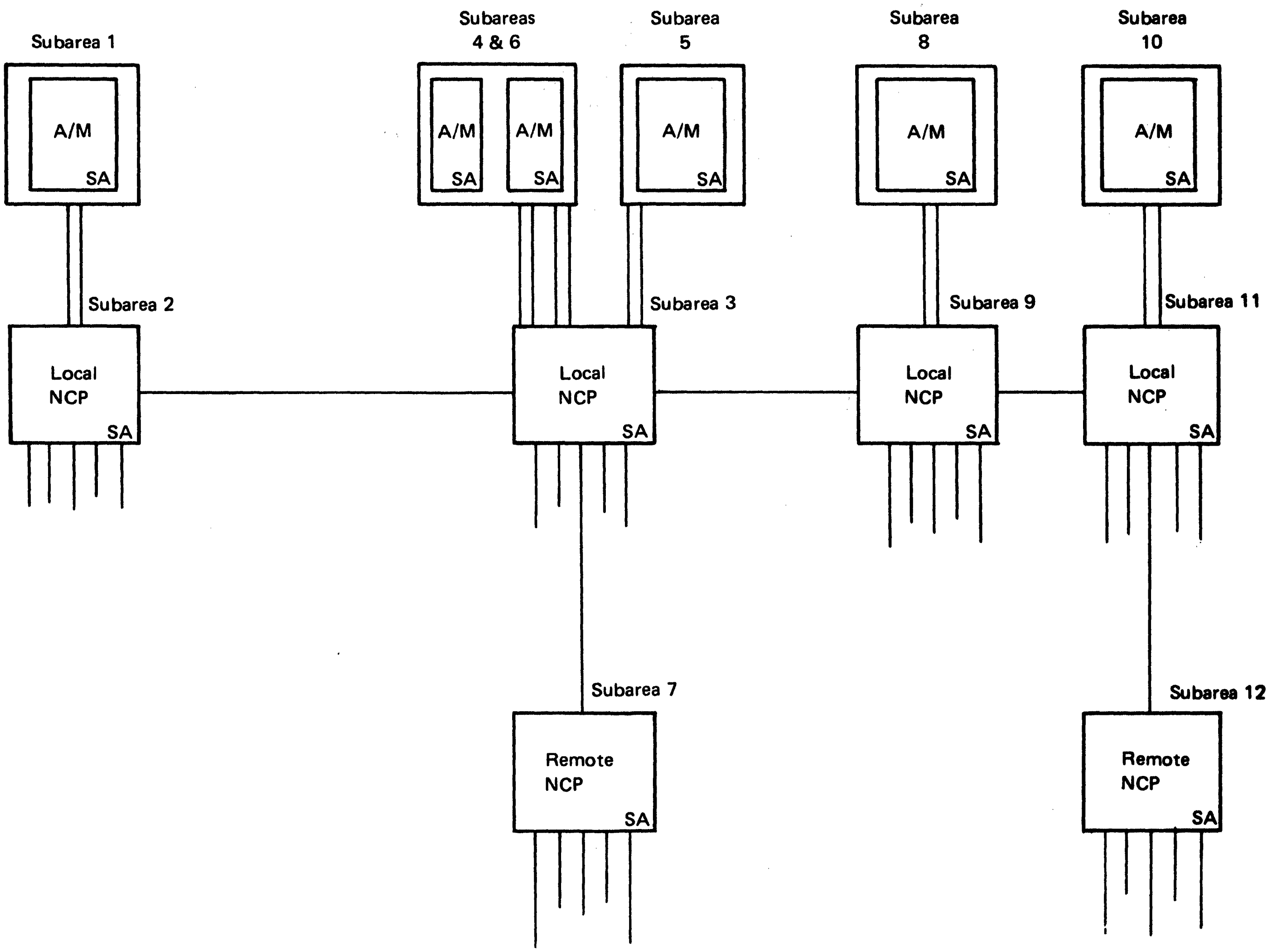


Figure 1-6. A Network with Twelve Subareas

The illustrations thus far have shown only the access method and NCP elements of the network. These elements are kinds of *network resources*. Each communication link, SDLC (SNA) physical unit, and each logical unit, as well as each non-SDLC (non-SNA) station (terminal, transmission control unit, or computer) in the network is also a network resource. (*Exception:* stations communicating with the network control program over lines operating in emulation mode are not referred to as network resources.) Figure 1-7 shows the various kinds of resources in a network. The resources (lines and stations) attached to a network control program constitute a part of that program's subarea, as depicted by the dashed lines in the figure.

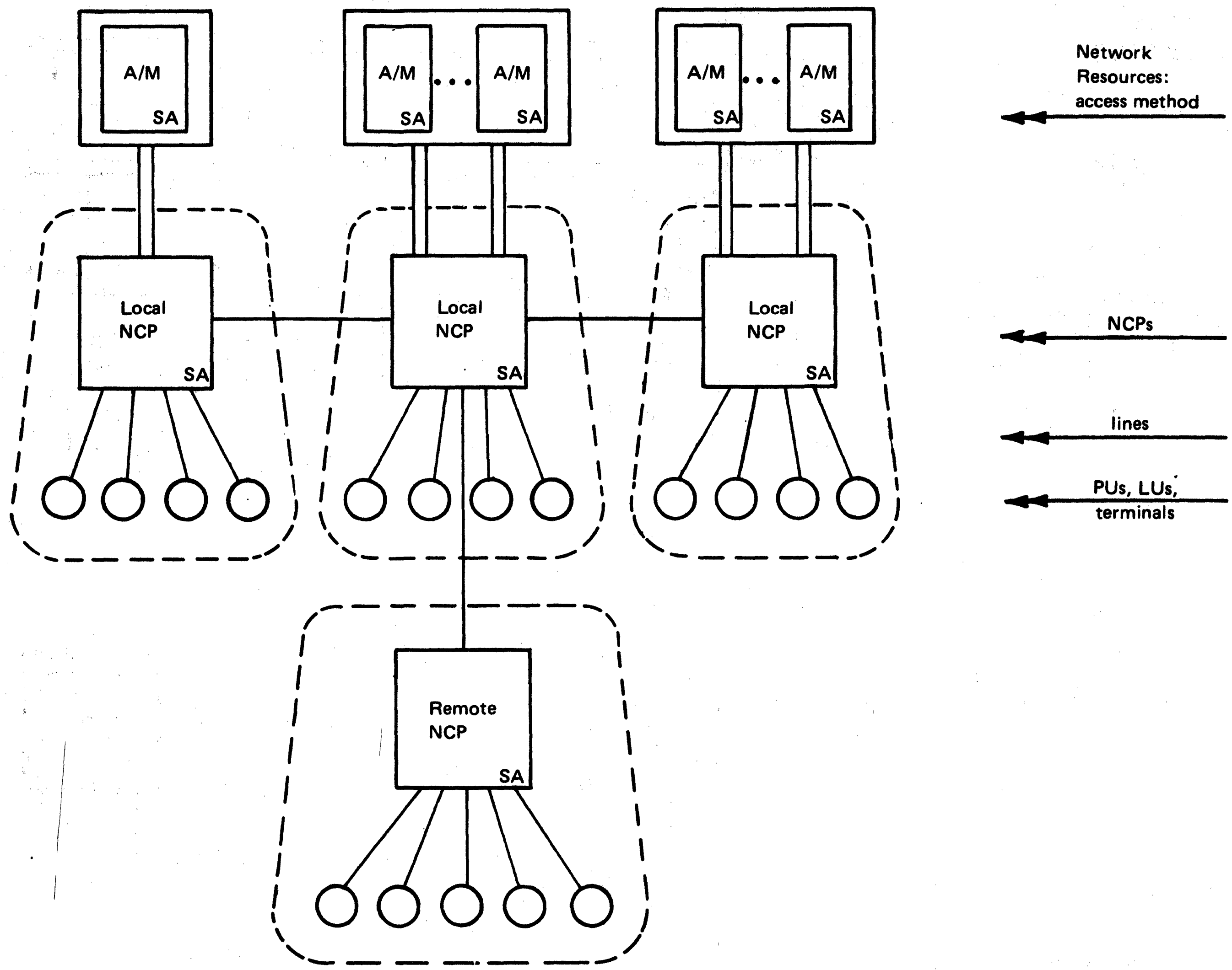


Figure 1-7. Network Resources

Each resource in a network must be assigned a symbolic name that identifies it to the subarea (NCP) that controls it. This name is specified in the name field of the NCP macro that represents the resource. The program generation procedure relates, in index tables, each resource name and its associated subarea address specified in the macros with a numeric value. This value is the *network address* of the resource. The access methods and network control programs in the network use these addresses to route messages to their destinations.

Resources are controlled at the access method level by a part of the SSCP (system services control point) that resides within the access method: the domain resource manager. A domain is the set of resources (links, physical units, logical units, and non-SDLC stations) associated with (owned by) an access method.

A single network may be divided into two or more domains, each of which is controlled by its own domain resource manager. Conversely, two or more existing, independent networks may be combined, by the addition of appropriate communication links, into a single network. Each of the existing networks may constitute a domain in the new network; or, new associations may be formed between the combined resources and the access methods to yield new domains.

Some resources can be "owned" by more than one domain resource manager; such resources therefore belong to more than one domain. Other resources must be owned by a single domain resource manager and therefore cannot be shared between domains.

In using this publication, it is important to be aware that establishing multiple domains and arranging for shared ownership of network resources are performed at the access method level, and require coordination in defining the access methods used in the network. See the *ACF/VTAM* and/or *ACF/TCAM Installation* manuals listed in the Preface for information on these subjects.

Communication between Controller and Host Processor

The network control or emulation mode interacts with one or more access methods executing in the host processor or in each of several host processors. The access method and any associated application programs must be designed to interact with the control program in the manner appropriate to the mode in which the lines served by the access method are to operate—that is, in network control mode or emulation mode, or both.

Communication in network control mode between the NCP and the access method generally consists of an exchange of requests issued by the access method and responses returned by the NCP. Each request and each response contains the control information necessary to identify the network resource to which it applies, the operation required (for example, read or write), and status information pertaining to that operation. Requests and responses also contain the text of messages to be passed between the access method and the network. Some responses from the network control program are unsolicited; that is, not returned in response to a request. Unsolicited responses report error conditions and status information that may develop during operation of the controller.

All requests and all responses for lines operating in network control mode pass between controller and host processor over the network control subchannel. This subchannel is represented by a subchannel address of the host channel to which the controller is attached. There is always one network control subchannel regardless of how many lines are operated in network control mode. If the controller is to perform only network control functions, this is the only subchannel required. Over the network control subchannel also pass load module data sent by the access method loader or independent loader and the contents of controller storage during the dumping process.

Communication in emulation mode between the access method and the NCP is essentially the same as between the access method and the transmission control unit being emulated. As is the case for transmission control units, each communication line operated in emulation mode requires its own host subchannel address. The subchannels associated with lines operated in this mode are called emulation subchannels. In defining a program that performs emulation functions, you associate each communication line with an emulation subchannel address. Use of the multi-subchannel line access (MSLA) feature of the program allows more than one subchannel address to be associated with a line, but communication is possible over only one subchannel at a time. The converse, however, is not true: more than one line cannot be associated with a single subchannel address. (An exception is a program option for emulation of a 2701 that is equivalent to the dual communications interface feature of the 2701. This option allows either of two lines to be associated alternately with a single subchannel.)

For either the MSLA or dual communications interface feature, the line with which a subchannel is to communicate is established by command from the access method. Selection is possible only among the line/subchannel associations established when the program was defined. The source program must be modified and reassembled to change these associations.

A program that performs both network control and emulation functions requires (1) a single network control subchannel for each channel adapter that the NCP is to operate and (2) one or more emulation subchannels for each line to be operated in emulation mode (even if such a line will sometimes operate in network control mode).

Defining the Network Control Program

Achieving an operating network control program is a three-step process. The first step, defining the program, is the most involved. Many different variables and options must be considered in preparing a program that meets the requirements of a particular network configuration and application.

A network control program is defined in the form of a source program consisting entirely of macro instructions called *control program generation macro instructions.* These include configuration macros for specifying the elements of the network and are similar to those used in some data communication access methods (for example, the LINE and TERMINAL macros). The source program, when punched into cards and preceded by the appropriate job control statements, forms the input to the next of three steps, the generation procedure.

Chapter 4 describes each of the characteristics of the network to be considered in defining a program that is to perform network control functions, or both network control and emulation functions for BSC and/or start-stop stations. Chapter 2 provides equivalent information for SDLC stations (network control functions only). Some of these characteristics are generally applicable to most networks. Other characteristics pertain to the capabilities and control techniques distinctive to the communications controllers and the network control program.

Chapter 2 describes each of the characteristics of the network to be considered when defining a program that is to perform network control functions for SDLC stations only. Chatper 3 describes additional functions that must be considered when BSC and/or start-stop stations are included.

Once you understand the characteristics and functions explained in Chapters 2 and 3, you should be able to determine the requirements and options best suited for your network control program. Chapter 5 describes the macro instructions needed to define the control program.

Generating the Network Control Program

After the network control program is defined in the form of a source program containing control program generation macro instructions, it is ready to be generated. Generating the program is a compilation procedure consisting of assembly and link-editing steps. The procedure may be executed in the host processor or in any other central processing unit that (1) can fulfill the operating system assembly and link-editing requirements and (2) has access to the IBM-supplied NCP module libraries. These libraries, supplied by the IBM Program Information Department, must be added to the operating system before any network control programs can be generated. Documentation provided with the libraries explains how to add them to the operating system.

The primary output of the generation procedure is a control program load module, ready for loading into the communications controller. Chapter 4 describes the generation procedure under OS/VS and DOS/VS.

VTAM Initialization Process

The VTAM initialization process requires as input the same control program source statements used for NCP generation, supplemented by several other source statements meaningful only to VTAM. These VTAM-only statements, which consist of separate macros and additional operands of existing NCP macros, may be placed in the network control program source deck either before or after the NCP is generated. Placing the VTAM-only statements in the deck before generating the network control program is recommended. Adding them to the deck after generating the program can introduce inadvertent errors such as misspelled operands or transposed cards. This causes the information given to the VTAM initialization procedure to differ from the generated NCP.

Chapter 5 lists these VTAM-only macros and operands, and indicates where they must appear in the NCP source deck used for VTAM initialization. However, their use is not explained; the *ACF/VTAM Installation* manual tells how to use these VTAM-only source statements. The program generation procedure checks only the keyword part of VTAM-only operands for proper spelling. The procedure does not check the accuracy of the parameters specified and does not verify the appropriateness of the operands coded.

> **CAUTION**
> Because the VTAM initialization does no validity checking of NCP parameters, it is imperative that the NCP source statements be entirely free of errors before being given to the VTAM initialization procedure. Therefore, before VTAM initialization, the network control program must be assembled, via stage 1 of the generation procedure, and reassembled if necessary until the stage 1 output listing shows no MNOTE statements having severity codes of 4 or 8.

Loading the Network Control Program

The final step in achieving an operating network control program is loading the program load module into the communications controller. For a local NCP, this loading procedure requires that a loader utility program be executed in the host processor, with the controller on-line to the processor. For a remote NCP, loading requires that a loader utility program be executed in the host processor and that a network control program be executed in the local controller to which the remote unit is connected. Apart from transferring load module data between the host processor and the controller being loaded, the local NCP does not participate in the loading process.

The loader utility program executed in the host processor may be an access method (VTAM or TCAM) facility or an independent utility program provided by IBM as part of the system support programs. The independent utility, use of which is explained in the *ACF/NCP Utilities* manual or the TCAM facility may be used only for loading a local controller. The VTAM loader facility *must* be used for loading a remote NCP and *may* be used for loading a local NCP. See the appropriate VTAM or TCAM publications, of those listed in the Preface, for information on the access method loading facility.

Obtaining the Contents of Controller Storage

A utility called the dump program allows a selected portion or all of the contents of the controller storage to be transferred from the controller to the host processor, which then prints the contents in hexadecimal format. The dump program, like the loader program, may be an access method facility or an independent program supplied as part of the system support programs. Either dump program has two modules, one of which the host processor transfers to the controller before the dumping process begins. The two modules then interact to transfer the contents of controller storage to the host processor; the host processor module then formats and prints the storage contents.

Executing the dump utility stops operation of the NCP. After the dumping process is completed, a control program must be reloaded into the controller before operations can resume. (An alternate means of obtaining the storage contents that does not require stopping the control program is explained under "Dynamic Dump Utility". This utility is available only in a network control program with the PEP extension.)

Dynamic Dump Utility

The dynamic dump utility is an optional utility program that allows the contents of controller storage to be transferred from the controller to the host processor without interrupting operation of the control program. A full storage dump or a dump of the trace tables for lines in emulation mode can be obtained. In addition, portions of storage can be displayed on the operator's console at the host processor. The utility can also activate or deactivate the emulation mode line trace function, which allows the selection of two program levels to be traced.

Chapter 2. Functions for SDLC Resources

This chapter describes the functions of a network with SDLC resources that you must identify to tailor an NCP to your particular installation.

The chapter is divided into six major sections. The first four define:

- Network characteristics
- Communications controller hardware configuration
- Data transfer between the communications controller and the host processor
- Procedural options governing message traffic between the controller and the network

The remaining two major sections explain (1) the optional diagnostic and service aid facilities that may be included in the NCP, and (2) the program generation options and data sets (files) that the generation procedure uses in creating an NCP load module. The description of each function and option is not exhaustive; it is intended to provide sufficient information to enable you to select the appropriate parameters when coding the program generation macro instructions.

For many functions, especially those relating to the equipment configuration, the decisions about what to code in the macro instructions have been made by the *system designer.* (This is the individual who determines the data communication equipment, network configuration, and communication services that constitute your network.) You need only determine what these functions are and code the appropriate macros and operands accordingly.

Other functions relate to resource, such as the size of the buffers in the buffer pool, or to procedural options, such as pacing. These affect the message-handling capacity and throughput of the communication system and require careful consideration before specifying the parameters.

Once you are familiar with the functions that apply to your configuration and applications, you are ready to code the program generation macro instructions to define your network control program.

Network Characteristics for SDLC Resources

This section applies only to SDLC operation. See "Network Characteristics for BSC and Start-Stop Resources" in Chapter 3 for information on BSC and start-stop operation.

Physical and Logical Units

In this book, *SDLC station* refers to the devices the NCP can communicate with over an SDLC link, such as the IBM 3601 and 3650 Control Units (called cluster controllers), IBM 3767 and 3770-series terminals, and IBM 3705 Communications Controllers.

To the network control program each SDLC station appears as a *physical unit.* A physical unit is a specific set of defined functions performed by programming or hardware. Each physical unit is represented by a PU macro that identifies it as a type 1, type 2, or type 4 physical unit. These type designations are used instead of numeric machine types. The type designations reflect the degree of program function performed by the physical unit. Functionally equivalent physical units have the same type designation. For example, IBM 3270 Models 11 and 12 and 3767 SDLC terminals are both type 1 physical units.

Other parameters that may be associated with the physical unit are the station address, the subarea address assigned to the unit, and procedural options that govern communication between the network control program and the physical unit.

Associated with type 1 and type 2 physical units may be one or more *logical units*. A logical unit on a nonswitched SDLC link is represented by an LU macro. An LU macro creates a logical unit control block (LUB) that defines the logical unit in the NCP.

For a logical unit on a switched SDLC link, the NCP allocates control blocks from a pool of logical unit control blocks. The pool is defined by an LUPOOL or LUDRPOOL macro. In this case, the logical unit parameters are defined and maintained in the host rather than in the NCP. After establishing a session with an SDLC station, the host passes the appropriate logical unit parameters to the NCP, which inserts them in the logical unit control block (LUBs) allocated for that session. These LUBs represent the logical unit for the duration of the session. When the session ends, the NCP returns the LUBs to the LU pool for reuse in other sessions.

The decision as to whether to use the LUPOOL macro or the LUDRPOOL macro depends on the access method(s) with which the NCP will communicate. One LUPOOL macro is required for each access method that *does not* support the Request Network Address Assignment (RNAA) command and will communicate concurrently with type 1 and/or type 2 physical units over switched lines. If the access method(s) *does* support RNAA, one LUDPROOL macro can be coded to include all of the logical units in one pool.

Communication Line Characteristics

A *communication line* as used in this book includes the entire transmission link between a station and the communications controller, including modems, physical conductors, microwave links, satellite links, etc. Communication lines over which synchronous data link control procedures are used are called *SDLC links.*

Line characteristics refer to the functional attributes of the transmission path (for example, whether the communication facility is half-duplex or duplex) and to related aspects of the line such as the physical address, within the communications controller, to which it is attached.

SDLC stations may communicate with a local or remote communications controller over a nonswitched point-to-point link, a nonswitched multipoint link, or a switched link. Each SDLC link must be represented within the network control program by a LINE macro. This macro specifies to the program certain characteristics of the link.

All SDLC links are treated as multipoint lines by the communications controller, regardless of the number of stations on the link. The controller contacts a specific station by sending a control character (physical address) that is recognized only by that station. The station receiving that character responds appropriately; the other stations ignore the transmission unit. SDLC links are specified by POLLED=YES in the LINE macro.

The network control program also requires a service order table for each SDLC link where the controller is the primary station. (The station that controls an SDLC link is the *primary station* for that link; the other station is the *secondary station.)* The service order table contains one or more entries for each SDLC station attached to the link.

A service order table is defined by a SERVICE macro. Code a SERVICE macro directly following each LINE macro that represents an SDLC link where the controller is the primary station.

Half-Duplex versus Duplex Links

The network control program must know whether a communication facility is half-duplex or duplex. This is specified in the DUPLEX operand of the LINE macro. This operand represents the characteristics of the entire communication path including common-carrier lines and equipment and the modems at both ends of the path. The operand does *not* specify the mode of data transfer over the line. It is important not to assume that a two-wire modem is necessarily a half-duplex modem. If the clear-to-send signal lead in the modem is continuously energized, the modem is duplex, regardless of whether it is a two-wire or four-wire modem. If in doubt, consult the installer or supplier of the modem.

Line Speeds and Clocking

The SPEED operand of each LINE macro specifies the data rate at which the line is to operate. This is the rate at which the station, controller, and modems are designed to transmit data over the link between the station and the controller.

If the modem that connects the line to the communications controller has two possible data rates, the DATRATE operand of the LINE macro specifies whether the line is to operate at the higher or lower of the two rates.

The CLOCKNG operand of the LINE macro, specifies whether internal (business-machine) clocking or external (modem) clocking is used for the communication line. Internal clocking is provided by the communication scanner that services the line. External clocking is provided by the modem, whether the modem is a separate unit or built into the controller.

Each communication scanner may have from one to four oscillators. The bit rates for each oscillator must be specified in the SPEED operand of the corresponding CSB macro.

Line Addresses

Each SDLC link attached to the communications controller is identified to the NCP by one or two physical line addresses. If a single line address is used for both transmitting and receiving over the SDLC link, specify that address in the ADDRESS operand of the LINE macro representing the link. If separate line addresses are used for transmitting and receiving, specify both addresses in the ADDRESS operand.

Modem New Sync Feature

Certain types of synchronous modems are equipped with a feature called new sync. This feature reduces the amount of line-turnaround time that is normally expended each time the direction of transmission on the line is reversed. The

NEWSYNC operand of the LINE macro specifies whether this feature is to be used.

NEWSYNC=YES is valid only if the modem at the controller has the new sync feature *and* if the communications controller is the multipoint master station for a duplex line that uses multipoint line control.

Consult your IBM representative or the modem installer or supplier to determine whether the modem has the new sync feature.

NRZI versus NRZ Bit Stream Encoding

The network control program transmits data over an SDLC link in either non-return-to-zero (NRZ) mode or non-return-to-zero-inverted (NRZI) mode. The mode to be used is specified in the NRZI operand of the LINE macro that represents the SDLC link. The choice is determined by the type of modems serving the link.

The modems at each end of the link must maintain synchronism with each other for the entire duration of message transmission. Some modems require bit transitions (0 to 1 or 1 to 0) at intervals in the data stream in order to maintain synchronism. (Such modems are said to be sensitive to transitionless bit streams.) When operating in NRZI mode, the data terminal equipment at the ends of the link manipulate the bit stream transferred to the modems in such a way that transitions are introduced into the bit stream even when the message data being transmitted is transitionless. (Such binary sequences are likely to occur in messages containing storage dump data or IPL data being sent to programmable controllers or terminals.) The transitions thus introduced ensure that the modems remain synchronized. The terminal equipment that receives the altered bit stream reconverts it to its original form.

In NRZI mode, sequences of zeros (000000...) are converted to alternating ones and zeros (101010...), thus satisfying those modems sensitive to transitionless bit streams. Certain other modems, however, are sensitive to just this alternating pattern (101010...) rather than to transitionless bit streams; they will lose synchronism if subjected to 101010 patterns of sufficient length. When the SDLC link is equipped with this type of modems, the terminal equipment must operate in NRZ mode rather than in NRZI mode.

The rules for selecting NRZI versus NRZ operation are:

- If internal (business-machine) clocking is used on an SDLC link, NRZI operation is required; specify NRZI=YES in the LINE macro representing the link.
- If external (modem) clocking is used on the SDLC link, specify NRZI=YES *unless* the modems on the link are sensitive to repeated 101010 patterns, in which case specify NRZI=NO in the LINE macro. Consult your IBM representative or the modem supplier or installer to determine whether the modems are sensitive to repeated 101010 bit patterns.

CAUTION
All stations on the same SDLC link must use the same encoding scheme. Mixing of modes on the same SDLC link will result in total lack of communication between stations on the link. Where the stations on the

SDLC link are remote 3705 controllers, the setting of the NRZI bit in the IPL configuration data set of the remote program loader must correspond to the program-specified option (NRZI=YES or NRZI=NO).

Dynamic Reconfiguration

Dynamic reconfiguration gives the host access method the ability to add and delete type 1 and type 2 physical units and logical units on a leased line to the network configuration without going through an NCP generation. The dynamic reconfiguration function is included in the NCP by coding the PUDRPOOL macro in the program generation.

Note: Dynamic reconfiguration is only for the addition or deletion of physical and logical units on an existing link. Links cannot be added or deleted with this function.

The PUDRPOOL and the LUDRPOOL macros create pools of null physical and logical unit control blocks. When a PU or an LU is dynamically added to the network, a physical or logical unit control block is created in place of one of the dummy control blocks in the pool. The information in the control block is supplied by parameters entered through the access method at the time the unit is added. When a unit is deleted, the control block is returned to the pool for future use. Only one PUDRPOOL macro and one LUDRPOOL macro may be included in the NCP generation. The LUDRPOOL macro creates two logical unit control block pools: one is for logical units added to a type 1 PU and the other is for logical units added to a type 2 PU.

If a generation-defined resource is deleted, in order for that resource to be reused, the assigned network address must be replaced by a network address from the resource vector table (RVT) extension. The RESOEXT operand of the BUILD macro defines the size of the RVT extension and should specify the anticipated number of generation-defined resources that may be deleted and then added at a later time via dynamic reconfiguration. If the RVT extension is depeleted, generation-defined resources can still be deleted but not reused. However, unlike the addresses defined during the NCP generation, the addresses in the RVT extension can be reused when they are deleted.

The MAXPU operand of the LINE macro and the MAXLU operand of the PU macro limit the number of PUs and LUs that can be added to a particular line. These two operands must reflect more than the generated number of physical or logical units if dynamic reconfiguration is to be used.

Physical and logical units that were defined in the NCP generation can be deleted with the dynamic reconfiguration function if the PUDR or LUDR operand was coded YES in their respective PU or LU macro. The dynamic reconfiguration function will not delete any unit unless PUDR=YES or LUDR=YES is coded.

When a physical unit is added to a line, an entry is placed in the service order table for that line. Therefore, the MAXLIST operand of the SERVICE macro must reflect a number greater than the number of entries in the ORDER operand. The difference between these two numbers is the number of physical units that can be added to that line.

The IBM 3270 Models 11 and 12 SDLC terminals need additional programming support over other SDLC terminals. When these 3270 terminals are not included in the initial NCP generation, but it is anticipated that they will be added through dynamic reconfiguration, you must code the DR3270 operand. This operand of the BUILD macro ensures that the proper 3270 support is generated into the NCP. If 3270s are included in the network when the program is generated, no additional support is necessary for dynamic reconfiguration, and the DR3270 operand is ignored.

The default for DR3270 is YES; therefore, when dynamic reconfiguration is specified and no 3270s are anticipated, this operand should be coded NO.

To dynamically reconfigure the NCP, the access method must define a dynamic reconfiguration data set (DRDS) consisting of ADD and/or DELETE statements and their associated PU and LU macros. The reconfiguration information in this DRDS is then used to modify NCP control blocks to reflect the changed configuration. Each time the NCP configuration is changed, a new DRDS must be defined that contains only the current modifications. For example, do not add and delete the same PU or LU to the network in one DRDS. If a device that was added through dynamic reconfiguration is to be deleted, define a new DRDS to delete it; do not put the DELETE statement in the same DRDS as the ADD statement.

The ADD statement directs the access method to ADD a PU or LU to an existing link or PU, respectively. The format of this statement is:

```
ADD        TO=symbol
```

where *symbol* is the name of the link or PU defined in the NCP generation that the new device will be added to.

The DELETE statement tells the access method to delete a PU or LU from an existing link or PU. The format of this statement is:

```
DELETE     FROM=symbol
```

where *symbol* is the name of the link or PU that the device will be deleted from. If a PU is deleted from a link, all LUs associated with that PU are also deleted.

If an ADD or DELETE statement specifies a link in the TO or FROM operand, it must be followed by a PU macro. The PU macro may then be followed by one or more LU macros.

If the ADD or DELETE statement specifies a PU in the TO or FROM operand, it must be followed by one or more LU macros.

For example, if PU6 and its associated LUs (LU1 and LU2) are to be moved from LINE6 to LINE7, the following statements must be placed in the DRDS:

```
name    DELETE    FROM=LINE6
PU6     PU        [operands] optional
name    ADD       TO=LINE7
PU6     PU        operands
LU1     LU        operands
LU2     LU        operands
```

As another example, if only one LU (LU1) is to be moved from PU3 to PU4, the DDRS must contain:

```
name    DELETE    FROM=PU3
LU1     LU        [operands] optional
name    ADD       TO=PU4
LU1     LU        operands
```

The *name* on the ADD and DELETE statement is optional.

When deleting a PU or LU from the configuration, only the *symbol* on the respective PU or LU macro needs to be coded. If any operands are coded, they will be ignored.

When adding a physical unit to a link, the PU macro must be coded as specified in the macro definition in Chapter 5. However, for dynamic reconfiguration, the DATMODE, MAXLU, and SUBAREA operands are ignored. Also, if the MAXDATA operand is omitted, it defaults to 266; and the RETRIES operand is always set to 7 for both the *t* and *n* parameters.

When adding a logical unit to a physical unit, the LU macro must be coded as specified in the macro definition in Chapter 5. The only exception is that the *m* parameter of the PACING operand must be 1; the *n* parameter remains as coded.

In defining the dynamic reconfiguration data set for an ACF/VTAM environment, the first statement must be:

```
VBUILD    TYPE=DR
```

This statement must precede the ADD and DELETE statements. ACF/VTAM stores the DRDS in the system definition library.

For ACF/TCAM, the DRDS must be assembled against the NCP macro library to create the delta resource resolution table (DRRT). The DRRT is then link-edited into the ACF/TCAM partitioned data set that contains the resource resolution tables for NCP load modules.

For more information on defining and using the DRDS with ACF/TCAM, see the *ACF/TCAM Installation Guide*, or for ACF/VTAM, see the *ACF/VTAM Installation* manual.

Communications Controller Hardware Configuration

Several characteristics that must be identified to the NCP reflect the system designer's choice of hardware options for the communications controller. These are (1) the size of storage installed in the controller; (2) the type and number of channel adapters that join the communications controller to the host processor(s); (3) the type, number, and oscillator bit rates of the communication scanners installed; and (4) the interrupt priority to be used for each line serviced by a scanner.

Specify the storage size in the MEMSIZE operand of the BUILD macro. Also specify the type and number of channel adapters in the CA operand of the BUILD macro.

A communications controller can be equipped with from one to four communication scanners. The IBM 3705 models A1, A2, and E1-E8 always have a single scanner. Models B1-B4 can have one or two scanners; models C1-C6, up to three scanners; and models D1-D8, up to four scanners. Models F1-F8 of the 3705-II have one or two scanners; models G1-G8 have three scanners; and models H1-H8 have four scanners. The number of lines serviced by each scanner depends upon the data rates (line speeds) at which the lines operate. Each scanner may be equipped with from one to four oscillators or internal clocks and can therefore provide internal clocking for up to four different speeds of lines. In addition, the scanner may service lines for which external modems (including integrated modems within the 3705) are used, without restriction as to the number of different external clock speeds used for those lines. To service a line that is externally clocked, however, a scanner must be equipped with an oscillator that operates at less than one-half of the data rate of that line. (This oscillator may be the same one that furnishes clocking for one or more of the internally clocked lines.) A scanner equipped with 600 bps and 1200 bps oscillators, for example, could service lines operating at these speeds, using *internal* clocking, and also service lines using *external* clocking at speeds exceeding 1200 bps; for instance, 2000 and 7200 bps. This scanner could not, however, service externally clocked lines of 1200 bps or less because, in this example, there is no oscillator that operates at less than one-half of 1200 bps.

For each scanner, you must specify to the network control program (1) the type of scanner, (2) the machine module in which it is installed, and (3) the bit rates of the oscillators with which each scanner is equipped. This information, like the storage size and channel information, should be obtained from the system designer before you code the program generation macro instructions. Specify the details of the scanners in the TYPE, MOD, and SPEED operands of a CSB macro--one macro for each scanner in the controller.

The NCP is interrupted by the line interface hardware of the controller each time a data character or a data buffer (depending on the type of scanner) is to be sent over or received from a communication line. To avoid character overrun or underrun, lines having a high data rate require service from the program more frequently than lines having lower data rates. Each line serviced by a given communication scanner is therefore assigned an interrupt priority relative to other lines serviced by the same scanner. If all lines on the scanner have the same data rate, the priority may be equal. If the lines have differing rates, however, those with high rates should be assigned higher priority than those with lower rates.

The priority may be 0, 1, 2, or 3 (3 is the highest priority). These priority values are specified in the INTPRI operand of the LINE macro.

Appendix E gives a method for determining the interrupt priority for each line in the network.

Communication Between Controller and Host Processor

Information on both the buffers within the access method and buffers within the NCP must be specified in the HOST macro to properly coordinate data transfers to and from the access method. Each access method with which the network control program concurrently communicates is represented by a separate HOST macro.

Data Transfer from Host Processor to Controller

The amount of data conveyed from the host processor to the communications controller during a single data transfer over the network control subchannel may vary over a wide range, depending on the number of requests and the amount of accompanying message data to be transferred. Efficient operation of the NCP requires that the program preallocate a suitable number of buffers for incoming data transfers, rather than allot buffers one at a time. Once the set of buffers is allocated, data transfer from the access method can proceed without further attention by the network control program's supervisory routine until the data transfer ends or all the preallocated buffers are filled. If the amount of data received during one transfer is insufficient to fill all of the preallocated buffers, the remaining buffers are used for subsequent data transfers until all are filled, at which point the program again allocates the same number of buffers.

The INBFRS operand of the HOST macro associated with an access method specifies the number of buffers the NCP is to allocate for data transfers over the network control subchannel from that access method. You should consider two factors when estimating a value for INBFRS.

If the size of a data transfer consistently exceeds the preallocated buffer space, the NCP supervisory routine is frequently interrupted to provide more buffers for the excess data. The time the program must spend in processing the interrupts reduces the time it can devote to servicing communication lines.

On the other hand, preallocating an excessive quantity of buffers for receiving messages from the access method may deplete the buffer pool to the point that insufficient buffers are available for receiving messages over the communication lines. Buffer depletion is especially likely when (1) the buffer pool is relatively small and (2) a low message rate over the channel from the access method causes the preallocated buffers to be filled slowly, thus unduly delaying return of these buffers to the pool.

In choosing a value for INBFRS, a reasonable balance between degraded program efficiency and unnecessary over-allocation of buffers must be achieved.

Data Transfer from Controller to Host Processor

There is a limit to the amount of data the access method can receive from the NCP during a single data transfer over the network control subchannel. This limit must be specified when defining the NCP so that the program does not attempt to send more data than the access method can accept. You specify this limit with the MAXBFRU and UNITSZ operands of the HOST macro. MAXBFRU designates the number of buffer units the access method allocates for a data transfer, and UNITSZ indicates the size of each unit in bytes. The size of the buffer used to receive a data transfer is the product of the two values. (A buffer unit is the smallest amount of contiguous storage area handled as buffer space; a buffer may consist of one or more units.)

In sending a series of response (or request) blocks to the host processor, the NCP causes the access method to begin receiving each successive block in a new buffer.

In some applications, the access method inserts prefixes in buffers ahead of the message data. An NCP option allows each new block sent to the host processor to be offset from the beginning of the access method buffer by enough space to allow the access method to insert the prefix. The amount of offset is specified in the BFRPAD operand of the HOST macro. OS/VS VTAM requires 28 bytes for the buffer pads; DOS/VS VTAM requires 15; ACF/VTAM requires 0; OS/VS TCAM requires a minimum of 17 bytes; ACF/TCAM requires 17-28 bytes.

Procedural Options

Several procedural options characterize the operation of SDLC links. These options include (1) the manner in which the NCP starts up and shuts down the network, (2) the amount of data to be transferred at one time between stations and the controller, and (3) the amount of data to be accumulated from a station before passing the data to the access method. By careful selection of these options you can customize a network control program to best meet the requirements of your data communication applications.

Some procedural options require no more than a simple yes/no choice as to whether the option is to be included. Other options require you to choose from a range of values, such as the size of NCP buffers or the maximum amount of data to be transferred at one time between the NCP and stations in the network.

Buffer Size

The NCP contains one buffer pool of fixed-size buffers. Buffers from this pool are used for all message data. Specify the size of the buffers to be in the BFRS operand of the BUILD macro. (The minimum is 56 (60, if on-line testing is supported); the maximum is 248. The buffer size is always a multiple of 4 bytes.

Buffer initialization occurs immediately after the NCP is loaded into the communications controller. During initialization, the NCP formats buffers in all controller storage remaining after the program is loaded. You can determine the number of buffers in the pool by dividing the buffer area by the buffer size plus 12 bytes (for buffer chaining fields).

Path Information Units

The basic unit of transmission in the network is the *path information unit* (PIU), which consists of network control and routing information and accompanying message text (optional). A PIU either requests a particular data communication operation (request PIU) or indicates the result of an operation (response PIU). Path information units associated with SDLC physical units can originate at either the host processor or the physical unit.

Under access method control, the NCP establishes physical and logical connections between the access method and the SDLC stations. While a connection is established, the NCP automatically controls the operation of each SDLC link in response to the data transfer and requested control operations.

Several parameters govern the amount of data the network control program sends to a station on an SDLC link.

The size of the buffer within an SDLC physical unit must exceed by at least 5 bytes (for a type 1 physical unit) or 9 bytes (for a type 2 physical unit) the size of a NCP buffer. In the MAXDATA operand of the PU macro representing a physical unit (type 1 or 2), you specify the maximum amount of data, in bytes, that the physical unit can accept.

The MAXOUT and PASSLIM operands of the LINE macro determine how PIUs are sent to an SDLC station. The MAXOUT operand allows you to specify how many PIUs or PIU segments (up to seven) can be outstanding at any given time (that is, the number that can be sent to the SDLC station before an acknowledgment is required from the station.

The maximum number of PIUs or PIU segments that can be outstanding is seven because of the sequence-numbering scheme used in SDLC to identify PIUs. Unless you specify a different value in the MAXOUT operand of the PU macro that represents the station, only one PIU is sent to each station before an acknowledgment is required. The higher the value of this parameter, the greater the utilization of the link. However, a high maximum-outstanding value also results in more PIU retransmission when an error occurs. Thus for links on which high error rates are experienced, you may wish to select a maximum-outstanding value lower than that used for links having lower error rates.

You may wish to cause some stations on an SDLC link to be serviced more frequently than others by representing them more times in the service order table for the link. By thus allowing more opportunities for contact with a particular station, relative to others, the total amount of data transferred to the station may be similarly increased. You may counter this effect, however, by using the PASSLIM operand to restrict the amount of data (number of PIUs) exchanged with the station for any one appearance of the station in the service order table.

Only one PIU per pass is sent or received unless you specify a larger value in the PASSLIM operand of the PU macro representing the station.

CAUTION
When choosing values for MAXOUT and PASSLIM, observe any restrictions imposed on these operands by specific types of SDLC stations. See the appropriate programming publications for the types of stations used in your network for such restrictions.

Pacing

The outbound pacing option limits the number of path information units sent to a logical unit before an acknowledgment is required. This option can be used to prevent needless transmission of PIUs to a logical unit that is momentarily unable to accept them. The PACING operand of the LU macro allows you to specify how many PIUs the NCP can send in each transmission.

When the specified number of PIUs has been transmitted, the NCP waits for a pacing response from the LU before sending any more PIUs. See Appendix G for an example of pacing.

A similar function, called inbound pacing, applies to message flow from the NCP to the access method. This function is controlled by the access method and does not require any parameters in the NCP source statements.

The NCP also allows primary and secondary LUs to negotiate certain session parameters through the Bind session command. Therefore, in some cases, pacing parameters may be changed for a particular session in order to better utilize the communication link. At the end of these negotiated sessions, all values return to those specified in the generation parameters.

When choosing a value for the PACING operand, restrictions may be imposed on this operand by specific types of SDLC stations. See the appropriate programming publications for pacing restrictions for the types of stations used in your network.

Half-Duplex versus Duplex Data Mode

Local-local and local-remote SDLC links may operate in either half-duplex or duplex data mode. If the SDLC link has two paths, as indicated by use of separate transmit and receive addresses, simultaneous sending and receiving on the link is possible. This is called operation in duplex data mode. If, on the other hand, the same address is used for both transmitting and receiving, these two functions must alternate (half-duplex data mode). If separate transmitting and receiving paths are available, the local NCP will operate the SDLC link to a remote or local network control program in duplex data mode, unless you specify DATMODE=HALF in the PU macro representing the remote NCP.

Network Slowdown

The NCP can receive message data from the access method and from the SDLC links only as long as it has buffers available for the data. The program normally receives and sends data at the same average rate, although momentary overloads can occur. When the program receives more data than it sends during a given time interval, it can exhaust its supply of buffers. To prevent this, the network control program continuously monitors its supply of buffers and, when the supply falls to a specified level, automatically enters *slowdown mode.* The level is specified as a percentage of the total number of buffers in the program.

When in slowdown mode, the program reduces the amount of data it receives from SDLC links and from the access method(s) but continues to send at the normal rate. Since the rate at which buffers are released exceeds the rate at which new buffers are obtained for receiving data, a net gain in the number of available buffers results. When the buffer supply is sufficiently replenished, the program automatically resumes normal operation.

The SLODOWN operand of the BUILD macro allows you to specify 12, 25, or 50 percent as the minimum percentage of available buffers. When this percentage is reached, program enters slowdown mode. However, during initialization, the NCP dynamically increases the percentage you specify if the minimum NCP buffer requirements cannot be met. The minimum number of buffers that the program must contain for the percentage values allowed is 80 buffers, for 12 percent; 40 buffers, for 25 percent; and 20 buffers for 50 percent.

Error Recovery and Recording

Transient noise on the communications line and intermittent hardware malfunctions are the most common causes of errors affecting communication networks. By implementing the appropriate error recovery procedures, most of these errors can be recovered from and go unnoticed. When an error is detected in received data, the NCP signals the station to retransmit the data. On the other hand, when the station detects an error, it informs the NCP and the NCP then retransmits data to the station.

The maximum number of retransmissions may be specified for each SDLC station in the network. If error-free transmission is not achieved before the retransmission limit is reached, the network control program indicates the fact in its response to the access method.

If the error is the kind that inherently cannot be recovered from (such as a modem error), the NCP does not attempt error recovery. In this case, the NCP records the error status and transmits the indication to the host processor.

The number of error recovery attempts for errors affecting message data is determined by the RETRIES operand of the LINE and PU macros representing the SDLC link and station. For errors occurring when receiving from a station, there can be one retry sequence. For errors occurring when transmitting to a station, there can be one or more sequences of retry attempts, with a pause between successive sequences. Specifying a pause of several seconds between sequences allows time for transient noise conditions on the link, which may be responsible for the repeated errors, to subside.

The number of sequences and the pause are specified in either the PU macro for the station or the LINE macro for the SDLC link. The number of retries per sequence is specified in the LINE macro.

Normally error recording is done only for permanent errors. The initial error status causing the error recovery to begin and the ending error status is recorded for transmission to the host. Operator commands can be issued to cause the NCP to implement *intensive-mode* error recording. This mode records all temporary errors as they occur. The expanded statistics made available by this function will often preclude the need to run specific link tests and thus provide more efficient, timely problem determination.

Note: Intensive-mode error recording puts an added burden on the buffer pool and should be used with caution.

Automatic Network Shutdown

Part or all of the network attached to a communications controller and currently operating in network control mode is shut down automatically, in an orderly manner, under any of several conditions as explained below. (Any lines currently operating in emulation mode are unaffected by shutdown of lines in network control mode.)

The orderly procedure is called *automatic network shutdown* (ANS). The ANS facility is included in the NCP unless you specifically exclude it by coding ANS=NO in the BUILD macro. The ANS facility is required if the network is divided into multiple (two or more) domains. See the *ACF/VTAM* or *ACF/TCAM Installation* manuals listed in the Preface for information on multiple-domain networks. (Apart from automatic network shutdown, individual lines and stations may be deactivated and reactivated by commands from the access method to the NCP.)

Automatic network shutdown is performed for network resources on behalf of the SSCP (system services control point) that currently owns the resources when the NCP loses its ability to communicate with that SSCP. The network control program may detect the loss of the SSCP if the SSCP is adjacent (residing in a host processor channel-connected to the NCP), or an adjacent NCP in the path to the affected SSCP may notify the present network control program of the loss of the SSCP.

Automatic network shutdown occurs for all or any part of the network under the following conditions.

Network controlled by local NCP:

- An adjacent access method fails to respond to the NCP within a specified interval after the NCP has presented an attention signal to the channel by which it communicates with that access method. This interval is specified in the TIMEOUT operand of the HOST macro that represents the access method.
- An adjacent network control program notifies the present NCP that it has lost contact with a subarea in the network.
- The SSCP ends its session with the network control program (by sending a Deactivate PU command).
- The network control program receives, from an SSCP with which it is currently in session, an unexpected request to establish a session with that SSCP. Such receipt of a second request may indicate that communication between the SSCP and the NCP was interrupted (possibly with resultant loss of data) without awareness by the NCP that the interruption occurred. (This can happen if, at the moment of interruption, the NCP did not have an attention signal present on the channel.)
- A shutdown request is entered at the control program of the communications controller.

Network controlled by remote NCP:

- The remote NCP detects a lapse in communication activity over the local-remote link it is presently using to communicate with the local network control program. The lapse may occur through outright failure of the link or through badly degraded performance of the link as indicated by exhaustion of error-recovery procedures performed by the local NCP. The lapse interval is determined by the value you specify in the ACTIVTO operand of the GROUP macro representing the SDLC link(s) to the local network control program. This interval must be sufficiently long for the local NCP to complete its error-recovery procedures for the link.
- The SSCP ends its session with the network control program (by sending a Deactivate PU command).
- The network control program receives, from an SSCP with which it is currently in session, an unexpected request to establish a session with that SSCP. Such receipt of a second request may indicate that communication between the SSCP and the NCP was interrupted (possibly with resultant loss of data) without awareness by the NCP that the interruption occurred.
- The local network control program, upon entering automatic network shutdown for the link to the remote controller, signals the remote NCP to shut down the network controlled by the remote NCP.

The NCP take the following action for each kind of line and station undergoing shutdown:

For SDLC links, the network control program:

- Dissociates the link from the owning SSCP with which communication has been lost.
- Disables the link, if it is a switched link, so that it cannot answer calls from stations.
- Cancels the line trace or online test operation if the link is currently being traced or is undergoing online testing.
- Breaks the switched connection, if any, to the station.

For SDLC stations where ANS=STOP is specified in the PU macro, the NCP:

- Sends a Disconnect command to, and stops polling of, the station.
- Dissociates the station from the owning SSCP with which communication has been lost.
- Cancels any sessions in which the station is currently active.
- Cancels any problem determination functions, intensive mode recording and link tests that are in progress with the station.

For SDLC stations where ANS=CONTINUE is specified in the PU macro, the NCP:

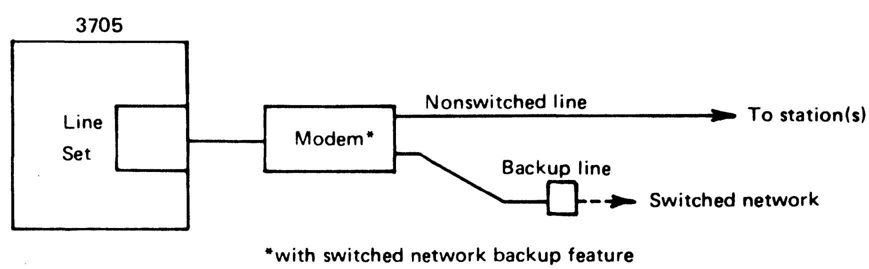
- Dissociates the station from the owning SSCP with which communication has been lost.
- Cancels the sessions, if any, between the stations and the SSCP with which communication has been lost.
- Permits to continue any existing sessions with logical units not affected by loss of the owning SSCP.

Switched Network Backup

The switched network backup facility of the NCP permits communication between a communications controller and an SDLC station over a temporary, switched communication path provided as an alternate, or backup, link to the usual (principal) nonswitched point-to-point or multipoint link. Provision of a backup link permits communication between controller and station to continue despite failure of the principal link.

The availability of a backup link requires the installation of appropriate equipment at the controller and stations.

The form of switched network backup used for SDLC links is called same-port backup. This technique requires the use of an IBM 3872, 3874, or 3875 modem equipped with the switched network backup feature to attach the controller to both the principal, nonswitched SDLC link and the switched telephone network, thus:



A similar arrangement is required at each station for which the backup path is to be provided. In the event the principal, nonswitched link fails, the operator at the host processor can (1) deactivate it by VTAM or TCAM operator command, (2) switch the operation of the modem at the controller to backup mode (and, by voice communication, cause the operator at the remote station to switch the station modem similarly), (3) reactivate the link (as represented to the NCP and access method--not the actual failed link) with a VTAM or TCAM operator command, and (4) establish the dialed backup connection with the station.

A single LINE macro represents the principal and the backup link to the NCP. The NCP is not aware of, and does not participate in establishing, the switched backup connection. Once the backup connection has been manually established, the program simply operates the link in the same way as it does the principal, nonswitched link.

In the case of a backup facility provided for a nonswitched *multipoint* link, the backup connection can be made with only one station on the principal link at a time. A separate backup connection must be made with each station in turn if several or all stations are to be contacted using the backup facility.

Restriction: Because a switched backup connection can operate only in half-duplex data mode, and the NCP operates the principal and the backup link in exactly the same manner, the principal link also must operate only in half-duplex data mode. Specify this mode by coding one line addresses in the ADDRESS operand of the LINE macro representing the principal link. Further, the physical operation of the link also must be half-duplex (DUPLEX=HALF specified in the LINE macro) to allow proper communication over the switched backup link. Line-turnaround time can be minimized by internally connecting the 3872, 3874, or 3875 modem for continuous-carrier operation.

Note: The value specified in the ENABLTO operand of the BUILD macro must be carefully chosen in order to avoid time-outs during the manual dialing operation. (See the description of the ENABLTO operand in Chapter 5.)

See the appropriate *ACF/VTAM* or *ACF/TCAM Operation* manuals for information on the operator commands used in establishing switched network backup operation.

Backup Local-Local and Local-Remote SDLC Links

To minimize the possibility of prolonged disruption of communication between two communications controllers because of failure of the SDLC link joining them, one or more alternate, or backup, SDLC links may be provided. The alternate links must be dedicated to backup use: they cannot be used for communication with other stations even if currently unneeded for backup use. Typically, for economic reasons, a backup link will comprise a switched line; however, the link can comprise one or two nonswitched point-to-point lines, just as the regular (principal) SDLC link does. Regardless of whether switched or nonswitched lines make up the SDLC link, nonswitched line control discipline is used on the link. (DIAL=NO must be specified in the GROUP macro for the alternate links.) In the case of a switched link, specifying DIAL=NO means that the connection between the controllers must be established by manual dialing.

Note: A backup link between two communications controllers cannot operate via the switched backup feature of modems equipped with this feature (for example, the IBM 3872 and 3874 modems). The backup link between two communications controllers must use a separate line interface address.

Switch-over from the principal to an alternate link is accomplished by the primary NCP for the link, the access method owning the link, and the system operator, as follows. (Assume that NCP 'A' is the primary network control program for the link and NCP 'B' is the secondary NCP for the link.)

1. Upon discovering (by exhausting all error recovery procedures) that the principal link has failed, NCP 'A' notifies the access method and returns

to the access method, with an indication of path error, all requests (currently pending or subsequently received) for communication with stations controlled by NCP 'B'.
2. The access method then informs the system operator, via console message, that the link has failed.
3. The operator must now select an alternate link and enter commands specifying that link and the controller ('B') to be associated with it. If the link is switched, he must also dial the telephone number of the controller.
4. The access method sends the appropriate commands to NCP 'A' to contact controller 'B' over the alternate link.
5. Once contact is established, normal communication is resumed between the two network control programs over the alternate link, provided that ANS=YES is specified in NCP 'B'.

If ANS=NO is specified instead (valid only if NCP 'B' is a remote NCP), NCP 'B' awaits resumption of traffic from NCP 'A' until the interval specified by the ACTIVTO operand in NCP 'B' expires, at which point controller 'B' enters IPL-required state and must be reloaded before it can resume communication with NCP 'A'. If you do not specify an activity time-out (ACTIVTO=NONE) in NCP 'B', that program waits indefinitely for traffic to resume from NCP 'A' over the local-remote SDLC link.

Note: If the alternate link is switched, the enable time-out specified in the ENABLTO operand of the BUILD macro must be carefully selected to allow the appropriate interval for the dial connection to be made. See the discussion of this subject under the ENABLTO operand (BUILD macro) in Chapter 5.

Reloading of a remote controller that is in IPL-required state can occur only after the remote program loader has been transferred into the controller storage from the diskette within the controller. The remote program loader monitors each local-remote SDLC link indicated as active within the IPL configuration data set. (The IBM customer engineer sets this indication for each SDLC link that is to be active.) When polled on the link selected by the operator, the remote program loader transmits a request for loading (or dumping) over that link. The access method than transmits the remote NCP load module to the remote controller.

An important point is that the secondary NCP for the link does not directly detect failure of the link because it performs no error-recovery actions for the link. Instead, the expiration of an activity time-out for the link resulting from link failure or shutdown of the link by the primary NCP causes the secondary NCP to perform any necessary automatic network shutdown (assuming that [ANS=YES] is specified in the program).

Note: Because the primary NCP cannot successfully contact the secondary NCP until the secondary program detects failure of the link, the activity timeout should be chosen to avoid an overly long wait before the secondary NCP detects failure. The timeout must, however, be long enough that failure is not indicated prematurely; that is, before the primary NCP has exhausted its error-recovery efforts on the link. Thus, the value for the ACTIVTO operand must be carefully selected; a formula for determining this value appears in the description of the ACTIVTO operand in the GROUP macro.

Use of the automatic network shutdown function in the remote NCP for a local-remote link, though not essential, is highly recommended. If you exclude this option, expiration of the activity timeout causes the remote program to abnormally end (abend). If you specify ACTIVTO=NONE, the program waits indefinitely for traffic from the local NCP over the link by which the remote controller was loaded.

After the principal link has been repaired, the system operator can initiate changeover from the alternate link back to the principal link. To do so, he first enters TCAM or VTAM commands to deactivate the alternate link (which stops traffic on this link) and then enters commands to activate the principal link. After the link is activated, the access method restarts the network, and normal communication traffic can resume over the principal link.

The alternate link need not have the same operating parameters as the principal link. It might, for example, operate at half the speed of the principal link.

If the backup link comprises a switched line, data transmission between primary and secondary network control programs, in both directions, is always in half-duplex mode even if the principal (nonswitched) link is duplex.

Both principal and alternate links are represented within the NCP by LINE macros.

For local-to-local operation, the principal link between the primary and secondary controllers is identified by specifying the SUBAREA operand of the associated PU macro in both the primary and secondary network control programs. For alternate backup links, omit the SUBAREA operand.

For local-to-remote operation, the principal link is identified by specifying the SUBAREA operand of the associated PU macro in *only* the local (primary) network control program. In the NCP for the remote controller, all links between the remote and local controllers are specified as alternate backup links (omit the SUBAREA operand). The principal link in the remote is determined at execution time.

Diagnostic and Service Aids

The network control program diagnoses difficulties in network operations by means of several diagnostic and service aids. These aids are useful in identifying malfunctions within the network and the NCP. Some aids are standard (always present in the program); others are optional. Inclusion of all service aids is recommended.

Online Link Testing

Online link testing is a diagnostic aid by which a terminal or console may request a variety of tests to be performed upon a communication link. The terminal operator requests the test by entering a test-request message having a defined format. The requested test is performed, and the results are printed at the terminal or console. This diagnostic aid, important in problem determination and online maintenance of communication links, is included in a NCP unless you exclude it via the OLT operand of the BUILD macro.

The NCP is only an intermediary on online test operations. It recognizes test-request messages, routes them to the access method, recognizes interpretive commands from the host processor, and executes data communication operations accordingly. Recognizing the message as a test request, the program sends it unchanged to the access method.

The access method detects that the message requests the online test function and interprets the parameters within the message to determine the kind of test to perform. The access method then selects the appropriate test modules and sends a series of interpretive commands to the NCP that indicate what data communication operations to perform. The NCP executes these operations and returns responses as necessary to the access method. Upon analyzing the responses, the access method determines what further operations to perform and sends the NCP the appropriate interpretive commands.

Online link test operations require buffer space to hold the interpretive commands and an online test control block. These buffers, which the program obtains from the same pool as for normal operations, are required only for the duration of the test operation.

The link test does not require disruption of traffic involving any station but the one participating in the test. Further, more than one link test can be run at the same time, each involving a different station on the link. Once each link test is completed, the result of the test is sent to the requestor of the test.

Link testing is important whenever degradation or failure of a communication line is suspected. The nondisruptive nature of the enhanced capability means that link test will more likely be run as soon as the problem is suspected rather than be deferred until normal operations are at a minimum.

Note: Diagnostic programs that communicate with the network control program via VTAM or TCAM (for example, TOTE) may impose restrictions on the NCP buffer size (specified in the BFRS operand of the BUILD macro). See the appropriate manuals for such diagnostic programs for restrictions that may apply.

The NCP can execute online testing operations concurrently for any number of communication lines operating in network control mode. Normal message traffic on lines not undergoing testing can continue as usual.

Address Trace Facility

Address trace is a service aid that records the contents of selected areas of controller storage and controller external registers when an interrupt occurs. Certain types of interrupts, or all interrupts, are designated to control the trace. The NCP records the trace data in a trace table within controller storage. When the desired data has been recorded, the contents of the trace table can be displayed on the 3705 control panel. The contents of controller storage can be transferred to the host processor by the dump program and the trace table examined in the listing of the dump.

The TRACE operand of the BUILD macro specifies whether the address trace option is to be included in the NCP and specifies the size of the trace table. The address trace function is performed only on lines operating in network control mode.

Line Trace Facility

The line trace facility is a service aid that permits detailed analysis of the operation of any communication line operating in network control mode. (All SDLC links operate only in network control mode.) This facility records operating parameters of a line each time a level 2 interrupt occurs for that line. The trace information is placed in buffers obtained as required from the buffer pool and is transmitted to the host processor. The host processor should accumulate these line trace records in a data set (file) to be printed for analysis.

A line trace can be initiated or ended at any time by a request from the host processor as long as the line is in network control mode.

The line trace activity does not interfere with normal operation of the communication line. Performance may diminish somewhat because of the additional processing needed each time a character-service or buffer-service interrupt occurs for the line being traced. The amount of decrease in performance depends upon how heavily the communications controller is currently loaded. The line trace facility has no effect on performance except when a line is actually being traced.

The NCP will accept and fulfill requests for tracing up to eight lines concurrently, as specified by the LTRACE operand of the BUILD macro. Any Activate Line Trace requests received when the specified limit has been reached are rejected. Each leg of a duplex line may be traced, but the traces are independent of each other.

The line trace facility for SDLC links is always present in the network control program.

Channel Adapter Trace Facility

Channel adapter trace is an optional service aid that stores in a trace table certain information about the channel adapters. Any combination of up to four channels can be traced.

An entry is placed in the table for each level 3 interrupt. After the last entry in the table is used, succeeding entries overlay previous entries, beginning with the first.

The CATRACE operand of the BUILD macro specifies whether the channel adapter trace option is to be included in the NCP, and specifies the size of the trace table. This trace can be activated or deactivated from the communications controller control panel. The channel adapter trace facility does not interfere with normal operation of the controller. Performance may diminish somewhat because of the additional processing needed. The amount of decrease in performance depends on how heavily the controller is currently loaded.

Abnormal End (ABEND) Facility

Programming errors detected during execution of nonsupervisory portions of the NCP (level 5 dispatched) cause abnormal termination of program execution. Examination of abend codes in a storage dump can help in locating the error.

The optional abend service aid extends detection of programming errors to the NCP *supervisor* (levels 1-4), thus causing termination of the program before a supervisor error can be propagated into nonsupervisory portions of the program. The abend code appearing in the storage dump, therefore, gives a better indication of the location of a supervisor error, if one should occur, than an error code reflecting a resultant error in the nonsupervisory portion would give. Including the abend option (by the ABEND operand of the BUILD macro) is recommended when you first begin using a network control program. Later, as experience demonstrates that your network operates routinely without abnormal termination, the abend option may be deleted from the program.

Panel Tests

Some testing of communication lines can be done from the control panel of the communications controller. These tests (called panel-initiated line tests or panel tests) are explained in the *Control Panel Guide* (see Preface). Using the test routines, the operator at the controller can perform many of the communication functions (such as polling, addressing, and data transfer) normally executed by the controller and its control program upon command from the access method.

The panel test function is always present in the NCP.

Program Generation Options and Data Sets (Files)

All of the options described thus far in this chapter have related to the operational characteristics of the network. Described in this section are several options affecting the generation procedure and the program data sets (files) used in the procedure.

Program Generation Options

Program generation options pertain to the type of communications controller (local or remote), complete versus partial generation procedure, and several assembly and link-editing options. All program generation options are specified in the BUILD macro.

Type of Program to be Generated

As explained in Chapter 1, when defining the control program for a local communications controller, you must decide whether the program is to perform network control functions only or both network control and PEP functions. Specify the choice in the TYPGEN operand of the BUILD macro. Also specify whether the program is to be a local NCP (TYPGEN=NCP) or a remote NCP (TYPGEN=NCP-R).

Model of Controller

The same NCP can be executed in an IBM 3705-I or 3705-II Communications Controller. However, differences in the addressing requirements between controller models require that you specify type of controller in the MODEL operand. Changing the value in this operand is the only modification required to allow a network control program originally defined for one type of controller to be executed in the other type, *provided* that the network and controller configurations are identical.

Partial Generation

Assembling and link-editing the modules making up an NCP requires substantial processing time. Once you have generated a complete network control program, however, modifications resulting from changes in network configuration and procedural options can be done in significantly less time via *partial generation.*

In a partial generation, only selected modules are reassembled. They are then link-edited with the modules that require no changes to produce the modified program.

To perform a partial generation, code PARTIAL=YES and specify the names of the modules requiring reassembly in the CONDASM operand. The modules requiring reassembly for each of various changes in the program functions are listed in Appendix B.

In a DOS/VS system, it is important that you retain all of the stage 1 and stage 2 assembly listings and the object library produced by the complete generation procedure. Ths DOS/VS object library is an essential part of the partial generation.

Other Options

Other program generation options and their operands of the BUILD macro are:

- Whether stage 2 of the generation consists of a single- or multiple-step job or a separate job for each step, and whether a job card is required

(JOBCARD).

- If a qualifier is added to conditional assemblies so that NCP generations with different names can run concurrently (OBJQUAL).
- Whether the JCL normally produced by the generation procedure is replaced by cataloged procedures. Procedures may be specified for assembly steps, post-assembly steps, and the linkage editor (OUTPUT).
- The region size for stage 2 linkage-editor job steps (LESIZE).
- Whether the generation procedure is to produce cross-reference listings for stage 2 assemblies (ASMXREF).
- Whether the macro expansion or NCP tables and NCP conditional assemblies are printed (PRTGEN).
- The value of the TIME parameter in stage 2 assembly EXEC statements (TIME).
- The type or class of devices used for utility data sets during stage 2 (UNIT).

Note: Some of these options apply only to OS/VS systems. See the macro descriptions in Chapter 5 for individual requirements.

Data Sets (Files) Used in the Generation Procedure

The names of various program data sets to be used in the generation procedure when generating under OS/VS are specified by the LOADLIB, MACLIB, OBJLIB, QUALIFY, UT1, UT2, UT3, and USERLIB operands of the BUILD macro. The NEWNAME operand specifies the name to be given to the generated NCP load module.

The only file name required when generating under DOS/VS is NEWNAME.

Chapter 3. Functions for BSC and/or Start-Stop Resources

Described in this chapter are the many aspects of a network that you must identify to the network control program to support BSC and/or start stop stations.

You should read this chapter if you wish to define a program capable of operating BSC and/or start-stop lines in network control mode only or in both network control mode and emulation mode. (Upon request from the access method, the NCP can change the operation of a line from network control mode to emulation mode, and vice versa, if start-stop or BSC stations are attached to the line.)

Many of the functions of the network covered in this chapter apply only to operation in network control mode. Others apply to both network control and emulation modes. In most cases, the functions are specified in exactly the same way for both modes. For example, the type of line control—start-stop or BSC—is specified in the same operand (LNCTL) of the same macro (GROUP) whether the lines in the group are to be operated in network control mode or in emulation mode, or in both modes alternately.

If the program you are defining is to operate all lines in network control mode, ignore any references to emulation mode in this chapter.

If the program you are defining is to operate some lines in network control mode and others in emulation mode or the same lines in both modes alternately, observe the references to emulation mode as well as those to network control mode.

This chapter explains the characteristics of the network with respect to:

- The stations and lines of the network
- The communications controller hardware configuration
- Data transfer between the communications controller and the host processor
- Procedural options governing message traffic between the controller and the network in network control mode
- Procedural options governing message traffic between the controller and the network in emulation mode
- Optional message processing within the controller (network control mode only)
- Diagnostic and service aid facilities
- Program generation options and data sets (files)

The description of each characteristic and option is not exhaustive; it is intended to provide sufficient information to enable you to select the appropriate parameters when coding the program generation macro instructions given in Chapter 5.

For many characteristics, especially those relating to the equipment configuration, the decisions about what to code in the macro instructions have been made by the *system designer*. (This is the individual who determines the data communication equipment, network configuration, and communication services that constitute your communication system.) You need only determine what these characteristics are and code the appropriate macros and operands accordingly.

Other characteristics relate to resources, such as the size of the buffers in the buffer pool, or to procedural options, such as the number of buffers of message data to be accumulated from a start-stop or BSC station before forwarding them to the host processor. Such characteristics, which affect the message-handling capacity and throughput of the communication system, require careful consideration before specifying the corresponding parameters in the program generation macro instructions.

Once you are familiar with those characteristics that apply to your equipment configuration and applications, you are ready to code the program generation macro instructions that define the NCP. At that point you should go on to Chapter 5, which provides detailed information on coding the macro instructions.

Network Characteristics for BSC and Start-Stop Resources

This section applies only to start-stop and BSC resources. (See Chapter 2 for information on SDLC resources.) The descriptions of the network characteristics give the names of the applicable macro instructions and operands. Unless otherwise indicated, the macros and operands named apply to both network control mode and emulation mode. For your convenience, the operand mentioned can often be specified in a macro instruction different from the one named, as explained in Chapter 5. The description of the operand always appears under the lowest level macro.

Station Characteristics

In this book, *station* refers to any equipment, regardless of type, that can transmit data onto, or receive data from, a communication line connected to the communications controller. For line operations in network control mode, this definition includes (1) computers; (2) communications control units such as the IBM 2701, 2703, and 3704; (3) other 3705 controllers; (4) the input/output units (keyboards, printers, tape and card readers, punches, and display screens) usually referred to as *terminals;* and (5) control units (such as IBM 3270 cluster control units) with input/output units attached.

Each start-stop or BSC station that communicates with the 3705 Communications Controller in network control mode is represented by a TERMINAL macro instruction. (Stations with which the controller communicates only in emulation mode are not represented by TERMINAL or CLUSTER macros.)

Type of Station

Type of station means the numerical designation by which the station is known, or an abbreviation thereof (for example, 1050, 2780, SYS3 [System/3]). Appendix A lists the types of stations the communications controller can communicate with in network control mode and in emulation mode. For network control mode, type of station is specified in the TERM operand of the TERMINAL macro or, for certain types of stations, in the CUTYPE operand of the CLUSTER macro. For operation in emulation mode, type of station is specified in the TERM or CUTYPE operand of the LINE macro.

Terminal Features

For some types of terminals and control units, the presence or absence of certain features with which the terminal or control unit may be equipped must be known to the NCP. The features that must be specified differ for network control mode and emulation mode. If the communications controller is to communicate with a terminal in network control mode, specify the appropriate features from the list of features for network control mode, and similarly, for emulation mode. If the controller is to communicate with the terminals in both modes, specify the appropriate features from both lists.

Features for Operation in Network Control Mode

The presence of some of the features below is specified in the FEATURE operand of the TERMINAL macro. For these, the suboperand that specifies the presence or absence of the feature appears in parentheses after the description. Other features are specified in the macros and operands indicated.

Transmit Interrupt (IBM 1050, 2741, 3767 in 2741 mode): If the terminal has this feature, the communications controller can interrupt a transmission from the terminal by sending the break signal. (BREAK or NOBREAK)

Buffered Receive (IBM 2740 Model 2, 2770, 2972 Models 8 and 11, 3270, 3780): If the terminal has this function, the NCP allows a time interval to elapse betweeen successive transmissions to the terminal. During the interval, the NCP can communicate with other terminals on the same multipoint line. The presence of the feature and the interval are specified by the BFRDLAY operand. See also the discussion of buffered terminals under *BSC and Start-Stop Operation* in this chapter.

Conversational Mode (IBM 1050, 2740 Models 1 and 2 with Record Checking feature, 2770 with Conversational Mode feature, and all IBM BSC stations except 2715 and 2780): A station equipped with this feature can receive messgae data, instead of the usual positive acknowledgment, in response to a message block sent by the station. The message block the station receives in reply serves as the positive acknowledgment. Exchanging message blocks in this way improves line utilization because the time normally spent in re-addressing (re-selecting) the station is eliminated. Conversational mode is specified in the CONV operand.

If you specify the conversational mode feature, the NCP automatically replies to a message block from the station with the next block it currently holds for sending to that station. If the program has no data to send, it replies with a positive acknowledgment.

Accelerated Carrier Return (IBM 1050): If your network includes IBM 1050 terminals having the accelerated carrier return feature, you should specify this in the FEATURE operand of the TERMINAL macro for each terminal so equipped. The communications controller then sends a fewer number of idle characters than if the terminal did not have the feature, thus saving a small amount of transmission time whenever the new line (NL) character occurs in message data. (ACR or NOACR)

Record Checking, Station Control, Transmit Control (IBM 2740): The command sequence the NCP uses to communicate with the IBM 2740 differs for each of these features, or combinations thereof. (CHECK or NOCHECK, SCTL or NOSCTL, XCTL or NOXCTL)

Interrupt (IBM 2741, 3767 in 2741 mode), Receive Interrupt (IBM 1050): If the terminal has this feature, it can interrupt the NCP while the program is sending to the terminal. (ATTN or NOATTN)

Features for Operation in Emulation Mode

The presence of the following features is specified in the FEATURE operand of each LINE macro.

Record Checking: Some start-stop stations have the record checking capability (also called longitudinal redundancy checking), and other do not. For each line operated in emulation mode, you must specify whether the terminals on that line have the record checking capability. If the terminal is an IBM 1050, 1060, 2260, 2265, 2845, 2848, or System/7--specify LRC in the FEATURE operand. Also specify LRC for an IBM 2740 (Model 1 or 2) if it is equipped with the Record Checking feature. For other types of start-stop terminals, specify NOLRC.

Downshifting on Space Characters: Some AT & T 83B3, Western Union 115A, and World Trade teletypewriter (teleprinter) terminals, upon sending or receiving a space character, automatically downshift so that subsequent message text is in lowercase, or downshifted, mode. Automatic downshifting avoids the need to send a LTRS character to effect downshifting. Specify SPACE in the FEATURE operand if the terminals have this feature.

Immediate End: Upon receiving an end-of-transmission character from a start-stop terminal (in emulation mode), the NCP normally delays ending the receive operation for several character times (the time required for the transmission of one character) until the line becomes electrically "quiet." The absence of further characters following the EOT verifies that the EOT character is valid and not a data character converted by line noise to a false EOT. Checking for false EOTs in this manner is appropriate for many applications. On some applications, however, the terminal continues to send data immediately after sending the EOT (as when the terminal is transmitting from a paper tape with data interspersed with EOTs). If the end of the receive operation were in this case delayed, the program would not recognize the EOT because of the immediately following data characters. In this instance, it is necessary to specify IMEND in the FEATURE operand; this causes the program to end the receive operation immediately upon detecting the EOT, without waiting to detect the presence or absence of any following characters.

Dual Code: Either of two transmission codes (EBCDIC and USASCII) can be transmitted on a binary synchronous communication line attached to an IBM 2701 Data Adapter Unit equipped with the Dual Code feature for that line. The code used is changed from one to the other by command from the access method. The same function can be performed when the IBM 3704 or 3705 is installed in place of the 2701. Specify DUALCODE in the FEATURE operand if the Dual Code feature was used for that line when the line was attached to the 2701. Otherwise, specify NODUALCD or omit the parameter. (In addition to EBCDIC and USASCII, transparent USASCII is supported as a dual code option for a line serviced by a type 3 scanner.)

End-of-Transmission Character

You may specify that the EOB character is to signify end of transmission for messages from terminals equipped to send EOB EOT ending sequences. If specified, the terminal operator can signal the end of each transmission by pressing only the EOB key rather than both the EOB and EOT keys. (In this case, each transmission from the terminal consists of a single block.) This option, which applies only to operation in network control mode, is specified in the ENDTRNS operand of the TERMINAL or COMP macro.

Printer Line Length and Carriage Return Rate

The network control program recognizes each carriage return (CR) character and horizontal tab (HT) character in text being sent to a nonbuffered start-stop terminal in network control mode. Upon detecting either character, the program sends a sequence of idle characters immediately following the CR or HT character. A sufficient number of idle characters delays further printing on a terminal printer until the movable printing mechanism (carriage, or carrier) has had sufficient time to reach the next printing position. The next position is one of the tab locations, in the case of the HT character, or the left margin on the next printing line, in the case of the CR character.

The appropriate number of idle characters to send is determined from (1) the maximum length of the line of print—that is, the number of character positions between the left margin and the rightmost printing position; and (2) the rate at which the printing mechanism moves, expressed as the number of character positions traversed by the mechanism for each idle character. From these two values, which you specify for each start-stop line in the network by the LINESIZ and CRRATE operands of the LINE macro, the generation procedure calculates the required number of idle characters. The values should be carefully selected to suit the type of terminal connected to the line. Too few idle characters sent following each CR or HT character will allow insufficient time for the mechanism to reach the next printing position, resulting in random printing of text characters on the print line. Too many idle characters, on the other hand, will cause excessive delay in the resumption of printing, resulting in wasted time on the communication line.

Thus, for example, if the terminals attached to a given line have a printer line length of 60, and a carrier return rate of 10 printing positions per idle character, you would specify the values 60 and 10, respectively , in the LINESIZ and CRRATE operands. If you do not specify the line size or return rate, the network control program uses the default values given under the description of these operands.

If the printers attached to the line have differing line lengths, specify in LINESIZ the maximum length used by any terminal on the line.

Communication Line Characteristics

Nonswitched Multipoint Line

Multipoint line control is typically used for a nonswitched line with several stations attached. The controller contacts a specific station by sending a polling character or addressing character assigned to and recognized only by that station. The one station recognizing that character responds appropriately; the other stations ignore the character.

Multipoint line control must also be used for a line with only one station attached, if that station must be polled or addressed by the controller before sending or receiving data. A multipoint line is therefore one on which a multipoint discipline must be used, regardless of the number of stations—several or only one—the controller communicates with over that line.

To specify a line as requiring a multipoint discipline, specify POLLED=YES in the LINE macro if the line is to be operated in network control mode. If it is to be operated only in emulation mode, you need not specify it as a multipoint line. (The access method is responsible for properly controlling multipoint lines in emulation mode.)

The network control program requires a service order table for each nonswitched start-stop or BSC communication line that requires a multipoint discipline and that is to be operated in network control mode. This table contains one or more entries representing each station and each component of a station with which the program can establish a session upon request from the access method. (Sessions are descibed later in this chapter under *BSC and Start-Stop Operation.)* The program attempts to establish sessions with stations and components in the same sequence as their respective entries appear in the service order table.

Directly following a LINE macro for a start-stop or BSC multipoint line (or a nonswitched point-to-point line that requires a multipoint discipline), code a SERVICE macro that defines the service order table to be used for that line.

Nonswitched Point-to-Point Line

To designate the communications controller as the secondary station on a BSC point-to-point line operated in network control mode, code YIELD=YES in the LINE macro (or omit the operand). To designate it as the primary station, code YIELD=NO.

Whichever choice you make, the staion at the other end of the line must be prepared to assume the complementary role (that is, primary or secondary).

Except for the YIELD operand, you need code no other operands to designate the type of line as nonswitched point-to-point. The line is assumed to be of this type unless you explicitly code operands that specify another type. A nonswitched point-to-point line is not identified as such if it is to be operated only in emulation mode.

Switched Point-to-Point Line

For each switched point-to-point line connection, or "port," over which the communications controller may call stations or receive calls from stations, you must code a LINE macro. In the GROUP macro that precedes the LINE macros for the switched line, code DIAL=YES. (The GROUP macro, rather than the LINE macro, indicates that the lines are switched lines. If any lines within the group are switched, all must be.) This operand is applicable whether the line is to be operated in network control mode or in emulation mode, or both.

A switched line port can receive calls from either BSC stations or start-stop terminals, but not from both. Any type of BSC station can call the controller over a line designated for use by BSC stations (provided that all use the same transmission code), as the line control discipline for all such stations is similar. On the other hand, a line designated for use by start-stop terminals can receive calls from only a single type of terminal except when multiple-terminal-access operation is specified for that line. (See *Multiple Terminal Access Facility* later in this chapter.)

If the line is to be operated in network control mode, you may designate in the CALL operand of the LINE macro, whether the line is to be used for receiving calls from stations (CALL=IN), for making calls to stations (CALL=OUT), or both (CALL=INOUT). Each line used for outgoing calls must be included in a dial set by means of the DIALSET macro. Dial sets are explained in *Switched Network Operation* under *BSC and Start-Stop Operation.*

Half-Duplex versus Duplex Lines

Half-duplex data transfer is always used for any start-stop or BSC station with which the controller can communicate.

Transmission Codes

The transmission code to be used for communicating with each station must be identified to the NCP. The program translates outgoing data characters from its internal processing code, EBCDIC, to the specified transmission code, and vice versa, for incoming data characters. (The transmission code used on a multipoint line must be the same for all stations attached to that line.)

Specify the required transmission code in the CODE operand of the LINE macro representing the communication line. (For BSC stations, the code you specify in the LINE macro also informs the network control program which line control scheme is to be used; the transmission code and line control scheme are related.)

Line and Subchannel Addresses

Each communication line attached to the communications controller is identified to the NCP by a line address. Whether the line is to be operated in network control mode or in emulation mode, specify this address in the ADDRESS operand of the corresponding LINE macro. If the line is to be operated in emulation mode, also specify the host processor subchannel address(es) corresponding to the line address. (Each line operated in emulation mode requires its own subchannel address(es) in the host processor. The multi-subchannel line access [MSLA] facility of the network control program with PEP extension permits two or more emulation subchannels to communicate alternately with the same communication line. The address of each subchannel to be associated with a line must be specified in the ADDRESS operand.)

Automatic Calling Units

Any switched network line that the NCP uses for calling stations may be equipped with an automatic calling unit (ACU). If a line is equipped with an ACU (whether the line is to be operated in network control mode or in emulation mode), specify the ACU address in the AUTO operand of the corresponding LINE macro.

The (ACU) address is determined by the physical location of the ACU hardware on the line interface base (LIB). A 1E line set is always required for the ACU.

Determine the ACU addresses from the *Teleprocessing Installation Record* or the system designer.

Ring Indicator Mode (not applicable in U.S. and Canada)

Certain European modems may require that their ring indicator signal line be energized (signifying that the modem is being called by a station) before the communications controller indicates its readiness to receive by energizing the modem's data terminal ready signal line. (These and other signal lines constitute the interface between the communications controller and the modem.) If this requirement applies to a modem in your network, code RING=YES in the LINE macro for the communication line attached to the modem, whether that line is to be operated in network control mode or in emulation mode. Most modems do not have this requirement, and for these you would specify RING=NO in (or omit the RING operand from) the LINE macro. Specifying RING=YES for a modem that does not have this requirement can result in unnecessary delay in establishing the connection.

Communication between Controller and Host Processor

For the NCP to operate lines in network control mode, information on both the buffers within the access method and buffers within the network control program must be specified. This information is specified in the HOST macro. Each access method the NCP concurrently communicates with in network control mode is represented by a separate HOST macro. Buffers for network control operation are allocated from a single pool of buffers used for all line and channel data transfers in network control mode.

Associated with each communication line specified as operable in emulation mode and serviced by a type 3 scanner is a pair of buffers contained within the control blocks related to the line. The size of each buffer in the pair is user specified as 4, 8, 16, 32, 64, 96, 128, 160, 192, or 224 bytes.

For a given amount of data passing over the line, use of larger buffers affords more protection against possible overruns than does use of smaller buffers. (Overruns can result from temporary slowdowns of channel operation or from momentary peaks in data traffic through the network.) Use of larger buffers also results in less interrupt-processing overhead for line operations and—up to 32 bytes—less interrupt-processing overhead for channel operations. The amount of data transferred across the channel is equal to *n*, up to 32 bytes. For values of *n* exceeding 32, the amount of data transferred over the channel is 32 bytes.

The size of the emulation mode buffers for a line seviced by a type 3 scanner is specified in the BUFSIZE operand of the LINE macro. If you do not specify a size, 32-byte buffers are provided for lines operating at speeds of 9600 bps or less, and 64-byte buffers are provided for lines operating at higher speeds (as specified in the SPEED operand of the LINE macros).

Transfer of data in emulation mode between the host processor and the line occurs in a manner equivalent to that provided by the IBM 2701, 2702, or 2703 being emulated. In the CU operand of the LINE macro, specify the type of transmission control unit to be emulated for that line—2701, 2702, or 2703.

Procedural Options for Operation in Network Control Mode

A number of procedural options characterize the operation of lines in network control mode. These options include (1) the manner in which the program starts up and shuts down the network, (2) the amount of data to be transferred at one time between stations and the controller, (3) the amount of data to be accumulated from a station before passing it to the access method, and (4) the number of sessions to be conducted concurrently on a start-stop or BSC multipoint line. (NCP sessions are described later in this chapter under *BSC and Start-Stop Operation.)* By careful selection of these options, you can customize a network control program to best meet the requirements of your data communication applications.

Logical Connection Stations

Each start-stop and BSC station connected to a communication line operated in network control mode is generally represented within the network control program by its own set of control blocks, each defined by a separate TERMINAL macro. However, for a switched line over which the controller receives calls from stations, known as a *call-in* switched line. The program maintains one set of control blocks for all stations that call in over that line, regardless of the number of terminals that may do so. The control blocks represent a dummy station called a *call-in logical-connection* station. The control blocks represent whichever station has called over that line at any given moment. A single TERMINAL macro in which CTERM=YES is specified represents a call-in logical-connection station.

BSC and Start-Stop Operation

This section describes those procedural options that apply in general to communication lines operated in network control mode, and not uniquely to binary synchronous stations or start-stop (asynchronous) terminals. Options specific to one or the other, but not both, of these categories of stations appear in subsequent sections. (Operation of SDLC links is covered in Chapter 2.)

Sessions

The ability of the NCP to conduct multiple sessions on the same start-stop or BSC multipoint line in network control mode depends on the fact that data transfer does not occur continuously for the duration of the session. For example, in interactive applications such as inquiry-response, the elapsed time between receiving a response from the host processor and entering the next inquiry typically exceeds the time required for transmission of the inquiry and response. The terminal operator typically needs 5 seconds or more think time after seeing the response to prepare his next inquiry, whereas transmission time for the inquiry and response together often consumes but 1 or 2 seconds. The interval during which the terminal is not using the line can profitably be used to service other terminals on the same line.

Buffered terminals are another example. With such terminals, the operator keys his message data into a buffer instead of directly on the communication line; the line is not needed for transmission until the entire message or block of a message has been accumulated in the buffer. The IBM 2740 Model 2 is an example of such a terminal. A terminal may also receive data from the line into a buffer rather than sending it directly to the printer or other output device. The 2740 Model 2 with the buffered receive feature works in this manner.

While its operator is keying message data into the buffer, on the one hand, and while the terminal is printing the contents of the buffer, on the other, the terminal has no need of the communication line. Since data transmission to and from a buffered terminal usually is much faster than the data entry or printing operations, the terminal requires the line for a relatively small proportion of the session period. Again, the line can be used for servicing other terminals in the interim.

Interleaving transmissions with several stations maximizes the utilization of a multipoint communication line, thus permitting more stations to share the same line than if only one session were possible. A direct result is reduced cost of communication lines and line attachment hardware within the communications controller.

The number of concurrent sessions to be conducted on a line depends on several factors. Among these are (1) the relative amount of time when a terminal is in use that it does not need the communication line and (2) the permissible delay between readiness to use the terminal and availability of the communication line.

The number of concurrent sessions is called the *session limit* and is specified in the SESSION operand of the LINE macro. The NCP not only limits the number of sessions to this value, but it also tries to maintain that many sessions in order to get maximum line utiliization. The number of sessions in progress will be less than the session limit whenever the NCP has requests for fewer devices than allowed by the session limit or when fewer devices are ready to communicate with the host processor.

In the case of clustered BSC stations such as the IBM 3270, the session limit is not applicable if general polling is used to solicit input from the attached terminals. The NCP cannot control the number of terminals that may respond to a general poll of the clustered station, and a separate session is established with each terminal that responds. However, for 3270 terminals, the session limit should equal the sum of the number of cluster controllers and the number of terminals attached to the line.

Service Order Table

The sequence in which the NCP attempts to establish sessions on a multipoint line operated in network control mode is determined by a *service order table* associated with the line. This table is defined by the SERVICE macro you code directly following the LINE macro representing the nonswitched multipoint line. Each station with which the host processor may request a session must be represented by at least one entry in the table. If the station consists of a control unit with one or more individually pollable or addressable components (for example, the IBM 3270 and 1050 terminals), each terminal or component must be represented in the service order table whether it is polled, addressed, or both. In addition, a cluster control unit must be represented in the table if general polling is to be used.

The same device may be represented by more than one entry in the service order table. Multiple entries are of value if you wish the program to attempt to begin sessions more frequently with some devices than with others.

The contents of the service order table for a BSC or start-stop line can be changed during program execution by a control request from the host processor. Control requests can cause the program to add or delete devices or change the order or frequency with which the devices are serviced. Thus the NCP can be kept responsive to application requirements if these should change from time to time.

Logical Connections

A session is said to be *active* when the NCP is communicating with, or is ready to communicate with, the associated device. The rest of the time the session is *suspended* (or *inactive*). The active and inactive portions of a session may be represented thus:

```
                                                             (end)
 ____             _________                            ____
      ------------         ---------------------
 time-->
```

The solid lines represent the session in its active state; the broken lines represent the suspended state.

Three concurrent sessions may be represented thus:

```
1  ____               _____                   _____
        --------------       -----------------      ---------

2       _____                _____                   _____
             ---------------      -----------------       ----

3            _____                _____                   ____
                  ---------------      -----------------
time-->
```

In most applications it is necessary to limit the amount of time a session is permitted to be active to prevent a device, once in session, from monopolizing the communication line to the exclusion of other devices. The period during which a session is active is called a *logical connection.* A logical connection exists when the NCP and the device in session are engaged in data transfer from one to the other. The length of a logical connection is limited by the *transmission limit,* which is the maximum number of transmissions that may be transferred in either direction between the NCP and the device during the logical connection. When the limit is reached, the NCP breaks the logical connection, thus suspending the session for the moment. The program is then free to service the next session or attempt to establish a new session if the session limit has not been reached.

Although the intent of the transmission limit is to restrict the time a session may be active, it does not represent a fixed amount of time or number of data characters. The actual length of a logical connection, in time, is determined by the number of transmissions, the number of data characters in each, and the speed of transmission.

The transmission limit will not always be reached, as the device in session may run out of data to send or become unable to receive or the NCP may run out of data transfer requests for the device before the transmission limit is reached.

The transmission limit can be individually specified for each device with which the host processor will establish sessions. This capability allows you to base the limit on the application requirements for the device. The limit is specified in the

XMITLIM operand of the TERMINAL or COMP macro representing the device. If the application warrants, you may specify that unlimited transmissions are to be allowed during an active session. This choice should be carefully considered, however, as it can result in monopolization of the communication line by a single device for lengthy periods, thus negating the benefits of having multiple sessions.

Once a session has been established, the network control program re-polls the device for each subsequent transmission to be solicited from the device. The logical connection is maintained during the polling operation. Unless you specify otherwise, the program polls a device only once to solicit the next transmission. If ready to transmit, the device will respond positively to this poll.

You may have the program repeat the polling operation one or more times if you wish to allow the device more time in which to respond with its next transmission. Specify the required number of polls in the POLIMIT operand of the LINE macro for the line. The network control program will then accept an equivalent number of negative responses to polling before breaking the logical connection. The value specified in the POLIMIT operand is accordingly called the *negative response limit*.

Allowing the program to poll the device more than once is often appropriate for interactive applications in which the terminal operator needs several seconds of think time in which to prepare his next transmission. Consider, however, that (1) no message data is communicated during the polling operation and (2) no other session can be serviced until the negative polling limit is reached, even if the terminal has no more data to transmit.

Once the polling limit is reached, the network control program can proceed to the next session. Unless you specify otherwise, the program breaks the logical connection and cancels the read request that caused polling. The program then goes on to service the next suspended session (or to resume service seeking).

There are two alternatives to this course of action, however. You may specify that the NCP (1) break the logical connection or (2) maintain the connection and notify the host processor that the negative polling limit was reached. In the latter case, the program also flags all subsequent requests that may be on the queue for the device to prevent them from being executed. It is then the responsibility of the host processor either to signal the NCP to go ahead with execution of the remaining requests or to cancel those requests and send other ones in their place. These options, like the negative polling limit, are specified in the POLIMIT operand of the LINE macro.

Sessions may be suspended in one other way. Most types of input/output errors that occur during an active session cause suspension of that session.

Session Servicing and Service Seeking

Establishing a new session on a start-stop or BSC line in network control mode is called *service seeking*. The NCP performs service seeking whenever the number of existing (suspended) sessions is less than the session limit established for the line. Thus, the program always tries to conduct the specified number of sessions.

Servicing existing sessions is called *session servicing*. Servicing a session consists of establishing a logical connection, then sending or receiving data. The logical connection ends when (1) a request from the host processor ends the session, (2) the transmission limit is reached, (3) the negative polling limit is reached, or (4) an input/output error occurs.

Session servicing and service seeking alternate in a sequence of operations called a *service cycle.*

A service cycle consists of both session servicing and service seeking whenever there is at least one session but the total number of sessions is less than the session limit. If no sessions exist at the moment, session servicing does not take place—the service cycle consists only of service seeking. Conversely, if the number of existing sessions equals the session limit, there is no need for service seeking—the service cycle consists exclusively of session servicing.

When session servicing, the NCP makes a single logical connection for each existing session, in the same sequence as the devices appear in the service order table.

When service seeking, the program attempts to establish a new session with one or more of the devices for which no session currently exists and for which the network control program currently contains a request to begin a session

As in session servicing, the sequence in which the network control program attempts to establish new sessions corresponds to the order in which the device entries appear in the sevice order table. Each service-seeking operation begins with the entry following the last entry handled in the previous service-seeking operation. The service order table is a "wraparound" table; that is, service seeking does not stop at the end of the table but resumes automatically with the first entry in the table.

The maximum number of devices with which the program attempts to establish a session during each service-seeking operation is called the *service limit.* If the device were always ready to engage in a session, only one service order table entry would require checking in each service-seeking operation because a new session would always be established with the device. This is not normally the case, however, and the network control program usually must make an attempt for each of several devices before successfully establishing a session.

Unless you specify otherwise, the network control program uses as the service limit one-half of the devices represented in the service order table.

You may instead designate in the SERVLIM operand of the LINE macro a specific maximum number of inactive devices for which the program is to attempt service seeking.

Several factors influence the distribution of the service cycle between session servicing and service seeking.

One major factor is the amount of data transferred between the communications controller and devices during logical connections. The longer the transmissions, the more time is spent in session servicing.

A second major factor is the value selected for the service limit. In periods when the network control program has data communication requests for few of the devices represented in the service order table, a large service limit can result in much service-seeking activity because the program will have to make numerous attempts before establishing a new session. On the other hand, in periods when the program does have data communication requests for most of the devices, it will be able to establish sessions much sooner. The value of the service limit would have less influence in this case since most often service seeking would end with establishment of a new session before the service limit was approached.

Another factor affecting the relative time spent in session servicing and service seeking is the *service priority.* This factor is effective only when the session limit exceeds the number of existing sessions by more than one. Unless you specify differently, the network control program returns to servicing existing sessions after one service-seeking operation—that is, after one attempt, successful or not, to establish a new session. This is referred to as giving priority to old sessions.

The alternative is to have the program perform the service-seeking operation more than once, the total number of operations equalling the difference between the number of existing sessions and the session limit. This is called giving priority to new sessions and is specified by coding SERVPRI=NEW in the LINE macro.

Assume, for example, that two sessions currently exist and that the session limit is 5. If priority is given to old sessions (SERVPRI=OLD), the NCP will perform a single service-seeking operation after servicing the two existing sessions. Then it will return to servicing the two sessions once again. But if new sessions have priority (SERVPRI=NEW), the program performs three service-seeking operations in succession, the value of 3 being the difference between the session limit and the number of existing sessions.

It can be seen that the larger the difference between the number of old sessions and the session limit, the more your choice of service priority affects the relative time spent in session servicing and service seeking. When the session limit exceeds the number of existing sessions by only one, the value in SERVPRI has no effect since in either case only one service-seeking operation will be performed.

A final factor influencing the distribution of the service cycle between session servicing and service seeking is the negative polling limit specified in the POLIMIT operand of the LINE macro. The higher the limit and the more often devices fail to respond promptly to polling once a logical connection has been established, the more time will be spent in session servicing.

During periods of low communication-line activity, there may be intervals when no sessions currently exist on a BSC or start-stop line operating in network control mode. The service cycle accordingly consists only of service seeking (provided the NCP currently contains at least one request to begin a session). Nonproductive polling and the resultant processing overhead can be minimized by specifying a *service-seeking pause* of from several seconds to many minutes. This pause, which you specify in the PAUSE operand of the LINE macro, is in effect at the end of each service-seeking operation. When at least one new session is established, the pause is inoperative, since to observe it would delay session servicing as well as minimize nonproductive polling. Since it is not in effect when the service cycle consists of both session servicing and service seeking, the service-seeking pause is not a factor in the distribution of the service cycle between these two activities.

During periods when no sessions currently exist and the program currently has no requests to begin a session, no service cycle exists; the line is idle. The service cycle resumes when the program receives from the host processor a new request to establish a session.

As indicated by the foregoing discussion, numerous factors influence the handling of sessions on a multipoint communication line. When specifying operations over start-stop and BSC lines, consider all of these factors in terms of their effect on your data communication applications.

Sessions on Point-to-Point Lines

The concept of a session as a defined sequence of data interchanges between host processor and device is valid for point-to-point lines as for multipoint lines.

However, the advantage of multiple concurrent sessions is not available for a point-to-point line; since the network control program communicates with only one station over the line, only one session can exist.

As mentioned earlier, the access method requests that the communications controller establish sessions on lines in network control mode without regard for the way in which the network control program will conduct them. The type of communication line—point-to-point or multipoint—over which the network control program will conduct the session, therefore, does not concern the host processor.

Because only one session at a time can exist, the session limit, service limit, service priority, service-seeking pause, transmission limit, and polling limit parameters are not applicable for a point-to-point line.

Switched Network Operation

The switched network facilities of the network control program are designed to permit a high degree of utilization of the switched network connections, or ports, available to the communications controller. Maximum utilization of these ports reduces the number required to support a given number of terminals, with attendant savings in line and controller hardware costs.

The network control program's switched network facilities (applicable only to communication in network control mode with BSC and start-stop stations) accommodate both *call-out* operation, in which the controller calls remote stations upon request from the host processor, and *call-in* operation, in which the controller answers calls from stations. Switched network connections (ports) may be designated for use in fulfilling call-out requests, call-in requests, or both.

Call-Out Operation

The network control program maximizes the utilization of call-out lines by dynamically allocating them to handle the call-out requests with a minimum of delay. To enable the NCP to fulfill call-out requests, you must (1) define a dial set consisting of switched lines having similar characteristics and each line designated as a call-out line and (2) specify the stations with which the NCP can communicate using lines in the dial set. (See "The Multiple Terminal Access Facility" later in this chapter for another method of increasing switched-line utilization.) Because the program allocates the lines dynamically, any line in the set may be used to communicate with any of the designated stations. You might, for example, establish a dial set of three lines to accommodate call-out requests for 20 stations. The program fulfills call-out requests in the order they are received from the host processor. If at the moment a request is received there have been no other requests received before it and a line is available, the program fulfills the request immediately. Otherwise, the program places the request on the queue for the dial set to be handled in its turn.

Lines to be used for handling call-out requests must be designated as CALL=OUT in the corresponding LINE macros; alternatively, they may be designated as CALL=INOUT if they are to be used interchangeably for both kinds of requests.

A station associated with the dial set is represented by its own TERMINAL macro, which represents that station and no other; the telephone number by which the program calls the station is contained within the program. The number is specified when defining the program via the DIALNO operand of the TERMINAL macro. It can be changed later, during program execution, by means of the dynamic control facility.

You may improve line utilization by ansigning as *alternate dial set* to the original dial set, which is accordingly called the *primary dial set.* The alternate dial set in effect helps to handle the load of call-out requests for the primary set when the alternate set is not fully occupied with its own call-out requests. The alternate dial set is in fact a primary dial set for its own group of stations.

Thus, for example, the switched network facilities for an installation might consist of three dial sets--*A, B, and C*--each of which services call-out requests for a group of 20 stations. Each is the primary dial set for its designated group. If B were also defined as the alternate dial set for A and C as the alternate for B, B could help with A's overloads, while C could help with B's overloads.

Dial sets are defined with the DIALSET macro. In addition to specifying the list of communication lines to make up the set, you may, in the DIALALT operand, specify the name of an alternate dial set.

Two other DIALSET macro operands—QLIMIT and QLOAD—determine the network control program's action when the program cannot immediately service a call-out request because all lines in the dial set are busy. Unless you specify a different value in QLIMIT, the program will place only one call-out request on the queue for the dial set. When this *queue limit*—whether it is 1 or a greater value that you specify—is reached, the program rejects any further call-out requests for that dial set. Or you may specify a queue limit of 0, in which case the program rejects *any* call-out request it cannot immediately service.

The other operand, QLOAD, specifies the number of unfulfilled requests the program will permit to accumulate on the queue for the dial set before using a line from the *alternate* dial set (if any) to service requests. The number must not exceed that specified in QLIMIT, for then the number of unfulfilled requests cannot reach the value that would cause the program to use the alternate dial set.

Call-In Operation

A switched line used for call-in (but not call-out) operation is not included in a dial set.

Since the host processor does not know which station will be calling from moment to moment, it directs its call-in requests to a logical-connection station associated with each line. Once a station has called, the logical-connection station represents that station until the logical connection ends.

To specify that a line is available for call-in requests, specify CALL=IN in the LINE macro. Alternately, you may designate the line as available for servicing either call-in or call-out requests (CALL=INOUT).

Using Lines in a Dial Set for Both Call-Out and Call-In Operations

If you want lines in a dial set to service both call-out and call-in requests. This is another way to maximize line utilization because the lines that might otherwise be idle in periods when the program receives few call-out requests can instead be used to service call-in requests.

To specify a line in a dial set as available for handling either incoming or outgoing calls, code CALL=INOUT in the corresponding LINE macro. A dial set may include any combination of lines specified as available for call-out operation or both call-out and call-in operation.

If all lines in a dial set are designated as available for both incoming and outgoing calls, the possibility exists that all may become occupied with outgoing calls, thus preventing any station from being able to call the controller. To prevent this, you may specify, in the RESERVE operand of the DIALSET macro, a minimum number of lines the program must hold in reserve for accommodating incoming calls.

Switched Network Backup

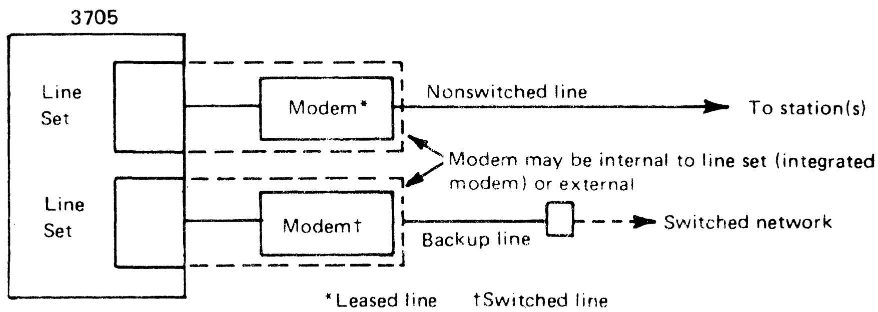
The switched network backup facility of the network control program permits communication between a communications controller and a BSC or SDLC station over a temporary, switched communication path provided as an alternate, or backup, line to the usual (principal) nonswitched point-to-point or multipoint line. Provision of a backup line permits communication between controller and station to continue despite failure of the principal line.

The availability of a backup line requires the installation of appropriate equipment at the controller and stations.

Two forms of switched network backup are available: same-port backup and alternate-port backup. The same-port backup technique can be used for either SDLC or BSC stations. (See Chapter 2 for details on same-port backup.) Alternate-port backup is available only for BSC stations.

Alternate-Port Backup

The alternate-port backup technique, available only for IBM 3270 (BSC or SDLC) and BSC stations, employs a switched-network connection (port), equipped with its own, separate, switched network modem, thus:



The principal and backup lines are represented to the NCP by separate LINE macros. The modem (3872, 3874, or 3875) at the remote station must be equipped with the switched network backup feature (or a separate switched network modem provided in addition to the regular modem, with provision for switching the station from one to the other).

If more than one station is attached to a modem equipped with the switched network backup and fanout features, all attached stations (up to three) can operate concurrently over the backup connection provided that all are switched to backup operation.

Only terminal components represented to the NCP and the access method by TERMINAL macros can participate in backup operation; components represented by COMP macros cannot participate.

In the case of a multipoint principal line, the NCP may continue to communicate normally with any stations on that line not affected by the line failure while concurrently communicating in backup mode with another station over the backup connection. Further, the program can communicate in backup mode with several stations affected by the failure of the principal line provided that sufficient backup ports are available.

Establishing the switched backup connection using the alternate-port technique requires that the operator at the host processor enter the appropriate access method commands. Either the operator or the NCP can select the particular switched port to be used. The operator at the remote station to be contacted must switch the station's modem to switched network operation (and back to normal operation when the principal line is restored to service); thus, voice contact is required between the host processor and the remote station when the backup connection is made or terminated.

Two alternatives are available for dialing the backup connection to a station: automatic calling and program-assisted manual dialing. (A backup connection can be made only from the host processor, not from a remote station.)

Automatic calling —that is, automatic dialing of the telephone number—is possible if the backup port is equipped with an automatic calling unit (ACU), the line address of the ACU is specified in the AUTO operand of the LINE macro for the backup port, and the dial digits are specified in the DIALNO operand of the TERMINAL macro representing the station.

Manual dialing is required if the backup port is not equipped with an ACU. The operator at the host processor must manually dial the telephone number of the station to be reached. If the number is specified in the DIALNO operand of the TERMINAL macro for the station, the access method will inform the operator of the telephone number and the specific line on which to make the call when he enters the command to switch to the backup line. If the number is not specified in the DIALNO operand, the access method tells him only which line is to be used for the call; he must consult a list of stations and their backup telephone numbers to determine which number to dial.

When communication with the remote station(s) is to be restored to the principal line, the operator at the host processor enters a command to switch the station(s) to the principal line. The NCP breaks the backup connection when all stations (if more than one are attached to the line via a fanout modem) have been switched back to the principal line.

Note: If the telephone number of a remote station is changed after the network control program is generated and the automatic calling technique is to be used, the new number *must* be specified in the DIALNO operand and the NCP tables reassembled via partial program generation in which CONDASM=TABLE is specified in the BUILD macro (or via complete program generation).

See Appendix J for an example of how alternate-port backup operation is specified in the network control program. Refer to the *ACF/TCAM Operation* manual for the commands required for backup operation.

Beginning Network Operation

After the NCP is loaded into the communication controller and begins execution, the communications access method must send control requests that logically activate the communication lines. After a line is activated, the NCP accepts requests for any devices connected to that line.

Type of Line Control

All types of stations with which the communications controllers can communicate use one of three line control schemes: binary synchronous (BSC), start-stop (or asynchronous), and synchronous data link control (SDLC). Each communication line attached to the controller uses one of these schemes; the same line never uses more than one. (Some stations can use two line-controls; however, only one type of line control can be defined for these stations in a network control program.)

When defining the NCP; specify which type of line control the program is to use for each line. All lines in a line group use the same line control; therefore the type is specified in the GROUP macro. The operand is LNCTL.

Network Slowdown

The NCP can receive message data from the access method (via the network control subchannel) and from lines in network control mode only as long as it has buffers available for the data. The program normally receives and sends data at the same average rate, although momentary overloads can occur when the program receives more data than it sends during a given time interval. Should the overload be protracted, however, the NCP can exhaust its supply of buffers. To prevent this condition, the NCP continuously monitors its supply of buffers and, when the supply falls to a specified level, automatically enters *slowdown mode.* The level is specified as a percentage of the total number of buffers in the program.

When in slowdown mode, the program reduces the amount of data it receives from lines in network control mode and from the network control subchannel(s), but it continues to send at the normal rate. Since the rate at which buffers are released after transmission of their contents exceeds the rate at which new buffers are obtained for receiving data, a net gain in the number of available buffers results. When the buffer supply is sufficiently replenished, the program automatically resumes normal operation.

The SLODOWN operand of the BUILD macro allows you to specify 12, 25, or 50 percent as the minimum percentage of available buffers below which the program enters slowdown mode. However, during initialization, the NCP dynamically increases the percentage you specified if the minimum NCP buffer requirements cannot be met. The minimum number of buffers that the program must contain for each percentage value is 80 buffers, for 12 percent; 40 buffers, for 25 percent; and 20 buffers, for 50 percent.

Terminal Time-Outs

The NCP normally observes for each communication line two *timeout* intervals of several seconds duration. One of these intervals is the *reply timeout,* which limits the amount of time the program will await a station's response to polling or response to message data sent to the station. The other interval is the *text timeout,* which limits the time that may elapse between receipt of successive message characters from the station after message transmission has begun. If the timeout expires before the response or the next message character is received, the program ends the read operation for that station and notifies the host processor of a timeout error. These timeouts apply to each line in the network whether the line is operating in network control mode or in emulation mode.

By observing these two timeout intervals, the NCP prevents a communication line from being idled indefinitely because of excessive delay in entering successive message characters at a terminal or because a malfunction or power failure at the station interrupts its transmission to the communications controller.

Unless you specify different values in the REPLYTO and TEXTTO operands of the GROUP macro, the NCP uses the timeout intervals indicated in the descriptions of these two operands for all lines in the group represented by that macro. Some applications may justify allowing unlimited intervals, that is, no timeout at all. This also may be specified in the REPLYTO or TEXTTO operands.

Conversational Response

Some BSC and start-stop stations equipped with parity checking accept message data as a positive response to a block of text the station has transmitted. The IBM 1050, 2740 Models 1 2 with Record Checking, 2770, 2972 without the Batched Message Input feature, and all BSC stations except the 2715 and 2780 are such devices.

Transmitting message data instead of the normal positive response eliminates the line turnaround time incurred when sending a positive response character followed by an addressing character and then receiving a response to addressing before sending message data to the station. The benefit of this conversational write operation is improved line utilization.

For each station or component capable of accepting message data as a positive response to text, you may specify conversational operation in the CONV operand of the TERMINAL or COMP macro representing the device. The NCP then withholds sending a positive response after executing a read (or invite) operation if the next request directed to the same device is a write request. Instead, the program sends the message data conveyed by the write request. (For BSC stations, the NCP sends message data as a response only if the data received in the preceding read or invite operation was a complete message [ended by an ETX character].) This option applies only to stations on lines operating in network control mode.

Polling and Addressing Characters

Certain types of start-stop and BSC stations must be polled or addressed by the communications controller in order for them to transmit to or receive from the controller. To receive data from the station, the controller sends a polling character (or sequence) assigned to and recognized by that station. Receipt of the polling character causes the station to unlock the keyboard, allowing the operator to enter data or to activate an input device such as a tape reader if he has previously readied the device to transmit data. Similarly, the controller sends a specific addressing character (or sequence) to signal the station to be ready to accept data from the controller.

If the station is a terminal having more than one input component, such as a card reader and/or a tape reader as well as a keyboard, a polling sequence may be assigned to each component. This allows the controller to solicit data from individual components. Or the polling sequence may specify activation of any input component that the operator has made ready to transmit data.

If the terminal has more than one output component--such as a display, a card punch and/or tape punch, as well as a printer--each may have its own addressing sequence, which allows the communications controller to send data to a specific unit.

Some BSC stations, such as the IBM 2770, recognize component selection characters within message data the station receives over the line. The output component to which data is sent may thus be changed as receipt of the message progresses.

Stations activated by polling and addressing characters are used in network configurations requiring attachment of more than one station to a communication line. Such a line is called a multipoint line, and each station is assigned polling and addressing characters different from those assigned to any other stations on that line. This allows a particular station to be activated while all others remain idle. In addition to the individual characters, a group addressing character (or sequences) may be assigned to certain stations to permit simultaneous transmission of data to all stations in the group. Or a broadcast address character may be assigned to all stations on the line to permit addressing all stations simultaneously.

Stations using polling and addressing are not limited to use on multipoint lines. The IBM 1050, for example, always must be polled and addressed, even if only one 1050 is attached to a line and regardless of whether that line is nonswitched or switched.

The NCP performs the polling and addressing functions for all stations connected to lines operating in network control mode. The access method performs these functions for all stations connected to lines operating in emulation mode. Therefore, in defining the NCP, you must specify the polling and addressing characters for each station with which the program communicates over a line operating in network control mode, but not for those stations on lines that will operate only in emulation mode. Specify the polling and addressing characters for lines in network control mode as follows.

In the POLL operand of each TERMINAL macro representing a station that must be polled, specify the polling character or characters recognized by that station. Similarly, in the ADDR operand of the same macro, specify the addressing character or characters recognized by that station. If a station is used for output only, you would specify only the ADDR operand; conversely, you would specify only the POLL operand for a station to be used for input only.

If a station has more than one individually addressable output component, the access method may establish individual sessions with each, provided that all components with which sessions are to be established are represented by a TERMINAL or COMP macro, the ADDR operands of which specify the addressing characters required. Likewise, if you wish to establish independent sessions with each of two or more input components, each must be represented by a TERMINAL or COMP macro, the POLL operands of which specify the appropriate polling characters. If a station has but one input and one output component, only the TERMINAL macro is required. Each additional component requires a COMP macro.

A terminal having multiple input and output devices requires only a TERMINAL macro if that the terminal has the common polling and addressing capability. (An example is the IBM 1050, for which the common polling character is 0 and the common addressing character is 9.) The common polling and addressing characters must be specified in the POLL and ADDR operands of the TERMINAL macro.

If your network includes start-stop terminals that permit group or broadcast addressing and you wish to use the facility (in network control mode), code an additional COMP or TERMINAL macro. This macro's ADDR operand specifies either the broadcast or group address.

If your network includes multiple component BSC stations connected to point-to-point lines, you may wish to specify selection sequences for each of the output components. Selection characters for one output component may be specified in the ADDR operand of the TERMINAL macro. Specify the selection sequences for any additional output components in separate COMP macros.

Station Telephone Numbers

To call a station (in network control mode) that is connected to the switched telephone network, the network control program must be given the telephone number of that station. You specify the dial digits in the DIALNO operand of the TERMINAL macro representing the station.

If necessary, you may change the number to a different number via the dynamic control facility.

When specifying a telephone number in the DIALNO operand, you may include a dialing pause, in multiples of one second, between successive digits. A dialing pause can allow time for receiving a secondary dial tone, as when an outside-line code or direct-distance-dialing network access code must be dialed before the station's telephone number. End-of-number and separator characters may be included in the sequence of dial digits specified in the DIALNO operand if the modem is designed to use these characters. (The use of these characters is explained in the description of the DIALNO operand.)

Number of Attempts to Dial a Station

Unless you specify otherwise, the NCP, on receiving a request to call a station in network control mode, automatically dials the station's telephone number up to four times in succession. If the last attempt is unsuccessful, the network control program returns to the host processor a response indicating the failure. To specify a different number of attempts to dial a station, code the desired value (up to 255) in the REDIAL operand of the LINE macro. (A value of 255 indicates that the NCP will redial the station indefinitely until the station answers or the access method resets the request.)

Preventing a Monopoly of NCP Buffers

The NCP fills all requests for buffers from a single buffer pool, and no BSC or start-stop station should monopolize the supply of buffers to the extent that other stations are prevented from communicating with the controller in network control mode. Such excessive buffer monopolization could occur if the network control program were to accumulate too much data from a station before forwarding the data to the host processor.

You can prevent buffer monopolization with the TRANSFR and CUTOFF operands of the LINE macro. The TRANSFR operand prevents excessive accumulation by the controller of message data *from* a station. The CUTOFF operand sets a maximum limit on the data received during a single logical connection.

Normally, the NCP routine that receives data from a station accumulates an entire block of a message before passing the data to the host processor. This is desirable because message processing routines within the host processor can then examine an entire block at once.

If the station were to send an excessively long block of data, as could happen if a terminal operator entered thousands of characters without sending an end-of-block character, an unreasonably large number of NCP buffers could be filled by the arriving data. To prevent all of these buffers from being tied up until the block is complete, the NCP restricts the number of buffers that can be filled with arriving data before passing their contents to the host processor. Reception of data from the station is not interrupted when this happens; the NCP continues to allocate buffers for the remaining data.

Each partial block the program passes to the access method is called a *sub-block,* and the NCP is said to be operating in *sub-blocking mode* with respect to the line over which the data is being received. The response header that precedes each partial block indicates to the access method that the data that follows is a sub-block, not a complete block.

Aside from the consideration of buffer monopolization, there is a limit to the amount of data the access method can receive from the NCP during a single data transfer operation. The program accordingly restricts a sub-block to only as much data as it can deliver to the host processor in a single data transfer operation over the network subchannel.

In the TRANSFR operand of the LINE macro, you may specify the size of a sub-block in terms of the number of buffers to be filled before forwarding to the access method. Specifying this parameter in the LINE macro allows you to set a different limit for each of various lines.

If you omit the TRANSFR operand or if you specify a number of buffers greater than the NCP can pass at one time to the access method, the NCP uses as the sub-block size the maximum number of buffers it can deliver to the access method.

Limiting the size of a sub-block in this manner usually prevents undue buffer usage by any one station. This assumes, however, that the NCP promptly transfers the contents of the sub-block buffers to the access method and then releases the buffers to the buffer pool. If for any reason the NCP cannot promptly transfer the sub-blocks to the access method (as when the host processor channel is slow to respond to the controller's signal for service), the data accumulating from the station can still cause monopolization.

For this reason, you may also set a limit on the number of sub-blocks the network control program will accept from a start-stop or BSC station during a single logical connection. If the station sends enough data to cause the limit to be reached, the program breaks the logical connection for that station and is then free to service another station on the same line, if a multipoint line. Limiting the number of sub-blocks also serves to prevent a station from monopolizing a multipoint line so as to delay servicing of other stations on the same line. In the CUTOFF operand of the LINE macro, you specify the number of sub-blocks to be accepted from a station. If you omit the CUTOFF operand, the NCP continues to accept sub-blocks from a station until the entire block is received. If any sub-block contains an error (indicated by a bit in the response header for the sub-block), the application program should normally discard all of the sub-blocks in the block, not only the one in error. There is no provision for requesting retransmission of sub-blocks in error.

Use of Buffer Delay for Buffered Terminals

Some types of IBM terminals receive incoming data into buffers at high speed and then print (or otherwise display) the data at a much slower rate. If the NCP has multiple data blocks to send to the same terminal, it must wait after sending each block for the terminal to print the contents of the terminal buffer. Then the program sends the next block. If the line is a multipoint line operating in network control mode, the NCP can use the time the line would otherwise be idle for communicating with other terminals. That is, at any given moment, the program can be sending to one of several terminals while the others are printing data received earlier.

For each start-stop or BSC terminal attached to a line operating in network control mode, specify in the BFRDLAY operand of the TERMINAL macro the delay in seconds that the NCP should wait after sending each block before sending the next one. The value you specify should equal the length of time the terminal needs to print or otherwise display the contents of its buffer.

The terminals for which specifying a buffer delay is appropriate are:

IBM 2740 Model 2 with Buffered Receive feature
IBM 2770
IBM 2980
IBM 3275
IBM 3277
IBM 3284
IBM 3286
IBM 3780

Error Conditions and Recovery Procedures

Communication between the communications controller and stations in the network is subject to *input/output* (I/O) errors, usually caused by the transient noise conditions to which communication lines are sometimes susceptible or by hardware malfunctions. The NCP takes no action for errors on lines operating in emulation mode. Any error-recovery procedures desired must be executed by the access method.

For lines operating in network control mode, if the error is of a kind that can be recovered from, the NCP makes the appropriate recovery efforts. For example, upon detecting a parity error in received data, the program signals the station to retransmit the data if the station is of the kind that can automatically retransmit, as from a buffer, or of the kind that can visually inform its operator to re-enter the same data. Conversely, upon being informed by the station that it has received data in error, the program retransmits the data.

The maximum number of retransmissions may be specified for each line operated in network control mode. If error-free transmission is not achieved before the retransmission limit is reached, the NCP indicates the fact in its response to the access method.

If the I/O error is of the kind that inherently cannot be recovered from (such as a modem error), the NCP makes no error-recovery attempt but immediately indicates in its response to the access method what kind of error occurred.

Once the NCP notifies the access method that it is unable to clear the error condition, it makes no further attempts at error recovery. Nor does the program attempt to execute any further requests for the device affected. Instead, it places the device in an *error-lock* status. The access method can then take appropriate action. For example, the access method can send a control request that causes the NCP to remove all unfulfilled requests from its queue for the device and return them to the access method, which can then modify the sequence of requests or build a new sequence and send it to the NCP. Or the access method can ignore the error condition and send a control request that reinstates the queued requests, thus removing the device from error-lock status.

Input/output errors can occur either during transmission of message data or while the NCP is performing a control function preparatory to or following message transmission. I/O errors are consequently divided into *text-mode* errors and *control-mode* errors.

The number of error-recovery attempts for text-mode errors is determined by the RETRIES operand of the LINE macro. The number you specify applies to all stations with which the NCP communicates over the line represented by the LINE macro. The number of error-recovery attempts for control-mode errors is set by the CRETRY operand of the GROUP macro. Thus, the same maximum number of recovery attempts applies to all stations on all lines making up the line group.

Text-mode errors may be further classified as *text-read* errors and *text-write* errors. Text-read errors are those occurring when receiving from a station; text-write errors occur when sending to a station.

For text-read errors you may specify a single sequence of retransmissions, from 1 to 254. Or you may specify unlimited retransmissions.

For text-write errors you may specify a single retransmission sequence or multiple sequences. When you specify multiple sequences, the program executes the first sequence immediately after detecting the error condition. If transmission is still unsuccessful at the end of the sequence, the program pauses and then executes the next sequence. This activity continues until successful transmission is accomplished or the limit on retransmissions is reached. Specifying a pause between retransmission sequences allows time for transient noise conditions on the line to subside. A problem that may be responsible for the repeated errors.

Unless you specify otherwise in the RETRIES or CRETRY operands, the NCP will perform a maximum of two recovery attempts for start-stop stations and seven attempts for BSC stations for text-read, text-write, and control-mode errors. As stated above, error recovery is attempted only for lines operating in network control mode.

Protecting against Failure of Modem or Automatic Calling Unit

Various malfunctions occurring within a modem or automatic calling unit (ACU), if not protected against, could cause the communication line attached to the modem or ACU to be tied up indefinitely without the knowledge of the NCP, thus effectively rendering the line unusable. To prevent such an occurrence, the NCP has three timeout values for lines operating in network control mode: *enable timeout, disable timeout,* and *dial timeout,*

If a switched line is not enabled (that is, the data set ready signal line in the modem is not turned on) within the enable timeout period, the enable operation is terminated abnormally. Similarly, if the switched line is not disabled (data set ready signal line is not turned off) within the disable timeout period, the disable operation terminates abnormally.

Unless you specify different values, the network control program uses an enable timeout of 2.2 seconds and a disable timeout of 3.0 seconds. These values are generally appropriate, but you may specify different ones in the ENABLTO and DSABLTO operands of the BUILD macro. The timeouts specified apply to all communication lines operating in network control mode.

The third timeout, dial timeout, protects against the failure of the automatic calling unit for a switched line to reach a called station within a reasonable period. This condition may result from failure of the called station to answer, from failure of the common-carrier equipment to complete the connection path, or from a malfunction within the ACU. Expiration of the dial timeout for any of these reasons causes abnormal termination of the dialing operation.

A dial timeout of 60 seconds is normally appropriate, and this is the value the NCP uses for lines in network control mode unless you specify a different one in the DIALTO operand of the BUILD macro.

Direction of Transmission

In most network configurations, the stations are capable of both sending and receiving message data. However, for each BSC or start-stop station attached to a line operating in network control mode, you may specify that the station be used for input only via the DIRECTN operand of the TERMINAL macro representing the station. If you specify input only, the NCP rejects any requests from the host processor to send to the station. Conversely, specifying output only causes the program to reject any requests to accept data from the station.

Erasing Critical Data in Buffers

Normally, the network control program releases buffers to the buffer pool after use without first clearing the buffer contents. As buffers are reallocated for subsequent operations, the old contents are overlaid with new message data.

For maximum assurance that security-sensitive data remaining in buffers returned to the buffer pool is not subsequently transmitted to an unintended destination, you may specify that the network control program clear buffers before returning them to the pool as follows. Specifying ERASE=YES in the BUILD macro causes the NCP to: (1) erase all data buffers received from the host processor but rejected because the network control program has entered slowdown mode; (2) erase all buffers containing data blocks received from the host processor in which the NCP detected a transmission error; and (3) erase data buffers containing PIUs sent to the host processor, after the host processor has indicated successful receipt of the data.

In addition, you may specify CDATA=YES in individual CLUSTER, TERMINAL, and COMP macros to cause the program to erase buffers that have been used for receiving from or sending to the corresponding start-stop or BSC stations; this function is not applicable for data received from or sent to SDLC stations.

Automatic Network Shutdown

For BSC and start-stop lines, the network control program:

- Cancels the command currently being executed for the line
- Breaks the switched connection if the line is a switched line
- Cancels the line trace or online test operation if the line is currently being traced or is undergoing online testing
- Dissociates the line from the owning SSCP with which communication has been lost

For BSC and start-stop stations, the network control program:

- Stops general polling of clustered stations
- Cancels any commands currently pending for the station
- Sends a predefined message to stations for which CRITSIT=YES is specified in the TERMINAL macro (see "Critical Situation Notification" in the following paragraph)
- Cancels any sessions in which the station is currently active
- Resets the station from monitor mode if that mode is currently in effect

Critical Situation Notification

During automatic network shutdown of lines in network control mode, the program can automatically send a predefined message to each active start-stop and BSC station that alerts the station to the impending shutdown. This critical situation message is sent to each active station in whose TERMINAL macro you have specified CRITSIT=YES, provided that the line to that station is currently operating in network control mode. Define the content of the message in the CSMSG and CSMSGC operands of the BUILD macro. A message header, if required for IBM 3270 terminals, may be specified in the CSMHDR and CSMHDRC operands of the BUILD macro.

The critical situation message always begins with the date and time (the latter is in 24-hour format) and ends with the text you specify in the CSMSG operand.

Example:

```
05/14/78 19.27.05 NETWORK SHUTDOWN IN PROGRESS-NO FURTHER
TRANSMISSIONS UNTIL NOTIFIED.
```

The NCP does not automatically send a message to notify stations when the network is again operational; this is the responsibility of the access method.

BSC Operation

The options described in this section apply only to binary synchronous (BSC) stations. Options common to both BSC and start-stop stations are described under the heading, *BSC and Start-Stop Operation.*

Transmission in Transparent Mode

The NCP is capable of sending and receiving data over a BSC line in *transparent mode.* Transparent mode allows transmission of message data containing any bit patterns, including those that the sending and receiving station normally recognize and act upon as line control characters.

For transmission to a station on a line operating in network control mode, the data communication request specifies whether the NCP is to send the message data in normal or in transparent mode. The program accordingly transmits, at the appropriate time, either the nontransparent control characters—STX, ETB, and ETX—or the transparent control sequences—DLE STX, DLE ETB, DLE ETX.

For transmission in network control mode from a station to the communications controller, the NCP automatically deletes the line control characters it detects in the received message data before forwarding the data to the host processor. The control information accompanying the data indicates to the host processor whether the NCP received the data in nontransparent mode or in transparent mode. If a station is on a line operating in emulation mode, the NCP does not insert and delete the required transparent control sequences. The access method must include all required control sequences in message data it delivers to the NCP for forwarding to a station. Conversely, the NCP delivers to the access method unchanged, transparent message data it receives from lines in emulation mode.

Intermediate Block Checking Mode

When receiving from a station in network control mode, the NCP automatically examines the block-check characters (BCC) that follow each intermediate-transmission-block (ITB) character, if any, in the received data. In the ITBMODE operand of the TERMINAL macro for the station, you specify whether the NCP is to send error information bytes (EIB) to the host processor following each ITB character. If you specify insertion of EIBs, the application program in the host processor can scan the received data for ITB— the error information byte—to determine whether an error occurred in the intermediate block.

When receiving from the host processor message data to be sent to a station in network control mode, the NCP can automatically remove EIBs, if any, before transmitting the message data. If you specify deletion, the NCP deletes the first character following each ITB it detects within the data to be sent. You should specify deletion only if each first-following character is an EIB; otherwise the receiving station will not receive the first data character of each intermediate block.

Intermediate Block Checking Mode for Transparent Text

If you specify XITB=YES in the BUILD macro, the NCP is capable of inserting ITB sequences and error information blocks in transparent text as well as in nontransparent text for BSC stations on lines operating in network control mode. Insertion of ITB sequences and error information blocks occurs only for those BSC stations whose TERMINAL macros specify, in the ITBMODE operand, use of the intermediate block-checking facility, as follows:

If the BUILD macro specifies XITB=YES and the *first* parameter of the ITBMODE operand specifies intermediate block checking, the NCP substitutes an error information block for each DLE ITB sequence in transparent text received from the station.

If the BUILD macro specifies XITB=YES and the *second* parameter of the ITBMODE operand specifies intermediate block checking, the NCP inserts DLE ITB sequences into transparent text being sent to the station. Special 2-byte fields within the text received from the host processor determine the intervals at which the DLE ITB sequences are inserted.

If the BUILD macro specifies XITB=NO (or you omit the XITB operand), no insertion of ITBs and EIBs is performed for any BSC stations.

This function is performed only if the communication line to a station is currently operating in network control mode.

ID Exchange and Verification

The NCP can receive an identification (ID) sequence from any BSC station that calls the communications controller over a switched line operating in network control mode and can either check that sequence against a list of valid sequences within the program or pass the sequence to the access method for checking. (If the access method is operating under DOS/VS, sequences *must* be passed to the access method for checking.) Upon failing to recognize the sequence as valid, the NCP does not proceed with message transmission. Instead it either breaks the line connection or maintains the connection but forwards the unrecognized sequence to the host processor for checking against a list kept there. In the latter case, the host processor can signal the NCP to proceed with message transmission or break the line connection.

After either the NCP or the access method checks the received sequence, the NCP can send an ID sequence in reply.

Advantages of NCP Verification versus Access Method Verification

If the NCP verifies a received ID sequence, message transmission can begin sooner than if the ID must be forwarded to the access method for checking. But the storage space needed within the NCP to maintain a list of ID sequences can be considerable if there are many sequences. Conversely, if the access method maintains the list, storage requirements within the controller are minimized, but ID sequence checking by the access method may take longer.

A compromise, if OS/VS VTAM or TCAM is being used, is to keep within the network control program the ID sequences for the stations that call most often, and to keep within the host processor those sequences representing stations that call less frequently. (If DOS/VS VTAM is the access method used, all sequences received from BSC stations *must* be passed to VTAM.)

The various ID verification options for checking by the NCP are specified with the IDSEQ operand of the TERMINAL macro and the NOMATCH operand of the IDLIST macro. ID verification by VTAM is specified by the (VTAM-only) VIDLIST macro. See the *ACF/TCAM Installation* manual for information on ID verification by TCAM.

Defining a Controller ID Sequence

If you choose to define an ID sequence within the controller to be sent in response to ID sequences received from BSC stations, specify the sequence in the CUID operand of the BUILD macro. The maximum length is 20 characters. However, because different types of stations may expect ID sequences of different lengths, you must specify the required length in the CUIDLEN operand of the TERMINAL macro for each station to which the sequence is to be sent. If you omit the CUIDLEN operand, no controller ID sequence is transmitted.

Controller ID sequences are not sent to stations on lines currently operating in emulation mode.

Sending and Receiving WACK Sequences

When receiving message data from a BSC station in network control mode, the NCP may need to temporarily defer further input from that station. This can happen, for example, when the NCP has no further read requests from the host processor for receiving additional message data. When this occurs, the program responds to the block just received with a WACK (wait-acknowledgment) sequence instead of the usual positive acknowledgment (ACK-0 or ACK-1). The WACK sequence informs the sending station that the NCP is deferring the positive acknowledgment until it is again able to receive from the station. Upon receiving the WACK sequence, the station replies with an ENQ character.

Exchanging of WACK and ENQ sequences can continue for as long as the NCP needs to defer input from the station or until the station breaks the connection. When the program is ready to resume receiving from the station, it sends the deferred positive acknowledgment (ACK-0 or ACK-1).

There is no limit to the number of WACK sequences the NCP will send. The program does, however, limit the number of WACK sequences it will accept from the station unless you wish to allow the program to receive them without limit.

Unless you specify otherwise, the NCP will accept up to 15 consecutive WACK sequences from a station. You may specify a different number, or specify unlimited acceptance of WACK sequences, in the WACKCNT operand of the GROUP macro.

Sending and Receiving Temporary Text Delay Sequences

When the NCP must temporarily suspend *sending* to a station in network control mode, it need not break the logical connection. Instead it can transmit a *temporary text delay* (TTD) sequence in lieu of the next message block. The TTD sequence informs the receiving station that the communications controller will continue sending after a short pause. The station replies to the TTD sequence with a NAK character. Exchange of TTD and NAK can continue as long as the program needs to defer transmission or until the station breaks the logical connection.

There is no limit on the number of TTD sequences the NCP will send. The program does, however, limit the number of TTD sequences it will accept from a station unless you wish to allow unlimited acceptance. When the limit is reached, the program breaks the logical connection.

Unless you specify otherwise, the program will accept up to 15 consecutive TTD sequences from a station. You may specify a different limit or specify that the program is to accept them without limit in the TTDCNT operand of the GROUP macro.

Frequency of Transmission of Synchronous Idle (SYN) Characters

In binary synchronous communications, a *synchronous idle* (SYN) character must be transmitted on the communication line at regular intervals to maintain the sending and receiving stations in synchronism. Binary synchronous stations transmit these characters periodically when sending message data. (A sequence of SYN characters is also transmitted when the line is otherwise idle.)

The NCP conforms to normal BSC practice by sending the SYN characters once each second. In rare circumstances it may be appropriate to change this interval; this can be done with the SYNDLAY operand of the GROUP macro. (The change will be effective only for lines serviced by a type 2 scanner. The type 3 scanner hardware sends SYN characters at one-second intervals.) This function applies only to lines operating in network control mode.

SDLC/BSC Path Function

The SDLC/BSC path function is an option by which the NCP can transmit data originating at an SDLC station directly to a specified BSC station without first sending the data to the host processor. (Normally, all data transmitted in the network passes through the access method in the host processor.) Only path information units (PIU) containing message data pass directly between the SDLC station and the BSC station; error and control messages associated with the SDLC station are sent to the access method.

To establish the logical path between the SDLC and the BSC station, specify the name of the TERMINAL macro representing the BSC station in the DATASW operand of the LU macro defining the SDLC logical unit.

Associated with each BSC station that is to participate in the SDLC/BSC path function there must be a block-handling routine that converts the BSC message data to the format required by the SDLC logical unit. This block-handling routine must be specified as executable at point 3 (after the data is received from the BSC station) and must be active whenever the SDLC-BSC path is to be available for data transfer. Format conversion for the message data transmitted in the opposite direction (SDLC to BSC format) is always present in the network control program.

An IBM-supplied block-handling routine that makes the proper conversion may be invoked by specifying an SPAFPT3 macro in the point 3 block handler associated with the BSC station. (See the description of the SPAFPT3 macro in Chapter 5.) Or you may code a user block-handling routine to convert the PIU format provided that it is functionally equivalent to the IBM-supplied routine. In this case, you would include a UBHR macro calling the user-written routine rather than a SPAFPT3 macro in the block-handler.

For more information about the SDLC/BSC path function, including requirements and restrictions about its use, see *SDLC/BSC Path Function System Programmer's Guide* (GC30-3029).

Start-Stop (Asynchronous) Operation

The options described in this section apply only to start-stop terminals (also called asynchronous terminals). Options common to both start-stop and BSC stations are described earlier in this chapter under the heading, *BSC and Start-Stop Operation.*

Multiple Terminal Access Facility

A major feature of the NCP is its ability to communicate in network control mode with a variety of dissimilar, commonly used start-stop terminals over the same switched network connection, or "port." This feature, called the *multiple terminal access* (MTA) facility, makes it unnecessary to reserve a separate port for each type of terminal as has often been the case in networks. This facility therefore serves to minimize the number of communication lines and their attendant modems and line-attachment hardware, resulting in lowered communication costs. At the same time, greater utilization of the remaining lines is achieved.

These types of terminals are accommodated by the multiple terminal access facility:

IBM 1050
IBM 2740 (basic)
IBM 2740 with Record Checking
IBM 2740 with Transmit Control
IBM 2740 with Transmit Control and Checking
IBM 2741
IBM 3767
Western Union TWX

The multiple terminal access facility allows the NCP either to call any MTA terminal over a line defined as a call-out line or to receive calls from MTA terminals over a line defined as a call-in line. The same line may be used for both call-out and call-in MTA operation. Lines used for call-out operation are included in dial sets, as for non-MTA operation.

MTA support for TWX terminals provides for discrimination of 110 baud and 300 baud terminals and can associate one of three translation tables (ASCII, DIC1, or DIC3) with a TWX terminal according to the first character entered at the terminal.

When answering calls over a line defined as an MTA line (specified by the MTALIST operand of the LINE macro), the program automatically determines the type of terminal in terms of its line control discipline and transmission code employed. Once these have been determined, the NCP carries on sessions and logical connections in the usual way.

In addition, by analyzing a code entered by the terminal operator when calling the controller, the NCP can distinguish among terminals that, while of the same type, require differing terminal or line operating parameters or procedural options.

These parameters and options are:

- The carriage return rate
- The presence or absence of the accelerated carrier return (ACR) feature for 1050 terminals
- The length of the print line used by the terminal printer
- The line speed, interrupt priority, modem data rate, and bit clocking options

- The maximum number of sub-blocks to be accepted from the terminal during a single logical connection
- The maximum number of attempts to recover from text-mode errors

To use the multiple terminal access facility, you (1) define the types of terminals to be handled as MTA terminals (these may be any combination of the terminals listed above) and (2) specify the lines with which the NCP will communicate with each of the types of terminals. All lines used for MTA operation must operate in network control mode.

Your requirements for the MTA facility are specified in the MTALCST, MTALIST, MTATABL, and MTAPOLL macros; the MTALIST operand of the LINE macro; the LCST and TERM operands of the TERMINAL macro; and the MTARTO and MTARTRY operands of the BUILD macro. An example of the use of MTA facilities appears in Appendix J.

Transmission of Attention Signals

The NCP, when transmitting to a terminal in network control mode, can respond to attention signals received from the terminal in either of two ways. (1) The program can interrupt its transmission to the terminal and immediately notify the host processor that the terminal has sent the attention signal. The program halts any remaining requests for that terminal. It is then up to the host processor to determine the next operation for the terminal. (2) The NCP can ignore the interrupt and continue sending to the terminal.

In the FEATURE operand of the TERMINAL macro for each terminal equipped to send attention signals (IBM 1050, 2741; AT & T 83B3; WU 115A, TWX), specify the ATTN parameter. If you wish the NCP to interrupt its transmission upon receiving the attention signal, specify ATTN=ENABLED in the TERMINAL macro. If you wish the program to ignore the signal, specify ATTN=DISABLED (or omit the ATTN operand). The foregoing applies to attention signals received while the NCP is transmitting to a terminal. The program can also monitor a communication line operating in network control mode for an attention signal or a disconnect condition detected while the program is momentarily executing no read or write commands for the line, provided that an active session is in progress with that terminal. The program notifies the access method that it has detected an attention signal or a disconnect condition.

Specify whether this function is required in the MONITOR operand of the LINE macro. The option is required if terminals on the line will communicate with TSO applications. (The NCP performs the monitoring function only if directed to do so by command from the access method. TSO causes the access method to send the required command.)

Logical Keyboard Lock for TWX Terminals

The keyboard of a TWX terminal, unlike those of other start-stop terminals, cannot be locked by the NCP when no read or write operation is in effect for the terminal. (Locking the keyboard prevents data from being entered when the NCP is not ready to receive from the terminal.)

Therefore, the NCP sends TWX terminals a character sequence that "jogs" the printing mechanism when the program is not ready to receive. This serves as a signal to the terminal operator not to enter data.

The character used to jog the TWX printer mechanism is specified in the KBDLOCK operand of the GROUP macro. Any character specified should be a non-printing, non-spacing character that jogs the printer mechanism.

Carriage Return Delay

A terminal operator may press the return (carriage return) key of the terminal at the end of a message block he is entering from the keyboard. If the NCP sends message data to the terminal immediately after receiving the block the terminal has just sent, the first several characters of the data the program sends may be printed randomly during the return motion of the terminal's printing mechanism. To prevent this from happening, you may specify that the program pause momentarily after completing a read operation before starting the next write operation. This allows time for the printing mechanism to return to the left margin. Specify this function in the CRDLAY operand of the TERMINAL macro that represents the terminal printer. The program pauses between the read and write operations only if the message block received from the terminal ended with a carriage return (new line) character or, for an IBM 1050 equipped with the automatic EOB feature, an EOB character.

This carriage return delay function is performed only for IBM 1050, 2740 Model 1, 2741 terminals, and TWX terminals (as specified in the TERM operand of the TERMINAL macro) and any multiple-terminal-access terminal (TERM=MTA).

Downshifting on Space Characters

Some AT & T 83B3, Western Union 115A, and World Trade teletypewriter terminals, upon sending or receiving a space character, automatically downshift so that subsequent message text is in lowercase mode. Automatic downshifting avoids the need to send a LTRS character to effect downshifting. In the LINE macro for each teletypewriter line, indicate whether the terminals are equipped with the downshift function. Specify this function in the SPSHIFT operand if the line is to operate in network control mode and in the FEATURE operand (SPACE parameter) if the line is to operate in emulation mode.

Deleting FIGS and LTRS Characters

Message data received from 83B3, 115A, and World Trade teletypewriter terminals contains the 2 case-shifting characters FIGS and LTRS. If the lines to which such terminals are attached operate in control mode, the NCP removes FIGS and LTRS characters from the data it transfers to the host processor unless you specify, in the FGSLTRS operand of the LINE macro for such terminals, that the program is to leave these characters in the data. (The characters are not deleted from message data received over lines operating in emulation mode.)

TWX ID Exchange and Verification

The NCP can recognize an identification (ID) sequence from any TWX terminal that calls or is called by the communications controller over the switched telephone network; the NCP can either check that sequence against a list of valid sequences within the program or pass the sequence to the access method for checking. (If the access method is DOS/VS VTAM, sequences must be passed to VTAM for checking.) Upon failing to recognize a sequence as valid, the NCP does not proceed with message transmission. Instead it either breaks the line connection, or it maintains the connection but forwards the unrecognized sequence to the access method for checking. In the latter case, the access method can signal the NCP to proceed with message transmission or to break the line connection.

After either the network control program or the access method checks the received sequence, the NCP can send an ID answerback sequence to the terminal before receiving text from the terminal.

Defining an Answerback Sequence

In the TWXID operand of the BUILD macro, you may specify the answerback sequence to be sent to TWX terminals. Two different sequences may be specified: one to be sent when a terminal calls the controller and the other to be sent when the controller calls a terminal. The maximum length of either sequence is 20 characters. In the CUIDLEN operand of the TERMINAL macro for each TWX terminal, you must specify the length of the answerback sequence to be sent. If you omit the CUIDLEN operand, the program does not send the answerback sequence to that TWX terminal.

Answerback sequences are not sent to TWX terminals on lines currently operating in emulation mode.

Options for World Trade Teletypewriter Terminals

In addition to the downshift-on-space character and FIGS/LTRS options mentioned above, there are two other procedural options for World Trade teletypewriter terminals.

Pad Characters

Some World Trade teletypewriter terminals have a motor that runs continuously whether or not the terminal is sending or receiving data. Others have motors that run only during data transmission; the motor stops automatically after about 10 seconds have elapsed since the terminal sent or received a character. Terminals of the latter type must receive several pad, or idle, characters before receiving message characters to allow sufficient time for the motor to reach operating speed. The number of characters required depends on the data rate on the communication line.

For lines running in emulation mode, the access method is responsible for including the appropriate number of idle characters in message data it sends to the terminal.

For lines running in network control mode, the NCP automatically sends the idle (pad) characters if you specify the required number in the PADCNT operand of the GROUP macro for the group of lines to which such terminals are attached.

EOB and EOT Sequences

You may specify the character sequence the NCP is to recognize as the end-of-block (EOB) and end-of-transmission (EOT) sequences when receiving from a terminal.

The EOB sequence may be either FIGS *x* or *nnnn*. *x* and *n* may be any code combination except a combination representing the FIGS or LTRS character. If the terminal is equipped to send who-are-you (WRU) sequences, *x* also may not be the letter D.

The EOT sequence may be FIGS *y* LTRS; *y* may be any code combination except one representing FIGS, LTRS, or the same *x* character used in the EOB sequence, FIGS *x*.

Specify the required EOB and EOT sequences in the EOB and EOT operands of the GROUP macro if any of the lines in the group are to operate in emulation mode and in the WTTYEOB and WTTYEOT operands of the GROUP macro if any of the lines are to operate in network control mode.

Procedural Options for Operations in Emulation Mode

There are three procedural options when defining a line that always operates in emulation mode: (1) the type of line control discipline to be used for each line; (2) the terminal timeouts required, and (3)—for World Trade teletypewriters only—the end-of-block and end-of-transmission sequences to be recognized by the program.

Type of Line Control

All types of stations with which the communications controller can communicate in emulation mode use one of two line control disciplines: binary synchronous (BSC) and start-stop (or asynchronous). Each line attached to the controller uses either BSC or start-stop line control; the same line never uses both types.

The type of line control discipline used is specified in the LNCTL operand of the GROUP macro. (All lines in a group must use the same line control discipline.)

Terminal Timeouts

The NCP normally observes for each communication line two *timeout* intervals of several seconds' duration. One of these intervals is the *reply timeout*, which limits the amount of time the program will await a station's response to polling or response to message data sent to the station. The other interval is the *text timeout*, which limits the time that may elapse between receipt of successive message characters from the station after message transmission has begun. If the timeout expires before the response or the next message character is received, the program ends the read operation for that station and notifies the access method of a timeout error. These timeouts apply to each line in the network and prevent a communication line from being idled indefinitely.

Unless you specify different values in the REPLYTO and TEXTTO operands of the GROUP macro, the program uses the timeout intervals indicated in the descriptions of these two operands for all lines in the group represented by that macro. Some applications may justify unlimited intervals, that is, no timeout at all. No timeout may also be specified in the REPLYTO or TEXTTO operands.

EOB and EOT Sequences for World Trade Teletypewriter Terminals

You may specify the character sequence the NCP is to recognize as the end-of-block (EOB) and end-of-transmission (EOT) sequences when receiving from a terminal.

The EOB sequence may be either FIGS *x* or *nnnn*. *x* and *n* may be any applicable telegraph code combination except a combination representing the FIGS or LTRS character. (If the terminal is equipped to send who-are-you (WRU) sequence, *x* also may not be the letter D.)

The EOT sequence may be FIGS *y* LTRS; *y* may be any applicable telegraph code combination except one representing FIGS, LTRS, or the same *x* character used in the EOB sequence, FIGS *x*.

Specify the required sequences in the EOB and EOT operands of each GROUP macro representing a World Trade teletypewriter (teleprinter) line group.

Note: Appendix H lists the transmission code bit patterns for the ITA2 and ZSC3 codes.

Multi-Subchannel Line Access Facility

The multi-subchannel line access (MSLA) facility of the PEP extension allows the program to communicate in emulation mode over two type 4 channel adapters concurrently. The channel adapters may both be attached to the same host processor or may be attached to separate processors. The MSLA facility further allows two or more host subchannels (on the same or different channels) to communicate alternately with the same communication line. In operation, a command issued over one of the subchannels seizes the line for use of that subchannel and the access method using that subchannel. The access method retains use of the line via that subchannel until it issues a Disable command, thus releasing the line for use by another subchannel. (Alternately, the 3705 control panel can be used to release a line from control of one subchannel in order to switch it to another subchannel. This action is required if the access method using the line does not issue Disable commands.)

Subchannel-to-line associations are established during program definition and can be changed only by regenerating the program.

The physical characteristics of the line (such as type of line control, line speed, etc.) remain constant regardless of which subchannel is currently using the line. The use of the line by each subchannel must be consistent with the line characteristics. Violation of this requirement will cause unpredictable results when the access method communicates with the line.

The MSLA facility can be used in the following ways:

- Load balancing—communication lines can be switched from one host processor to the other during high-traffic periods to balance the load on the processors.
- Host processor backup—communication lines can be switched to a backup host processor if the original host processor, channel, or access method fails. Execution of the control program does not end, and the program need not be reloaded into the communications controller.
- Line sharing—two access methods in the same or different host processors can share the same communication line alternately. The same line can thus be assigned to different applications at different times of day.

The description of the ADDRESS operand of the LINE macro explains how to associate subchannels with a line.

Block-Handling Options

Block handling refers to the optional message processing of message data within the communications controller. The NCP can process either message data from the host processor before sending it to a start-stop or BSC station or message data received from a station before sending it to the access method. Processing is possible only when the message data between controller and station is transmitted in network control mode.

The IBM-supplied network control program modules provide two standard message-processing functions. Each is performed by a block-handling routine invoked by a program generation macro instruction. In addition, user-coded block handling routines may be added to the network control program during the generation procedure. A program generation macro, UBHR, allows you to invoke the user block-handling routines in the same way as IBM-provided block-handling routines. (The IBM-supplied block-handling routines cannot be included in a user block-handling routine.)

Two optional IBM-supplied block-handling routines allow for insertion of date and time-of-day into messages and automatic correction of incorrect message text. An additional IBM-supplied routine may be used to convert message data from BSC to SDLC format when the SDLC/BSC data path function is used. (See *SDLC/BSC Path Function* under *BSC Operation* in this chapter.)

Insertion of Date and Time

The NCP can insert the current date, or time of day, or both, into message blocks it receives from the access method over the network control subchannel or from a station over a line in network control mode.

The date may be in any of four formats: (1) month/day/year, for example, 10/21/78; (2) year followed by day of year, for example, 78.294 (October 21, 1978); (3) year/month/day (78/10/21): or (4) day/month/year (21/10/78).

The time of day is in the format hh.mm.ss. (hours, minutes, seconds). The continental (24-hour) form is used. For example, 09.17.25 and 21.17.25 represent 9:17:25 a.m. and 9:17:25 p.m., respectively. (Each format is preceded by an EBCDIC blank character.)

The date and time may be placed in the first block of each message or transmission, or in every block of the message or transmission.

Date and time insertion is specified with the DATETIME macro.

Automatic Text Correction

Automatic text correction is an editing function by which the NCP replaces text incorrectly entered from a terminal keyboard with the corrected characters the terminal operator subsequently sends. The program does this by scanning each block for predefined characters called text canceling characters. The NCP deletes from the block each such character it finds, plus 1 preceding text character. For example, if the program finds a sequence of 3 canceling characters, it 3 characters plus the 3 immediately preceding characters.

A keyboard operator may, for instance, enter COMMUNCIATE and, seeing that he has misspelled it, enter 5 backspace characters to "back up" to the first erroneous character. Then he re-enters the corrected characters, thus:

COMMUNCIATE bksp bksp bksp bksp bksp ICATE

If you have specified the text correction option and designated *backspace* as the text-canceling character, the text correction block-handling routine deletes the five backspace characters and C I A T E. The remaining characters form the correctly spelled word COMMUNICATE.

The text-canceling character need not be a backspace character. Any other character (except a line control character) is adequate if it is not used in any other way within message text. For example if / is the character chosen and a keyboard operator enters ATLANITC///TIC, the text correction block-handling routine corrects the word to ATLANTIC.

The EDIT macro specifies the text correction function.

SDLC/BSC Path Function Block-Handling Routine

Use of the SDLC/BSC path function for message transmission between an SDLC logical unit and a designated BSC station (specified by the DATASW operand of the LU macro) requires that the network control program convert path information units (PIU) transmitted between the BSC station and the logical unit from the BSC format to the SDLC format, and vice versa. You may invoke an IBM-supplied conversion routine by specifying a SPAFPT3 macro in a block-handling routine that you assign to the BSC station. The block handler must be specified as executable at point 3; that is, process (conversion) of the data is done after receipt from the BSC station. (The code necessary to convert from the SDLC to the BSC format is always present in the network control program, so no macro is provided to call this routine.) Alternatively, you may code a user block-handling routine to do the BSC-to-SDLC format conversion and include it in a point 3 block handler via a UBHR macro.

User Block-Handling Routines

Any block-handling routine you provide is referred to as a *user* block-handling routine. You code a user block-handling routine using the communications controller assembler language (similar to the operating system assembler language), assemble it using the controller assembler, and then place the routine in a data set available to the NCP generation procedure. Then you include in the program generation source statements a UBHR macro instruction that specifies the name of your routine and the point at which the network control program is to execute it.

Guidelines for Writing User Block-Handling Routines

User block-handling routines permit you to add certain data-handling functions to the network control program. These routines typically examine and manipulate incoming or outgoing data contained in NCP buffers. If you have a good general understanding of the network control program and the access method, you may add such routines to the program with little likelihood of disrupting the NCP code. On the other hand, routines that perform more complex functions, such as leasing and releasing buffers or scheduling input/output operations, require that you have an intimate understanding of the internal operation of the network control program and the access method. Adding such routines must be approached with caution to avoid disrupting the network control program logic.

Coding user block-handling routines requires knowledge of the information in these publications: *IBM 3704 and 3705 Communications Controllers Principles of Operation, IBM 3704 and 3705 Communications Controller Assembler Language, ACF/NCP Logic Manual.*

The assembled object modules containing the user block handling routines must be placed in the data set specified by the USERLIB operand of the BUILD macro. During stage 2 of the program generation procedure, user block-handling routines that you have specified in the appropriate UBHR macros are included in the generated NCP load module.

Specific rules and guidelines to be observed in coding user block-handling routines are as follows:

- All registers may be used in a user block-handling routine. Before passing control to a user routine, the network control program saves all registers; when receiving control from the routine, the NCP restores all registers.
- At entry to a user block-handling routine, register 2 points to the queue on which the block being handled is enqueued.
- A POINT (2), (3) macro instruction causes the address of the block at the head of the queue to be returned in register 3.
- You may use a SCAN macro instruction to scan the text in chained NCP buffers containing the block being processed.
- You may use the DEQUE, ENQUE, and INSERT macros to dequeue, enqueue, or insert the block whose address was returned by the POINT macro instruction.
- You may use the LEASE macro to obtain NCP buffers; you must release any buffers thus obtained with a RELEASE macro.
- Use a SYSXIT macro to return control from the user block-handling routine to the network control program.
- If a user block-handling routine is to be executed for more than one BSC or start-stop station or line, code it such that it is serially reusable.
- The ACF/NCP Logic manual describes the macro operand formats and gives details for use of supervisory macros.
- If the user routine changes the amount of message text accompanying a BTU (basic transmission unit), the routine must accordingly update the BCUTLEN field of the BTU and the data count fields of the buffer prefix areas.
- User block-handling routines should not modify any part of the first 34 bytes (the BTU) of a header buffer or the first 4 bytes of any other buffer.
- Logic errors encoutered in user block-handling routines can cause the NCP to end abnormally (abend). For many kinds of logic errors, a dump listing of the NCP will reveal: (1) The level 5 instruction address register (IAR—register 0) will point within the user block-handling routine. (2) Bit 4 of storage location X'685' (indicating block handlers in execution) will be on. (3) The abend code will appear at location X'760'. (Abend codes are described in the *ACF/NCP Program Reference Summary*).

Associating Block-Handling Routines with Stations

The requirements of the application determine how the NCP is to process messages before sending them to the network or the access method. The requirements may vary for different stations or for different components of a station. You may wish, for example, to provide the text-correction function for messages entered from a terminal keyboard but not for messages received from a paper tape reader. Or, you may wish to insert time and date information in messages received from station A but not in those received from station B.

Network control program generation macro instructions provide a means of grouping individual block-handling routines into block handlers and for combining block handlers into block-handler sets. Block-handler sets can then be associated with individual stations or station components.

Each block handler within a set can be executed at a different logical point in the flow of message data through the NCP. For instance, one block handler in the set can be executed immediately upon arrival of a message from the host processor before the NCP has obtained the use of a communication line for transmitting the message to the station. Another block handler in the same set can include routines that process message data from the host processor *after* the program has obtained use of a line. This block handler may also include routines that process message data from a station before the NCP releases the line over which it received the data.

A third block handler in the set may be assigned to process message data received from a station after the program has released the line for use in communicating with another station.

The network control program generation macro instructions for grouping block handling routines into block handlers are STARTBH and ENDBH. A third macro, BHSET, combines block handlers into sets.

To assign block handler sets to stations or station components, you code the name of the set in the BHSET operand of the appropriate TERMINAL, COMP, or CLUSTER macro instruction. In the BHEXEC operands of the same macros you specify which block handlers within the set are to executed at the logical points in the message flow.

Diagnostic and Service Aids for Emulation Mode

Two additional service aids are available with the PEP extension: (1) line trace for emulation lines and (2) dynamic dymp facility. The dynamic dump facility is optional; however, you should include it in the network control program.

Line Trace Facility for Emulation Mode

The emulation mode line trace facility of the program is a service aid that permits detailed analysis of the operation of any communication line currently operating in emulation mode. This facility records operating parameters of a line each time a level 2 interrupt (except bit-service interrupt) or level 3 interrupt occurs for that line. (Level 2 is the program level at which bit service or character service for the communication line is performed. Level 3 is the program level at which the servicing of channel interrupts is performed.) The program accumulates this information within controller storage. The emulation mode line trace, unlike the line trace for network control mode, does not accumulate the trace information in buffers and does not automatically transfer the buffer contents to the host processor. The contents of the controller storage must be dumped to make the line trace records available or dynamically dumped using the dynamic dump utility.

The line trace facility does not interfere with normal operation of the communication line. Performance may diminish somewhat because of the additional processing needed each time a character service or level 3 interrupt occurs for the line or lines being traced. The amount of decrease in performance depends upon how heavily the communications controller is currently loaded and how many lines are being concurrently traced. The line trace facility has no effect on performance except when a line is actually being traced.

Line traces of lines in emulation mode are initiated by the controller operator at the control panel of the communications controller or via the dynamic dump utility. Any number of lines may be traced concurrently.

The line trace facility for emulation mode is always present in a network control program with the PEP extension. The number of lines to be traced and the size of the trace table are specified in the LINETRC operand of the BUILD macro.

Dynamic Dump Facility

The dynamic dump facility is a service aid that transmits communications controller storage contents to the host processor over an emulation subchannel without stopping execution of the NCP. A full storage dump or a dump of the trace table can be obtained. Additionally, the emulation mode line trace can be activated, deactivated, or modified. Portions of controller storage can also be displayed on the operator's console at the host processor.

The DYNADMP operand of the BUILD macro specifies whether the dynamic dump option is to be included in the NCP and specifies the emulation subchannel address(es) over which the controller storage contents are to be dumped. Each channel adapter in the controller can have one host subchannel address assigned for this purpose; the assigned subchannel(s) cannot be used for communicating with any line in the network.

The dynamic dump facility can be used only with a network control program that includes the PEP extension.

Chapter 4. Generation and Catalog Procedures

Program Generation under OS/VS

The control program generation under OS/VS is a two-stage process. The procedure creates a load module that executes in the 3705 according to your particular configuration needs. NCP generation can be done on any host processor. It does not have to be done on the processor that will control the network.

Stage 1 is an assembly job using either the communications controller assembler (CWAX00) or an OS/VS assembler. This stage uses the generation macro instructions that you have coded to prepare a job stream for input to stage 2. The output may be placed on cards, tape, or a direct-access device. The stage 1 output data set contains (1) data constants, (2) macros that cause stage 2 to generate the control tables and conditionally assemble the required program modules, (3) job control statements for stage 2, and (4) linkage editor control statements.

Stage 2 first uses the communications controller assembler (CWAX00) to assemble the control tables and program modules that require conditional assembly. The resulting object modules are placed on the object library. Stage 2 then link-edits these modules, along with other preassembled modules (located in SYS1.OBJ3705), into an NCP load module and places it on the load library. From this library, either the access method loader or the independent loader provided in the system support programs can load the control program into the communications controller. The object library and the load library used in these steps are the libraries specified in the OBJLIB and LOADLIB operands in the BUILD macro.

Stage 2 also produces a *resource resolution table* load module and, if you have coded any block-handling routines, a *block handler set resolution table* load module. Stage 2 then places them on the library specified in the LOADLIB operand. These load modules contain information required by the access method.

Note: If user-written code is included in the NCP via the INCHI operand of the BUILD macro, only two load modules are generated. The block-handler set modules and the user-written modules become part of the NCP load module.

Operator intervention is required between the two stages of program generation. Diagnostic messages produced at the end of stage 1 indicate any errors that may have occurred. If these are serious errors, no job stream is produced. The source statements must be corrected, and stage 1 must be re-executed. If no serious errors occur in stage 1, the operator initiates the second stage, using the stage 1 output as input.

Caution (VTAM Users): Because the VTAM initialization does no validity checking of parameters, it is imperative that the NCP source statements be entirely free of errors before being given to the VTAM initialization procedure. Therefore, the NCP must be reassembled, if necessary, until the stage 1 output listing shows no MNOTE statements having severity codes of 4 or 8.

Figures 4-1 and 4-2 show the contents of the stage 1 input job stream and stage 1 output (stage 2 input) job stream using the communications controller assembler.

The partial generation procedure is the same as for a complete generation. The only difference is that fewer modules are conditionally assembled in a partial generation; in some cases only the control tables are reassembled.

The same source deck used to generate a given control program may be modified and used to generate a different program. Care must be exercised in specifying program and library names associated with the subsequent program. If the new program is to replace the original one and the original object modules in the object library will not be needed, the NEWNAME, LOADLIB, and OBJLIB operands of the BUILD macro in the source deck need not be changed. The subsequent control program load module will be cataloged in place of the original. If both the original and the subsequent load modules will be needed, either change the NEWNAME operand to specify a different name or change the LOADLIB operand to specify a different library.

The object modules associated with the original load module should be saved before submitting the modified source deck to the generation procedure to create a new load module. Failure to save the old object modules will cause them to be replaced in the object library with the modified object modules; this would prevent your later regenerating the original program should you wish to do so. To save the object modules, specify a different data set name in the OBJLIB operand of the BUILD macro before submitting the revised source deck to the subsequent generation procedure.

An alternate method of saving the object modules is by using the OBJQUAL operand of the BUILD macro. This operand allows you to specify a unique name to the object modules of each generation. (See the OBJQUAL operand of the BUILD macro.)

Another step is necessary if the subsequent generation is partial: you must copy the original object modules onto the new data set you specify in the OBJLIB operand of the modified source deck. This step allows a partial generation procedure to obtain the original modules for updating and link editing into the new load module.

Providing User Job Cards

The format of the stage 2 job cards produced when you specify JOBCARD=YES or JOBCARD=MULTI in the BUILD macro is:

```
//NCPGENnn   JOB     1,'NCPSYSGEN',MSGLEVEL=1
```

where *nn* is a sequential identification number provided by the program generation procedure. The job card may be changed by using the OS/VS IEBUPDTE utility program before executing the stage 1 assembly. (Refer to the *OS/VS Utilities* manual, GC35-0005, for information on the IEBUPDTE program.)

When you modify the job statement, the *name* parameter must be *jobname* & SNOA. *jobname* consists of 1 to 6 alphanumeric characters (including @, #). The first character must be alphabetic, @, or #. & SNOA is a counter that is incremented by the program generation procedure to provide unique job names. (See the *OS/VS Job Control Language* manual, GC28-0618, for information on the job statement.)

The following example illustrates how the job statement may be changed:

```
//CHNGJOB    JOB      (G40,060,SG,-,2),name,MSGLEVEL=1
//UPDATE     EXEC     PGM=IEBUPDTE
//SYSPRINT   DD       SYSOUT=A
//SYSUT1     DD       DSN=SYS1.GEN3705,DISP=OLD
//SYSUT2     DD       DSN=SYS1.GEN3705,DISP=OLD
//SYSIN      DD       *
./           REPL     NAME=JOBCARD,LIST=ALL
./           NUMBER   NEW1=100,INCR=100
             PUNCH    '//mygen&SNOA  JOB (81,3,B62),myname,'
             PUNCH    '// MSGLEVEL=(1,1),CLASS=A'
./           ENDUP
/*
```

```
//STAGE1       JOB             MSGLEVEL=1
//STEP1        EXEC            PGM=CWAX00
//SYSLIB       DD              DSN=SYS1.GEN3705
         .
         .
    (JCL statements for assembler)
         .
         .
//SYSIN        DD              *
         .
         .
               BUILD           ...
                               ...
         .                                  Control
         .                                  Program
        (etc.)                              Generation
         .                                  Macro
         .                                  Statements
               GENEND          ...
               END
/*
```

Figure 4-1. OS/VS Generation Stage 1 Input

```
//NCPGEN1  JOB     MSGLEVEL=1,...
//S1       EXEC    PGM=CWAX00,... (Communications controller
                                   assembler)
//SYSLIB   DD      DSN=SYS1.MAC3705
                                    Two assembly job steps
                                    for control tables plus
  (JCL statements for assembler)    one assembly job step
                                    for each program module
                                    requiring conditional
//SYSIN    DD      *                assembly (JOB card is
                                    provided for each
                                    assembly job step only if
  (Data for conditional assembly)   JOBCARD=MULTI is coded
                                    in BUILD macro

/*
//S2       EXEC    PGM=CWAX00,...
//SYSLIB   DD      DSN=SYS1.MAC3705

  (JCL statements for assembler)

//SYSIN    DD      *

  (Data for conditional assembly)
/*

//Sn       EXEC    PGM=IEWL,... (First linkage editor job step)

  (JCL statements for OS/VS linkage editor)

//SYSIN    DD      *

  (INCLUDE statements for linkage editor)

/*
//

  (INCLUDE statements specify object modules obtained from
  SYS1.OBJLIB and object modules obtained from the library
  specified in the OBJLIB operand of the BUILD macro.  The
  load module is placed on the library specified in the
  LOADLIB operand.)
```

Figure 4-2. OS/VS Generation Stage 1 Output (Stage 2 Input)

Program Generation Under DOS/VS

The control program generation under DOS/VS is a three-stage process. The procedure creates a load module that executes in the 3705 according to your particular configuration needs. NCP generation can be done on any host processor. It does not have to be done on the processor that will control the network.

Stage 1 of the generation procedure is a series of assembly jobs using the communications controller assembler (IFZASM) to prepare a job stream (sequential file) for input to stage 2. The file may be placed on cards, tape, or a direct-access device. The stage 1 output (stage 2 input) contains (1) data constants, (2) macros that cause stage 2 to generate the control tables and conditionally assemble the required program modules, (3) job control statements for stage 2, and (4) an assembly step that punches stage 2 statements.

Stage 2 assembles the control tables and those program modules that require conditional assembly; it then punches job control and linkage editor control statements.

Stage 3 catalogs the tables and modules assembled in stage 2 and link-edits them into a load module. This module is placed on the core image library. From there the CSERV utility must be used to move it to a user-defined file.

In addition to the load module produced by the linkage editor, unresolved external references may also be produced. These references will not cause program execution problems. See the section "Unresolved External References" in this chapter for additional information.

The VTAM loader facility or the independent loader utility can now obtain the load module from the file and load it into the communications controller.

Stage 3 also produces a *resource resolution table* load module and, if you have coded any block-handling routines, a *block handler set resolution table* load module. These load modules contain information VTAM needs for its initialization process; VTAM must obtain them from the library where they were placed in stage 3.

Note: If user-written code is included in the NCP via the INCHI operand of the BUILD macro, only two load modules are generated. The block-handler set modules and the user-written modules become part of the NCP load module.

Operator intervention is required between the stages of program generation. Diagnostic messages produced at the end of each stage indicate any errors that may have occurred. If these are serious errors, no job stream or partial job stream is produced. The source statements must be corrected, and the stage must be re-executed. If no serious errors occur, the operator initiates the next stage, using the output of the previous stage as input.

CAUTION
Because the VTAM initialization does no validity checking of NCP parameters, the NCP source statements must be entirely free of errors before being given to the VTAM initialization procedure. Therefore, stage 1 of the network control program must be reassembled, if necessary, until the output listing shows no MNOTE statements having severity codes of 4 or 8.

Figures 4-3, 4-4, and 4-5 show the contents of each input job stream using the communications controller assembler.

The partial generation procedure is the same as for a complete generation. The only difference is that fewer modules are conditionally assembled in a partial generation; in some cases, only the control tables are reassembled.

Partial generation is possible only if the relocatable library previously used during the complete generation procedure is available. You should, therefore, always save this library and the stage 1, stage 2, and stage 3 assembly listings produced by the complete generation.

Providing User Job Cards

The format of the stage 2 job cards produced when you specify JOBCARD=YES or JOBCARD=MULTI in the BUILD macro is:

```
// JOB  module name
```

If you specify JOBCARD=MULTI, you may provide different job cards before initiating the program generation procedure by submitting the following job:

```
// JOB      CHGJOBCD
// EXEC     ESERV
   GENEND
   DSPCH    F.ASMJCL
)  COL 73, 4
)  REP 1228
   PUNCH    '// JOB name'
)  END
/&
```

This job creates a new ASMJCL macro. You must then assemble the macro with IFZASM and catalog it.

Only the second and any subsequent job cards may be changed; the format of the first job card is unchanged.

```
//  JOB      jobname
//  PAUSE               (Before executing stage 1, assign appropriate
                        source statements and private relocatable libraries)
//  EXEC     IFZASM

    BUILD

    •                   NCP generation
    •                   macro
    •                   statements
    GENEND
    END
/&
```

Figure 4-3. DOS/VS Generation Stage 1 Input

This input job stream is produced automatically by stage 1 of the generation procedure.

```
//   JOB      jobname
//   PAUSE              (Before executing stage 2,
                         assign appropriate libraries)
//   OPTION   DECK
//   EXEC     IFZASM
     PUNCH    '// JOB   jobname'
     PUNCH    '// EXEC  MAINT'
     PUNCH    ' CATALR  module name'

       (Source code for conditionally assembled modules)
     END
/*
//   EXEC     IFZASM
     PUNCH    ' CATALR module name'

       (Source code for conditionally assembled modules)
     END

/*
     •
     • (Other conditional assemblies as in above step)
     •

//   EXEC     IFZASM
     PUNCH    ' CATALR  INITINCS'
     PUNCH    ' INCLUDE ...' (INCLUDE statements for UBHR
                              modules, if specified)
     PUNCH    ' INCLUDE ...' (INCLUDE statements for initial-
                              ization routines and tables)
     PUNCH    ' CATALR  LOADINCS'
     PUNCH    ' ACTION  MAP,NOAUTO'
     PUNCH    ' PHASE   NCP001,+0'
     PUNCH    ' INCLUDE ...' (INCLUDE statements for remainder
     •                 •      of program modules)
     •                 •
     •                 •
     •                 •
     PUNCH    '/*'
     PUNCH    '// OPTION CATAL' (If program includes UBHRs,
                                 the option is LINK)
     PUNCH    ' INCLUDE LOADINCS'
     PUNCH    ' INCLUDE INITINCS' (Omit this statement if
                                   program contains UBHRs)
     PUNCH    '// EXEC  LNKEDT'
     PUNCH    '/&'
     END
/&
```

Figure 4-4. DOS/VS Generation Stage 2 Input

This input job stream is produced automatically by stage 2 of the generation procedure.

```
//   JOB        jobname
//   PAUSE      (Before executing stage 3,
                 assign appropriate libraries)
//   EXEC       MAINT
     CATALR     module name
        .
        . (Object code)
        .
     CATALR     module name
        .
        . (Object code)
        .
        (etc.)
        .
        .
        .
     CATALR     INITINCS
     INCLUDE    ...(INCLUDE statements for UBHR modules, if
                    present)
     INCLUDE    ...(INCLUDE statements for initialization
                    routines and tables)
     CATALR     LOADINCS
     ACTION     MAP,NOAUTO
     PHASE      phasename,+0  (INCLUDE statements for
     INCLUDE    module-1       remainder of
     INCLUDE    module-2       object modules)
        .
        .
        .
     INCLUDE    module-n
/*
//   OPTION     CATAL  (If program includes UBHRs, the option
                        is LINK)
     INCLUDE    LOADINCS
     INCLUDE    INITINCS (This statement omitted if program
                          contains UBHRs)
//   EXEC       LNKEDT
/&
```

Figure 4-5. DOS/VS Generation Stage 3 Input

Including User-Written Modules

To include user-written code in the NCP load module, the code must be pre-assembled and placed on the appropriate object library. If you are operating under DOS/VS, the object modules must be placed on the relocatable library. If your system is OS/VS, the object modules must be placed on the NCP object library (SYS1.OBJ3705) or a user object library. Modules on SYS1.OBJ3705 are in load module format. (See the USERLIB operand in the BUILD macro.)

An INCLUDE statement must be coded for each object module. These INCLUDE statements are then separated into the following categories:

1. Level 2 and 3 code that must reside in the lower 64K of controller storage.
2. Level 2 and 3 code that may reside anywhere in storage.
3. Code that is not level 2 or 3 and must reside in the lower 64K of controller storage.
4. Code that is not level 2 or 3 and may reside anywhere in storage.
5. Initialization code that is overload by the buffer pool when initialization is complete.

Place the INCLUDE statements on the macro library that contains the user control block source modules. You may place all of the INCLUDE statements within one category in one member of the macro library.

On an OS/VS system, each CSECT name in the user-written modules must be identified on a linkage editor ORDER statement. These ORDER statements must be placed on the macro library in the same manner as the INCLUDE statements. That is, you may place all of the ORDER statements within one of the previously named categories in one member of the macro library. Stage 1 of the generation procedure creates an INCLUDE statement for each member that contains an ORDER statement and inserts it into the proper place in the linkage editor job stream.

Any CSECT not specified on a linkage editor ORDER statement will be placed at the end of the load module and will be overlaid by the buffer pool once the initialization procedure is complete.

The names of the members that contain the INCLUDE and ORDER statements in the macro library must then be specified in the appropriate operand in the GENEND macro. (See the GENEND macro in Chapter 5.)

Including user-named control blocks in the NCP stage 2 generation causes an exposure to a duplicate label problem. The situation pertains only to the control block assembly, not to the assembly of individual user modules. It exists because the NCP was not designed with this inclusion in mind, and prefixes were not reserved for label used in the assembly. In order to eliminate this exposure, you should not use any of the following prefixes:

$	ACB	ATB	BCB	BCH
BCO	BCT	BCU	BOQ	CCB
CM	CRP	CTB	CX	DAE
DRS	DVB	DVI	DVQ	ICW
IOB	IRN	LCB	LCS	LCW
LGT	LKB	LXB	MDR	OLL
PAD	PCB	RCV	RG	RH-digit
RU-digit	SCB	SYS	STQ	TH
TVS	UIB	UNASGN	UI	X
CY	RN-number	R-number	ATP	PIU

The most effective solution would be to establish your own unique prefix for all of your control blocks.

pecial Considerations for DOS/VS Link-Edits

When link-editing under DOS/VS with user-written code included in the NCP load module, it is necessary to perform one or two temporary link-edits before the final link-edit in order to obtain a load module for the core image library. The first link-edit is performed using modules that must reside in the low 64K of storage. The resulting load module from the first link-edit resides in the relocatable library. If the user has incorporated, into the first link edit, user written "low core" CSECTS then, the ALIGN macro needs to be assembled using the size of the linked module as the value of the HICORE operand. The ALIGN macro creates a dummy CSECT which causes a link-edited module to be padded to the next 2K boundary. If user code was included in the first link edit, the use of the ALIGN macro becomes necessary, since there is no way of determining the size of the user's CSECTs. A first or subsequent link edit must be performed, using as input the "low core" modules, and the first pad module, if applicable. The output from this link edit resides in the relocatable library. The ALIGN macro is assembled using the size of the link edited module as the HICORE operand value. The dummy CSECT, which is produced by the assembly of the ALIGN macro, is created so that user "high core" modules will begin on a 2K boundary. The modules in "high core" include initialization plus user block handler routines. The final link edit is performed, using as input the "low core" modules, the first pad CSECT, if applicable, the SAT, the second pad CSECT and the "high core" modules. The load module obtained from the final link edit is cataloged in the core image library. An explanation of the link edit process and the name(s) of the pad module(s) can be found in the link edit portion of the STAGE-1 listing. The format of the ALIGN macro is:

ALIGN HICORE=value

where *value* is the hexadecimal *hicore* value found in CXTEND of the stage 3 link edit. The following job stream is used to create the dummy object module.

```
//     JOB name
//     OPTION DECK
//     EXEC IFZASK
       ALIGN HICORE=value
       END
/&
```

Unresolved External References

Due to the many options available when generating a network control program, unresolved external references can appear in the linkage editor listings. These references are the result of certain options' not being selected. The following list shows external references that may appear in your listing as unresolved. If any of these references do appear and you have not selected the option they apply to, the reference should be ignored.

	Unresolved	*references:*
For initial test and phase 1 loader:	SEL01	OLT03
	SEL2	OLT04
	SELFF	OLT05
	OLT00	OLT06
	OLT01	

	Unresolved references for: *Type 1 Channel Adapter*	*Type 4 Channel Adapter*
If trace option is not specified:	CYATRCEI	CYATRCEI CYETRCRS CYETRCSP
If trace option is not specified and dynamic dump option is specified:	CYASETRC CYATABLE	CYASETRC CYATABLE
If trace option is not specified and BSC terminals are specified:		CYATRCL2 CYETRCL2
If dynamic dump option is specified:	$DSCCB	
If dynamic dump option is not specified:	CYADSTRT	CYADSTRT
If panel test option is not specified:	CYAPANLT	CYAPANLT
If panel test option is specified and no EBCDIC lines are serviced by a type 2 scanner:		CYAEBCDT
If panel test option is specified and no ASCII lines are serviced by a type 2 scanner:		CYAASCDT
If panel test option is specified and no ASCII lines are serviced by a type 2 scanner:		CYAASCDT
If only one type 4 channel adapter is specified:		CYECHCB2 CYECHVT2

	Unresolved references for: *Type 1 Channel Adapter*	*Type 4 Channel Adapter*
If panel test option is specified and no BSC terminals are specified:	CYAEBCDT	
	CYAASCDT	
If panel test option is specified and no start-stop terminals are specified:	CYAXTABL	
If start-stop terminals are specified but no display terminals are specified:	CYAATDA4	CYAATDA4
	CYAB28CL	CYAATDA5
	CYAB2848	CYAB28CL
		CYAB2848
If start-stop terminals are specified but no teletypewriter (83B3, 115A) or TWX terminals are specified:	CYATDONE	CYATSTYE
		CYASRCH
If start-stop terminals are specified but no DELAY or QUIETCT operand is specified:	CYADAT1	CYADAT1
	CYADAT2	CYADAT2
If no start-stop terminals are specified:	CYABARP1	CYABARP1
	CYABTDA0	CYABTDA0
	CYACBKPL	CYACBKPL
	CYACBRES	CYACBRES
	CYACPOLS	CYACPOLS
	CYACPRES	CYACPRES
	CYACRDCL	CYACRDCL
	CYACREAS	CYACREAS
	CYACSEAS	CYACSEAS
	CYACWRIS	CYACWRIS
	CYAMTBFR	CYAMTBFR
	CYAQUIET	CYAQUIET
	*CYASPCFA	CYASTPER
	*CYASPCFB	CYATDONE
	*CYASPCF8	CYATRN
	*CYASRCVT	SYATSTYE
	CYASTPER	
	CYATRN	
	*CYAXSSTT	
*These modules are unresolved only if a type 1 scanner is specified		
If BSC terminals are specified:		
If no type 2 scanner is specified:		CYARARS0
		CYATAPD0
		CYATAX10
		CYATBSWR
		CYATSTMW
		CYATXDA0

	Unresolved Type 1 Channel Adapter	*references for Type 4 Channel Adpater*
If no type 3 scanner is specified:		CYEABRTW
		CYEPRPRC
		CYERCVN
		CYERCVN1
		CYERCVP
		CYERCVPS
		CYERCVP1
		CYETXEND
		CYEXITB
		CYEXMITN
		CYEXMSYN
		CYEXMTEN
		CYEXPOLL
		CYEXTEND
If no BSC terminals are specified:		
	*CYABPCFA	CYABSHIO
	*CYABPCF8	CYABSTOP
	CYABSHIO	CYACADPB
	CYABSTOP	CYACPOLB
	CYACADPB	CYACPREB
	CYACPOLB	CYACREAB
	CYACPREB	CYACSEAB
	CYACREAB	CYACSETB
	CYACSEAB	CYACWRIB
	CYACSETB	CYARARS0
	CYACWRIB	CYATAPD0
	*CYAPCF45	CYATBSPL
	CYARARS0	CYATBSPR
	*CYARCDTA	CYATBSRD
	CYATAPD0	CYATBSSM
	CYATBSPL	CYATBSWR
*These modules are unresolved	CYATBSPR	CYA
only if a type 1 scanner	CYATBSRD	CYA
is specified	CYATBSSM	CYE
	CATBSWR	CYE
	CYATSTMW	CYE
		CYE
		CYE
		CYE
If no ALC lines (LNCTL=ALC)		CYE
are specified:		CYE
		CYE

Notes:

1. External reference TM598 is defined in the CYASL210/310/320 modules.
2. External reference CYAATDA5 is defined in the CYADSP10/20 modules.

Coding Sequence for Generation Macros

This section shows the required sequence of NCP generation macro instruction statements in the stage 1 input job stream. (See Appendix I for examples of several sample network programs.)

There are five distinct groups of program generation macro instructions. These groups must be submitted in the following sequence:

1. System Macro Instructions
2. Configuration Definition Macro Instructions
3. Network Configuration Macro Instructions
4. Block Handling Macro Instructions
5. Generation Delimiter Macro Instructions

The block handling macro instruction group is valid only if start-stop and/or BSC lines are included in the network configuration macro instructions.

The following charts show the names of all macros within a group and shows:

- Whether the macro is required or optional, and the number that must or can be coded.
- The operands are always required.
- The position of the macro within the group, when a specific sequence is required.

Only those operands that must always be coded are shown; the configuration and characteristics of the data communications network and the procedural options needed determine which other operands are needed. Refer to Chapters 2, and 3 for explanations of the characteristics and options; refer to Chapter 5 for descriptions of each of the macro instructions and operands.

(VTAM Users Only): The PCCU, VIDLIST, and VTERM macros are VTAM-only macros. They provide information only to the VTAM initialization process. Their presence in the network control program generation deck is not required (but is permissible) for the generation procedure, but must be included when the same deck is used as input to the VTAM initialization process. PCCU is always required; whether VIDLIST and VTERM are required depends upon VTAM application requirements. See the VTAM *System Programmer's Guide* for details of the meaning and use of these macros.

1-SYSTEM MACRO INSTRUCTIONS

Usage	*Macro*	*Required Operands*	*Macro Position*
Required for VTAM initialization: 1	PCCU (VTAM Users Only)	(See VTAM System Programmer's Guide for required operands)	VTAM-only macro: must precede BUILD macro for VTAM initialization.
Required: 1	BUILD (for TYPGEN=NCP or TYPGEN=PEP)	LOADLIB= (OS/VS only) MAXSUBA= MEMSIZE= OBJLIB= (OS/VS only) SUBAREA= TYPGEN=	first macro in NCP source statements
Required: 1	SYSCNTRL	OPTIONS=	directly following BUILD macro
Optional: 1 for each Network Addressable Unit	NCPNAU	NAUFVT=	directly following SYSCNTRL macro

2-CONFIGURATION DEFINITION MACRO INSTRUCTIONS

Usage	*Macro*	*Required Operands*	*Macro Position*
Required: 1 for each access method to communicate with local NCP over channel. Not required for remote NCP.	HOST	INBFRS= MAXBFRU= UNITSZ=	macros in this group (2) may appear in any sequence except as indicated for MTALCST and MTALIST macros
Required: 1 for each scanner in controller	CSB	SPEED=	
Optional: 1 for each list of ID sequences to be checked by NCP	IDLIST	IDSEQ=	
Optional: 1 for each list of ID sequences to be checked by VTAM (required only for VTAM initialization)	VIDLIST (VTAM Users Only)	(See VTAM System Programmer's Guide for required operands)	
Optional: 1 for each LCST entry to be defined	MTALCST	GROUP= SPEED=	must appear in one sequence
Optional: 1 for each list of line control types for MTA lines	MTALIST	LCTYPE=	must appear in one sequence
Optional: 1 for each unique combination of line control types and transmission codes for MTA lines	MTATABL	LCST=	
Optional: 1 only, for all 1050s that may call controller over any MTA line	MTAPOLL	POLL=	
Optional: 1 for each dial set to be defined	DIALSET	LINES	
Optional: 1 or more for all logical units to be associated with SDLC stations on switched links.	LUPOOL	NUMBER=	

Optional: 1 each for Dynamic Reconfiguration	LUDRPOOL PUDRPOOL		
Optional: 1 for each path from NCP being defined to a non-adjacent subarea.	PATH	ADJSUB= DESTSUB=	must precede first GROUP macro
Optional: 1 for each SDLC selection table needed.	SDLCST	GROUP=	must precede first GROUP macro

3-NETWORK CONFIGURATION MACRO INSTRUCTIONS

Usage	*Macro*	*Required Operands*	*Macro Position*
Required: 1 for each physical line group	GROUP		at beginning of line group definition
Required: 1 for each line within group	LINE	ADDRESS= SPEED=	directly following GROUP macro or another line definition
Required: at least 1 if line uses multipoint discipline; more as needed to accommodate all stations (omit if LINE macro is coded in remote NCP and represents SDLC link to remote controller)	SERVICE	ORDER=	directly following LINE macro

For all start-stop and all BSC stations except IBM 2972, 3271, 3275:

Required: 1 for each station to be identified to NCP; represents first or only input component and/or first or only output component	TERMINAL	TERM=	directly following SERVICE macro (if present) or LINE macro
Optional: Used only for call-in MTA terminals to be associated with specific VTAM application programs.	VTERM (VTAM Users Only)	(see VTAM System Programmer's Guide for required operands)	directly following TERMINAL macro in which TERM=MTA, CTERM=YES are coded.
Required: 1 for each additional input or output component (one COMP macro can specify both one input and one output component)	COMP		directly following the TERMINAL macro or another COMP macro

OR
For IBM 2972, 3271, 3275:

Required: 1 for each BSC cluster-type station (2972, 3271 [BSC], 3275 [BSC])	**CLUSTER**	**directly following SERVICE macro (if present) or LINE macro; or following another station definition**
Required: 1 for each terminal address on cluster control unit	**TERMINAL TERM=**	**directly following CLUSTER macro or another station definition**

OR
For SDLC stations:

Required: 1 for each SDLC station (e.g., 3270 [SDLC] 3600, 3650, 3660, 3767, 3770])	**PU PUTYPE=**	**directly following SERVICE macro or another PU macro**
Required: 1 for each logical unit associated with physical unit, if physical unit is on nonswitched link.	**LU LOCADDR=**	**directly following PU macro or another LU macro associated with same cluster.**

OR
For 3705:

Required: 1 for each 3705.	**PU PUTYPE=**	**directly following SERVICE macro, when coded in primary NCP for link; directly following LINE macro, when coded in secondary NCP for link.**

4-BLOCK HANDLING MACRO INSTRUCTIONS (OPTIONAL)

Usage	*Macro*	*Required Operands*	*Macro Position*
Required: 1 for each block handler to be defined	STARTBH		at beginning of block handler
Optional: 1 allowed in each block handler	EDIT		preceding DATETIME, if DATETIME present
Optional: 1 allowed in each block handler	DATETIME		following EDIT, if EDIT present
Optional: no limit on number allowed	UBHR	NAME=	anywhere between STARTBH and ENDBH: may be inter-mixed with EDIT and DATETIME macros
Required: 1 for each block handler	ENDBH		at end of block handler
Optional: 1 for each block handler set needed (limit 255)	BHSET		following all block handler macros

5-GENERATION DELIMITER MACRO INSTRUCTION

Usage	*Macro*	*Required Operands*	*Macro Position*
Required: 1	GENEND		last macro in NCP source statements

Chapter 5: NCP Generation Macro Instructions

This chapter gives detailed descriptions of the macro instructions used to define a network control program. The order of presentation is arranged in a logical order of progression for defining a complete program. These coded macro instructions become the input to stage 1 of the NCP generation procedure.

The sequence of the macros, when they are assembled, may vary with the network configuration. Also, some macros may be coded only once, while others may require more than one. See Chapter 4 for the sequence requirements and conventions.

Figure 5-1 lists all the macro instructions in the order that they appear in this chapter. The figure may be used as a quick reference to determine which macros apply to your network configuration.

Macro Instruction Coding Conventions

The following conventions are used in the descriptions of the macro instructions.

- Capital letters represent values you code directly, without change.
- Small letters represent parameters for which you must supply a value.
- Brackets [ ] enclose operands or symbols that are either *optional* or *conditional.*

An optional operand is one that you may code or omit, independent of other operands. Depending on the operand, omitting it may cause a default value to be given. The assumed value is always given.

A conditional operand is one that you may need to code or to omit, depending on how you code (or omit) other operands in the same or other macros.

- Braces { } indicate that you must choose from the enclosed items.
- An ellipsis (...) indicates that you may code a sequence of values, within parentheses.
- An underlined value represents the default value of the operand; that is, the network control program will use that value if you omit the operand.
- Parentheses ( ) must enclose a sequence of values coded in one operand.
- Quotes must be used to frame a character string if it can be confused with a keyword value for an operand. This is to avoid preventing your use of certain names as symbols.

Symbols coded in the name field of a macro instruction must not begin with a $ character.

Within the macro instruction formats and descriptions, operands that are always required appear first, in alphabetical order. Then, operands that are conditional or optional follow in alphabetical order. These are enclosed in brackets—[].

Data set (file) names must begin with an alphabetic character or $, @, or #.

Macros	*SDLC*	*BSC*	*S/S*	*VTAM only*	*Comments*
PCCU				X	
BUILD	R	R	R		
SYSCNTRL	R	R	R		
NCPNAU	X				Programmed Resources only
HOST	R	R	R		Not required for remote NCP
CSB	R	R	R		One for each scanner
IDLIST		X	X		
VIDLIST		X	X	X	
LUPOOL	X				
LUDRPOOL	X				
PUDRPOOL	X				Dynamic reconfiguration only
PATH	X				
SDLCST	X				
DIALSET		X	X		
MTALCST			X		Multiple terminal access only
MTALIST			X		Multiple terminal access only
MTAPOLL			X		Multple terminal access only
MTATABL			X		Multiple terminal access only
GROUP	R	R	R		
LINE	R	R	R		
SERVICE	X	X	X		
PU	X				
LU	X				
CLUSTER		X			
TERMINAL		X	X		
VTERM		X	X	X	
COMP		X	X		
STARTBH		X	X		Block handling routines only
BHSET		X	X		Block handling routines only
DATETIME		X	X		Block handling routines only
EDIT		X	X		Block handling routines only
SPAFPT2		X	X		Block handling routines only
UBHR		X	X		Block handling routines only
ENDBH		X	X		Block handling routines only
GENEND	R	R	R		

Figure 5-1. NCP Generation Macros

System Definition Macro Instructions

The system definition macro instructions—PCCU, BUILD and SYSCNTRL—to be used in defining a network control program. (PCCU is a VTAM-only macro instruction.)

PCCU Macro Instruction (VTAM Only)

The PCCU macro instruction identifies for VTAM the 3705 communications controller in which the NCP being defined is to be loaded and executed. The macro must precede the BUILD macro at the beginning of the NCP generation input deck before the deck is provided to the VTAM initialization process. You may include the PCCU macro(s) in the deck provided to the NCP generation procedure, but this is not required.

See the *ACF/VTAM Installation* manual for a complete description of the macro and its operands.

The format of the PCCU macro is:

```
[symbol]        PCCU            [operands]
```

Operands

```
AUTODMP=
AUTOIPL=
AUTOSYN=
BACKUP=
CONFGDS=
CONFGPW=
CUADDR=
DUMPDS=
INITEST=
MAXDATA=
NCPLUB=
OWNER=
RNAME=
SUBAREA=
```

BUILD Macro Instruction

The first macro instruction in the program source statements is BUILD. This macro specifies:

- The type of controller (3705-I or 3705-II) that is to execute the NCP and whether the program is to control the 3705 as a local or a remote controller.
- The controller storage size.
- The size of buffers in the buffer pool.
- The name that is to be assigned to the network control program and resource resolution table load modules.
- The type, number, and status (active or inactive) of channel adapters in the communications controller.
- The subarea address to be assigned to the NCP being defined.
- The upper limit of the range of subarea addresses assigned to subareas in the network.
- Certain optional facilities that may be included in the NCP.
- Certain program generation options that may be desired.
- The names of program data sets used in the generation process.
- Whether a complete or a partial program generation is to be performed.

The format of the BUILD macro is:

```
[symbol]        BUILD        operands[,operands]
```

Operands

```
LOADLIB=dsname,
MAXSUBA=n,
MEMSIZE=n,
OBJLIB=dsname,
SUBAREA=n,
        {NCP   }
TYPGEN={NCP-R}
        {PEP   }
[,ABEND={YES} ]
         {NO }
[,ANS={YES} ]
       {NO }
           {SHORT}
[,ASMXREF={YES   } ]
           {NO    }
[,BFRS={size}]
        {60  }
[,CA=(adapter0[,adapter1[,adapter2[,adapter3]]])]
[,CATRACE={(YES[,{count}])}]
           {       { 10  }   }
           {NO               }
[,CHANTYP=type]
[,CONDASM={TABLE         }]
           {(value1,...)}
[,CSMHDR=chars]
[,CSMHDRC=chars]
[,CSMSG=chars]
[,CSMSGC=chars]
[,CUID=chars]
[,DIALTO={count} ]
          {60.0 }
[,DR3270={NO }]
          {YES}
[,DSABLTO={count} ]
           {3.0  }
                   {addr1}     {addr2}
           {(YES,[{NSC   }][,{NSC   }])}
[,DYNADMP={       {NONE  }    {NONE  }  }]
           {NO                            }
```

Operands

```
[,ENABLTO={count} ]
          {2.2   }
[,ERASE={YES}]
        {NO }
[,HICHAN=([addr1][,addr2])]
[,ITEXTTO={count}]
          {NONE }
          {YES  }
          {YES  }
[,JOBCARD={NO   } ]
          {MULTI}
[,LESIZE=size]
[,LINETRC=([YES][,lines][,entries]])]
[,LOCHAN=([addr1][,addr2])]
[,LTRACE={count} ]
         {2     }
[,MACLIB=dsname]
[,MAXSSCP=count]
        {3705-2}
[,MODEL={3705  } ]
[,MTARTO={count}]
         {1.0  }
[,MTARTRY={count}]
          {0    }
[,NCPCA=([status0],[status1],[status2],[status3])]
          {symbol}
[,NEWNAME={NCP001}]
          {PEP001}
[,OBJQUAL={symbol}]
          {CG    }
[,OLT={YES}]
      {NO }
[,OPCSB2={YES}]
         {NO }
[,OUTPUT=([asm],[post-asm],[link])]
[,PARTIAL={YES}]
          {NO }
[,PRTGEN=({NOGEN}[,{NOGEN}])]
          {GEN  }  {GEN  }
```

Operands

```
[,PWROFF={YES}]
         {NO }
          {symbol}
[,QUALIFY={NONE  }]
          {SYS1  }
[,REMLOAD={YES}]
          {NO }
[,RESOEXT={count}]
          {0    }
          {12}
[,SLODOWN={25}]
          {50}
[,TIME=integer]
        {NO                }
[,TRACE={(YES[,{ size}])}]
                { 10 }
[,TRANSFR=count]
[,TYPSYS={OS }]
         {DOS}
[,TWXID=([inchars][,outchars])]
[,UNIT=unit type]
[,USERLIB=dsname]
[,UT1=dsname]
[,UT2=dsname]
[,UT3=dsname]
[,XBREAK={integer}]
         {NONE    }
[,XITB={YES}]
       {NO }
```

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

LOADLIB=dsname

(OS/VS only)

Specifies the name of a partitioned OS/VS data set that will contain the NCP load module and resource resolution table module. (The data set name may be qualified, as determined by the QUALIFY operand of this macro.) The unqualified name may be up to 8 characters long; the first character must be alphabetic or $, #, or @. This data set must be cataloged.

The LOADLIB operand is required for generation under OS/VS and is not applicable for generation under DOS/VS.

MAXSUBA=n

Specifies the upper limit of the range of subarea addresses used within the *network* you are defining. The same value is required for all MAXSUBA operands in all network control programs that may be active at the same time.

VTAM Note: This value must be the same as specified in the MAXSUBA VTAM start parameter.

The maximum subarea address value is always a power of 2 minus 1, within the range of 3 through 255 (3,7,15,31,63,127,255). If you specify a value that is not one of those listed, the generation procedure rounds the specified value to the next higher such value. (For example, any value between 16 and 30 will be rounded to 31.)

The total number of resources that can be associated with any subarea address depends on the value of MAXSUBA, as follows:

Value of MAXSUBA	*Maximum Number of Resources Possible*
3	16382
7	8190
15	4094
31	2046
63	1022
127	510
255	254

Note: Specifying an unnecessarily high value for *n* will waste NCP storage. (Storage is assigned for all subarea addresses whether used or not.)

For example, assume that the network includes one host access method, assigned subarea address 1; two local communications controllers, assigned subarea addresses 2 and 3; and four remote controllers, assigned addresses 4 through 7. The highest address being 7, you would specify an upper limit of at least 7. If you wish to allow for adding more access methods or controllers to the network, you would specify a value greater than 7. A value of 31 would allow up to 30 access methods and controllers (addresses 2 through 31) to be included in the network.

This operand is required.

MEMSIZE=n

Specifies the storage size of the controller in K (1,024) bytes.

The value of *n* must be one of the following:

For 3705-I (MODEL=3705): 80, 112, 144, 176, 208, or 240
For 3705-II (MODEL=3705-2): 96, 128, 160, 192, 224, 256, 320, 384, 448, or 512

This operand is required.

If the controller storage size is 64k or less, specify any one of the above values for n. If the storage size is greater than 64K, specify the exact size. Do not include the "K" when specifying storage size.

OBJLIB=dsname

(OS/VS only)

Specifies the name of a partitioned OS/VS data set that will contain the output from all assemblies during stage 2 of the generation procedure. (The data set name may be qualified, as determined by the QUALIFY operand of this macro.) The unqualified name may be up to 8 characters long; the first character must be alphabetic or $, #, or @. This data set must be cataloged.

This operand is required for generation under OS/VS and is not applicable for generation under DOS/VS.

SUBAREA=n

Specifies the subarea address assigned to the network control program you are defining.

The minimum subarea address is SUBAREA=1. The maximum address is the value specified in the MAXSUBA operand of this macro.

This operand is required.

```
        {NCP  }
TYPGEN= {NCP-R}
        {PEP  }
```

Specifies (1) whether the program you are defining is to operate the 3705 as a local or as a remote communications controller; (2) whether the program can communicate with a remote network control program; and (3) whether the program will include PEP functions in addition to network control functions. Select the appropriate parameter from the following:

Parameter	*The program: will include emulation functions*	*Can communicate with a remote NCP*	*Will operate the controller as a:*
NCP	No	Yes	Local
NCP-R	No	No	Remote
PEP	Yes	Yes	Local

Note: The TYPGEN operand of a BUILD macro assembled under the non-ACF NCP generation procedure may specify NCP-LR instead of NCP, or PEP-LR instead of PEP. The same program may be reassembled under the ACF/NCP/VS generation procedure without respecifying the TYPGEN operand. An ACF/NCP/VS in which the TYPGEN operand specifies NCP, NCP-LR, PEP, or PEP-LR is able to communicate with a remote NCP.

This operand is required.

```
[ABEND={YES} ]
       {NO  }
```

Specifies whether the abend facility is included in the network control program.

```
[ANS={YES} ]
     {NO  }
```

Specifies whether automatic network shutdown is to be included in the network control program.

Code ANS=YES if you wish the facility to be included. Code ANS=NO if you do not wish it included.

ANS=YES is required if the program you are defining is a local NCP and (1) this program will operate more than one channel adapter concurrently or (2) any PU macro in this program represents a local network control program [PUTYPE=(4,LOCAL)].

```
         {SHORT}
[ASMXREF={YES  } ]
         {NO   }
```

Specifies whether you want the generation procedure to produce cross-reference listings for the stage 2 program assemblies.

(OS/VS Only): Specify ASMXREF=SHORT if you want a cross-reference table of all symbols that are referred to in the assembly. Any symbols defined, but not referred to, are not included in the table.

Specify ASMXREF=YES if you want a cross-reference table of all symbols used in the assembly. This includes symbols that are defined but never referred to.

Specify ASMXREF=NO if you do not want cross-reference listings produced.

[BFRS={size}]
      {60   }

Specifies the size, in bytes, of the buffers in the network control program buffer pool.

Specify *size* as a multiple of 4 bytes. The minimum size is 56 bytes. If you include the on-line testing facility in the NCP (OLT=YES) the minimum size is 60 bytes. The maximum is 248. The generation procedure rounds the size specified to the next higher multiple of 4 bytes if the value is not a multiple of 4.

The generation procedure adds 12 bytes to each buffer for control use. These 12 bytes are never used to contain message data. If you omit the BFRS operand, the buffer length is 72 (60 + 12 bytes for control use).

Note: Diagnostic programs that communicate with the network control program via VTAM or TCAM (for example, TOTE) may impose restrictions on the values specified for the BFRS operand. See the appropriate manuals for such diagnostic programs for restrictions that may apply.

[CA=(adapter0[,adapter1][,adapter2][,adapter3])

Specifies the type(s) of channel adapter installed in the communications controller in which the program being defined will be executed.

adapter0

Specifies the type of adapter in channel adapter position 0. *adapter0* may be specified as TYPE1, TYPE2, TYPE3, or TYPE4.

Channel adapter position 0 is located in the base module of the controller.

adapter1

Specifies the type of adapter in channel adapter position 1 and indicates whether the adapter is installed in the base module or the first expansion module of the controller.

A type 2 or type 4 channel adapter in position 1 can be installed in either the base module or the first expansion module; a type 3 adapter in position 1 is always installed in the first expansion module.

TYPE2-0 or TYPE4-0 specifies that the adapter is installed in the base module; TYPE2-1 or TYPE4-1 specifies that it is installed in the first expansion module. TYPE2 or TYPE4 specifies the type of adapter but does not indicate the module in which it is installed. If two adapters are installed [CA=(adapter0,adapter1)], the generation procedure assumes that the second adapter is installed in the first expansion module; if three or four adapters are installed [CA=(adapter0,adapter1,adapter2) or CA=(adapter0,adapter1, adapter2,adapter3)], the generation procedure assumes that the second adapter is in the base module.

adapter2

Specifies the type of adapter in channel adapter position 2. *adapter2* may only be specified as TYPE4. Code this parameter only if the first two channel adapter positions contain type 4 adapters.

adapter3

Specifies the type of adapter in channel adapter position 3. *adapter3* may only be specified as TYPE4. Code this parameter only if the first three channel adapter positions also contain type 4 adapters.

If you omit this operand and the CHANTYP and NCPCA operands are omitted, CA=TYPE2 is assumed.

If you omit this operand and the NCPCA operand is omitted, and the CHANTYP operand is specified, the assumed values for the CA operand are determined by the CHANTYP operand.

To specify that a remote 3705-II is equipped with one or more channel adapters, you must code the CA operand. Omitting the operand causes the program generation procedure to assume that the remote 3705-II has no channel adapters.

This operand is invalid for a remote network control program (TYPGEN=NCP-R) to be executed in a 3705-I controller.

Figure 5-2 shows how the CA, NCPCA, and CHANTYP operands may be coded for each possible channel adapter configuration.

```
[CATRACE={(YES[,{count}])}]
         {      {10   }  }
         { NO            }
```

Specifies whether the network control program is to include the channel adapter trace facility.

count specifies the number of 32-byte entries the channel adapter trace table is to include. The minimum is 1; the maximum is 255.

If you omit this operand or specify CATRACE=YES without the *count* parameter, the channel adapter trace option is included and the trace table will contain ten 32-byte entries.

[CHANTYP=type]

Specifies the type of adapter to be used to communicate with the host processor in network control mode over a single channel adapter. *type* may be specified as TYPE1, TYPE2, TYPE3, or TYPE4.

If the NCP is to communicate with the host processor in network control mode over more than one channel adapter, omit this operand and code the NCPCA operand. Do not code both the CHANTYP and the NCPCA operands.

The network control program must be loaded over the type of channel adapter specified. The loader disables all adapters, except the one over which the NCP is loaded. (See the description of the CA operand for the possible channel adapter configurations.) The network control program operates only the channel adapter over which it was loaded.

Note: A network control program of an earlier (non-ACF) version in which the CHANTYP operand specifies two parameters [for example, CHANTYP=(TYPE3,TYPE2)] will assemble correctly using the generation procedure for the current (ACF) version. The adapter specified by the second parameter is disabled by the network control program.

Channel Adapter Configuration				CA, NCPCA, and CHANTYP Operands
Base Module		Expansion Module		Note: In the BUILD macro, code the CA operand that represents the installed channel adapter configuration and either (1) one of the NCPCA operands shown below or (2) one of the CHANTYP operands that is appropriate. Do not code both an NCPCA and a CHANTYP operand in the BUILD macro.
Channel Adapter Type:				
1	–	–	–	CA=TYPE1 CHANTYP=TYPE1
2	–	–	–	CA=TYPE2 CHANTYP=TYPE2
3	–	–	–	CA=TYPE3 CHANTYP=TYPE3
4	–	–	–	CA=TYPE4 CHANTYP=TYPE4
1	–	2	–	CA=(TYPE1,TYPE2) CHANTYP=TYPE1 CHANTYP=TYPE2
1	–	3	–	CA=(TYPE1,TYPE3) CHANTYP=TYPE1 CHANTYP=TYPE3
2	–	2	–	CA=(TYPE2,TYPE2) NCPCA=(ACTIVE,ACTIVE) CHANTYP=TYPE2
2	–	3	–	CA=(TYPE2,TYPE3) NCPCA=(ACTIVE,ACTIVE) CHANTYP=TYPE2 CHANTYP=TYPE3
3	–	2	–	CA=(TYPE3,TYPE2) NCPCA=(ACTIVE,ACTIVE) CHANTYP=TYPE3 CHANTYP=TYPE2
3	–	3	–	CA=(TYPE3,TYPE3) NCPCA=(ACTIVE,ACTIVE) CHANTYP=TYPE3

Figure 5-2. CA, NCPCA, and CHANTYP Operands for 3705 Channel Adapter Configurations (1 of 2)

Channel Adapter Configuration				CA, NCPCA, and CHANTYP Operands
Base Module		Expansion Module		
4	2	–	–	*CA=(TYPE4,TYPE2-0) CHANTYP=TYPE4 CHANTYP=TYPE2
4	–	2	–	CA=(TYPE4,TYPE2-1) CHANTYP=TYPE4 CHANTYP=TYPE2
4	–	3	–	CA=(TYPE4,TYPE3) CHANTYP=TYPE4 CHANTYP=TYPE3
4	4	–	–	*CA=(TYPE4,TYPE4-0) NCPCA=(ACTIVE,ACTIVE) CHANTYP=TYPE4
4	–	4	–	CA=(TYPE4,TYPE4-1) NCPCA=(ACTIVE,ACTIVE) CHANTYP=TYPE4
4	4	–	4	*CA=(TYPE4,TYPE4-0,TYPE4) NCPCA=(ACTIVE,ACTIVE,INACTIVE) NCPCA=(ACTIVE,INACTIVE,ACTIVE) NCPCA=(INACTIVE,ACTIVE,ACTIVE) CHANTYP=TYPE4
4	–	4	4	CA=(TYPE4,TYPE4-1,TYPE4) NCPCA=(ACTIVE,ACTIVE,INACTIVE) NCPCA=(ACTIVE,INACTIVE,ACTIVE) NCPCA=(INACTIVE,ACTIVE,ACTIVE) CHANTYP=TYPE4
4	4	4	4	*CA=(TYPE4,TYPE4-0,TYPE4,TYPE4) NCPCA=(ACTIVE,ACTIVE,INACTIVE,INACTIVE) NCPCA=(ACTIVE,INACTIVE,ACTIVE,INACTIVE) NCPCA=(ACTIVE,INACTIVE,INACTIVE,ACTIVE) NCPCA=(INACTIVE,ACTIVE,ACTIVE,INACTIVE) NCPCA=(INACTIVE,ACTIVE,INACTIVE,ACTIVE) NCPCA=(INACTIVE,INACTIVE,ACTIVE,ACTIVE) NCPCA=(ACTIVE,ACTIVE,ACTIVE,INACTIVE) NCPCA=(ACTIVE,ACTIVE,INACTIVE,ACTIVE) NCPCA=(ACTIVE,INACTIVE,ACTIVE,ACTIVE) NCPCA=(INACTIVE,ACTIVE,ACTIVE,ACTIVE) NCPCA=(ACTIVE,ACTIVE,ACTIVE,ACTIVE) CHANTYP=TYPE4

Note: In the NCPCA operand, a comma may be substituted for each ACTIVE parameter; thus, for example, NCPCA=(,INACTIVE,,INACTIVE) is equivalent to NCPCA=(ACTIVE,INACTIVE,ACTIVE,INACTIVE).

*This configuration is invalid if a 3705-II is equipped with both a channel adapter and a remote program loader in the base module.

Figure 5-2. CA, NCPCA, and CHANTYP Operands Valid for Each 3705 Channel Adapter Configuration (2 of 2)

```
[CONDASM={TABLE        }]
         {(value1,...)}
```

Specifies conditionally assembled network control program modules to be reassembled during a partial generation.

This operand is valid only if you specify PARTIAL=YES in the BUILD macro.

TABLE

Specifies that only the modules containing network control program tables are to be reassembled.

`value1,...`

Identifies specific modules to be reassembled. value1,... represents a sequence of 2-digit numbers corresponding to the last two digits of the names of the modules to be assembled. For example, to reassemble modules SYSCG007 and SYSCG00A, you would code CONDASM=(07,0A).

The modules that may be individually reassembled, and the corresponding values to be coded in the CONDASM operand, are:

SYSCG000	00	SYSCG009	09
SYSCG001	01	SYSCG00A	0A
SYSCG002	02	SYSCG00B	0B
SYSCG003	03	SYSCG00C	0C
SYSCG006	06	SYSCG00D	0D
SYSCG007	07	SYSCG00E	0E
SYSCG008	08	SYSCG010	10

The NCP tables are always assembled, regardless of which specific modules you specify in CONDASM=(value1,...).

Appendix B lists the module that must be reassembled for various changes in program functions.

`[CSMHDR=chars]`

(IBM 3270 only)

Specifies the header of the critical situation message to be sent to any IBM 3270 terminals in the network controlled by this NCP. A header must be specified for these terminals if the program is to send critical situation messages. The header must contain the appropriate device control characters and may also include any other valid characters desired. (See the IBM *3270 Component Description* manual, GA27-2749, for the required device control characters.) The header specified by this operand is sent only to 3270 terminals.

Code *chars* as the hexadecimal representation of the EBCDIC characters to be sent.

You may specify up to 127 EBCDIC characters in this operand. If the header required exceeds this length, code a CSMHDRC operand for the remaining characters, up to a combined total of 238 characters.

Note: Although CSMHDR and CSMHDRC may together specify up to 238 characters of header and CSMSG and CSMSGC may together specify up to 238 characters of text, the combined total of header and text specified in these four operands may not exceed 238 characters.

[CSMHDRC=chars]

(IBM 3270 only)

Specifies up to 111 additional characters of header for the critical situation message specified by the CSMHDR operand. This operand is valid only if CSMHDR is specified. See the note on total header and text length under the CSMHDR operand.

[CSMSG=chars]

(start-stop and BSC stations only)

Specifies the text of the critical situation message to be sent to the active start-stop and BSC stations on lines in network control mode before automatic network shutdown occurs. The message will be sent to each active station whose TERMINAL macro specifies CRITSIT=YES. Code *chars* as the hexadecimal representation of the EBCDIC characters desired.

If this operand is omitted, the NCP will not notify stations before automatic network shutdown occurs.

You may specify up to 127 EBCDIC characters in this oeprand. If the message required exceeds this length, code a CSMSGC operand of the remaining characters, up to combined total of 238 characters.

Note: Although CSMSG and CSGMSGC may together specify up to 238 characters of text and CSMHDR and CSMHDRC may together specify up to 238 characters of header, the combined total of header and text specified in these four operands may not exceed 238 characters.

[CSMSGC=chars]

(start-stop and BSC stations only)

Specifies up to 111 additional characters of text for the critical situation message specified by the CSMSG operand. This operand is valid only if CSMSG is specified. See the note on total header and text length under the CSMSG operand.

[CUID=chars]

(BSC stations only)

Specifies the identification characters the NCP sends to BSC stations on switched lines operating in network control mode. Code *chars* as the hexadecimal representation of the EBCDIC characters to be sent. You may specify a maximum of 20 EBCDIC characters. The NCP sends these characters to each station with the CUIDLEN operand coded in the TERMINAL or COMP macro. The characters are sent each time the NCP calls the station or answers a call from the station.

If this operand is omitted, the network control program can verify station ID sequences it receives, but will not send a controller ID sequence in return.

```
[DIALTO={count}]
        {60.0 }
```

Specifies the timeout used by the NCP in detecting the failure of an automatic calling unit's "abandon call and retry" (ACR) signal. Specify the timeout either as an integral number of seconds (DIALTO=30) or to tenths of a second (DIALTO=40.4).

The default value (60 seconds) is recommended unless the system designer recommends a different one.

The maximum timeout you may specify is 1632 seconds.

Note: See the section "Restriction on Number of Time Intervals" following the description of the GROUP macro.

```
[DR3270={NO }]
        {YES}
```

(dynamic reconfiguration only)

Specifies whether the NCP is to include SDLC 3270 Model 11 and 12 terminal support when dynamic reconfiguration is supported and no 3270s are being defined during system generation. This operand should be coded YES if 3270 terminals may be added at a later time through the dynamic reconfiguration function. The default for this operand is YES if a PUDRPOOL macro is coded.

This operand is effective only if dynamic reconfiguration is included in the NCP. If it is not included, the operand is ignored. However, if BNNSUP=3270 is coded in the PU macro, 3270 support is always included regardless of the value coded for this operand.

```
[DSABLTO={count}]
         {3.0  }
```

Specifies the timeout used by the NCP in detecting failure of the "data set ready" signal line of the modem to be turned off when the line attached to the modem is disabled. Specify the timeout either as an integral number of seconds (DSABLTO=5) or to tenths of a second (DSABLTO=7.5).

The maximum timeout you may specify is 1632 seconds.

The line remains disabled for the period specified, regardless of whether the "data set ready" signal line is turned off within the period.

Note: See the section "Restriction on Number of Time Intervals" following the description of the GROUP macro.

```
                     {addr1}    {addr2}
[DYNADMP={(YES[,{NSC   }][,{NSC   }])}]
         {           {NONE }    {NONE }   }
         {NO                              }
```

(emulation mode only)

Specifies whether the dynamic dump facility is included in the network control program. The dynamic dump allows the storage contents of the communications controller to be transferred to the host processor without interrupting execution of the program. This operand also specifies the subchannel(s) to be available for the transfer.

addr1 is the address of an emulation subchannel in the channel adapter in adapter position 0 over which dynamic dump data can be transferred to the host processor.

addr2 is the address of an emulation subchannel in the channel adapter in adapter position 1 over which dynamic dump data can be transferred to the host processor.

NSC specifies that the native subchannel of the channel adapter is to be used for dump data transfer. NSC in the first and second address positions refers to the native subchannel of channel adapters positions 0 and 1, respectively.

NONE specifies that no subchannel of the channel adapter is to be used for dump data transfer. NONE in the first and second address positions refers to channel adapters positions 0 and 1, respectively.

Examples:

1. If you wish to allow dynamic dump data to be transferred over the emulation subchannel of a single type 4 channel adapter, code DYNADMP=(YES,*addr*).
2. If the controller has a type 4 channel adapter in adapter position 0 and a type 2 or type 3 adapter in adapter position 1, code DYNADMP=(YES,NSC) to allow dump data transfer over the native subchannel of the type 4 channel adapter; code DYNADMP=(YES,*addr1*) to allow dump data transfer over an emulation subchannel of the type 4 adapter.
3. If the controller has two type 4 channel adapters, code (DYNADMP=(YES,*addr1*,*addr2*) to allow dump data transfer over a specified emulation subchannel of each of the channel adapters; code DYNADMP=(YES,NONE,*addr2*) to allow transfer over only the specified subchannel of the second channel adapter.

The following rules govern which type of subchannel (native or emulation) can be used to transfer dynamic dump data.

For type 1 channel adapter: (1) If the native subchannel of the adapter is used for network control operations, an emulation subchannel of this adapter can be used for dynamic dump data transfer; the native subchannel cannot be used for this purpose. (2) If the native subchannel of the adapter is *not* for network control operations, it can be used for dynamic dump data transfer.

For type 2 and type 3 channel adapter: These types of channel adapters cannot be used for dynamic dump data transfer.

For type 4 channel adapter: (1) An emulation subchannel can be used for dynamic dump data transfer regardless of whether the native subchannel is used for network control operations. (2) The native subchannel can be used for dynamic dump data transfer if it is not used for network control operations.

Figure 5-3 shows how the DYNADMP operand can specify subchannels used for transfer of dynamic dump data to the host processor.

If CA=	*and CHANTYP=*	*Then to allow dynamic dump data transfer over these subchannels*:*	*Code DYNADMP=*
TYPE1	TYPE1	ESC_1	(YES,*addr4*)
(TYPE1,TYPE2)	TYPE1	ESC_1	(YES,*addr4*)
	(TYPE1,TYPE2)	ESC_1	(YES,*addr4*)
	(TYPE2,TYPE1)	NSC_1	(YES,NSC)
(TYPE1,TYPE3)	TYPE1	ESC_1	(YES,*addr4*)
	(TYPE1,TYPE3)	ESC_1	(YES,*addr4*)
	(TYPE3,TYPE1)	NSC_1	(YES,NSC)
TYPE4	TYPE4	ESC_1	(YES,*addr4*)
(TYPE4,TYPE4) or (TYPE4,TYPE4-0) or (TYPE4,TYPE4-1)	TYPE4	ESC_1	(YES,*addr4*)
		ESC_2	(YES,NONE,*addr2*)
		ESC_1 and ESC_2	(YES,*addr4*,*addr2*))
(TYPE4,TYPE2) or (TYPE4,TYPE2-0) or (TYPE4,TYPE2-1)	TYPE2	ESC_1	(YES,*addr4*)
		NSC_1	(YES,NSC)
(TYPE4,TYPE3)	TYPE3	ESC_1	(YES,*addr4*)
		NSC_1	(YES,NSC)

*ESC - Emulation subchannel
NSC - Native subchannel

Subscripts indicate channel adapter position:
$_1$Adapter position 0
$_2$Adapter position 1.

Figure 5-3. Subchannel Address Specification for Dynamic Dump Data Transfer

```
[ENABLTO={count} ]
        {2.2   }
```

(network control mode only)

Specifies the timeout used by the NCP in detecting the failure of the "data set ready" signal line of the modem to be turned on when the communication line attached to the modem is enabled (for nonswitched lines) or when a dialing operation is completed (the automatic calling unit has signalled a connection for switched lines). Specify the timeout either as an integral number of seconds (ENABLTO=3) or to tenths of a second (ENABLTO=3.2).

The maximum timeout you may specify is 1632 seconds.

For a nonswitched line or a switched line where calls are made by an automatic calling unit (ACU), the default value of 2.2 seconds is usually appropriate. It is not appropriate, however, if the local NCP being defined includes any switched backup SDLC links to a remote controller. Such a link requires an enable timeout long enough to allow the system operator to dial the telephone number, receive an answer, and place the modem (data set) in data mode. This process may typically take from 30 seconds to more than a minute. On the other hand, the timeout value should be no greater than necessary; otherwise it will needlessly extend the time required to shut down the NCP if shutdown is initiated while a line is being activated by command from the access method. (The NCP does not shut down until all lines are deactivated.)

Note: See the section "Restriction on Number of Time Intervals" following the description of the GROUP macro.

```
[ERASE={YES}]
       {NO }
```

Specifies whether the NCP is to include the buffer erase function.

ERASE=YES is required if you specify CDATA=YES in any CLUSTER, TERMINAL, or COMP macro within the program.

```
[HICHAN=( [addr1][,addr2] )]
```

(emulation mode only)

Specifies the highest subchannel address on each channel adapter associated with any line operating in emulation mode (or the address of the subchannel used for the dynamic dump operation). The address must equal or exceed the highest emulation subchannel address specified in the ADDRESS operand of any LINE macro (or the address specified in the DYNADMP operand of the BUILD macro).

If the controller has a single type 1 or type 4 channel adapter, specify HICHAN=*addr1*. For example, HICHAN=2B. If the controller has two type 4 adapters, specify HICHAN=(*addr1*,*addr2*), where *addr1* is the highest subchannel address associated with the first type 4 channel adapter and *addr2* is the highest subchannel address associated with the second type 4 adapter. The value of *addr1* and *addr2* must be one of the following hexadecimal addresses:

03	07	0B	0F	83	87	8B	8F
13	17	1B	1F	93	97	9B	9F
23	27	2B	2F	A3	A7	AB	AF
33	37	3B	3F	B3	B7	BB	BF
43	47	4B	4F	C3	C7	CB	CF
53	57	5B	5F	D3	D7	DB	DF
63	67	6B	6F	E3	E7	EB	EF
73	77	7B	7F	F3	F7	FB	FF

The range of subchannel addresses specified by HICHAN and LOCHAN must not include any addresses associated with shared UCWs (unit control words) in the host processor.

This operand is required if the program includes emulation functions.

The subchannel address is the same address specified in the unit channel block (UCB) for OS/VS or in the physical channel block (PUB) for DOS/VS.

Note: See the description of the LOCHAN operand.

```
[ITEXTTO={count} ]
        {NONE  }
```

(start-stop and BSC stations only)

Specifies the text timeout interval used by the NCP for any terminal or component for which INHIBIT=TEXTTO is specified in the TERMINAL or COMP macro.

The maximum timeout you may specify is 1632 seconds.

ITEXTTO=NONE specifies that no timeout is to occur.

Note: See the section "Restriction of Number of Time Intervals" following the description of the GROUP macro.

```
         {YES  }
[JOBCARD={NO   }]
         {MULTI}
```

Specifies whether the program generation procedure is to provide a job card for the stage 2 input stream and specifies whether the input stream will consist of more than one job.

The job card provided is in the form:

```
//NCPGENnn JOB 1,'NCP GENERATION',MSGLEVEL=1 (OS/VS)
// JOB ASSEMBLE MAINT JCL FOR STAGE 3 (DOS/VS)
```

If you specify JOBCARD=YES or omit the operand, a single job card is provided and the program generation input stream consists of a single, multiple-step job. The job card label is // NCPGEN00.

If you specify JOBCARD=MULTI, a job card is provided for each step and the input stream consists of multiple jobs. The job card labels are //NCPGENnn, where *nn* is a sequential identification number provided by the generation procedure.

Note (OS/VS Users): If you code JOBCARD=YES or JOBCARD=MULTI, you may specify a job card different from the one shown by using the IEBUPDTE utility program to change the job statement information in the stage 1 macro library. See Chapter 4 for information on this procedure.

Note (DOS/VS Users): If you code JOBCARD=MULTI, you may specify job cards different from the one shown by using the ESERV utility to change the job statement information. (Only the second and any subsequent job cards may be changed; the first job card has the format shown above.) See Chapter 4 for information on this procedure.

[LESIZE=size]

(OS/VS only)

Specifies the OS/VS region size, in K (1024) bytes, to be used by all linkage editor job steps during stage 2 of program generation. The number you specify is reduced by 10 and used as *value1* of the linkage editor SIZE parameter. *value2* of the SIZE parameter is always 48(K), regardless of what you specify in the LESIZE operand.

size must exceed 10 and be less than 16 384 (16 384K bytes).

If you omit the LESIZE operand, the EXEC card for the linkage editor job steps will have a REGION parameter of 384K and PARM parameter values of 374 (for *value1*) and 48 (for *value2*).

[LINETRC=([YES] [[,lines] [,entries]])]

(emulation mode only)

Specifies the maximum number of lines in emulation mode that can be traced concurrently and the number of trace table entries provided. The line trace functions may be initiated from the control panel of the communications controller. The *Control Panel Guide* explains the use of the line trace option.

lines

Specifies the maximum number of lines that are to be traced concurrently. If you omit this parameter, all lines currently operating in emulation mode can be traced at once. The minimum value of *lines* is 1; the maximum is 352.

entries

Specifies the number of 8-byte entries in the trace table. The minimum is 12 entries; the maximum is 23,680.

If you omit this parameter, the trace table will contain 200 eight-byte entries.

[LOCHAN=([addr1] [,addr2])]

(emulation mode only)

Specifies the low end of the range of subchannel addresses associated with the channel adapter(s) installed in the communications controller.

If the controller has a single type 1 or type 4 channel adapter, specify LOCHAN=*addr1*. If the controller has two adapters, specify LOCHAN=(*addr1,addr2*), where *addr1* is the lowest subchannel address associated with the first channel adapter and *addr2* is the lowest subchannel address associated with the second adapter. The value of *addr1* and *addr2* must be one of the following hexadecimal addresses.

00	10	20	30
40	50	60	70
80	90	A0	B0
C0	D0	E0	F0

Note: Specifying an address that is not listed causes an MNOTE warning message to appear in the assembly listing.

The range of subchannel addresses specified by HICHAN and LOCHAN must not include any addresses associated with shared UCWs (unit control words) in the host processor. Optimum storage utilization is achieved by a contiguous assignment of all emulation subchannels. Each unassigned subchannel address between the values specified by the LOCHAN and HICHAN operands adds 10 bytes to the control program storage requirements.

CAUTION

1. All commands (except Sense, Test I/O, and I/O No-Op) issued to unassigned subchannels within the LOCHAN to HICHAN range are rejected. (Unassigned subchannels are those not specified in the ADDRESS operand of any LINE macro or in the DYNADMP operand of the BUILD macro.)
2. Although the channel adapter recognizes as valid any commands issued for a subchannel address that is outside the LOCHAN to HICHAN range, the emulation program does not recognize the address and, therefore, ignores any such commands received from the host channel. A permanently busy ("hung") subchannel results.
3. If a unit control block (UCB) exists for a device associated with a subchannel outside the LOCHAN to HICHAN range but within the channel adapter's address range, initial program load (IPL) of the operating system in the host processor cannot be completed because Test I/O and Sense commands—though accepted by the controller—are ignored.

The address range specified by the LOCHAN and HICHAN operands applies only to emulation subchannels. The network control (native) subchannel address may, but need not, lie within this range. *Exception:* The network control subchannel address must not be within the LOCHAN-HICHAN range if the communications controller is equipped with two type 4 channel adapters.

It is recommended that values specified by the LOCHAN and HICHAN operands correspond to the low and high subchannel addresses actually installed within the controller. However, the program generation procedure does not verify that this is the case. Failure to follow this recommendation may cause the following:

- All commands (except Sense, Test I/O, and I/O No-Op) issued to unassigned subchannels within the LOCHAN to HICHAN range will be rejected.
- All commands issued to subchannels outside the LOCHAN to HICHAN range will result in a permanently busy ("hung") subchannel.

The subchannel address is the same address specified in the unit channel block (UCB) for OS/VS or in the physical channel block (PUB) for DOS/VS.

```
[LTRACE={count}]
        {2     }
```

Specifies the maximum number of lines that the NCP is to trace concurrently. The minimum is two lines; the maximum is eight.

If you omit this operand, the NCP will allow only one or two lines at a time to be traced. (Line traces are requested from the host processor.)

The line trace facility is a service aid always included in the network control program.

Note: The performance of a line being traced may diminish somewhat because of the additional processing required each time a character-service or buffer-service interrupt occurs for the line. The performance of the network control program may be similarly affected to a lesser extent. In both cases, the amount of decrease in performance depends upon the type of scanner servicing the line and the degree to which the communications controller is currently loaded.

[MACLIB=dsname]

(OS/VS only)

Specifies the name of the partitioned OS/VS data set to contain the user-written source code to be included in the stage 2 generation assembly. If user-written is included in the NCP, the linkage editor INCLUDE and ORDER statements are also contained in this library. If the MACLIB operand is omitted and SRCHI or SRCLO is specified in the GENEND macro, stage 1 of the generation assumes that the user source code and the INCLUDE and ORDER statements are in the NCP macro library (SYS1.MAC3705).

The data set name may or may not be qualified, depending on the QUALIFY operand in this macro. An unqualified name may be up to 8 characters long. The first character must be alphabetic or $, #, or @. This data set must be cataloged.

Note: The NCP stage 2 macro library follows the user macro library in the generation job control statements. Therefore, if a member name in the user library is the same name as a member in the NCP library, the member in the NCP library is used.

[MAXSSCP=count]

Specifies the maximum number of system service control points (SSCP) the NCP can be in session with concurrently. These sessions can be conducted over channel adapters or over SDLC communication links, or a combination of the two.

A local NCP can be in session with as many as either SSCPs concurrently. The maximum value of *count* is 8. The program can be in session with as few as one SSCP. However, the minimum value of *count* is the number of channel adapters you have specified in the NCPCA operand as active. If you do not code the NCPCA operand, the minimum value of *count* is 1.

For example, if the 3705 has three channel adapters and you specify that two channel adapters are to be currently active in network control mode, the minimum value for count is 2. If you specify that only one adapter is to be active (by coding the CHANTYP operand instead of the NCPCA operand), the minimum value of *count* is 1; the maximum is again 8.

If you omit this operand, the value assumed for MAXSSCP equals the number of concurrently active channel adapters as specified in the NCPCA operand (or

1, if the CHANTYP operand is coded instead of the NCPCA operand). In this case, no SSCP sessions can take place over SDLC links.

Do not code the MAXSSCP operand for a remote NCP or if user-written code is included in a local NCP.

```
[MODEL={3705   }]
       {3705-2}
```

Specifies whether the network control program is to be loaded into and executed by a 3705-II (MODEL=3705-2) or a 3705-I (MODEL=3705). (A 3705-I may be specified as either MODEL=3705 or MODEL=3705-1.)

Note: This information is needed only by the generation procedure. The network control program itself does not differ for the three machine types.

```
[MTARTO={count}]
        {1.0  }
```

(multiple-terminal-access lines only)

Specifies the reply timeout used when a terminal on a multiple-terminal-access line calls the controller. Specify the timeout either as an integral number of seconds (MTARTO=5) or to tenths of a second (MTARTO=5.5).

This operand is valid only if the network includes lines defined as multiple-terminal-access lines (see the MTALIST operand of the LINE macro.)

Note: See the section, "Restriction on Number of Time Intervals" following the description of the GROUP macro.

```
[MTARTRY={count}]
         {0    }
```

(multiple-terminal-access lines only)

Specifies the number of times the NCP is to retry the multiple-terminal-access sign-on procedure after identifying the type of MTA terminal that called this controller.

The maximum number of retries is 255.

This operand is valid only if the network includes lines defined as multiple-terminal-access lines (see the MTALIST operand of the LINE macro).

```
[NCPCA=([status0],[status1],[status2],[status3])]
```

Specifies the active or inactive status of each channel adpater that operates in the network control mode.

status0, status1, status2, and *status3* are positional parameters and correspond to the channel adapter positions as specified in the CA operand.

If the adapter is to be active, code ACTIVE as the corresponding *status* parameter. A comma may be substituted for each ACTIVE parameter; for example, NCPCA=(,INACTIVE,,INACTIVE) is equivalent to NCPCA=(ACTIVE,INACTIVE,ACTIVE,INACTIVE).

If the adapter is to be inactive, code INACTIVE as the corresponding *status* parameter. Channel adapters used in emulation mode only, must be coded as inactive.

Code this operand only if two or more channel adapters of the same or equivalent types are to be active. Type 2 and type 3 are equivalent types for the purpose of this operand. If only one adapter is to be active, omit this operand and code the CHANTYP operand.

Any channel adapters you specify as INACTIVE are disabled for communication with the network control program.

Examples:

(1) For CA=(TYPE2,TYPE2), NCPCA=(ACTIVE,ACTIVE) is valid. Both adapters are of the same type.
(2) For CA=(TYPE3,TYPE2), NCPCA=(ACTIVE,ACTIVE) is valid. The adapters are of equivalent types.
(3) For CA=(TYPE4,TYPE4-0,TYPE4), NCPCA=(ACTIVE,ACTIVE,INACTIVE) or NCPCA=(ACTIVE,INACTIVE,ACTIVE) or NCPCA=(INACTIVE,ACTIVE,ACTIVE) is valid.
(4) For NCPCA=(ACTIVE), NCPCA=(INACTIVE,ACTIVE), or any other value of NCPCA that includes only one ACTIVE parameter is invalid.
(5) For CA=(TYPE4,TYPE3), NCPCA=(ACTIVE,ACTIVE) is invalid because the two adapters are not of the same or equivalent types.

Note: The network control program must be loaded into the 3705 over one of the adapters you have specified as active.

```
          {symbol}
[NEWNAME={NCP001}]
          {PEP001}
```

Specifies the name to be given to the generated network control program load module.

Code NEWNAME=*symbol*, where *symbol* is any valid symbol that does not exceed 7 characters. The generation procedure automatically assigns the name you specify, followed by the letter R, to the resource resolution table load module that corresponds to the NCP load module. If a block handler set resolution table load module is generated, its name is the name you specified as *symbol*, followed by the letter B.

Alternatively, specify NCP001 or PEP001 in this operand, whichever is appropriate. If you omit the operand, the name assigned is NCP001 if TYPEGEN=NCP or PEP001 if TYPEGEN=PEP.

```
[OBJQUAL={symbol}]
         {CG     }
```

Specifies a 2 character alphanumeric symbol to uniquely name conditionally assembled object modules. By giving the generated object modules different names, multiple NCP generations can be run concurrently. The value specified in this operand is used as the fourth and fifth characters of the object module names. For example, if OBJQUAL=X1, then the NCP conditional assemblies will be SYSX1001, SYSX1002, etc.

symbol may be any valid alphabetic or numeric character, including $, #, and @.

An object module qualifier eliminates the necessity of having multiple object libraries for the NCP conditionally assembled object modules. When an object qualifier is used for a partial generation, OBJQUAL must be the same as was specified in the complete generation that is being used as a base.

```
[OLT={YES}]
     {NO }
```

Specifies whether the optional online terminal test and online line test facilities (for lines in network control mode) are to be included in the NCP. Code OLT=NO to omit the facilities; code OLT=YES (or omit the operand) to include them.

```
[OPCSB2={YES}]
        {NO }
```

(emulation mode only)

Specifies that a 20-byte data buffer is to be provided for communication lines (1) that are serviced by a type 2 communication scanner, (2) which you have specified as operable in emulation mode, and (3) for which you have specified CHNPRI=HIGH in the LINE macro. These buffers are permanently assigned to the line and provide extra protection against overruns that can result from temporary slowdowns in channel operation or temporary peaks in data traffic in the network. Lines serviced by a type 2 scanner where OPCSB2=YES is not specified, have two 4-byte buffers.

The 20-byte buffer is used only when the line is operating in emulation mode.

If you omit this operand, OPCSB2=NO is assumed for lines associated with subchannels on a type 1 channel adapter; OPCSB2=YES is assumed for lines associated with subchannels on a type 4 channel adapter for which CHNPTRI=HIGH is specified.

If you specify OPCSB2=YES, do not specify *both* CHNPRI=HIGH and TADDR=*address* in the same LINE macro. (Either, alone, may be specified.)

```
[OUTPUT=([asm],[post-asm],[link])]
```

(OS/VS only)

Specifies the names of cataloged procedures to be used in place of the normally generated JCL when doing an NCP generation. The parameters of this operand are positional; therefore, if all of the parameters are not specified, the commas are still required. S*n* is the step number of the procedure.

asm

Specifies the name of a cataloged procedure to be used in the assembly steps of the generation. A cataloged procedure allows you to direct the assembly output to a media other than print. When asm is specified, the cataloged procedure should use the symbolic parameter &MOD. &STEP may be used to conditionally execute the assembly steps. When the assembly output is directed to tape, &FILE must be used as the symbolic parameter for the tape label operand in the procedure.

The cataloged procedure should provide the data set name for the object library but not specify the SYSIN DD statement. The SYSIN DD statement is generated in the JCL job stream. The OBJLIB operand of the BUILD macro also should not be coded.

An example of the JCL statement generated when *asm* is specified would be:

```
//S1 EXEC ASMPROC,MOD=NCP001,STEP=S1,FILE=1
```

FILE is the number of the print file generated by the assembly step.

post-asm

Specifies the name of the cataloged procedure to be inserted in the JCL job stream after the assembly step. This provides the ability to retrieve the assembly output as specified in the cataloged procedure. For example, you may want to print the assembly output only under centain error conditions.

An exmample of the JCL statement generated when *post-asm* is specified would be:

```
//S2 EXEC POSTPROC,STEP=S2,FILE=1
```

FILE is the number of the print file generated the by assembly step.

link

Specifies the name of a cataloged procedure to be used in the linkage editor step of the generation. The OBJLIB operand of the BUILD macro should not be coded, but the data set name of the object library should be provided by the cataloged procedure. The LOADLIB operand should be coded and the data set name specified in the procedure. The SYSLIN DD statement is generated in the job stream and therefore should not be specified in the cataloged procedures.

An example of the JCL statement generated when *link* is specified would be:

```
//S3 EXEC LINKPROC,STEP=S3,NAME=LOADLIB
```

Figure 5-4 shows an example of cataloged procedures that may be used as a result of the OUTPUT operand. The ASMPROC procedure causes the assembly output to be written on a nonlabeled tape. The POSTPROC procedure checks the condition code setting of the assembly step. If the condition code of the assembly is 0 or 4, the POSTPROC procedure does not retrieve and print the assembly output produced by the ASMPROC procedure. If the condition code setting of the assembly step is 8 or greater, then the assembly output is retrieved and printed. This gives you the ability to selectively print the assembly output.

NCP generation macro specification:

```
NCPGEN BUILD TYPSYS=OS,OUTPUT=(ASMPROC,POSTPROC,LINKPROC)
```

Cataloged Procedures:

```
ASMPROC       PROC
//S&FILE      EXEC   PGM=CWAX00,PARM=(DECK,XREF)
//SYSPRINT     DD    DCB=(DEN=3,LRECL=121,BLKSIZE=3025,RECFM=FB),
//                   UNIT=2400,LABEL=(&FILE,NL),DISP=(,PASS),VOL=SFR=ALPHA1
//SYSUT1       DD    UNIT=SYSSQ,SPACE=(1700,(800,800),DCB=(OPTCD=C)
//SYSUT2       DD    UNIT=SYSSQ,SPACE=(1700,(800,800),DCB=(OPTCD=C)
//SYSUT3       DD    UNIT=SYSSQ,SPACE=(1700,(800,800),DCB=(OPTCD=C)
//SYSLIB       DD    DSN=SYS1.MAC3705,DISP=SHR,UNIT=3330-1,VOL=SER=BETA
//SYSPUNCH     DD    DSN=SYS1.OBJ730D(&MOD),DISP=OLD
//SYSUDUMP     DD    SYSOUT=A
              PEND
POSTPROC      PROC
//&STEP       EXEC   PGM=IEBGENER,COND=(8,GT,S&FILE)
//SYSPRINT     DD    SYSOUT=A
//SYSIN        DD    DUMMY
//SYSUT1       DD    DCB=(DEN=3,LRECL=121,BLKSIZE=3025,RECFM=FB),
//                   UNIT=2400,LABEL=(&FILE,NL,DISP=(,PASS),VOL=SER=PE0081
//SYSUT2       DD    SYSOUT=A
              PEND
LINKPROC      PROC
//&STEP       EXEC   PGM=HEWL,REGION=450K,
//                   PARM='LIST,LET,DC,NCAL,XREF,SIZE=(440K,48K),ALIGN2'
//SYSPRINT     DD    SYSOUT=A
//SYSUT1       DD    UNIT=SYSSQ,SPACE=(1700,(800,800)),DCB=(OPTCD=C)
//SYSLMOD      DD    DSN=SYS1.LOADLIB,DISP=SHR
//OBJ3705      DD    DSN=SYS1.OBJ3705,DISP=SHR
//SYSPUNCH     DD    DSN=SYS1.OBJ730D,DISP=SHR
              PEND
```

Figure 5-4. Example of Cataloged Procedures

```
[PARTIAL={YES}]
         {NO }
```

Specifies whether a partial program generation is to be performed.

If you code PARTIAL=YES, only the tables and conditionally assembled modules specified in the CONDASM operand are assembled and link-edited with the remaining object modules. The conditionally assembled modules not specified by the CONDASM operand are obtained from the library specified by the OBJLIB operand. The modules assembled by the partial generation procedure replace the corresponding modules from the previous generation.

Appendix B lists the modules that must be reassembled for various changes in program functions.

If you code PARTIAL=NO (or omit the operand) a complete program generation is performed.

```
[PRTGEN=( {NOGEN} [,{NOGEN} ] )]
          {GEN  }    {GEN  }
```

Specifies whether macro generated statements are printed for the NCP table assemblies and NCP conditional assemblies. The first parameter is for the two NCP table assemblies. The second parameter is for all of the conditional assemblies.

If GEN is specified (or the parameter is omitted) the generation procedure prints all of the generated statements produced by the stage 2 assembly.

Specifying NOGEN reduces the amount of printed output from stage 2 during a partial generation. NOGEN suppresses printing of the assembled statements with the exception of MNOTEs. MNOTEs are always printed.

```
[PWROFF={YES}]
        {NO }
```

(local NCP only; VTAM users only)

Specifies whether the program in a local communications controller will turn off the remote controller's power upon command from VTAM. PWROFF=YES is valid only if the remote controller is equipped with the remote power off feature. (Power can be turned on again only at the control panel of the remote controller.)

```
         {symbol}
[QUALIFY={NONE  }]
         {SYS1  }
```

(OS/VS only)

Specifies the first-level qualifier for OS/VS data sets specified by the LOADLIB, OBJLIB, USERLIB, UT1, UT2, and UT3 operands of this macro. The data set name is formed by appending the characters SYS1, or *symbol*, to the name specified *dsname* in each of the previously mentioned operands.

symbol

Specifies the qualifier as from 1 to 8 alphanumeric characters; the first character must be alphabetic (including $, @ and #). (Omit the period that separates the qualifier and the data set name; the generation procedure appends the period to the qualifier you specify.)

NONE

Specifies that no qualifier is to be placed before the simple name specified by *dsname*.

SYS1

Specifies that SYS1 is to be used as the qualifier.

```
[REMLOAD={YES}]
         {NO }
```

Specifies whether the remote program loader feature is installed on an 3705-II.

REMLOAD=NO is invalid for a remote network control program (TYPGEN=NCP-R) that is to be executed in a 3705-II.

If you specify TYPGEN=NCP-R and MODEL=3705-2, and omit this operand, REMLOAD=YES is assumed, otherwise REMLOAD=NO is assumed.

Note: A 3705-II equipped with channel adapter(s) and a remote program loader may operate at different times as either a local or a remote controller. Separate network control programs are required for the two modes: a program in which TYPGEN=NCP (or NCP=LR) is specified to operate the controller as a local controller and a program in which TYPGEN=NCP-R is specified to operate the controller as a remote controller. (Specify REMLOAD=YES if the controller has a remote program loader even if the program you are defining is a local network control program.)

```
[RESOEXT={count}]
         {0    }
```

Specifies the number of network addresses in the resource vector table (RVT) extension. The size of the RVT extension determines the number of generation-defined resources that can be deleted and resued.

When a generation-defined physical or logical unit is deleted, the resource is returned to its respective pool. In order for that resource to be reused, the generation-defined network address must be replaced by a network address from the RVT extension. If the RVT extension is depleted (all addresses used), generation-defined resources can still be deleted but not reused. However, unlike the network addresses defined during the NCP generation the addresses in the RVT extension can be reused once they are deleted.

The maximum value of *count* is subject to the maximum number of resources that can be defined in a particular NCP. (See the MAXSUBA operand of the BUILD macro.)

```
         {12}
[SLODOWN={25}]
         {50}
```

Specifies the minimum percent of NCP buffers that are available (not in use) before the network control program enters slowdown mode. When the percent of buffers available drops below this value, the program reduces the amount of data it accepts from lines operating in network control mode and from the network control subchannel, but it continues to send data over the lines and the subchannel. This procedure reduces the number of buffers in use.

Slowdown mode is entered when fewer than one-half (SLODOWN=50), one-quarter (SLODOWN=25) or one-eighth (SLODOWN=12, or operand is omitted) of the buffers are available.

During initialization, the NCP dynamically increases the percentage you specify if the minimum NCP buffer requirements cannot otherwise be met. The minimum number of buffers that the program must contain for each percentage value allowed is 80 buffers for 12 percent; 40 buffers for 25 percent; and 20 buffers for 50 percent.

If the number of buffers contained in the NCP is less than 20, the program abends.

```
[TIME=integer]
```

(OS/VS only)

Specifies the time value, in minutes, to be used as the TIME parameter in the stage 2 assembly EXEC statements. *integer* must be greater than 0 and less than 1441.

If you omit this operand, no TIME parameter is used for the stage 2 assembly EXEC statements.

```
[TRACE={(YES[,{ size}])}]
               { 10  }   }
        { NO             }
```

Specifies whether the address trace option is to be included in the NCP. Code TRACE=YES to include the option; code TRACE=NO (or omit the operand) to omit the option.

If you code TRACE=YES, you may also specify the number of 16-byte entries the trace table is to contain, from 10 [TRACE=(YES,10)] to 256 [TRACE=(YES,256)]. If you omit the number or specify fewer than 10, the table will contain 10 entries.

```
[TRANSFR=count]
```

Specifies the number of NCP buffers corresponding to the maximum amount of data that the network control program is to send to a destination host processor during a single data transfer operation.

The path between the NCP being defined and the destination host processor may comprise a channel connection to the attached host processor or a combination of SDLC links between network control programs and a channel connection to a distant host processor.

The principal use of this operand when used with the Build macro, is to limit the amount of line trace data accumulated in NCP buffers to no more than the access method in the destination host processor (and intermediate access methods, if any), can accept in a single channel operation (or to no more than the data transfer limit imposed by a subsequent NCP in the path).

You may also use this operand to specify the data transfer limit for any communication line where the TRANSFR operand is omitted from the LINE macro. See the description of the TRANSFR operand in the LINE macro.

To calculate the value of *count*, proceed as follows:

1. Determine which host processors in the network will receive the line trace data.
2. Calculate the data transfer limit, in bytes, imposed by the access method in each destination host processor. To do so, multiply the values specified by the MAXBFRU and UNITSZ operands and then subtract the value of the BFRPAD operand of the HOST macro that represents the access method to the NCP.
3. Determine which of the data transfer limits thus calculated is the smallest, and divide that limit by the buffer size specified in the BFRS operand of the BUILD macro in the present network control program. The result (ignoring any fractional remainder) is the maximum value of *count* you should specify in this TRANSFR operand.

See the description of the TRANSFR operand in the LINE macro for an example showing how to calculate the data transfer limits. Also see this description for determining the *minimum* value of *count* if you use this TRANSFR operand to specify the data transfer limits for any communication line for which you omit the TRANSFR operand from the LINE macro.

It is most convenient to establish the same data transfer limit for line trace data as for message data received from communication lines by the present network control program. Because the buffers used for both purposes are of the same size, the *count* values you specify in the TRANSFR operands of the BUILD and LINE macros would be the same.

If you omit the TRANSFR operand from some LINE macros, be sure that the value you specify in this TRANSFR operand (in the BUILD macro) is appropriate for the corresponding lines.

The minimum you may specify in this operand is one buffer (TRANSFR=1); the maximum is 255 (TRANSFR=255).

If you omit this TRANSFR operand, the program generation procedure determines a value in one of the following ways:

- If one or more HOST macros appear in this program, the generation procedure computes the data transfer limit(s) from the MAXBFRU, UNITSZ, and BFRPAD operands as described. It then divides the result (lowest result, if there is more than one HOST macro) by the NCP buffer size, rounds down to the next lower integer, and uses the resulting value (or a maximum of 255) as the maximum buffer count for the line trace data.
- If the program contains no HOST macro(s), the generation procedure assumes an arbitrary value of 7 buffers for the TRANSFR operand.

Upon filling the specified number of buffers with line trace data, the NCP transfers the data to the destination host processor and continues to accumulate trace data in other buffers. The cycle of obtaining buffers, filling them, and transferring their contents to the host processor continues until the line trace function is ended by the operator.

```
[TWXID=( {inchars}{,outchars} )]
        {xon     }{,xon       }
```

(TWX terminals in network control mode only)

Specifies the characters used for both the initial ID answerback and subsequent prompting sequences for all TWX terminals on lines in network control mode.

inchars specifies the answerback and prompting sequences to be sent to all TWX terminals when the switched connection is terminal initiated.

outchars specifies the answerback and prompting sequences to be sent to all TWX terminals when the switched connection is controller initiated.

inchars and *outchars* must be specified as the hexadecimal representation of the EBCDIC characters to be used. Each may contain up to 20 EBCDIC characters. Any EBCDIC character may be specified, including those that translate into TWX carriage return or line feed.

The last character specified (presumably an X-on or WRU) is not transmitted during the initial ID answerback or on the subsequent prompting sequences that are sent as part of each TWX terminal read operation. Instead, the NCP generates and sends an X-on character in lieu of the last character specified.

Note: Since the prompting sequence is sent at the beginning of *every* terminal read operation, you may want to specify a short sequence to avoid an excessive amount of time required in transmitting and printing the prompting sequence.

This operand has no meaning if communication with TWX terminals is only in emulation mode.

```
[TYPSYS={OS }]
        {DOS}
```

Specifies whether stage 2 of the NCP generation procedure is to be run under OS/VS or DOS/VS.

```
[UNIT=unit type]
```

(OS/VS only)

Specifies the type of device used for the assembler and linkage editor utility data sets during stage 2 of program generation under OS/VS. You may specify either an actual device type (for example, UNIT=3330) or the name of a class of devices (for example, UNIT=SYSDA). The maximum number of characters you may specify is 8.

If you omit this operand, SYSSQ is assumed to be the unit type for the assembly steps and SYSDA is assumed for the link-editing steps.

Note: The utility data set for the linkage editor must reside on a direct-access device.

```
[USERLIB=dsname]
```

(OS/VS only)

Specifies the name of the partitioned OS/VS data set to contain user-written object code and block handler routines. If this operand is omitted and the generation instructions call for user-written object code, the generation procedure assumes that the user code is in the NCP object library (SYS1.OBJ3705).

The data set name may or may not be qualified, depending on the QUALIFY operand in this macro. An unqualified name may be up to 8 characters long. The first character must be alphabetic or $, #, or @. The data set specified by *dsname* must be cataloged.

`[UT1=dsname]`

(OS/VS only)

Specifies the name of a sequential OS/VS data set to be used as work space for the assembly steps (SYSUT1). The data set name may or may not be qualified, depending the QUALIFY operand of this macro. An unqualified name may be up to 8 characters long; the first must be alphabetic or $, #, or @. This data set must be preallocated and cataloged.

If you omit this operand, a temporary data set will be created during each assembly step using the type of device specified by the UNIT operand. The data set space provided is equivalent to SPACE=(1700,(800,800)).

`[UT2=dsname]`

(OS/VS only)

Specifies the name of a sequential OS/VS data set to be used as work space for the assembly steps (SYSUT2). The data set name may or may not be qualified, depending on the QUALIFY operand of this macro. An unqualified name may be up to 8 characters long; the first must be alphabetic or $, #, or @. This data set must be preallocated and cataloged.

If you omit this operand, a temporary data set will be created during each assembly step using the type of device specified by the UNIT operand. The data set space provided is equivalent to SPACE=(1700,(800,800)).

`[UT3=dsname]`

(OS/VS only)

Specifies the name of a sequential OS/VS data set to be used as work space for assembly (SYSUT3) and link-edit (SYSUT1) steps. The data set name may or may not be qualified, depending on the QUALIFY operand of this macro. An unqualified name may be up to 8 characters long; the first must be alphabetic or $,#,or @. This data set must be preallocated and cataloged.

If you omit this operand, temporary data sets will be created during each assembly step and each linkage edit step using the type of device specified by the UNIT operand. The data set space provided is equivalent to SPACE=(1700,(800,800)).

```
[XBREAK={integer}]
       {NONE    }
```

(start-stop lines in network control mode only)

`integer`

Specifies the number of character times the NCP is to place the break signal on the line to interrupt transmission from the terminal. This is applicable only for lines in network control mode and terminals for which the LINE and TERMINAL macros specify DUPLEX=FULL and FEATURE=BREAK, respectively.

The minimum value is 3; the maximum is 255.

NONE

Specifies that the NCP will not send break characters.

If you omit the XBREAK operand, a value of 3 is assumed if the network includes any duplex lines (DUPLEX=FULL in the LINE macro) terminals having the break function are attached (FEATURE=BREAK in the TERMINAL macro).

If the network does not include such lines and terminals, XBREAK=NONE is assumed if you omit the XBREAK operand.

```
[XITB={YES} ]
      {NO  }
```

(BSC stations in network control mode only)

Specifies whether the NCP is to insert transparent ITB sequences and error-information blocks in transparent text sent to stations which have intermediate block checking specified.

If you specify XITB=YES and the *first* parameter of the ITBMODE operand of the TERMINAL macro for the station specifies intermediate block checking, the program substitutes an error information block for each DLE ITB sequence in transparent text received from the station.

If you specify XITB=YES and the *second* parameter of the ITBMODE operand specifies intermediate block checking, the program inserts DLE ITB sequences into transparent text being sent to the station.

If you specify XITB=NO (or omit the XITB operand), no insertion of DLE ITB sequences and EIBs is performed for any BSC station.

SYSCNTRL Macro Instruction

SYSCNTRL specifies the dynamic control facilities included in the network control program. These facilities allow the NCP to execute requests from the access method to change certain NCP parameters or to determine the status of resources such as lines and stations.

Figure 5-5 shows which options are required by VTAM and TCAM.

This macro is required and must appear directly following the BUILD macro.

The format of the SYSCNTRL macro is:

```
[symbol]        SYSCNTRL     operands
```

Operands
OPTIONS=(entry,...)

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

OPTIONS=(entry,...)

Specifies which of the dynamic control facilities are to be included in the network control program. *entry* may be any value listed in Figure 5-5.

Entry	Facility
Always required by VTAM:	
BHSASSC	Modify block handler set
ENDCALL	Physical disconnect (required for switched lines only)
MODE	Change line speed
RCNTRL	Request control mode reset
RCOND	Reset conditional
RECMD	Reset at end of command
RIMM	Reset immediate
Required by VTAM if ANS=YES is specified in the BUILD macro:	
NAKLIM	Change line negative polling response limit
SESSION	Change session limit
SSPAUSE	Change service-seeking pause
XMTLMT	Change device transmission limit
Always required by TCAM:	
RCOND	Reset conditional
RIMM	Reset immediate
Required by TCAM for certain operator control functions or application program macros:	
BACKUP	Switched network backup
BHSASSC	Modify block handler set
DVSINIT	Change device session initiation information
LNSTAT	Display line status
MODE	Change line speed
RECMD	Reset at end of command
SESINIT	Change line session initiation information

Figure 5-5. Dynamic Control Facilities Required by VTAM and TCAM

NCPNAU Macro Instruction

The NCPNAU macro instruction defines the names of the user-defined control blocks and function vector tables associated with a network addressable unit (NAU). It also assigns a specific name (symbol) to the NAU and tells the NCP whether the NAU is to function as a system services control point or a logical unit.

Each network addressable unit must be represented by a separate NCPNAU macro. The NCPNAU macro must immediately follow the SYSCNTRL macro.

This macro is valid only when the NCP contains user-written interrupt code and control blocks. If user-written code is not included, the NCPNAU macro *must* be omitted.

The format of the NCPNAU macro is:

```
[symbol]        NCPNAU        operands[,operands]
```

Operands

```
NAUFVT=(symbol1[,symbol2]...[,symbol9]))]
[,NAUCB=(symbol1[,symbol2]...[,symbol9]))]
[,TYPE={SSCP }]
       {NCPLU}
```

symbol

Specifies the resource name for the network addressable unit. *symbol* may be any valid assembler-language symbol, but it may not begin with a $ as the first character. A symbol is required on all NCPNAU macro instructions.

NAUFVT=(symbol1[,symbol2]...[,symbol9]))]

Specifies the names of the functional vector tables associated with this network addressable unit. At least one functional vector table is required. The symbols in this operand are positionally related to the symbols in the NAUCB operand.

An EXTRN statement for each unique FVT symbol must be included in the user source code that is copied into the tables during assembly. (See the SRCHI and SRCLO operands in the GENEND macro.)

Note: Functional vector tables must be preassembled and included in the user object modules. (See the INCHI and INCLO operands in the GENEND macro.)

[NAUCB=(symbol1[,symbol2]...[,symbol9]))]

Specifies the names of user-defined control blocks associated with this network addressable unit.

The symbols in the NAUCB operand are positionally related to the symbols in the NAUFVT operand. If a symbol is specified in NAUCB, a corresponding symbol *must* be specified in the same symbol position in the NAUFVT operand.

Note: The user control block must be assembled with the NCP in stage 2 of the generation. (See the SRCHI and SRCLO operands in the GENEND macro.)

```
[TYPE={SSCP }]
      {NCPLU}
```

Specifies whether the network addressable unit is to function as a system services control point or as an NCP logical unit. If this operand is omitted, the generation procedure assumes SSCP.

Configuration Definition Macro Instructions

This section contains the configuration definition macro instructions (HOST, CSB, IDLIST, LUPOOL, LUDRPOOL, PUDRPOOL, PATH, SDLCST, DIALSET, and MTA macros) used in defining a network control program.

HOST Macro Instruction

The HOST macro specifies characteristics of the access method that the NCP will communicate with. The NCP must know these characteristics in order to conduct the channel operations necessary to communicate with the access method.

This macro also specifies the subarea address to identify the access method to the control program. More than one subarea (access method) can be represented by the same HOST macro if their parameters are the same, but only one can communicate with the NCP at any moment. A separate HOST macro must be defined for each access method that can communicate with the NCP concurrently.

The number of HOST macros is not limited to the maximum number of access methods that can communicate concurrently with the control program. You may code as many macros as you need to associate subareas with their related access method parameters.

The HOST macro instruction specifies:

- The number of NCP buffers to be allocated for receiving a data transfer from the access method
- The size of the access method buffer unit used to receive data from the NCP
- The number of buffer units the access method will allocate for receiving a data transfer
- The number of bytes in the header prefixes used by the access method
- The amount of time the NCP waits for a response from the host processor to an attention signal
- The amount of time the NCP waits before presenting an attention signal on the channel after data becomes available for transfer to the host processor

A local network control program requires at least one HOST macro.

A remote NCP (TYPGEN=NCP-R) should not include a HOST macro. A PATH macro coded in a remote program represents the subarea of the local NCP with which the remote program communicates.

The HOST macro(s) must appear among the configuration macros that follow the system macros (BUILD, SYSCNTRL, etc.).

The format of the HOST macro is:

```
[symbol]        HOST          operands[,operands]
```

Operands

```
INBFRS=count,
MAXBFRU=count,
UNITSZ=length
[,BFRPAD={n }]
        {28}
[,DELAY={count}]        (local NCP only)
        {0    }
[,STATMOD={YES}]        (local NCP only)
          {NO }
[,SUBAREA={(subarea1[,subarea2]...[,subarean])}]
          { 1                                  }
[,TIMEOUT={count}]      (local NCP only)
          {NONE }
```

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

INBFRS=count

Specifies the number of controller buffers initially allocated for each data transfer received from the host processor. The minimum value is 1; the maximum is 255. If a data transfer requires additional buffers, they are allocated in the quantity specified by INBFRS.

This operand is required.

MAXBFRU=count

Specifies the number of buffer units the access method will allocate for receiving data from the NCP. The minimum is 1; the maximum is 255.

This operand is required.

See the description of the UNITSZ operand for a sample calculation.

VTAM Note: The value specified in this operand must be less than the value of the *baseno* subparameter of the LFBUF (DOS/VS) or IOBUF (OS/VS) parameter of the VTAM start parameter list.

UNITSZ=length

Specifies the size of the access method buffer units used for data transfers from the NCP. *length* is the sum of (1) the length of the message data, (2) 13 bytes for control information, and (3) the number of pad characters specified in the BFRPAD operand.

The access method must use one buffer unit size for all transfers from the network control program. A buffer unit is the smallest unit of contiguous storage handled as buffer space. A buffer may consist of one or more units.

The NCP generation procedure accepts values between 1 and 65535, inclusive. However, the access method imposes its own minimum and maximum values. See the *ACF/VTAM* or *ACF/TCAM Installation* manuals for the proper values for your perticular network.

VTAM Note: This operand must specify the same value as the *bufsize* subparameter of the IOBUF (OS/VS) or LFBUF (DOS/VS) parameter of the VTAM start parameter list.

The maximum number of characters the NCP will transfer to the host processor in a single operation over the network control subchannel equals MAXBFRU times UNITSZ, minus the value specified in the BFRPAD operand. For example, if you code MAXBFRU=5, UNITSZ=84, and BFRPAD=(28), the maximum number of characters sent to the host processor during a single channel operation is 5 x 84 - 28 = 392 bytes. (This value includes the 13 bytes of control information associated with each response sent by the NCP.)

CAUTION
The maximum amount of data that the access method can send to the NCP should not exceed the amount that the NCP can return to the access method.

```
[BFRPAD={n }]
        {28}
```

Specifies the number of pad characters the NCP transmits to the access method immediately preceding the control information for the response.

Placing pad characters at the beginning of each access method buffer allows the access method to insert message header and message text prefixes.

The range of values for this operand is 0-255. However, the value you specify in the BRFPAD operand should equal the size of the prefixes that the access method uses. The required value for OS/VS VTAM is 28 bytes; for DOS/VS VTAM, 15 bytes; for ACF/VTAM, 0 bytes; for OS/VS TCAM, a minimum of 17 bytes; for ACF/TCAM, 17-28 bytes.

```
[DELAY={count}]
       {0    }
```

(local NCP only)

Specifies the interval that the NCP will delay between the time the NCP has data available for the host processor and the time the NCP presents an attention signal to the host processor.

count

Specifies the delay, to the nearest tenth of a second. The minimum delay is 0 seconds and the maximum is 420.0 seconds.

0

Specifies that an attention signal is to be presented to the host processor as soon as data is available.

If the amount of data is sufficient to fill the buffers allocated by the host processor, the attention signal will be presented before the delay count has been reached.

This operand is invalid for a remote network control program.

```
[STATMOD={YES}]
        { NO }
```

(local NCP only)

Specifies whether the NCP is to use the status modifier option for sending responses to the access method. Use of this option when the attention delay option is used minimizes the number of asynchronous channel interrupts during data transfer. This reduction of channel interrupts occurs because each time the access method sends data to the NCP, the NCP returns any accumulated response data to the access method as part of the same channel operation instead of sending a later attention interrupt to the host processor.

Use of the status modifier option is required if the host access method is run under OS/VS. To maximize the increase in channel performance, also specify an attention delay of at least 0.1 second in the DELAY operand. If the access method is DOS/VS VTAM, the status modifier option is optional for release 33 (or higher) of DOS/VS and release 2 (or higher) of VTAM. For lower levels of VTAM or DOS/VS, the option is not available and STATMOD=NO should be specified.

This operand must be specified the same (either STATMOD=YES or STATMOD=NO) in all HOST macros included in the NCP being defined.

```
[SUBAREA={(subarea1[,subarea2]...[,subarean])}]
        { 1                                   }
```

Specifies the subarea address(es) representing the access method(s) in one or more adjacent host processors that the NCP will communicate with.

If you specify more than one subarea address in this operand, the NCP can communicate with only one of the subareas (access methods) at a time. All subareas specified by this operand must be functionally equivalent; that is, the access method buffering, status modifier, and time-out parameters specified in this HOST macro must apply identically to all of those subareas.

The network control program transfers path information units (PIU) to the currently active subarea (access method) if the destination of the PIUs is:

- One of the subareas specified in this SUBAREA operand, or
- One of the subareas specified in the DESTSUB operand of a PATH macro in which the ADJSUB operand specifies the subareas specified in this SUBAREA operand.

For example, if in this present NCP you specify SUBAREA=(2,3,7) in this HOST macro and ADJSUB=(2,3,7) and DESTSUB=(4,5) in a PATH macro, then the NCP will send PIUs whose destination is 2, 3, 4, 5, or 7 to the access method corresponding to subarea 2, 3, or 7 (whichever is currently active).

Any subarea address you specify in this HOST macro cannot be specified in another HOST macro, in the DESTSUB operand of any PATH macro, in any PU macro, or in the BUILD macro. It can be specified in the ADJSUB operand of a PATH macro.

Each subarea in the entire network (whether the network is controlled by one or multiple network control programs) must have a unique subarea address. The only exceptions are (1) a subarea defined to one NCP that never communicates with an NCP having the same address in a different subarea and (2) a configuration in which more than one access method has the same address but only one is communicating with the network at a time.

Note: The program generation procedure establishes a logical association of a host subarea, as represented by the HOST macro, with the NCP. The physical association between the host subarea and the NCP begins when the access method in the host subarea initiates communication with the NCP.

The smallest subarea address you may specify is 1. The largest is the value specified in the MAXSUBA operand of the BUILD macro. That operand specifies the maximum subarea address that can exist within the entire network, including those assigned to nonadjacent access methods and other network control programs.

```
[TIMEOUT={count}]
         {NONE  }
```

(local NCP only)

Specifies the interval that the NCP waits for a response to an attention signal it has sent to the host processor before initiating automatic network shutdown. This operand applies only if ANS=YES is specified in the BUILD macro. Express this interval either as an integral number of seconds (TIMEOUT=15), or to the nearest tenth of a second (TIMEOUT=12.5).

The minimum value is 0.2 second; the maximum is 420.0 seconds.

NONE

Specifies that the NCP is to wait indefinitely for a response from the host processor.

If the automatic network shutdown facility is included in the program (ANS=YES) and you omit the TIMEOUT operand, TIMEOUT=420.0 is assumed.

Upon failure of the host processor to respond to the network control program, the action the NCP takes depends on how you have specified the ANS and TIMEOUT operands: (1) If you specify ANS=YES and TIMEOUT=*count*, the NCP performs automatic network shutdown and then waits for data over the channel. (2) If you specify ANS=NO and TIMEOUT=NONE, the NCP does not perform automatic network shutdown, but instead waits indefinitely for message traffic over the channel. (3) If you specify ANS=YES and TIMEOUT=NONE, you have to initiate ANS from the control panel.

This operand is invalid for a remote network control program (TYPGEN=NCP-R).

Note: The interval specified in the TIMEOUT operand is not counted as among the 16 possible time intervals permitted in the network control program. (The limit of 16 applies only to communication line time-outs and intervals—not to channel time-outs.)

CSB Macro Instruction

The CSB macro specifies:

- The internal oscillator (business machine clock) rates for the scanner
- The location of the scanner within the controller
- The type of communications scanner
- The line address for receiving test data in emulation mode

Each scanner in the controller must be represented by a CSB macro.

The format of the CSB macro is:

```
[symbol]        CSB            operands[,operands]
```

Operands

```
SPEED=(rate,...)
[,MOD={n}]
      {0}
[,TYPE={TYPE2}]
       {TYPE3}
[,WRAPLN=line addr]
```

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

SPEED=(rate,...)

Specifies the internal oscillator (business machine clock) rates for up to four oscillators installed in the communication scanner. (Do not confuse this SPEED operand, which specifies the oscillator bit rates, with the SPEED operand of the LINE macro, which specifies the data rate for the communication line.) The speeds must be specified in the same order that the oscillators are installed on the scanner, in ascending order according to speed. Standard oscillator bit rates are shown in Figure 5-6.

Note: If external (modem) clocking is used for any line attached to this scanner (CLOCKNG=EXT is specified in the LINE macro), one of the oscillator bit rates you specify must be less than one-half of the lowest modem clocking rate specified in the SPEED operand of any LINE macro representing the attached lines.

A type 3 scanner is always equipped with an oscillator that provides 150, 600, and 1200 bps bit-rates, and may optionally have a 2000 or 2400 bps oscillator. Therefore, if this CSB macro represents a type 3 scanner, specify SPEED=(150,600,1200) or SPEED=(150,600,1200,2000) or SPEED=(150,600,1200,2400), as appropriate.

This operand is required.

Rate	*Represents:*	*Rate*	*Represents:*
45	45.5 bps	150	150.0 bps
50	50.0	200	200.0
56	56.89	300	300.0
74	74.2	600	600.0
75	75.0	950	950.0
100	100.0	1200	1200.0
110	110.0	2000	2000.0
134	134.5	2400	2400.0

Figure 5-6. Communication Scanner Oscillator Bit Rates

```
[MOD={n}  ]
     {0}
```

Specifies the location of the communication scanner. The line interface addresses valid for each scanner type and module location are shown in Figure 5-7.

		Line Interface Addresses	
If scanner is in:	*Code MOD=*	*Type 2 Scanner*	*Type 3 Scanner*
3705 base module	0	020-05F	020-04F
3705 first expansion module	1	0A0-0FF	0A0-0DF
3705 second expansion module	2	120-17F	120-15F
3705 third expansion module	3	1A0-1FF	1A0-1DF

Figure 5-7. Communication Scanner Line Interface Addresses

```
[TYPE={TYPE2}  ]
      {TYPE3}
```

Specifies whether the communication scanner is type 2 or type 3. (ACF/NCP/VS does not support a type 1 scanner.)

Valid designations for scanner type are:

If controller is a 3705-I (MODEL=3705 or 3705-1): TYPE2
TYPE3 (MOD-1, 2, or 3 only)

If controller is a 3705-II (MODEL=3705-2): TYPE2
TYPE3

If you omit this operand, the scanner is assumed to be type 2.

[WRAPLN=line addr]

Specifies the line address the controller will use to sent test data when executing a *wraparound test*. Specify the hexadecimal line address without framing characters; for example, WRAPLN=02F. The address specified must be within the range shown in Figure 5-7, and must appear in the ADDRESS operand of one of the LINE macros.

The line specified need not be dedicated to the wraparound test operation; it can be any line that can be conveniently closed to normal message traffic when a wraparound test is needed. Both the specified line and the line to be tested must be closed to normal message traffic for the duration of the test. The online test (OLT) program selects the line to be tested.

Note: If any of the lines serviced by the scanner represented by this macro are BSC lines, the address you select for WRAPLN must be the line interface address for a BSC line.

This operand is required if the program includes emulation functions (TYPGEN=PEP is specified).

IDLIST Macro Instruction

The IDLIST macro specifies:

- A list of identification sequences for BSC or TWX stations that call or are called by the controller over a switched line operated in network control mode
- The maximum size of the ID list
- The action the NCP performs when it receives an ID sequence that does not match any sequence in the list

Note: The IDLIST macro is used only in a network control program that communicates with OS/VS VTAM or OS/VS TCAM. It is invalid for use with DOS/VS VTAM.

An identification list is optional for each switched line that BSC or TWX stations will call or be called by the controller in network control mode. If you provide a list, the network control program checks ID sequences it receives from a station against the sequences in the list. If you do not provide a list, no check is made. (The same list may be used for more than one line.)

Note: You may specify that ID sequences received from stations calling the controller be checked by the access method instead of (or in addition to) the NCP. Refer to descriptions of the VIDLIST (VTAM-only) macro and IDSEQ operand of the TERMINAL macro. See the *ACF/TCAM Installation* manual for information about ID verification by TCAM. ID sequences received during call-out operations are never passed to the access method.

The format of the IDLIST macro is:

```
[symbol]        IDLIST        operands[,operands]
```

Operands

```
IDSEQ={(chars,...)              }
      {((chars,termname),...)}
[,MAXLEN=count]
[,NOMATCH={PASS}]
          {STOP}
```

[symbol]

Provides a name for the ID list and is required except as indicated in the description of the IDSEQ operand. *symbol* may be any valid assembler-language symbol. If *symbol* is required, the first character may not be $.

```
IDSEQ={(chars,...)           }
      {((chars,termname),...)}
```

Specifies the identification sequence for each station that may call or be called by the controller. It also may specifiy the name of the TERMINAL macro for the station from which the sequence is expected. Either (*chars*,...) or *((chars,term name),...)* may be specified in this operand; the two cannot be intermixed.

chars

Specifies the identification sequence the NCP will recognize as valid. Code *chars* as the hexadecimal representation of the EBCDIC characters to be recognized. You may specify a maximum of 20 EBCDIC characters in one sequence.

Note: Any EOT, ENQ, or ACK characters sent by a TWX terminal as part of its ID sequence are deleted by the NCP as it receives the sequence into a buffer. It is therefore necessary to omit any of these three characters when specifying the ID sequence in the IDSEQ operand. Failure to omit them will cause the program never to recognize the received sequence.

[termname]

Specifies the name of the TERMINAL macro representing the station associated with the ID characters. (Do not specify a TERMINAL macro in which CTERM=YES is coded.)

If you omit *termname*, the network control program will recognize the sequence as valid, but it will not recognize it as the sequence for a specific station.

You may code a maximum of 255 characters in the IDSEQ operand, including the beginning and ending parentheses and all commas. This limit applies regardless of how many entries you code within the operand. If you need more than 255 characters, code additional IDLIST macros (omitting the symbol field of each) directly following the first IDLIST macro. (Only the first IDLIST macro may include the MAXLEN and NOMATCH operands.)

[MAXLEN=count]

Specifies the maximum size of the list, in bytes (*not* the number of entries). This value includes the total number of bytes in all entries, plus the control field that precedes the list. This operand should be specified only if the access method uses the dynamic control facility either to add entries to the list or to increase the size of one or more existing entries in the list.

Calculate the value for MAXLEN by adding:

For the list header: 4 bytes

For each entry in the list:

The number of bytes required to contain the ID sequence, plus either:

2 bytes if you omit *termname* in the IDSEQ operand; or
4 bytes if you specify *termname* in the IDSEQ operand.

Round the size of each entry to the next higher fullword.

The maximum number of ID sequences (entries) the list may contain is 256.

```
[NOMATCH={PASS} ]
         {STOP}
```

Specifies the action the NCP performs if it does not receive an ID sequence from a calling station or it receives an ID sequence it does not recognize as valid (a sequence not defined in this IDLIST macro).

If the network control program is to send any unrecognized ID sequences to the host processor, code NOMATCH=PASS. If the NCP receives no ID sequence, this fact is indicated by the response returned to the access method.

If the NCP is not to send unrecognized sequences to the host processor, code NOMATCH=STOP. Then, upon receiving an ID sequence it does not recognize, the NCP breaks the line connection.

VIDLIST Macro Instruction (VTAM only)

The VIDLIST macro instruction specifies a list of identification sequences for BSC or TWX stations that call the communications controller over a switched line operated in network control mode. Either IDLIST or VIDLIST macros, or both, may be coded for each communication line within your network. Use VIDLIST if you want VTAM to check the received identification sequence; use IDLIST if you want the NCP to check the sequences; and use both macros if both programs are to check ID sequences.

This macro must appear between the SYSCNTRL macro and the first GROUP macro.

See the *ACF/VTAM Installation* manual for a complete description of the macro and its operand.

The format of the VIDLIST macro is:

```
[symbol]        VIDLIST     operands
```

Operands

```
VIDSEQ=((chars,termname),...)
```

LUPOOL Macro Instruction

The LUPOOL macro instruction specifies a pool of logical unit control blocks (LUBs) that the NCP uses in communicating with logical units associated with type 1 and type 2 physical units reached over switched SDLC links. (Logical units for physical units on nonswitched links are defined by LU macros associated with the PU macro that represents the physical unit.)

On establishing a connection with a switched type 1 or type 2 physical unit, the network control program allocates a LUB from the pool for each logical unit contained within that physical unit. The access method tells the NCP the number of LUBs to allocate and also supplies the logical unit parameters for each LUB. On completion of the transmission between the network control program and the physical unit, the program releases each LUB to the pool.

One LUPOOL macro is required for each access method that does not support the Request Network Address Assignment command and will communicate with type 1 and/or type 2 physical units on switched lines. Do not code an LUPOOL macro for any access method with request network address assignment command support. (See the LUDRPOOL macro.)

More than one access method may use the same logical unit pool as long as only one of them is active at a time. For example, three subareas specified in the HOST macro as SUBAREA=(2,3,5) could use the same logical unit pool because only one subarea (access method) can be actively communicating with the NCP at a time.

The format of the LUPOOL macro is:

```
[symbol]        LUPOOL        operands
```

Operands

```
NUMBER=count
```

VTAM-only operand:

```
OWNER=
```

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity. A name is required if you define more than one LUPOOL macro.

NUMBER=count

Specifies the number of logical unit control blocks (LUBs) to be included in the pool. The minimum is 1. The maximum is the total number of resources that can be defined in the network control program (as determined by the value chosen for the MAXSUBA operand of the BUILD macro) minus the number of resources defined in the LINE, PU, LU, and other LUPOOL macros.

For example, if the MAXSUBA operand specifies 31 subareas, the maximum number of resources controlled by the NCP is 2045. If the total number of resources (equivalent to the total number of macros listed above) is 1000, the maximum value you may specify in the NUMBER operand of all LUPOOL macros is 1045 (2045-1000).

To determine the highest value you should specify in this operand, assume that (1) the NCP is communicating simultaneously with type 1 and type 2 physical units over all of the switched links associated with this access method; and (2) the physical units involved are those having the highest number of logical units. The total number of logical units in all connected physical units represents the greatest possible demand that can be placed on the logical unit control block (LUB) pool. By specifying this value, you can ensure that the pool is never depleted.

In practice, the value of NUMBER can be less, since the assumptions above represent extreme conditions that seldom, if ever, occur. Specifying too low a value, however, may result in depletion of the pool. If, after a physical connection is established with a switched physical unit, the NCP is unable to supply sufficient LUBs from the pool, the program will break the physical connection before data transmission has begun. Then a new connection will have to be made and LUB allocation attempted again. Occasional occurrences of this kind may be tolerable, but frequent occurrences demonstrate the need to increase the size of the pool.

VTAM-Only Operand

The VTAM-only operand listed at the beginning of the LUPOOL macro description conveys no information to the NCP generation assembly process. This operand must appear in the NCP generation input deck that serves as input to the VTAM initialization process. See the *ACF/VTAM Installation* manual for the description of this operand and for information on the VTAM initialization process.

LUDRPOOL Macro Instruction

The LUDRPOOL macro defines two pools of null logical unit control blocks. One pool is for LUs added to type 1 physical units and the other is for LUs added to type 2 physical units. These logical unit control blocks are used when an LU is added to a physical unit by the dynamic reconfiguration process. When LUs are deleted, the storage allocated for their control blocks is returned to the pool for later use.

The LUDRPOOL macro is also used in place of the LUPOOL macro if the access method supports the Request Network Address Assignment (RNAA) command for SDLC switched links. If the NCP communicates with more than one access method, and all of the access methods include the RNAA command support, one LUDRPOOL macro should be coded to include all of the logical units in one pool.

Both LUDRPOOL and LUPOOL macros can be included in the same NCP generation. (See the description of the LUPOOL macro.)

This macro is used only if logical units are to be added to Type 1 or Type 2 physical units. Only one LUDRPOOL macro is allowed in a generation; it must precede the first GROUP macro.

The format of the LUDRPOOL macro is:

```
[symbol]        LUDRPOOL      [operands]
```

Operands

```
[NUMTYP1={count}]
         {0    }
[NUMTYP2={count}]
         {0    }
```

`[symbol]`

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

```
[NUMTYP1={count}]
         {0    }
```

Specifies the number of logical units to be included in the LU pool for type 1 PUs. The maximum number of entries is limited by the network resources defined by the MAXSUBA operand of the BUILD macro minus the number resources defined by the LINE, PU, LU, TERMINAL, LUPOOL, and PUDRPOOL macros.

To conserve controller storage, the count should be based on the anticipated dynamic resource requirements, not the maximum number of subareas.

```
[NUMTYP2={count}]
        {0    }
```

Specifies the number of logical units to be included in the LU pool for type 2 PUs. The maximum number of entries is determined the same as for the NUMTYP1 operand.

PUDRPOOL Macro Instruction

The PUDRPOOL macro defines a pool of null physical unit control blocks available to the NCP for the dynamic reconfiguration function. These physical unit control blocks are used when a PU is added to the network. When PUs are deleted, the storage allocated for their control blocks is returned to the pool for later use.

This macro is required if the dynamic reconfiguration function is to be used in the NCP. Only one PUDRPOOL macro is allowed in a generation and it must precede the first GROUP macro. This macro is valid only if ANS=YES in the GROUP macro.

The format of the PUDRPOOL macro is:

```
[symbol]        PUDRPOOL     [operands]
```

Operands

```
[NUMBER={count}]
        {0    }
[,MAXLU={count}]
        {1    }
```

```
[symbol]
```

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

```
[NUMBER={count}]
        {0    }
```

Specifies the number of physical units to be included in the physical unit pool. The maximum number of entries is the difference between the number of resources specified by the MAXSUBA operand of the BUILD macro and the number of resources defined by the LINK, PU, LU, TERMINAL, LUPOOL, and LUDRPOOL macros.

To conserve controller storage, the count should be based on the anticipated dynamic resource requirements, not the maximum number of subareas.

```
[MAXLU={count}]
       {1    }
```

Specifies the maximum number of logical units that can be dynamically added to one of the physical units in the dynamic reconfiguration pool. The maximum number of logical units is 255.

Note: The local address of any local unit added to a type 1 PU cannot exceed the value of *count* minus 1. The local address of any logical unit added to a type 2 PU cannot exceed the value of *count*.

PATH Macro Instruction

The PATH macro instruction defines, for an adjacent subarea, the destination subarea(s) with which the NCP can communicate via an adjacent subarea. The adjacent subarea may be an access method the program communicates with over a channel connection, or it may be a network control program reached by the present NCP over an SDLC link.

The destination subarea may be any nonadjacent subarea the NCP will communicate with. Every destination subarea must be represented by having its address specified in the DESTSUB operand of a PATH macro. No such address may appear in a HOST macro or in any other macro in the present NCP. You may code as many PATH macros as you need to specify all of the nonadjacent subareas.

Appendix K gives several examples of how PATH macros may be used to establish paths between the NCP you are defining and nonadjacent subareas elsewhere in the network.

The format of the PATH macro is:

```
[symbol]      PATH      operands
```

Operands

```
ADJSUB=(subarea1[,subarea2]...[subarean]),
DESTSUB=(subarea1[,subarea2]...[,subarean])
```

`[symbol]`

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

`ADJSUB=(subarea1[,subarea2]...[,subarean])`

Specifies the address(es) of one or more adjacent subareas through which path information units (PIU) to nonadjacent (destination) subareas will pass. (Only one of the subareas specified may communicate with the NCP at any given moment.)

If this ADJSUB operand represents an adjacent *local* network control program, specify the subarea address of that NCP. (An adjacent *remote* network control program cannot be specified in this operand.)

If this operand represents an adjacent access method, specify the subarea address of that access method. If the NCP being defined will communicate with any one of a set of access methods represented by a HOST macro (each access method having a different subarea address), specify all of the addresses in this ADJSUB operand exactly as they appear in the SUBAREA operand of the HOST macro. (If they are not specified identically, all destination subareas you specify in the DESTSUB operand will be associated with the first subarea in this ADJSUB operand.)

The smallest subarea address you may specify in this operand is 1; the largest is the value specified in the MAXSUBA operand of the BUILD macro.

If you are defining a local network control program, each subarea address in the ADJSUB operand must also be specified in the SUBAREA operand of either a HOST macro (for an adjacent access method) or a PU macro (for an adjacent NCP).

If you are defining a remote network control program, each subarea address specified in the ADJSUB operand must *not* appear in any other macro.

`DESTSUB=(subarea1[,subarea2]...[,subarean])`

Specifies the address(es) of one or more destination subareas the NCP being defined can communicate with via an adjacent subarea specified in the ADJSUB operand.

List each subarea in the network that (1) is not an adjacent subarea and (2) requires that the NCP communicate with it via one of the adjacent subareas.

The smallest subarea address you may specify in this operand is 1; the largest is the value specified in the MAXSUBA operand of the BUILD macro.

SDLCST Macro Instruction

The network control program can cause a controller to function as a primary station or a secondary station on an SDLC link attached to another controller.

If a controller is to function only as a primary station or only as a secondary station on a given SDLC link, you define the parameters for the link only in the LINE and PU macros.

If, however, a controller functions initially as a secondary station and then subsequently functions as a primary station, you use the SDLCST macro in addition to the LINE and PU macros to define parameters for the link. For each link, two SDLCST macros are required: one to define the parameters to be used when the controller becomes the primary station and one to define the parameters to be used when the controller again becomes the secondary station after having been the primary station. A SERVICE macro is not required for a secondary station even if the secondary might become a primary. Service order tables are automatically generated for secondary stations and are used if the secondary becomes a primary.

Example: Refer to Figure 5-8. The parameters specified in the LINE and PU macros govern operation of the link for as long as the present network control program operates controller A as a secondary station after NCP A is loaded and activated. These parameters are replaced by those you specify in the SDLCST macro for the primary station the first time (and each subsequent time, if any) controller A becomes the primary station on the link. Similarly, these parameters are in turn replaced by those you specify in the SDLCST macro for the secondary station the first time (and each subsequent time, if any) controller A subsequently becomes the secondary station.

Switching controller A from a secondary to a primary station on a link occurs only in the circumstances illustrated in Figure 5-8: local controller B, joined by a local-local link to controller A, becomes unable to communicate with its attached host processor and is reloaded (over the link) with a remote network control program. (Reloading over a link is possible only if controller A is equipped with a remote program loader.) Controller B was serving as the primary station on the link and now becomes the secondary station. (A remote controller can only be the secondary station on a link to a local controller.) Similarly, local controller A, formerly the secondary station, must now be the primary station on the link.

Each SDLCST macro defines one entry—set of parameters—in an SDLCST selection table. The entry represents one combination of parameters to be used by an SDLC link. To associate a table entry with a particular link, specify, in the SDLCST operand of the corresponding LINE macro, the name (*symbol)* of the SDLCST entry that defines the entry. More than one LINE macro may refer to the same set of parameters as defined in a pair of SDLCST macros.

Code one SDLCST macro for each unique combination of line control parameters needed. All SDLCST macros must precede the first GROUP macro in the program.

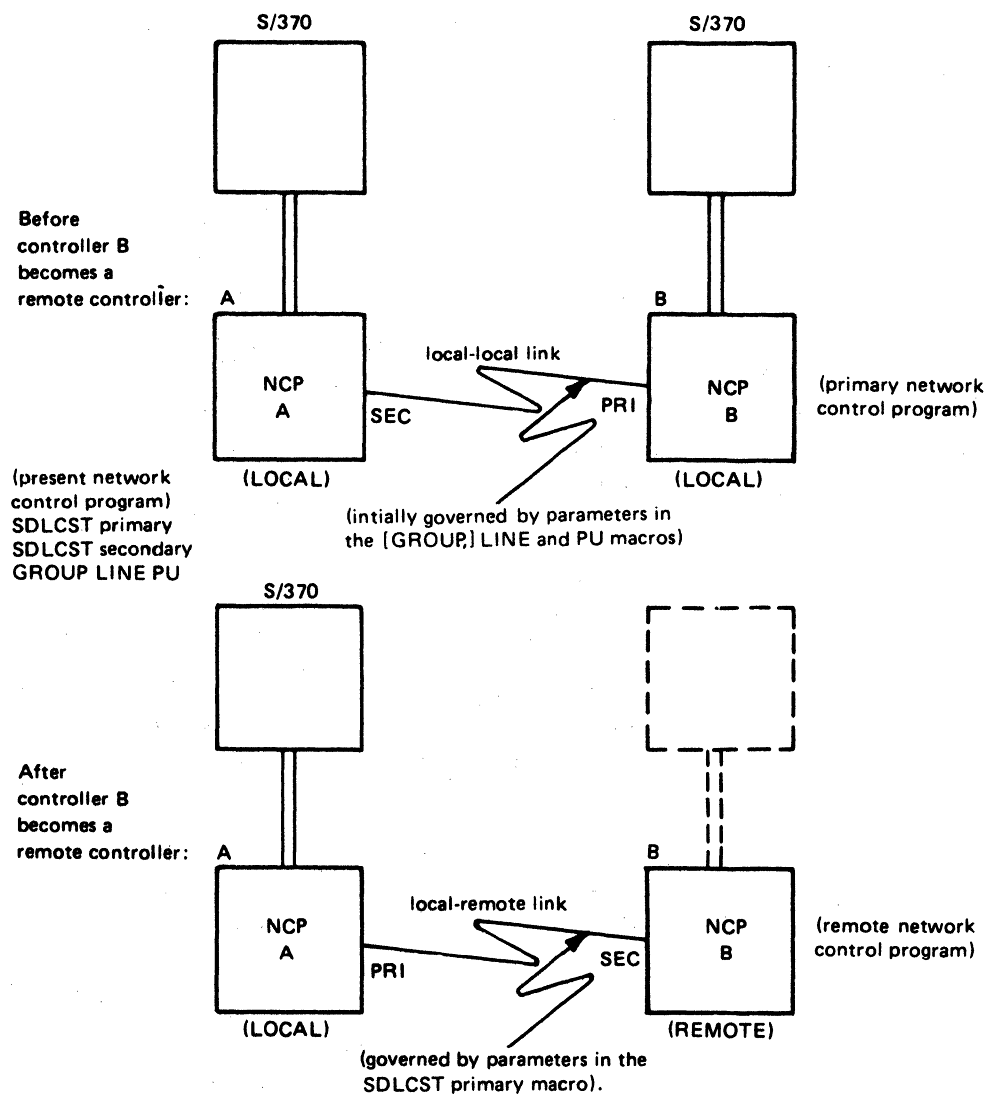


Figure 5-8. Exchange of Primary and Secondary Roles by 3705s on a Local-Local Link

The format of the SDLCST macro is:

```
symbol      SDLCST      operand[,operands]
```

Operands

```
GROUP=group name,
[,ADDR=chars]
[,IRETRY={YES}]
         {NO }
[,MAXOUT=n]
[,PASSLIM={n}]
          {1}
[,POLLED={YES}]
         {NO }
[,RETRIES={NONE        }]
          {(m[,t[,n]])}
[,SERVLIM={count}]
          {4    }
[,TADDR=chars]
[,TRANSFR=count]
```

(In the format above, NO under IRETRY, 1 under PASSLIM and 4 under SERVLIM are underlined on the page.)

`symbol`

Provides a name for the line control selection table entry. *symbol* may be any valid assembler-language symbol. This field is required and is referred to by the SDLCST operand of the LINE macro.

`GROUP=group name`

Specifies the name of the GROUP macro that contains the link parameters to be associated with the SDLC selection table entry defined by this SDLCST macro. The parameters of the GROUP macro associated with the table entry are POLLED, LNCTL, DIAL, ACTIVTO, and REPLYTO. (The PUTYPE operand, if coded in the GROUP macro, is ignored.) The GROUP macro must specify LNCTL=SDLC and DIAL=NO. Both the GROUP and the SDLCST macros must specify POLLED=YES or POLLED=NO, as appropriate.

group name may be the name of either a regular GROUP macro (followed by LINE and other lower-level macros) or a stand-alone GROUP macro (discussed at the beginning of the GROUP macro description).

`[ADDR=chars]`

Specifies the hexadecimal representation of the 8-bit address of the physical unit on the link. This operand is required (and is valid) only if you specify POLLED=YES in this macro. Addresses 00 and FF are invalid.

```
[IRETRY={YES}]
        {NO }
```

Specifies whether the network control program, when an idle detect time-out condition follows a polling operation, is to immediately retry the operation. If you specify IRETRY=YES, the program repolls the physical unit; otherwise, the program services the station represented by the next entry in the service order table.

This operand is valid only if you specify POLLED=YES in this macro. If you omit this operand and specify POLLED=YES, IRETRY=NO is assumed.

```
[MAXOUT=n]
```

Specifies the maximum number of path information units (PIUs) (or PIU segments if the program divides PIUs into segments) the NCP will send to the physical unit on the link before requesting a response from the physical unit.

The valid range for *n* is 1 to 7.

If you omit this operand, MAXOUT=1 is assumed if you specify POLLED=YES, or MAXOUT=7 is assumed if you specify POLLED=NO.

```
[PASSLIM={n}]
         {1}
```

Specifies the maximum number of consecutive path information units (PIUs) or PIU segments the NCP will send at one time to the physical unit on the link.

The minimum is 1 PIU or segment. The maximum is 254. The default value assumed if you omit this operand is 1 if POLLED=YES is specified and 254 if POLLED=NO is specified.

```
[POLLED={YES}]
        {NO }
```

Specifies whether the NCP being defined will cause the 3705 to function as a primary station (POLLED=YES) or a secondary station (POLLED=NO) relative to the link with which the entry specified by this macro is associated. Code POLLED=YES if the present network control program must poll and address the station(s) on that link. Code POLLED=NO if polling and addressing by the present NCP are not required.

```
[RETRIES={NONE          }]
         {(m[,t[,n]])}
```

Specifies the number of attempts to recover from errors occurring during transmission over the link associated with this SDLCST macro.

Note: (This note applies only to a local controller). To avoid losing contact with a remote controller while it accesses its disk (for example, when the remote controller storage is being dumped), specify RETRIES and REPLYTO in the GROUP macro for a minimum retry time of 30 seconds.

An error condition exists when the NCP does not receive a positive indication that a frame it has sent has been successfully received, and the NCP retransmits that frame. If the error recurs, retransmission is repeated. The NCP persists in retransmitting until it successfully sends the frame or until a maximum number of retries has been attempted.

When an error occurs while the network control program is receiving from a station, the NCP sends the station a command that causes the station to retransmit the frame in error. If the error recurs, the NCP repeats this command. The program persists in this way until (1) it has successfully received the frame, (2) it has retried the received operation a user-specified maximum number of times, or (3) the station sends an abnormal response; for example, a request for initialization.

The repetitive retries (retransmissions of data or control commands) are called a retry sequence. The maximum number of retries in the sequence is specified as *m*, which may be from 0 to 128. (RETRIES=0 is equivalent to RETRIES=NONE, resulting in no retry attempts at all.)

You may also specify that the network control program pause after completing the retry sequence and then begin a new retry sequence. This second sequence continues until the frame is successfully transmitted or received or the maximum, *m*, is again reached. Alternation or retry sequence and pause continues until the error is cleared or the maximum number of retry sequences is reached. The pause, specified by the *t* parameter, may be from 1 to 255 seconds. The maximum number of retry sequences, specified by *n*, may be from 1 to 127. See the RETRIES operand of the LINE macro for additional information on coding this operand.

```
[SERVLIM={count}]
         {4    }
```

Specifies the maximum number of regular scans of the service order table that the NCP will make for normal servicing of the physical unit on the link before it makes a special scan of the table. The regular scan of the table accommodates normal transmission of path information units between the access method or host application programs and the physical unit on the link. In the special scan of the service order table, the NCP determines whether there are any outstanding commands from the access method to interrogate or alter the online status of any physical units on the link. If so, the program fulfills the first such command and then resumes regular scans of the table to perform normal servicing. If no status commands are outstanding, the program immediately resumes regular scans unless, in the previous regular scan, the program found the physical unit was not in the contacted state (not presently active). In this case, resumption of regular scans occurs after a delay of 2.2 seconds. If more than one status command is outstanding, only one is honored each time the special scan is made; the remaining status commands are fulfilled one at a time, in turn, during subsequent special scans of the table.

Upon completing a regular scan, the program begins the special scan when one of the following occurs:

- in the regular scan just completed, the program found that the station was not active
- the maximum number of regular scans specified by SERVLIM has been reached

Specifying a low value in SERVLIM gives the NCP more frequent opportunities to fulfill accumulated status commands than does specifying a higher value. Such status commands can be fulfilled more promptly, but at the cost of frequent interruptions to normal servicing. Conversely, specifying a higher value in SERVLIM causes fewer interruptions to normal servicing of stations than does a lower value, but delayed fulfillment of the status commands is more likely to result. The relative number of status commands the access method will issue for the link served by the service order table, the relative importance of the alternatives described above, and experience should influence your selection of value for the SERVLIM operand.

CAUTION
The network control program will perform a time-out for any status command issued for the physical unit of an SDLC station whose power is off. The duration of this time-out interval is as specified in the REPLYTO operand of the GROUP macro.

If you omit this operand, and POLLED=YES is specified, a value of 4 (four regular scans of the service order table) is assumed.

This operand is valid only if you specify POLLED=YES in this macro.

`[TADDR=chars]`

Specifies the 2-digit hexadecimal representation of a single EBCDIC SDLC station address; that is, the address of the secondary statino (3705) in which the present NCP will be executed. You may assign as the address any bit configuration except hexadecimal 00 or FF. However, the address you specify in the PU macro within the NCP executing in the primary station (3705) must be identical to the address you specify in this TADDR operand.

`[TRANSFR=count]`

Specifies the number of NCP buffers corresponding to the maximum amount of data (the data transfer limit) that the network control program is to receive from the SDLC link associated with this SDLCST macro during a single data transfer operation for retransmission to a destination host processor.

See the TRANSFR operand of the LINE macro for information regarding use of this operand. References to the TRANSFR operand of the LINE macro should be construed to refer to the same operand in the SDLCST macro defining the line control selection table entry associated with the link.

DIALSET Macro Instruction (BSC and Start-Stop Lines Only)

The DIALSET macro instruction specifies the switched point-to-point lines that make up a dial set. (A dial set is the group of lines from which the NCP selects a line to call a station.) A dial set may contain any number of lines, but all must have similar characteristics and all must operate in network control mode. This allows the NCP to use any of the lines to call a station of a specific type. See Appendix J for an example of how this macro is used to establish operations over lines used for switched and multiple-terminal-access facilities.

Dial sets cannot be specified for SDLC links.

The format of the DIALSET macro is:

```
[symbol]        DIALSET      operand[,operands]
```

Operands

```
LINES=(line name,...)
[,DIALALT=dialset name]
[,QLIMIT={count}]
         {1    }
[,QLOAD={count}]
        {0    }
[,RESERVE={count}]
          {0    }
```

(Default values 1, 0, 0 are underlined.)

[symbol]

Provides a name for the dial set and is required except as indicated in the description of the LINES operand. *symbol* may be any valid assembler-language symbol; the first character may not be $. The name is referred to by the DIALSET operands of the LINE and TERMINAL macros and the DIALALT operands of the LINE and DIALSET macros.

LINES=(line name,...)

Specifies the switched lines of which the dial set is to consist; *line name* is the name of the LINE macro for a line to be included. Only a line whose LINE macro specifies CALL=OUT or CALL=INOUT may be included in a dial set. No line may appear in more than one dial set.

This operand is required.

You may specify a maximum of 255 characters in the LINES operand, including the beginning and ending parentheses and all commas. This limit applies regardless of how many line names you code within the operand. If you need more than 255 characters to complete the list, code one or more additional DIALSET macros (omitting the symbol) after the first. Code the remaining line names in the LINES operand. No other operands may be specified in the additional macros.

[DIALALT=dialset name]

Specifies a dial set that is an alternate to the dial set you are defining. *dialset name* is the name of the DIALSET macro for the alternate dial set. The alternate dial set must consist of the same type of lines as the primary dial set.

The DIALSET macro specified by the DIALALT operand must immediately follow the DIALSET macros defining this dial set. Omit the DIALALT operand from the last of a chain of DIALSET macros. Do not use the operand to specify the name of the first DIALSET macro in the chain.

```
[QLIMIT={count}]
        {1    }
```

Specifies the maximum number of requests the NCP will allow to accumulate on the queue for the dial set. When this limit is reached, the NCP rejects (returns to the host processor) any further call-out requests it receives. Rejection will continue until the number of requests becomes less than the queue limit value specified. The minimum queue limit is 0; the maximum is 255. 0 specifies that the NCP will reject *any* request when the program does not immediately have a line available.

```
[QLOAD={count}]
       {0    }
```

Specifies the number of call-out requests the NCP will allow to accumulate on the queue for this dial set before using a line from the alternate dial set. If you code QLOAD=0 (or omit the operand), and an alternate dial set is specified, the program uses a line from the alternate dial set if no line in the primary dial set is available.

The maximum number of requests is 255. The value specified in QLOAD must be less than the value in QLIMIT; otherwise the queue of unfulfilled call-out requests cannot use the alternate dial set.

```
[RESERVE={count}]
         {0    }
```

Specifies the number of lines in the dial set to be reserved for incoming calls from stations. If you code RESERVE=0 no lines will be reserved. When all lines are busy with outgoing calls, no stations will be able to call the communications controller.

The maximum value for RESERVE is 255.

MTALCST Macro Instruction

If any switched lines in the network are specified as multiple-terminal-access (MTA) lines for start-stop terminals, a line control selection table is generated within the network control program. MTALCST macros are used to define entries in the table. Each entry represents a particular combination of terminal operating parameters for a terminal that may call the NCP on an MTA line. The MTALCST macro specifies terminal operating parameters such as line speed, transmission code, type of line control, length of print line, carriage return rate, text error retry limit, buffer cutoff limit and a mask and a compare character for TWX terminals.

Code one MTALCST macro for each distinct combination of parameters. All MTALCST macros must be grouped together in a single sequence, preceding all MTALIST and MTATABL macros. The maximum number of MTALCST macros you may code is 63.

See Appendix J for an example of how this and other MTA macros are used to establish multiple-terminal-access operations.

The format of the MTALCST macro is:

```
symbol        MTALCST      operands[,operands]
```

Operands

```
GROUP=entry,
SPEED=rate
[,ACR={YES}]
      {NO }
[,CLOCKNG={INT}]
          {EXT}
       {BCD  }
       {BCD2 }
       {EBCD }
[,CODE={COR  }]
       {COR2 }
       {DIC1 }
       {DIC3 }
       {ASCII}
[,COMPARE=chars]
[,CRRATE=integer]
[,DATRATE={HIGH}]
          {LOW }
[,INTPRI={0}]
         {1}
```

```
            { 1050 }
            { 2740A}
[,LCTYPE={ 2740D}]
            { 2740E}
            { 2740F}
            { 2741 }
            { 3767 }
            { TWX  }
[,LINESIZ=integer]
[,MASK=chars]
             {m    }
[,RETRIES={2    }]
             {NONE}
[,TRANSFR=count]
```

symbol

Provides a name for the line control selection table entry defined by this macro and is required. *symbol* may be any valid assembler-language symbol; the first character may not be $.

symbol is referred to by the LCST operand of the MTATABL macro.

GROUP=entry

Specifies the name of the GROUP macro for any line group whose terminal characteristics are the same as the characteristics of the terminal that will call the controller over the multiple-terminal-access line. *entry* must not be the name of a GROUP macro that represents a group of multiple-terminal-access lines.

This operand is required.

SPEED=rate

Specifies the data rate (in bits per second) for the multiple-terminal-access lines associated with this entry.

If CLOCKNG=INT, this rate must be one of the four oscillator rates specified for the communication scanner the lines are attached to (SPEED operand of the CSB macro). Specify the line speed in bits per second, omitting any fractional part. For example, specify a line speed of 110 bps as SPEED=110; specify a line speed of 134.5 bps as SPEED=134 (omitting the decimal point and fraction).

If CLOCKNG=EXT, this rate must be the clocking rate of the modem attached to the line (which is not necessarily one of the oscillator bit rates specified for the scanner). However, the SPEED operand of the CSB macro for the scanner must specify a scanner bit rate *less than one-half* of the modem clocking rate you specify in this SPEED operand.

This operand is required.

```
[ACR={YES}]
     {NO }
```

Specifies whether the 1050 terminals that may call the controller on this line are equipped with the accelerated carrier return feature. Specify ACR=YES only if all 1050 terminals that may call the controller over this line are equipped with the feature.

```
[CLOCKNG={INT}]
         {EXT}
```

Specifies whether the communication scanner or the modem (data set) is to provide clocking. This may be determined from the system designer.

If the scanner provides clocking, code CLOCKNG=INT. If the modem (whether external to or contained within the controller) provides clocking, code CLOCKNG=EXT.

```
       {BCD  }
       {BCD2 }
       {EBCD }
[CODE={COR  }]
       {COR2 }
       {DIC1 }
       {DIC3 }
       {ASCII}
```

Specifies the transmission code used to communicate with the type of terminal represented by this macro. Associated with each transmission code is a translation table in the network control program. The contents of the translation tables are defined in the *ACF/NCP Program Reference Summary.*

Valid transmission codes for each type of terminal are as follows. (The underscored values indicate the code assumed if you omit this operand is omitted.)

Terminal	CODE=	Transmission Code
IBM 1050	EBCD	Extended BCD code
	BCD	BCD code 1
	BCD2	BCD code 2
IBM 2740	EBCD	
	BCD	
	COR	Correspondence code 1
IBM 2741	COR	
	COR2	Correspondence Code 2
	BCD	
	BCD2	
	EBCD	
TWX	DIC1	Data interchange code 1
	DIC3	Data interchange code 3
	ASCII	American Standard Code for Information Interchange

[,COMPARE=chars]

(TWX terminals only)

Specifies the hexadecimal representation of the compare character used during the LCST entry selection process. The mask character specified by the MASK operand is logically ANDed with the first character recieved from the terminal. If the result equals the compare character specified by this COMPARE operand, the contents of the LCST entry defined by this MTALCST macro are used to initialize the operating characteristics for the TWX terminal.

chars should be specified as 2 hexadecimal digits.

The following table contains recommended pairs of values for the COMPARE and MASK operands.

TWX Terminal Speed	First Received Character	COMPARE=	MASK=
110 baud	@[1]	00	7F
	N	1E	7F
	0	60	7F
	∧[2]	3E	7F
300 baud	@[1]	78	7B
	N	79	7B
	0[1]	7A	7B
	∧[2 3]	7B	7B

If no value is specified for the COMPARE operand, a value of 01 is used. This value will, along with the default value for the MASK operand, properly identify the WRU character transmitted at 110 baud.

1 The letter P is used for emulation mode speed selection and may be desired for compatability reasons. If the letter P is used for 110 baud, specify COMPARE=00, MASK=5F, and do not use the character @ and 0.

2 ASCII X'5E' character. This character is represented on terminal keyboards as either ∧, ¬, or ↑.

3 This character should not be used if a 3767 can call on the same line as a TWX terminal, because NCP cannot distinguish between this character and the initial end-of-address (circle D) character automatically sent by the 3767.

[CRRATE=integer]

Specifies the number of print positions that the carriage printers return for each idle character the network control program sends. The minimum is 1; the maximum is 255.

If you omit this operand, a rate of 40 print positions per idle character is assumed for TWX terminals; 13 is assumed for IBM 1050 terminals equipped with the accelerated carrier return feature; and 10 is assumed for IBM 2740 and 2741 terminals and for 1050 terminals without the ACR feature.

```
[DATRATE={HIGH} ]
        {LOW }
```

Specifies the data rate used on the modem (data set) that attaches the multiple-terminal-access line to the controller. This operand is valid only if the modem has a dual data rate.

Code DATRATE=HIGH if the high data rate is to be used. Code DATRATE=LOW (or omit the operand) if the low data rate is to be used.

Note: DATRATE=HIGH is invalid for modems attached to line sets 1A, 1B, 1C, 2A, 3A, 4A, 4B, and 4C and, if specified, may cause a feedback-check error condition.

If the modem has only one data rate, specify DATRATE=LOW or omit the operand.

```
         { 1050 }
         { 2740A}
[LCTYPE={ 2740D}]
         { 2740E}
         { 2740F}
         { 2741 }
         { 3767 }
         { TWX  }
```

Specifies the type of terminal and line control used.

Entry	*Type*
1050	IBM 1050
2740A	IBM 2740 basic
2740D	IBM 2740 with transmit control feature
2740E	IBM 2740 with transmit control and checking features
2740F	IBM 2740 with checking feature
2741	IBM 2741
3767	IBM 3767 in 2741 compatability mode
TWX	Western Union TWX

```
[LINESIZ=integer]
```

Specifies the length of the print line, in number of print positions, for printer-type devices connected to the line represented by this MTALCST macro. The minimum value for *integer* is 1; the maximum is 255.

If you omit this operand, a line length of 72 print positions is assumed for TWX terminals; a line length of 130 is assumed for 1050, 2740, and 2741 terminals.

```
[MASK=chars]
```

(TWX terminals only)

Specifies the hexadecimal representation of the mask character used during the LCST entry selection process. The mask character is logically ANDed with the first character received from the terminal. If the result equals the compare character specified by the COMPARE operand, the contents of the LCST entry are used to initialize the operating characteristics for the TWX terminal.

chars should be specified as 2 hexadecimal digits. The mask character must never have a binary 0 specified in the same bit position that the compare character has a binary 1 specified. Refer to the COMPARE operand description for a table of recommended mask and compare character pairs.

If no value is specified for the MASK operand, a value of 31 is used. This value will, along with the default value for the COMPARE operand, properly identified the WRU character transmitted at 110 baud.

```
            {m    }
[RETRIES={2    }]
            {NONE}
```

Specifies the number of attempts to recover from text errors in message data sent to or received from an IBM 1050 or an IBM 2740 with record checking. (Other types of multiple-terminal-access terminals are not capable of retransmission.) *m* is the number of attempts, from 1 to 255. Any value less than 255 specifies the exact number of attempts; 255 specifies unlimited attempts.

If you code RETRIES=NONE, no error recovery is attempted for read-text or write-text errors.

If you omit the operand, a maximum of two recovery attempts will be made for terminals capable of retransmission. If the terminal is not capable of retransmission, *m* is assumed to be 0 and no recovery is attempted.

```
[TRANSFR=count]
```

Specifies a limit on the number of buffers that the NCP will obtain to receive message text from a terminal before transferring filled buffers to the host processor. If the NCP receives the specified number of buffers, it transfers them to the host processor as a sub-block but continues to receive message text from the station until it receives an end-of-block or end-of-transmission character.

The minimum value of *count* is 1. The maximum is the smaller of the following values:

1. 255 or
2. the result of multiplying the values of the MAXBFRU and UNITSZ operands of the HOST macro, subtracting the value of the BFRPAD operand of the HOST macro, and dividing the result by the buffer size specified in the BFRS operand of the BUILD macro.

This calculation of *count* can be expressed as a formula:

$$\text{count} \leq \frac{\text{(MAXBFRU)(UNITSZ)-BFRPAD}}{\text{BFRS}}$$

If you omit this operand, the network control program uses the smaller of 255 or the result of the calculation expressed by the preceding formula.

MTALIST Macro Instruction

After a call has been established on a multiple-terminal-access (MTA) line, the NCP uses a list of terminal identification procedures defined by an MTALIST macro to determine the terminal type of the calling terminal. All terminal types which may call the MTA line should be specified by the MTALIST macro.

Each MTA line is required to refer to an MTALIST macro, by the MTALIST operand of the LINE macro. The same list may be referred to by more than one LINE macro if the lines need the same list of terminal identification procedures.

MTALIST macros must be coded immediately after the MTALCST macros. See Appendix J for an example of how MTALIST and other MTA macros are used.

The format of the MTALIST macro is:

```
symbol          MTALIST      operands
```

Operands

```
LCTYPE=(type,...)
```

`symbol`

Provides the required name of the multiple-terminal-access list referred to by the MTALIST operand of the LINE macro. *symbol* may be any valid assembler-language symbol; the first character may not be $.

`LCTYPE=(type,...)`

Specifies the terminal types to be included in a list of terminal identification procedures. This list may be referred to by the MTALIST operands of one or more LINE macros of MTA lines.

Any of the following types may be specified, singly or in combination:

LCTYPE=	*Type of Terminal*
1050	IBM 1050
2740A	IBM 2740 (basic)
2740D	IBM 2740 with transmit control feature
2740E	IBM 2740 with transmit control and record checking features
2740F	IBM 2740 with record checking feature
2741	IBM 2741
3767	IBM 3767 in 2741 compatability mode
TWX	Western Union TWX

MTAPOLL Macro Instruction

The MTAPOLL macro instruction specifies the polling characters used by IBM 1050 terminals that may call the controller over any call-in multiple-terminal-access line. Both common and specific polling characters may be included.

Only polling characters for 1050 terminals that will call the controller over a call-in multiple-terminal-access line need be specified.

Only one MTAPOLL macro may be specified in the NCP generation input statements.

The format of the MTAPOLL macro is:

```
[symbol]        MTAPOLL       operands
```

Operands

```
POLL=(chars,...)
```

`[symbol]`

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

`POLL=(chars,...)`

Specifies the list of polling characters. Code the hexadecimal representation of the EBCDIC polling characters.

For example, if the network includes four IBM 1050 terminals, each having a keyboard and tape reader as input devices and the polling characters for the eight devices are A5, A6, B5, B6, C5, C6, D5, D6, code POLL=(C1F5,C1F6,C2F5,C2F6,C3F5,C3F6,C4F5,C4F6). Each of the eight hexadecimal sequences represents one of the EBCDIC polling sequences.

MTATABL Macro Instruction

For each unique combination of line control discipline and transmission code specified by an MTALCST macro, an MTATABL macro is required. In this macro you specify the line control and code used, and the name of each MTALCST macro that defines a set of operating parameters for that particular line control and code.

After a call has been established on a multiple-terminal-access (MTA) line and the NCP has identified the terminal type, the terminal operator can select a unique combination of terminal operating characteristics from the MTA table. The table is composed of names of 1 to 10 MTALCST macros, each of which has specified a unique combination of terminal operating characteristics. However, the table can only contain names of MTALCST macros which have the same line control and code combination as specified on the MTATABL macro.

For IBM terminals, selection of the appropriate combination of terminal operating characteristics is determined from the repeated character entered by the terminal operator during the terminal sign-on. For TWX terminals, selection is based upon the results of applying mask and compare characters to the first character entered by the TWX terminal operator. The mask and compare characters used are those specified for MTALCST macros named by the LCST operand and their order of use is determined by the order that the MTALCST macros are named.

For example, if you have coded a single MTALCST macro represent a 2741 using BCD code,

```
MTA1     MTALCST LCTYPE=2741,CODE=BCD,...
```

you would code one MTATABL macro that specifies the same line control and code:

```
         MTATABL      LCST=(MTA1),LCTYPE=2741,CODE=BCD
```

If, on the other hand, you have coded two MTALCST macros, both representing a 2741 using BCD code but each defining a different set of operating parameters,

```
MTA1     MTALCST LCTYPE=2741,CODE=BCD,LINESIZ=72,...
MTA2     MTALCST LCTYPE=2741,CODE=BCD,LINESIZ=90,...
```

you would code an MTATABL macro that specifies the same line control and transmission code and that names both MTALCST macros:

```
         MTATABL      LCST=(MTA1,MTA2),LCTYPE=2741,
                      CODE=BCD
```

Code only one MTATABL macro for a given combination of line control type and transmission code. In one MTATABL macro you may specify the names of up to ten MTALCST macros.

See Appendix J for an example of how this and other MTA macros are used to establish multiple-terminal-access operations.

The format of the MTATABL macro is:

```
[symbol]        MTATABL      operand[,operands]
```

Operands

```
LCST=(mtalcst name,...)
        {BCD }
[,CODE={EBCD}]
        {COR }
          { 1050 }
          { 2740A}
          { 2740D}
[,LCTYPE={ 2740E}]
          { 2740F}
          { 2741 }
          { 3767 }
          { TWX  }
```

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check it for validity.

LCST=(mtalcst name,...)

Specifies the name of one to ten MTALCST macros.

Only those MTALCST macros that have the same combination of code and line control as this MTATABL macro may be named. This restriction does not apply for TWX line control. Any MTALCST macro with LCTYPE=TWX specified may be named here regardless of whether its CODE operand has been specified as CODE=DIC1, CODE=DIC3, or CODE=ASCII.

COR2 is considered equivalent to COR and BSC2 is considered equivalent to BCD for purposes of this operand specification. For example, suppose that you have specified two MTALCST macros with LCTYPE=2741, but one with CODE=COR and the other with CODE=COR2. If you specify LCTYPE=2741 and CODE=COR for this MTATABL macro, then both MTALCST macros may be named on this operand.

```
      {BCD }
[CODE={EBCD}]
      {COR }
```

Specifies the transmission code used by the type of terminals represented in this table. Specify the same type of code as you specified in the MTALCST macro named in the LCST operand of this macro.

Do not specify this operand if LCTYPE=TWX is specified.

This operand defaults according to the LCTYPE specification (or LCTYPE default) as follows:

LCTYPE=	*CODE DEFAULT*
1050	EBCD
2740	EBCD
2741	COR
3767	COR

```
         { 1050 }
         { 2740A}
         { 2740D}
[LCTYPE={ 2740E}]
         { 2740F}
         { 2741 }
         { 3767 }
         { TWX  }
```

Specifies the type of line control used by the terminals represented in this table. Specify the same line control type as you have specified in the MTALCST macros named in the LCST operand of this MTATABL macro.

Network Configuration Macro Instructions

A network configuration may be viewed as a logical arrangement of elements, each kind of element occupying a different hierarchical level within the arrangement. Each element is represented by a specific macro instruction.

The highest element is the *communication line*. Each line in the network is represented by a LINE macro. This is true whether start-stop, BSC, or SDLC stations are connected to the line and whether the line is nonswitched multipoint, nonswitched point-to-point, or switched point-to-point.

If the line is to operate only in emulation mode, no other macros are required to represent the stations connected to the line. Operation in network control mode, however, requires that the LINE macro be followed by a macro for each of the remaining elements in the hierarchy.

Start-Stop Terminals and Nonclustered BSC Stations

Each start-stop terminal and each BSC station (exclusive of cluster-type stations) is generally represented by a TERMINAL macro. Each pollable or addressable component of a terminal may be represented by a COMP macro.

Consider a network consisting of three lines, six terminals, and seventeen components arranged as shown in Figure 5-9.

If all the lines are to operate only in emulation mode, only LINE macros are required to represent the network:

Macro:	*Represents:*
LINE	Line 1
LINE	Line 2
LINE	Line 3

On the other hand, if all lines are to operate in network control mode, whether they may sometimes operate instead in emulation mode, TERMINAL macros are required as well:

```
 Macro:          Represents:

LINE            Line 1
  TERMINAL        Terminal A, components A1, A2, A3, A4
  TERMINAL        Terminal B, components B1, B2
  TERMINAL        Terminal C, components C1, C2
LINE            Line 2
  TERMINAL        Terminal D, components D1, D2
LINE            Line 3
  TERMINAL        Terminal E, components E1, E2, E3, E4, E5
  TERMINAL        Terminal F, components F1, F2
```

The foregoing macro sequence sufficiently repesents the network if only one polling sequence and one addressing sequence are required for each station. This is the case if the station is a computer; a transmission control unit (such as an IBM 2701); a terminal having only one input component and one output component, typically a keyboard and a printer (for instance, IBM 2741) or a keyboard and a display screen (for instance, IBM 2260). One TERMINAL macro also suffices to represent a terminal having multiple input and/or output components if only a general polling and a general addressing sequence are required.

If, however, more than one polling or addressing sequence, or both, are required, each additional pair of polling and addressing sequences must be specified by a COMP macro. The conditions under which COMP macros may be required are explained further in the description of the COMP macro in this chapter.

Each COMP macro can represent one input component and one output component. Therefore, assuming that components A3 and E3 are input (polled) components and A4 and E4 are output (addressed) components, one COMP macro for each TERMINAL macro is required to accommodate these added components. The third additional component of terminal E (component E5) requires another COMP macro.

Assume now that individual polling and addressing sequences are needed for each terminal in the configuration of Figure 5-9. Three COMP macros must be added to the preceding macro sequence, resulting in this sequence:

```
Macro:                  Represents:

LINE                    Line 1
  TERMINAL                Terminal A, components A1, A2
    COMP                    Components A3, A4
  TERMINAL                Terminal B, components B1, B2
  TERMINAL                Terminal C, components C1, C2
LINE                    Line 2
  TERMINAL                Terminal D, components D1, D2
LINE                    Line 3
  TERMINAL                Terminal E, components E1, E2
    COMP                    Components E3, E4
    COMP                    Component E5
  TERMINAL                Terminal F, components F1, F2
```

COMP macros should be coded only if needed to specify polling and addressing sequences beyond the first polling and addressing sequences specified in the TERMINAL macro. Avoiding COMP macros where possible conserves storage space within the communications controllers.

Clustered BSC Stations

The preceding arrangement differs for clustered BSC stations such as the IBM 3270 and 2972. For such stations, the macro sequence is LINE, CLUSTER, and TERMINAL, rather than LINE, TERMINA, and COMP.

Consider, for example, a line connected to two terminal control units (for example, 2972), to each of which are attached three terminals (for example 2980).

You would represent this configuration by the sequence:

```
LINE
  CLUSTER
    TERMINAL
    TERMINAL
    TERMINAL
  CLUSTER
    TERMINAL
    TERMINAL
    TERMINAL
```

The highest level in the hierarchy (communication lines) is represented by the LINE macro, the next level (cluster controller) by CLUSTER macros, and the lowest by the TERMINAL macros.

SDLC Stations

The macro hierarchy for SDLC stations other than communications controllers (for example, IBM 3600, 3767, 3771, 3776) is LINE, PU, LU, for nonswitched links and LINE, PU for switched links. A single PU macro represents the physical unit within the SDLC station.

If the NCP communicates with the physical unit over a *nonswitched* SDLC link, one or more LU macros following the PU macro represent the logical units associated with the physical unit. (A logical unit is an application program within the physical unit together with the devices associated with that program.)

For example, if one physical unit having four logical units were attached to a nonswitched SDLC link, the following sequence would be required:

```
LINE
  PU
    LU
    LU
    LU
    LU
```

If the NCP communicates with the physical unit over *switched* facilities, the logical units are not represented by LU macros, and the macro sequence is simply LINE, PU. The NCP control blocks representing logical units are allocated dynamically from the pool of control blocks defined by the LUPOOL and LUDRPOOL macros.

Communications Controller Attached to SDLC Link

Each communications controller connected by an SDLC link to another communications controller is represented by a PU macro following the LINE macro.

In NCP for controller 'A':

LINE
    PU *(represents controller 'B')*

In NCP for controller 'B':

LINE
    PU *(represents controller 'A')*

Line Groups

To the hierarchies of LINE, PU, LU, TERMINAL, and COMP macros must be added one or more GROUP macros. The GROUP macro, unlike the others, represents not a physical element of the network but a grouping of lines having certain characteristics in common. The grouping is referred to as a *physical* line group because it contains lines having certain physical attributes in common, such as the type of stations attached to them.

Each LINE macro in the foregoing macro sequences must be associated with a GROUP macro that precedes it. Assume, for example, that in the configuration shown in Figure 5-9, the first two lines have similar attributes allowing them to be in the same line group, but that the third line has different characteristics, which require it to be in a different line group. (A single line can constitute a line group.) Further, although not shown above, a SERVICE macro must directly follow each LINE macro representing a line on which polling and addressing are required. Since this is true of all three lines in the example, each requires a SERVICE macro.

The complete macro sequence would then be:

```
GROUP
    LINE
    SERVICE
        TERMINAL
            COMP
        TERMINAL
        TERMINAL
    LINE
    SERVICE
        TERMINAL
GROUP
    LINE
    SERVICE
        TERMINAL
            COMP
            COMP
        TERMINAL
```

If all of the lines were dissimilar, each of the three LINE macros would have to be preceded by a GROUP macro. If all were alike, all could be in the same line group, with only one GROUP macro immediately preceding the first LINE macro.

For the clustered BSC stations, adding the GROUP and SERVICE macros gives the sequence:

```
GROUP
    LINE
    SERVICE
        CLUSTER
            TERMINAL
            TERMINAL
            TERMINAL
        CLUSTER
            TERMINAL
            TERMINAL
            TERMINAL
```

Adding the GROUP and SERVICE macros to the LINE, PU, LU sequence shown above under *SDLC Stations* results in:

```
GROUP
    LINE
    SERVICE
        PU
            LU
            LU
            LU
            LU
```

Similarly, the SDLC link between the communications controllers would be represented as follows, assuming that the NCP in controller 'A' is to control (perform polling and addressing on) the link:

In NCP for controller 'A':

```
GROUP
     LINE
     SERVICE
          PU
```

In NCP for controller 'B':

```
GROUP
     LINE
          PU
```

Notice that the SERVICE macro appears only in the NCP that controls the link (its communications controller is the primary station on the link). The SERVICE macro must be omitted following the LINE macro in the other NCP, executed in the controller serving as the secondary station on the link.

Note: For a local-remote SDLC link, the local controller is always the primary station and the remote controller is always the secondary station on the link. For a local-local SDLC link, either of the controllers may serve as the primary station; the other controller then serves as the secondary station on the link.

See the description of the GROUP macro later in this chapter for the attributes that communication lines must have in common in order to appear within the same line group.

Example of Network Configuration Macro Instructions

Refer to the network configuration shown in Figure 5-10. The network attached to the local communications controller comprises the following communication lines:

Line 1: Nonswitched multipoint, five IBM 1050 terminals—each having keyboard/printer and one 1056 card reader (start-stop terminals)

Lines 2 and 3: Nonswitched point-to-point lines, each communicating with an IBM 2701 (nonclustered BSC station)

Line 4: Switched point-to-point line over which the local communications controller can call two IBM 1130s and one IBM 2780 (nonclustered BSC stations)

Line 5: Nonswitched multipoint SDLC link, two IBM 3600 systems, with four logical units each (SDLC stations)

Line 6: Nonswitched point-to-point SDLC link, one 3705 communications controller (remote communications controller)

The network attached to the remote communications controller comprises:

Line R1: Nonswitched multipoint line, two IBM 3270 systems, each comprising a 3271 controller, four 3277 displays, and a 3284 printer.

Line R2: Switched point-to-point line over which any number of IBM 2741 terminals can call the remote controller.

Line R3: Nonswitched multipoint SDLC link, three IBM 3790 systems, with three logical units each.

The network configuration macros required to represent the network attached to the *local* communications controller are as follows (not all required operands are shown):

```
G1     GROUP LNCTL=SS
L1        LINE   POLLED=YES,TERM=1050
S1        SERVICE ORDER=(T1,C1,T2,C2,T3,C3,T4,C4,T5,C5)
T1           TERMINAL
C1              COMP
T2           TERMINAL
C2              COMP
T3           TERMINAL
C3              COMP
T4           TERMINAL
C4              COMP
T5           TERMINAL
C5              COMP

G2     GROUP LNCTL=BSC
L2        LINE   TERM=2701
T6           TERMINAL
L3        LINE   TERM=2701
T7           TERMINAL

G3     GROUP LNCTL=BSC,DIAL=YES
L4        LINE
T8           TERMINAL TERM=1130
T9           TERMINAL TERM=1130
T10          TERMINAL TERM=2780

G4     GROUP LNCTL=SDLC
L5        LINE   POLLED=YES
          SERVICE ORDER=(PU1,PU2)
PU1          PU PUTYPE=2
LU1             LU
LU2             LU
LU3             LU
LU4             LU
PU2          PU PUTYPE=2
LU5             LU
LU6             LU
LU7             LU
LU8             LU

G5     GROUP LNCTL=SDLC
L6        LINE   POLLED=YES
S2        SERVICE ORDER=(PU3)
PU3          PU PUTYPE=(4,REMOTE)
```

The network control program defined for the *remote* communications controller would include:

```
RG1    GROUP LNCTL=BSC
RL1       LINE   POLLED=YES,CUTYPE=3271
RS1       SERVICE ORDER=(RCL1,RT1,RT2,RT3,RT4,RT5,RCL2,RT6, X
                RT7,RT8,RT9,RT10)
RCL1         CLUSTER TERM=3277
RT1             TERMINAL
RT2             TERMINAL
RT3             TERMINAL
RT4             TERMINAL
RT5             TERMINAL TERM=3284
RCL2         CLUSTER TERM=3277
RT6             TERMINAL
RT7             TERMINAL
RT8             TERMINAL
RT9             TERMINAL
RT10            TERMINAL TERM=3284
```

```
RG2     GROUP LNCTL=SS
RL2        LINE
RT11          TERMINAL TERM=2741,CTERM=YES
RG3     GROUP LNCTL=SDLC
RL3        LINE   POLLED=YES
RS3        SERVICE ORDER=(RPU1,RPU2,RPU3)
RPU1          PU PUTYPE=2
RLU1A            LU
RLU1B            LU
RLU1C            LU
RPU2          PU PUTYPE=2
RLU2A            LU
RLU2B            LU
RLU2C            LU
RPU3          PU PUTYPE=2
RLU3A            LU
RLU3B            LU
RLU3C            LU
RG0     GROUP LNCTL=SDLC
RL0        LINE   POLLED=NO
RN0           PU PUTYPE=(4,LOCAL)
```

These network examples include each of the types of hierarchical arrangements, as well as the three types of line connections available: nonswitched point-to-point, nonswitched multipoint, and switched point-to-point.

See Appendix I for other sample programs that illustrate macro coding sequences.

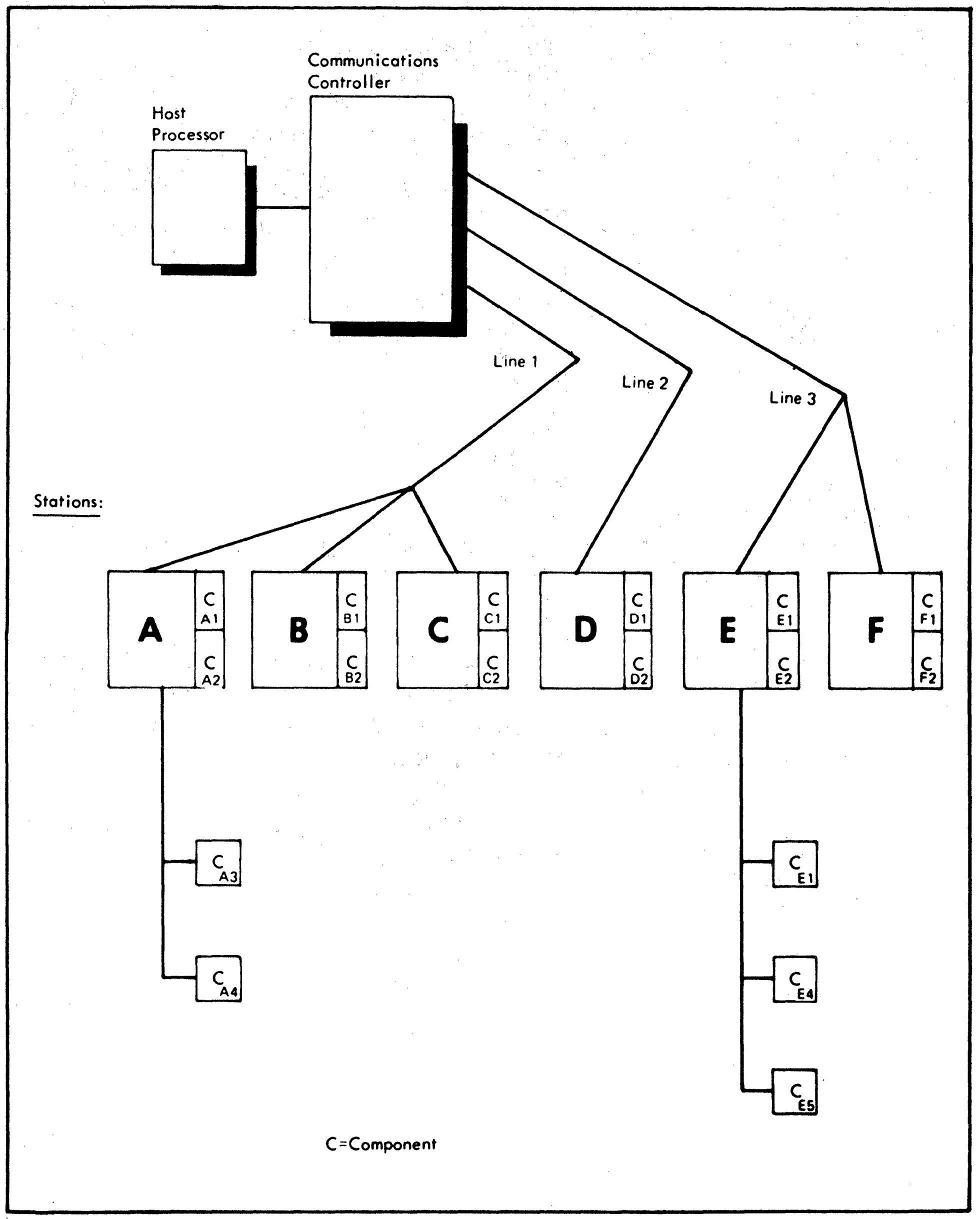


Figure 5-9. Example of Line Groups

Coding Operands on Higher Level Macros

The principal advantage of coding operands on higher level macros is that it saves coding effort. For example, five characteristics common to fifteen terminals on a line can be specified once in the LINE macro rather than in each of the fifteen individual TERMINAL macros. Five operands are coded instead of 75 (15 x 5).

The characteristic need not be identical for *all* of the elements at a level to specify it at a higher level. You may code the exceptions at the lower level. Any characteristic you code at the lower level automatically overrides the characteristic specified at a higher level.

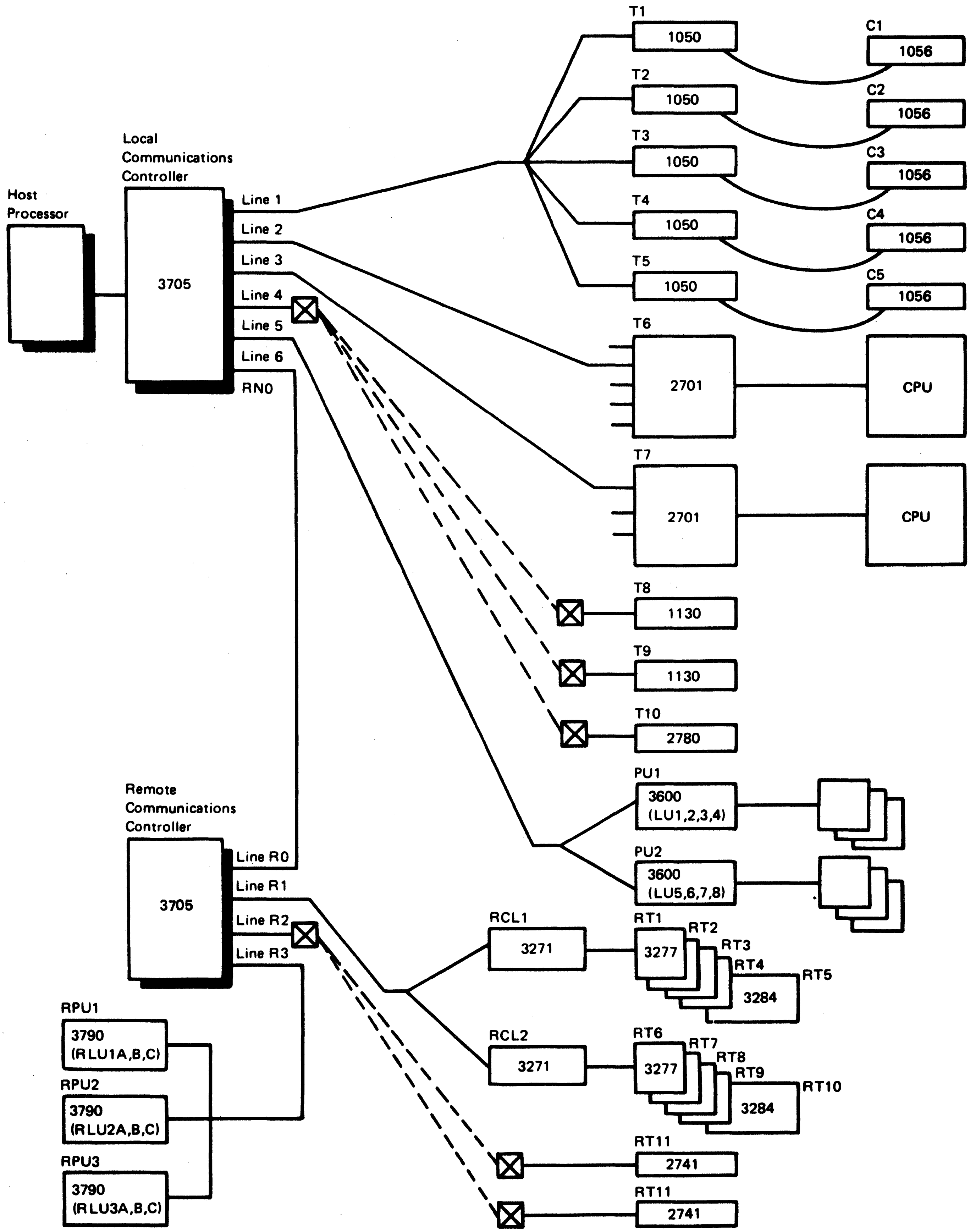


Figure 5-10. Example of Network Configuration

If, for example, only twelve of the fifteen terminals mentioned previously have the same characteristic in common, you could still specify that characteristic in the LINE macro; then you would specify the differing characteristics in the TERMINAL macros for the three exceptional terminals.

If you code one parameter in an operand that can have two or more, that operand completely overrides an equivalent operand specified in a higher-level macro. The default values are assumed for the parameters omitted at the lower level.

Summary of Macro Instructions and Operands

Figure 5-11 lists all of the operands of the network configuration macros. The asterisk (*) indicates the macro where the description of the operand appears. The bullet (•) indicates other macros in which the operand may be coded.

Note: This summary does *not* indicate the conditions (for example, type of line control, type of station) under which use of the operand is appropriate. For this information, see the individual macro descriptions later in this chapter.

VTAM-Only Operands

Some operands convey no information to the NCP generation assembly process but must nonetheless appear in the NCP generation input deck that serves as input to the VTAM initialization procedure. Such operands are called *VTAM-only* operands. The macro assembly step of the NCP generation procedure permits each VTAM-only operand to appear in the macros indicated by a V in the table of Figure 5-11. The assembly process does not check these operands for proper syntax or verify that any related operands are present or absent. For the meanings of VTAM-only operands, see the *ACF/VTAM Installation* manual.

	Macro Instruction								
	For Network Control Mode							*For Emulation Mode*	
	GROUP	LINE	CLUSTER	PU	LU	TERMINAL	COMP	GROUP	LINE
ACTIVTO	*								
ADDR				*		*	•		
ADDRESS		*							*
ANS	•	•		*					
ANSTONE	•	*							
ANSWER	V	V							
ATTN	•	•				*			
AUTO		*							*
AUTUAC B	•	*							
BATCH	•	•			*				
BFRDLAY	•	•	•			*			
BHEXEC	•	•	*			*	•		
BHSET	•	•	*			*	•		
BNNSUP	•	•		*					
BUFLIM	V	V		V	V	V	V		
BUFSIZE								•	*
CALL	•	*							
CDATA	•	•	*			*	•		
CHAREC	*							*	
CHECK								•	*
CHNPRI								•	*
CLOCKING	•	*						•	*
CODE	•	*						•	*
CONFIG	•	*							
CONV	•	•				*	•		
CRDLAY	•	•				*	•		
CRETRY	*								
CRITSIT	•	•	•			*			
CRRATE	•								
CTERM						*			
CU								•	*
CUIDLEN	•	•				*			
CUTOFF	•	*							
CUTYPE	•	•	*					•	*
DATASW					*				
DATMODE	•	•		*					
DATRATE	•	*						•	*
DELAY								*	
DEVICE						V	V	*	
DIAL	*								
DIALALT	•	*							
DIALNO						*			
DIALSET	•	*				*			
DIRECTN	•	•				*			
DISABLE								•	*
DISCNT	V	V		V					
DLOGMOD	V	V	V	V	V	V	V		
DUALCOM									*
DUPLEX	•	*						•	*
ENDTRNS	•	•				*	•		
EOB								*	
EOT								*	
EXEC	•	•	•			*	•		

V indicates the macros in which the VTAM-only operand at the left may be coded. See the ***VTAM System Programmer's Guide*** for a description of this operand.

Figure 5-11. Summary of Operands for Configuration Macro Instructions (Part 1 of 3)

	Macro Instruction								
	For Network Control Mode							*For Emulation Mode*	
	GROUP	LINE	CLUSTER	PU	LU	TERMINAL	COMP	GROUP	LINE
FANOUT						*			
FEATURE[1]	•	•	*			*		•	*
FEATURE2	V	V	V	V	V	V			
FGSLTRS	•	*							
GPOLL			*						
HDXSP	•	*							
IDSEQ	•	•				*			
INHIBIT	•	•	•			*	•		
INTPRI	•	*						•	*
IRETRY	•	•		*					
ISTATUS	V	V	V	V	V	V	V		
ITBMODE	•	•	•			*	•		
KBDLOCK	*								
LCST	•	•				*			
LEVEL2	*								
LEVEL3	*								
LEVEL5	*								
LGRAPHS	•	•	*			*	•		
LINECB	•	*		•	*				
LINEFVT	•	*		•	*				
LINESIZ	•	*							
LNCTL	*							*	
LNQTCNT	•	*							
LOCADDR					*				
LOGAPPL	V	V	V	V	V	V	V		
LOGTAB	V	V	V	V	V	V	V		
MAXDATA	•	•		*					
MAXLU	•	•		*					
MAXOUT	•	•		*					
MAXPU	•	*							
MODEM								•	*
MODETAB	V	V		V	V				
MONITOR	•	*							
MPTALT	•	*							
MTALIST	•	*							
MULTI								•	*
NEGPOLP	•	*							
NEWSYNC	•	*						•	*
NRZI	•	*							
OWNER	V	V							
PACING	•	•	•	•	*				
PAD								•	*
PADCNT	*								
PARCHK	•	*							
PASSLIM	•	•		*					
PAUSE	•	*							
PECHAR	*								
POLIMIT	•	*							
POLL						*	•		
POLLED	•	*							

[1] Only BATCH or NOBATCH may be specified in the FEATURE operand of the CLUSTER macro, for the IBM 2972 General Banking Terminal System.

V indicates the macros in which the VTAM-only operand at the left may be coded. See the *VTAM System Programmer's Guide* for a description of this operand.

Figure 5-11. Summary of Operands for Configuration Macro Instructions (Part 2 of 3)

Macro Instruction

	For Network Control Mode							*For Emulation Mode*	
	GROUP	LINE	CLUSTER	PU	LU	TERMINAL	COMP	GROUP	LINE
POLLTO	•	*							
PT3EXEC	•	•	*			*	•		
PU	V	V	V						
PUCB	•	•		*					
PUFVT	•	•		*					
PUTYPE	•	•		*					
QUIET								•	*
QUIETCT								*	
REDIAL	•	*						*	
REPLYTO	*								
RETRIES	•	*		*					
RING	•	*						•	*
SCLSET	•	*							
SDLCST	•	*							
SECURE	•	*							
SERVLIM	•	*							
SERVPRI	•	*							
SESSION	•	*							
SPAN	V	V	V	V	V	V	V		
SPDSEL	•	*							
SPEED	•	*						•	*
SPSHIFT	•	*							
SSCPFM	V	V		V	V				
SUBAREA				*					
SYNDLAY	*								
TADDR	•	*						•	*
TERM	•	*	•	V	V	*		•	*
TEXTTO	*							*	
TIMER	*								
TRANSFR	•	*							
TTDCNT	*								
TYPE	*	*							
UACB	•	*							
UNITXC								•	*
USE	•	*							
USSTAB	V	V		V	V				
UTERM						V			
VIDSEQ	V	V							
VIRTUAL	*								
VPACING	V	V	V	V	V				
WACKCNT	*								
WAKDLAY	*								
WTTYEOB	*								
WTTYEOT	*								
XIO	*								
XMITLIM	•	•	*			*	•		
YIELD	•	*							

V indicates the macros in which the VTAM-only operand at the left may be coded. See the *ACF/VTAM* for a description of this operand. (See the *NOSP Installation Manual* for a description of the SPAN operand.)

Figure 5-11. Summary of Operands for Configuration Macro Instructions (Part 3 of 3)

GROUP Macro Instruction

The GROUP macro indicates the beginning of a series of macros that define lines and devices having common characteristics and specifies:

- Whether the lines are switched or nonswitched.
- Whether the lines are used for start-stop, binary synchronous, or SDLC communication.
- Optional or variable characteristics that all lines in the group must have in common.
- Certain procedural options to be applied to all lines in the group.

A communication line group consists of lines that have the following characteristics in common:

- All lines in the group are nonswitched point-to-point, nonswitched multipoint, or switched point-to-point.
- All lines in the group are polled, or all are nonpolled.
- All stations connected to lines in the group are start-stop stations, all are binary synchronous stations, or all are SDLC stations.
- If the stations are binary synchronous, they may be different types—for example, IBM 2770, IBM 2780, and IBM 1130—but all must use the same transmission code. All BSC stations use a uniform line control scheme.
- If the stations are SDLC, they may be different types. All SDLC stations use a uniform transmission code and line control scheme.
- If the stations are start-stop, all must be of the same type—for example, they may be IBM 1050 or IBM 2741, but not both—and all must use the same transmission code. If they are IBM 2740 terminals, they must have certain features in common. For example, a line group cannot include both 2740s with the record checking feature and 2740s without this feature. An exception is a line group consisting of multiple-terminal-access lines. These lines can accommodate IBM 1050 2740, and 2741 terminals, using the same or different transmission codes, and Western Union TWX terminals.

One GROUP macro is required for each line group. No line may be included in more than one line group.

All GROUP macros for SDLC line groups must appear in the source program following any GROUP macros for BSC and/or start-stop line groups.

Defining a Stand-Alone Line Group for SDLC

To define a stand-alone SDLC line group (a GROUP macro followed by no LINE and lower-level macros) for an SDLC selection table, specify the following operands in the GROUP macro:

LNCTL=SDLC
TYPE=NCP
DIAL=NO
ACTIVTO=count (valid only if POLLED=NO is specified)
REPLYTO=count
POLLED=YES or POLLED=NO (the POLLED operand must be coded the same in the GROUP and in the SDLCST macros)

See the SDLCST macro description for information on using an SDLC selection table.

Defining a Stand-Alone Line Group for MTA

A stand-alone GROUP macro (a GROUP macro not followed by any LINE and TERMINAL macros) is needed for MTA support only when there is no non-MTA GROUP macro that has the same values specified for the TERM, POLLED, and FEATURE operands. To define this stand-alone group, specify the following operands in the GROUP macro:

DIAL=YES
LNCTL=SS
TERM=1050 2740-1, 2741, or TWX
POLLED=YES, only for TERM=2740-1
FEATURE=(XCTL,CHECK), only for TERM=2740-1

Other GROUP macro operands may be specified for the stand-alone GROUP macro. However, any LINE or TERMINAL macro operands that may normally be coded for a GROUP macro should not be coded for a stand-alone GROUP macro. Any LINE or TERMINAL operands coded for a stand-alone GROUP macro are ignored.

(VTAM Users Only): Appearing at the end of the list of operands below are the VTAM-only operands that may be coded in this macro instruction. These provide information only to the VTAM initialization process and are not required (though are permissible) in the card deck used as input to the NCP generation procedure. See the *ACF/VTAM Installation* manual for descriptions of these operands and for information on the VTAM initialization process. (See the *NOSP Installation Manual* for the description of the SPAN operand.)

See Appendix J for an example of stand-alone line groups in defining multiple-terminal-access operations.

The format of the GROUP macro is:

```
symbol          GROUP          [operands]
```

Operands

```
[ACTIVTO={count}]
         {NONE }

         {[XONOFF][,chars]}
[CHAREC=( {XON,chars       })]
         {XOFF,chars      }
         {NO,chars        }

[,CRETRY=count]

        {600 }
[,DELAY={1200}]
        {NO  }

[,DIAL={YES}]
       {NO }

[,EOB=(char[,F])]

[,EOT=(char[,F])]

[,FRAMING={10}]
          {11}
```

```
[,KBDLOCK=chars]
[,LEVEL2=symbol]
[,LEVEL3=symbol]
[,LEVEL5={NCP  }]
         {USER}
[,LNCTL={SS   }]
        {BSC  }
        {SDLC}
        {USER}
[,PADCNT={count}]
         {0     }
[,PECHAR={chars}]
         {FF    }
[,QUIETCT={count}]
          {0     }
[,REPLYTO={count}]
          {NONE  }
[,SYNDLAY={count}]
          {1.0   }
[,TEXTTO={count}]
         {NONE  }
[,TIMER=(error,ras,stap,lstap)]
[,TTDCNT={count}]
         {15    }
[,TYPE={NCP}]
       {PEP}
[,VIRTUAL={YES}]
          {NO }
[,WACKCNT={count}]
          {15    }
          {count}
[,WAKDLAY={2.2   }]
          {NONE  }
[,WTTYEOB=chars]
[,WTTYEOT=chars]
[,XIO=(line,setmode,immed[,LINK])]
```

VTAM-only operands:

```
ANSWER=
BUFLIM=
CALL=
DISCNT=
DLOGMOD=
FEATUR2=
ISTATUS=
LOGAPPL=
LOGTAB=
OWNER=
PU=
SPAN=
SSCPFM=
USSTAB=
VIDSEQ=
VPACING=
```

symbol

Provides a name for the line group and is required. *symbol* may be any valid assembler-language symbol. The first character may not be $.

[ACTIVTO={count}]
{NONE }

(SDLC only)

Specifies the time that the secondary NCP allows to elapse since communication from the primary NCP. When this time-out value is reached, the secondary NCP abnormally ends or begins automatic network shutdown, depending on the values of the ACTIVTO and ANS operands (see Figure 5-12). If an abend occurs, the controller must be reloaded (IPL).

count

Specifies the time-out value, to the nearest tenth of a second, that the secondary NCP will wait for communication from the primary NCP before initiating shutdown. The minimum value is 60.0 seconds; the maximum is 420.0 seconds.

count is valid only if you specify ANS=YES in the BUILD macro.

NONE

Specifies that the secondary NCP will wait indefinitely for communication from the primary NCP; automatic network shutdown will not occur.

This operand is valid only if this program is the secondary NCP for the link and PUTYPE=(4,LOCAL) is specified in the PU macro.

If you omit this operand and specify ANS=YES (or you omit the ANS operand), ACTIVTO=420.0 is assumed. If you omit this operand and specify ANS=NO, ACTIVTO=NONE is assumed.

The following expression is a guide in determining the appropriate ACTIVTO value:

$$\text{MO} - \text{IPL} \geq \text{ACTIVTO} \geq t*n \geq \text{SDO}$$

MO is the maximum time the controller may be out of service due to link failure.

IPL is the interval normally required to load (IPL) the controller; if loading is not required, the value of IPL may be 0.

ACTIVTO is the value specified in the ACTIVTO operand.

t and *n* are the values specified in the RETRIES operand of the PU macro of the primary network control program.

SDO is the average duration of *short-duration outages* of the local-local or local-remote link (outages caused by momentary loss of modem carrier, excessive noise, power surges, or other disruptive line conditions).

Action taken by secondary (remote) NCP upon failure of SDLC link to primary (local) NCP:

ACTIVTO=	*ANS=*	*Number of SDLC Local-Remote Links*	*Action*
count	YES	1	ANS occurs; remote NCP then awaits message traffic from local controller over the SDLC link.
count	YES	2 or more	ANS occurs; remote NCP then waits for message traffic from the local controller over any one of the available SDLC links.
count	NO	1	Expiration of time-out causes abnormal end (abend) of remote NCP; reloading (IPL) occurs over the link (after repair).
count	NO	2 or more	Expiration of time-out causes abnormal end (abend) of remote NCP; reloading (IPL) occurs over any available link.
NONE	NO	1 or more	Remote NCP waits indefinitely for traffic from local controller.
NONE	YES	1 or more	(Invalid combination)

Action taken by secondary (local) NCP upon failure of SDLC link to primary (local) NCP:

ACTIVTO=	*ANS=*	*Number of SDLC Local-Remote Links*	*Action*
count	YES	1 or more	Secondary NCP notifies adjacent access method(s) and performs ANS for affected network resources.
count	NO	1 or more	(Invalid combination)
NONE	NO	1 or more	(Invalid combination)
NONE	YES	1 or more	(Invalid combination)

Figure 5-12. Effect of ACTIVTO and ANS Options on Secondary to Primary NCP Like Failure Recovery

```
         {[XONOFF][,chars]}
[CHAREC=({XON,chars        })]
         {XOFF,chars       }
         {NO,chars         }
```

(83B3, 115A lines in emulation mode only)
(all TWX lines)

Specifies the message-ending characters recognized by the network control program when receiving data from an AT & T 83B3, WU115A, or TWX terminal.

When these characters are received from a line in emulation mode, the program signals channel-end and device-end status to the emulation subchannel associated with the line.

Specify this operand as follows:

For TWX terminals:

The NCP allows the following options when selecting the ending characters.

- XONOFF indicates either XON or SOFF and no other character.
- (XONOFF,chars) indicates either XON, XOFF, or one or two user-defined characters.
- (XON,chars) indicates XON only or one or two user-defined characters.
- (XOFF,chars) indicates XOFF only or one or two user-defined characters.
- (NO,chars) indicates one or two user-defined characters but not XON or XOFF.

Coding CHAREC=(,*chars*) is the same as CHAREC=(XONOFF,*chars*).

Note: The program will not recognize any characters other than WRU, XON, XOFF, and EOT unless specified in this operand.

For 83B3 and 115A terminals (emulation mode only):

The ending sequence for these terminals is FIGS H LTRS and do not transmit or receive XON and XOFF characters. The program will recognize the FIGS H LTRS sequence if you specify CHAREC=XONOFF or omit the operand. CHAREC=(XONOFF,chars) is *not* valid.

If the terminals in your network send an ending character or sequence other than FIGS H LTRS, specify the character or sequence as CHAREC=(,chars). *chars* must be the hexadecimal representation of the transmission code bit patterns shown in Appendix H.

[CRETRY=count]

(start-stop and BSC lines only)

Specifies the maximum number of error recovery attempts the network control program is to make when recoverable errors occur on any line in control mode. (Errors in control mode are usually those that occur during polling or addressing.) The minimum number of retires is 0; the maximum is 254 (CRETRY=254). If you code CRETRY=255, the network control program retries the operation indefinitely. (The network control program makes error-recovery attempts only when the line is operating in network control mode.)

If you omit this operand, the NCP attempts error recovery twice for start-stop lines or seven times for BSC lines.

```
         {600 }
[DELAY={1200}]
         {NO  }
```

(World Trade teletypewriters in emulation mode only)

Specifies whether the teletypewriter (teleprinter) terminals in this GROUP macro require a line turnaround time of 70 to 80 milliseconds.

Specify DELAY=600 if the terminals require the delayed turnaround and operate at 600 bps; specify DELAY=1200 if the terminals require the delayed turnaround and operate at 1200 bps.

If delayed turnaround is not required, specify DELAY=NO or omit the operand.

[DIAL={YES}]
{NO}

Specifies whether the lines in the group require switched line control procedures. If they do, code DIAL=YES. If they do not, code DIAL=NO or omit the operand.

Note: SDLC local-local and local-remote backup links, whether they comprise switched or nonswitched lines, use nonswitched line control procedures (DIAL=NO).

[EOB=(char[,F])]

(World Trade teletypewriter lines in emulation mode only)

Specifies the end-of-block (end-of-message) sequence received from any World Trade teletypewriter terminal or certain U.S. and Canadian teletypewriter terminals. These terminals typically send either a sequence of four identical characters or a sequence of FIGS *character* LTRS.

If the terminal transmits a four-character sequence, specify the character used as EOB=*char*, where *char* is the hexadecimal representation of the character transmitted. For example: if the terminal sends the sequence MMMM (in letters shift), code EOB=1C.

If the terminal transmits a FIGS *character* LTRS sequence, code *char* as the hexadecimal representation of the character sent and also code the F following the character. For example, FIGS M LTRS would be coded as EOB=(3C,F).

Note: Appendix H lists the transmission code bit patterns for the ITA2 and ZSC3 codes.

The end-of-block sequence may be specified in either the EOB or the CHAREC operand, but not in both.

[EOT=(char[,F])]

(World Trade teletypewriter lines in emulation mode only)

Specifies the character sequence the program is to recognize as the end-of-transmission sequence when received from any World Trade teletypewriter terminal or certain (specially modified) U. S. and Canadian teletypewriter terminals. These terminals typically send, as an EOT indication, either a sequence of four identical characters or a sequence of FIGS *character* LTRS.

If the terminal transmits a four-character sequence, specify the character used as EOB=char, where *char* is the hexadecimal representation of the character transmitted. For example: if the terminal sends the sequence AAAA (in letters shift), code EOT=18.

If the terminal transmits a FIGS *character* LTRS sequence, code *char* as the hexadecimal representation of the character sent and code the F following the character. For example, FIGS A LTRS would be coded as EOT=(38,F).

Note: Appendix H lists the transmission code bit patterns for the ITA2 and ZSC3 codes.

The standard teletypewriter ending sequence is FIGS H LTRS. If you omit the EOT operand, this is the sequence the program recognizes as the EOT sequence when receiving from a teletypewriter terminal.

[,FRAMING={10}]
{11}

(TWX lines in network control mode only)

Specifies the number of bits per character the NCP sends and expects to receive from TWX terminals in the group. 11 specifies that 2 stop bits are to be used; 10 specifies that only one stop bit is to be used. Each TWX character is: 1start bit, 7data bits, 1 parity, and 1 or 2 stop bits.

[KBDLOCK=char]

(TWX lines in network control mode only)

Specifies the hexadecimal representation for the nonprinting, nonspace character the network control program sends to TWX terminals to signal terminal operators to stop using the keyboard. This technique, called logical keyboard lock, is used to avoid data entry by the terminal operator before the program is ready to receive. The signal is send only when the TSO/TCAM monitor mode is active for the line.

If you omit this operand or code KBDLOCK=80, the program does not send any keyboard lock characters. If the null is sent, the line is put in a mark state (X'FF's are sent with the secondary control field (SCF) pad flag on).

This operand is valid only if you specify DIAL=YES and LNCTL=SS in the GROUP macro, MONITOR=YES and POLLED=NO in the LINE macro, and TERM=TWX in the TERMINAL (or higher-level) macro.

[LEVEL2=symbol]

(user-written line control only)

Specifies the entry point of the user-written level 2 interrupt code. This operand is required if LNCTL=USER is specified. Level 2 interrupt code is optional if LNCTL=SS, BSC, or SDLC.

If the LEVEL2 operand is specified, the LEVEL3 operand *must* also be specified.

[LEVEL3=symbol]

(user-written line control only)

Specifies the entry point of the user-written level 3 interrupt code. This operand is required if LEVEL 2 is specified.

[LEVEL5={NCP }]
{USER}

(user-written code only)

Specifies whether the level 5 code is provided by the NCP or is user-written. This operand is valid only if LEVEL2 and LEVEL3 are specified. If this operand is omitted and LEVEL2 and LEVEL3 are specified, LEVEL5=USER is assumed.

LEVEL5=NCP specifies that the level 5 code is provided by the NCP. In this case, LNCTL *must* equal SS, BSC, or SDLC; and the LINE, CLUSTER, TERMINAL, COMP, PU, and LU macros must follow the same requirements as for line groups supported totally by NCP-supplied code.

LEVEL5=USER specifies that the level 5 code is user-written and may be specified for any type of line control (LNCTL). When this option is selected, the NCP generation procedure assumes that level 5 code will be supplied separately, and therefore no level 5 control blocks are generated for this group.

If LEVEL5=USER and VIRTUAL=NO, only LINE macros may be specified in this GROUP macro. A separate GROUP macro with VIRTUAL=YES is specified must be included to attach the lines to the network. If VIRTUAL=YES in this GROUP macro, the LINE, SERVICE, PU, and LU macros must follow the same requirements as for line groups supported by NCP-supplied code.

```
        {SS  }
[LNCTL={BSC }]
        {SDLC}
        {USER}
```

Specifies the type of line control used for the lines in this group. A line group may contain lines of only one type. Start-stop and BSC line groups may appear in the program generation deck in any sequence. SDLC line groups must appear after all start-stop and BSC line groups. For example:

```
G1  GROUP   LNCTL=SS,...
    *
G2  GROUP   LNCTL=BSC,...
    *
G3  GROUP   LNCTL=BSC,...
    *
G4  GROUP   LNCTL=SS,...
    *
G5  GROUP   LNCTL=SDLC,...
    *
G6  GROUP   LNCTL=SDLC,...
```

(* represents intervening lower-level macros)

If you omit the LNCTL operand from a GROUP macro, a line group is assumed to be (1) a start-stop line group if no GROUP macros in which SDLC is specified precede the macro or (2) an SDLC line group if a GROUP macro in which LNCTL=SDLC is specified does precede the macro.

LNCTL=SDLC is valid for a line that is operated only in network control mode. (TYPE=NCP is specified in the GROUP macro.)

All lines in an SDLC line group must be specified as POLLED=YES, or all must be specified as POLLED=NO. Both options cannot be included in the same line group.

If user-written line control is included for this group (LEVEL2 and LEVEL3 operands specified) and VIRTUAL=NO is coded, LNCTL=USER must be specified. If VIRTUAL=YES, LNCTL=SDLC must be specified.

Note: If LNCTL=USER is specified in this macro, only LINE macros may be specified in this group. A separate GROUP macro, with LNCTL=SDLC and VIRTUAL=YES specified is required.

If LNCTL=SS, BSC, or SDLC, the LEVEL5 operand specifies whether the NCP or the user supplies the level 5 support.

```
[PADCNT={count}]
       {0     }
```

(World Trade teletypewriter lines in network control mode only)

Specifies the number of idle characters sent to a World Trade teletypewriter terminal to permit its motor to reach full speed before receiving data. This is required only for a line where terminals are not equipped with a continuously running motor. A sufficient number of characters must be specified to create a 1.5-second delay on the line. The characters are sent only when the line is in network control mode.

The minimum value of *count* is 0; the maximum is 255.

```
[PECHAR={chars}]
       {FF    }
```

(TWX lines in network control mode only)

Specifies the hexadecimal representation of the data character that the network control program uses to overlay data received from a TWX terminal when the program detects incorrect parity. If you omit this operand, the program overlays with the X'FF' character.

This operand is valid only if PARCHK=ODD or PARCHK=EVEN is specified in one or more LINE macros associated with this group.

```
[QUIETCT={count}]
        {0     }
```

(start-stop lines in emulation mode only)

Specifies the number of character times that elapse between the end of a receive operation and the beginning of a transmit operation on a start-stop line. The elapsed time allows the line to electrically "quiesce" following the receive operation.

The default value of 0 is appropriate for most start-stop lines operating at speeds under 1200 bps. For lines operating at 1200 bps or more, one or more extra character times may be necessary to ensure quieting of the line. The recommended value for 1200 bps start-stop lines is 5 (QUIETCT=5).

The minimum you may specify is 0; the maximum is 10.

Note: The interval (number of character times) following a normal receive operation equals the value specified plus 2. The interval following receipt of a negative response to polling equals the value you specify. Thus, if you specify QUIETCT=5, a normal receive operation is followed by seven character times and a negative response to polling is followed by five character times before the next transmission begins.

```
[REPLYTO={count}]
        {NONE  }
```

Specifies the reply time-out value for the lines in this group. If the NCP does not receive a response to a poll, selection, or message text before the specified time expires, a time-out error is indicated, and no further attempt is made to communicate with the terminal. Except for BSC lines in emulation mode, this timeout applies to all lines in either network control or emulation mode.

If this GROUP macro represents an SDLC line group, the REPLYTO operand is valid only if the lines in the group are polled (POLLED=YES in the LINE macro). It is invalid in a remote program (TYPGEN=NCP-R in the BUILD macro).

Specify this value as an integral number of seconds or to the nearest tenth of a second.

If you specify REPLYTO=NONE, no time-out occurs regardless of the time that elapses between sending to the station and receiving a response.

Reply time-outs for start-stop lines are resolved to the nearest half second; reply time-outs for BSC lines, to the nearest tenth of a second. For example, if you specify a value of 4.6 for a BSC line group, the reply time-out value will be 4.6 seconds. If you specify 4.6 for a start-stop line group, the reply timeout value will be 4.5 seconds.

Reply timeouts for BSC terminals using conversational replies must be long enough to allow the conversational text to be received.

The maximum value is 1632 seconds for lines in network control mode and 51.1 seconds, for lines in emulation mode.

If you omit the REPLYTO operand, the NCP uses a time-out of 23.5 seconds for TWX terminals in network control mode, 1.0 second for SDLC stations, and 3.0 seconds for all other types of stations. However, if a TERMINAL macro in the group specifies:

1. IBM 2740 Model 1 terminals (TERM=2740-1) without at least one of these features: checking (FEATURE=CHECK), transmit control (FEATURE=XCTL), or station control (FEATURE=SCTL)
2. multiple-terminal-access terminals (TERM=MTA) or
3. World Trade teletypewriter terminals (TERM=WTTY)

The value specified for REPLYTO is ignored for that terminal and no time-out is provided.

For most networks, the default value is appropriate.

Note: If any IBM 1050 in the line group includes a paper tape punch, specify a value of at least 23.5 seconds in this operand. If you specify a lesser time, the NCP may time out after sending data to the paper tape punch.

```
[SYNDLAY={count}]
         {1.0  }
```

(BSC lines in network control mode only)

Specifies the interval, in seconds, between transmissions by the network control program of the BSC synchronizing characters (SYN) on a line that is in text-transmit mode (and in network control mode).

Specify this interval as an integral number of seconds or to the nearest tenth of a second.

For example, for an interval of 2 seconds, code SYNDLAY=2 (or 2.0); for 1.5 seconds, code SYNDLAY=1.5.

The maximum interval you may specify is 1632 seconds. For most networks, the default value of 1 second is recommended and appropriate.

Note: This operand determines the interval for lines serviced by a type 2 communication scanner only. The interval for lines serviced by a type 3 scanner is determined directly by the scanner.

```
[TEXTTO={count} ]
       {NONE }
```

(BSC and start-stop lines only)

Specifies the text timeout value, in seconds, for the lines in the line group. If the interval between any two successive message characters received from a station exceeds this value, the NCP ends the read or invite operation with a text timeout error indication. For start-stop lines, this action occurs whether the line is operating in network control mode or in emulation mode. For BSC lines, it occurs only when the line is in network control mode.

Specify this value as an integral number of seconds or to the nearest tenth of a second.

Text timeouts are resolved to the nearest half-second. For example, if you specify a value of 20.2, the text timeout value will be 20 seconds (nearest half second to 20.2). If you specify TEXTTO=NONE, no timeout occurs regardless of the interval that elapses between receipt of successive characters.

Note: The value specified for *count* is a nominal value. The actual elapsed interval may vary anywhere between the nominal value and twice the nominal value. A nominal value of 30 seconds, for example, will result in an actual interval of 30 to 60 seconds.

If TERM=2741 is specified in the TERMINAL macros of this line group and you omit the TEXTTO operand, a value of NONE is assumed. For any other type of terminal, a value of 23.5 seconds is assumed for lines operating in network control mode and 25.6 seconds, for lines operating in emulation mode. (If TERM=MTA is specified, the TEXTTO operand has no meaning and should be omitted.) For most networks the default value of 23.5 (or 25.6) seconds is recommended and appropriate.

The maximum value is 1632 seconds for lines operating in network control mode and 51.1 seconds for lines operating in emulation mode.

```
[TIMER=(error,ras,stap,lstap)]
```

(user-written line control only)

Specifies the entry points of user-written timer service routines. This operand is required if the LEVEL2 and LEVEL3 operands are specified. If level 2 and level 3 user-written code is not included, this operand must be omitted.

All four parameters of this operand are required.

error specifies the entry point of the user-written timer error service routine.

ras specifies the entry point of the user-written timer RAS service routine used to make periodic checks for certain error conditions.

stap specifies the entry point of the user-written time shoulder tap service routine that receives control when a user-initiated timer reaches 0.

lstap specifies the entry point of the user-written lagging shoulder tap timer routine.

```
[TTDCNT={count} ]
        {15    }
```

(BSC lines in network control mode only)

[TTDCNT={count}]
{15 }

(BSC lines in network control mode only)

Specifies the maximum number of times the BSC temporary text-delay (TTD) sequence may be accepted from a station before the operation is aborted. The TTD sequence notifies the controller that the station is temporarily unable to send the next block of data. The maximum count is 255. Any value less than 255 specifies the maximum number of times the sequence is to be accepted; 255 specifies that the sequence is to be accepted without limit. This operand applies only to line operation in network control mode.

For most networks, the default value of 15 TTD transmissions is recommended.

[TYPE={NCP}]
{PEP}

Specifies, for the line group represented by this GROUP macro, whether all lines operate in network control mode, some lines operate in one mode, and some operate in another, or alternately in either network control or emulation mode.

Note: A line group consisting of SDLC lines can operate only in network control mode.

Specify TYPE=NCP if all lines in the group are always to operate in *network control* mode.

Specify TYPE=PEP if some lines in the group are to operate in *network control* mode and others in *emulation* mode, or if any lines are to operate alternately in both modes.

The value choices for this operand depend on whether the program can perform only network control functions, or both network control and emulation functions, as specified in the TYPGEN operand of the BUILD macro:

If...	*Then valid choices are...*
BUILD: TYPGEN=	**GROUP: TYPE=**
NCP (or NCP-LR) or NCP-R	NCP
PEP (or PEP-LR)	PEP[1] NCP

[1] Not valid if GROUP macro represents an SDLC line group.

Default values: If you omit the TYPE operand from the GROUP macro, the following values are assumed:

BUILD: TYPGEN=	**Default for GROUP: TYPE=**
NCP (or NCP-LR) or NCP-R	NCP
PEP (or PEP-LR)	PEP if line group is start-stop or BSC (GROUP: LNCTL=SS or BSC) NCP if line group is SDLC (GROUP: LNCTL=SDLC)

Note: If you specify TYPE=PEP, all valid operands for operation in network control mode or in emulation mode may be specified in this GROUP macro. If you specify TYPE=NCP, only valid operands for operation in network control mode should be coded; operands applicable to emulation mode will be ignored.

```
[VIRTUAL={NO }]
         {YES}
```

(user-written code only)

Specifies whether programmed resources are supported in this group. This operand is valid only when LNCTL=SDLC.

```
[WACKCNT={count}]
         {15   }
```

(BSC lines in network control mode only)

Specifies the maximum number of times the BSC wait-before-transmit positive acknowledgment (WACK) may be accepted from a nonbuffered BSC station (in text mode only) before the operation is to be aborted. The WACK sequence notifies the controller that the station is temporarily not ready to receive. Any value less than 255 specifies the maximum number of times the sequence is to be accepted; 255 specifies that the sequence is to be accepted without limit. This operand applies only to line operation in network control mode.

Note: This operand has no effect for buffered BSC stations because the network control program, upon receiving a WACK sequence from such a station, transmits an EOT to end the transmission. (See the description of the BFRDLAY operand under the TERMINAL macro.)

The maximum count is 225; the minimum is 1.

For most networks, the default value of 15 WACK transmissions is recommended.

```
         {count}
[WAKDLAY={2.2  }]
         {NONE }
```

(BSC lines in network control mode only)

Specifies the maximum interval that is to elapse before the network control program responds to message text received from a station on any line in the group operating in network control mode. If the network control program has been unable to respond normally (with a positive acknowledgment) by the time this interval has elapsed, it will send a WACK sequence instead.

If you code WAKDLAY=NONE, the NCP will not send a WACK sequence when unable to send a positive acknowledgment.

Specify this delay as an integral number of seconds or to the nearest tenth of a second. For example, code a delay of 12 seconds as WAKDLAY=12 (or WAKDLAY=12.0); for 12.5 seconds, you code WAKDLAY=12.5.

The maximum number of seconds is 1632 (1632.0) seconds.

For most networks the default interval of 2.2 seconds is recommended.

```
[WTTYEOB=chars]
```

(World Trade teletypewriter lines in network control mode only)

Specifies, in hexadecimal, the end-of-block sequence required for World Trade teletypewriters (teleprinters) connected to lines in the group operating in network control mode. Specify up to 8 hexadecimal characters (4 EBCDIC characters). Any valid character may be included in the sequence, but the FIGS and LTRS characters may be included only once. If the terminal is equipped to send who-are-you (WRU), avoid using the letter D in the sequence. A typical EOB sequence is FIGS *x*, where *x* is any valid character (except FIGS). See Appendix H for the World Trade Teletypewriter transmission codes.

`[WTTYEOT=chars]`

(World Trade teletypewriter lines in network control mode only)

Specifies, in hexadecimal, the end-of-transmission sequence required for World Trade teletypewriters connected to lines in the group operating in network control mode. Any valid character may be included in the sequence, but the FIGS and LTRS characters may be included only once. If the terminal is equipped to send who-are-you (WRU), avoid using the letter D in the sequence. A typical EOT sequence is FIGS *y* LTRS, where *y* is any valid character (except FIGS or LTRS).

This operand is required for all World Trade teletypewriters. See Appendix H for the World Trade Teletypewriter transmission codes.

`[XIO=(line,setmode,immed[,link])]` *(user-written line control only)*

Specifies the entry points of user-written execute I/O service routines. This operand is required if the LEVEL2 and LEVEL3 operands are specified. If level 2 and level 3 user-written code is not included, this operand must be omitted.

line specifies the entry point of the user-written execute I/O line service routine. This entry point is required.

setmode specifies the entry point of the user-written execute I/O set mode service routine. This entry point is required.

immed specifies the entry point of the user-written execute I/O immediate service routine. This entry point is required.

link specifies the entry point of the user-written execute I/O link service routine. This entry point is required only if the user-written code issues an XIO LINK command. If no XIO LINK commands are issued, this parameter may be omitted.

Restriction on Number of Time Intervals Specified

A maximum of 16 different time intervals may be specified in the network control program. This includes seven standard intervals (including default values) that are common to all network control programs and intervals that you explicitly specify in the REPLYTO, SYNDLAY, TEXTTO, and WAKDLAY operands of the GROUP macro. If the total number of different time intervals is 16, the default values are used for all subsequent time specifications the generation procedure encounters in processing the source statements. An example of different time-out values is REPLYTO=10, WAKDLAY=4. The same time value specified in any number of the four operands mentioned constitutes only a single time interval.

The standard time intervals for all network control programs are:

0.0 (immediate action required)
1.0 seconds

2.2 seconds
3.0 seconds
23.5 seconds
60.0 seconds
NONE (infinite time delay)

Specifying Lower-Level Operands in the GROUP Macro

In addition to the preceding operands, most operands of the LINE, CLUSTER, PU, LU, TERMINAL, and COMP macros can be specified in the GROUP macro instead of in the individual macros mentioned. Figure 5-11 shows which of the lower-level operands you may specify in the GROUP macro.

LINE Macro Instruction

Each LINE macro represents one start-stop or BSC communication line or SDLC link. Some of the operands apply only to certain types of lines or stations. The descriptions of the individual operands indicate the conditions under which they are coded.

One LINE macro must be coded for each start-stop or BSC communication line connected to the controller, whether it is to operate only in network control mode or alternately in either network control or emulation mode. For an SDLC link, a single LINE macro represents either one or two communication lines. All LINE macros representing lines in a physical line group must appear between the GROUP macro representing that group and the next GROUP macro.

Note: LINE macros representing SDLC links must appear in the program generation input deck following LINE macros representing BSC and start-stop lines. (An example appears in the description of the LNCTL operand of the GROUP macro.)

A remote communications controller may have one principal SDLC link to the local controller and from one to three alternate (backup) SDLC links to that controller. Each link is represented within the remote NCP by a LINE macro; therefore, up to four LINE macros representing SDLC links to the local controller may appear in the generation input deck for the remote program.

If no TERMINAL macros follow a LINE macro for a start-stop or BSC line, CALL=OUT (VTAM only) and the device type (TERM operand) must be specified in the LINE or GROUP macro. The FEATURE operand of the TERMINAL macro may also be coded in the LINE macro. All other TERMINAL macro operands are ignored. (See Appendix J for an example of switched network operation.)

(VTAM Users Only): Appearing at the end of the following list of operands are the VTAM-only operands that may be coded in this macro instruction. These provide information only for the VTAM initialization process and are not required (though are permissible) in the card deck used as input for the NCP generation procedure. See the *ACF/VTAM Installation* manual for descriptions of these operands and for information on the VTAM initialization process. (See the *NOSP Installation Manual* for the description of the SPAN operand.)

The format of the LINE macro is:

```
symbol          LINE          operands[,operands]
```

Operands

```
         {line addr                              }
ADDRESS={(line addr,subchan addr1[,subchan}
                 addr2,...,subchan addrn])
         {(xmt addr,rcv addr)                    }
,SPEED=(rate[,rate])
[,ANSTONE={YES}]
          {NO }
[,AUTO=address]
[,AUTUACB=symbol]
[,BUFSIZE=n]
        {IN   }
[,CALL={OUT  }]
        {INOUT}
[,CHECK={DCD  }]
         {NODCD}
[,CHNPRI={NORMAL}]
          {HIGH  }
[,CLOCKNG=({INT}[,{INT}])]
           {EXT}   {EXT}
        {EBCDIC  }
        {USASCII }
        {BCD     }
        {EBCD    }
        {COR     }
[,CODE={KATAKANA}]
        {ITA2    }
        {ZSC3    }
        {DIC1    }
        {BCD2    }
        {COR2    }
        {DIC3    }
        {ASCII   }
[,CONFIG={SW   }]
          {NONSW}
[,CRRATE=integer]
      {2701}
[,CU={2702}]
      {2703}
[,CUTOFF={count}]
          {NO   }
          {2972}
[,CUTYPE={3271}]
          {3275}
[,DATRATE=({HIGH}[,{HIGH}])]
           {LOW }   {LOW }
[,DIALALT={dial set name}]
           {NONE         }
[,DIALSET={dial set name}]
           {NONE         }
```

Operands

```
[,DISABLE={YES}]
          {NO }

          {(line address,{A})}
[,DUALCOM={               {B} }]
          {NONE               }

[,DUPLEX=({HALF}[,{HALF}])]
          {FULL}  {FULL}

[,FEATURE=([{DUALCODE}]
            {NODUALCD}

           [,{IMEND   }]
             {NOIMEND}

           [,{LRC   }]
             {NOLRC}

           [,{SPACE   }])]
             {NOSPACE}

[,FGSLTRS={IN }]
          {OUT}

[,HDXSP={YES}]
        {NO }

           {0}   {0}
[,INTPRI=({1}[,{1}])]
           {2}   {2}
           {3}   {3}

[,LINECB=symbol]

[,LINEFVT=symbol]

[,LINESIZ=integer]

[,LNQTCNT={count}]
          {1    }

[,MAXPU=count]

        {OPTION1}
[,MODEM={OPTION2}]
        {NTT    }

[,MONITOR={YES}]
          {NO }

[,MPTALT={YES}]
         {NO }

[,MTALIST=entry]

[,NEGPOLP={n   }]
          {NONE}

[,NEWSYNC=({YES}[,{YES}])]
           {NO }  {NO }

[,NRZI={YES}]
       {NO }

[,PAD={YES}]
      {NO }

         {ODD    }
[,PARCHK={EVEN   }]
         {NOCHECK}
```

Operands

```
              {EVEN}
[,PARGEN=    {ODD }]
              {MARK}

[,PAUSE={t}]
        {0}

                       {WAIT   }
[,POLIMIT=([{n}][,{NOWAIT}])]
            {1}       {QUEUE }

[,POLLED={YES}]
          {NO }

[,POLLTO={ERROR   }]
          {NEGRESP}

[,QUIET={YES}]
         {NO }

[,REDIAL={(,[t1][,n[,t2]])      }]
          {([m][,t1][,n[,t2]])}

[,RETRIES={NONE          }]
           {(m[,t[,n]])]}

[,RING={YES}]
        {NO }

[,SCLSET={YES}]
          {NO }

[,SDLCST=(plcst,slcst)]

[,SECURE={YES}]
          {NO }

[,SERVLIM=count]

[,SERVPRI={OLD}]
           {NEW}

[,SESSION={count}]
           {1    }

[,SPDSEL={YES}]
          {NO }

[,SPSHIFT={YES}]
           {NO }

[,TADDR={char}]
         {NONE}

[,TERM=type]

[,TRANSFR=count]

        {EP }
[,TYPE={NCP}]
        {PEP}

[,UNITXC={YES}]
          {NO }

[,USE={NCP}]
       {EP }

[,YIELD={YES}]
         {NO }
```

Operands

VTAM-only operands:

ANSWER=
BUFLIM=
CALL=
DISCNT=
DLOGMOD=
FEATUR2=
ISTATUS=
LOGAPPL=
LOGTAB=
MODETAB=
OWNER=
PU=
SPAN=
SSCPFM=
USSTAB=
VIDSEQ=
VPACING=

symbol

Provides a name for the communication line and is required. *symbol* may be any valid assembler-language symbol. The first character may not be $.

```
        {line addr                                                }
ADDRESS={(line addr, subchan addr1[,subchan addr2,...,subchan addrn])}
        {(xmt addr,rcv addr)                                      }
```

Specifies the line interface address for the line represented by this LINE macro. It also specifies the emulation subchannel address if the line is to be operable in emulation mode.

The line interface address is determined by the physical location of the line interface hardware on the line interface base (LIB) and the type of scanner installed in the controller. The range of valid addresses depends on the type of scanner installed (see Figure 5-13).

For start-stop and BSC lines:

If the line operates only in network control mode, specify a single line interface address (ADDRESS=*line addr*). *line addr* is the hexadecimal controller address to which the line is attached. For example, ADDRESS=02F.

Lines with a data rate of 14 400 bps or more require two consecutive controller addresses. Specify only the even address of the even-odd pair. If the addresses assigned to a 14 400 bps line are 022 and 023, you would code ADDRESS=022.

If the line operates in emulation mode, specify both the line address and the corresponding emulation subchannel address(es) as ADDRESS=(*line addr,subchan addr1* [*,subchan addr2,...,subchan addrn*]). The subchannel address is the same address specified in the UCB (unit channel block) for OS/VS or in the PUB (physical channel block) for DOS/VS. Specify 3 hexadecimal digits for the line address. Specify 2 hexadecimal digits, each followed by -0 or -1, for each associated subchannel address: -0 to indicate that the channel adapter is installed in channel adapter position 0; -1 to indicate that the adapter is installed in channel adapter position 1. (Adapter position 0 is assumed if a -0 or -1 is not specified.) *Examples:* (1) ADDRESS=(02F,2E-0): line address 02F, subchannel address 2E, channel adapter position 0; (2) ADDRESS=(02F,2E-0,17-1,1C-0): line address 02F, subchannel addresses 2E and 1C, in channel adapter position 0 and subchannel address 17 in adapter position 1.

The subchannel address must be associated with a type 1 or type 4 channel adapter. The subchannel address specified for this line must not be the same as a subchannel address associated with any other line or used by any other program facility (such as the dynamic dump utility). Also, the subchannel address used for this line must not be the native subchannel (NSC).

Note: If the line address is for a line on a type 3 communication scanner, the subchannel address must be associated with a type 4 channel adapter. More than one subchannel address is valid only if the controller has one or two type 4 channel adpaters.

The subchannel address must be within the range specified by the HICHAN and LOCHAN operands of the BUILD macro.

For SDLC links:

If the communications controller transmits and receives over the *same* path, specify a single line interface address (ADDRESS=line addr) regardless of whether the communication facility is half-duplex or duplex (for example, ADDRESS=020).

If the communications controller transmits and receives over *separate* paths, specify *two* line interface addresses [ADDRESS=(*xmt addr, rcv addr*)]. For example, ADDRESS=(020,021).

Note: For lines attached to line set 1T or 1U, both addresses must be even addresses.

The line interface address is determined by the physical location of the line interface hardware on the line interface base (LIB) and the type of scanner installed in the controller. The range of valid addresses depends on the type of scanner installed (see Figure 5-13).

This operand is required.

If line is attached to:	*Range of Valid Addresses is:*	
	020-05F	(Base module)
Type 2 scanner	0A0-0FF	(First expansion module)
	120-17F	(Second expansion module)
	1A0-1FF	(Third expansion module)
	020-04F	(Base module)
Type 3 scanner	0A0-0DF	(First expansion module)
	120-15F	(Second expansion module)
	1A0-1DF	(Third expansion module)

Figure 5-13. Line and Auto-Call Addresses for LINE Macro

SPEED=(rate[,rate])

Specifies the data rate for this line in bits per second. This is the rate at which the stations communicate with the communications controller.

If the modem at the controller is a dual-rate modem, specify the higher of the two rates.

If one line interface address is specified in the ADDRESS operand of this macro, specify SPEED=*rate*. If this LINE macro represents an SDLC link and if two line interface addresses (*xmt addr, rcv addr*) are specified in the ADDRESS operand, code SPEED=*rate* if the speeds for both sides of the link are the same. Code SPEED=(*rate,rate*) for the transmit and receive addresses, respectively, if the rates differ.

If internal (business machine) clocking is used (see the CLOCKNG operand), the rate must be one of the four oscillator rates specified for the communication scanner where this line is attached (SPEED operand of the CSB macro). Specify the line speed in bits per second, omitting any fractional part. For example, specify a line speed of 1200 bps as SPEED=1200; specify a line speed of 134.5 bps as SPEED=134.

If external (modem) clocking is used, the rate must be the clocking rate of the modem, which is not necessarily one of the oscillator bit rates specified by the SPEED operand of the CSB macro. However, the SPEED operand of the CSB macro for the scanner must specify a scanner oscillator bit rate *less than one-half* of the modem clocking rate specified in this SPEED operand.

If the external modem has more than one data rate, the lowest SPEED operand of the CSB macro must be *less than one half* of the lowest modem speed that will be used.

The maximum speed you may specify is 57 600 bps.

If this LINE macro represents a multiple-terminal-access line used to communicate with terminals of different speeds, specify the speed of any of the terminals. For example, if terminals operating at speeds of 110 bps and 134.5 bps use this line, you may specify either SPEED=110 or SPEED=134.

This operand is required.

[ANSTONE={YES}
{NO}

(call-in switched lines in network control mode only)

Specifies whether the NCP is to transmit an *answer tone* to the calling station to signify completion of the line connection.

It is necessary for the program to send the tone only when the modem does not send an answer tone. Most modems in the United States and Canada do provide the tone; therefore ANSTONE=NO is normally appropriate. If in doubt, consult the supplier or installer of the modem.

This operand is valid only for incoming calls on switched links.

[AUTO=address]

Specifies that the auto call facility is installed and gives the automatic calling unit (ACU) interface address.

A line used to call a station (CALL=OUT or CALL=INOUT in this macro) may or may not be equipped with the ACU facility. If the line has the ACU facility, code the ACU interface address in hexadecimal, without framing (' ') characters (AUTO=020). If the line does not have the ACU facility, omit the operand.

A 1E line set is required for the ACU. Figure 5-7 gives the range of valid addresses.

[AUTUACB=symbol]

(user-written code only)

Specifies the name of the user adapter control block associated with the CSB address specified in the AUTO operand of this macro. This operand is required if user-defined line code is used to support a switched line using the auto-call facility. If auto-call or user-defined line code is not supported, this operand is invalid.

Note: The user adapter control block must be assembled with the NCP control blocks in stage 2 of the generation. (See the SRCHI and SRCLO operands in the GENEND macro.)

[BUFSIZE=n]

(type 3 scanner; emulation mode only)

Specifies the size of the emulation mode buffers for a line serviced by a type 3 communication scanner. Each line has two buffers for data being sent to or received from the line. This operand also determines the amount of data (up to 32 bytes) transferred over the channel to the host processor without program interrupt.

The value of *n* may be any of the following:

4 (see note)	96
8	128
16	160
32	192
64	224

Specifying larger buffers offers more protection against possible overruns than does specifying smaller buffers. (Overruns can result from temporary slowdowns of channel operation or from momentary peaks in data traffic through the network.) Use of larger buffers also results in less interrupt-processing overhead for line operations and up to 32 bytes less interrupt-processing overhead for channel operations. The amount of data transferred across the channel is equal to *n* up to 32 bytes. For values of *n* exceeding 32, the amount of data transferred over the channel is 32 bytes.

If this operand is omitted, two 32-byte buffers (*n*=32) are provided if the line speed is 19 200 bps or less. Two 64-byte buffers (*n*=64) are provided if the line speed exceeds 19 200 bps.

Note: Do not specify 4-byte buffers (*n*=4) if the line is polled (POLLED=YES) and the size of the poll entries in the service order table is 6 bytes or more. A poll entry comprises the address characters, ENQ, and index byte used to poll one station on the line (for example, C1 C1 40 40 ENQ *index*).

```
       {IN    }
[CALL={OUT   }]
       {INOUT}
```

(switched BSC and start-stop lines only)

Specifies whether stations or the NCP, or both, can initiate calls.

If the line is to be used only for incoming calls (stations call the controller), code CALL=IN, or omit the operand.

If the line is used only for outgoing calls (controller calls stations), code CALL=OUT. Also specify the name of the dial set for this line in the DIALSET operand.

If the line is to be used for both incoming and outgoing calls, code CALL=INOUT and specify the dial set name in the DIALSET operand.

This operand is valid only if you specify DIAL=YES in the GROUP macro, and it applies only to line operation in network control mode.

VTAM Note: For SDLC links, this operand provides information only to the VTAM initialization procedure and has no effect on the generation of the network control program. See the *ACF/VTAM Installation* manual for information on use of this operand for SDLC links.

```
[CHECK={DCD   }]
        {NODCD}
```

(switched, duplex, start-stop lines in emulation mode only)

Specifies whether the controller is to use the data carrier detect option for the line represented by this LINE macro. Use of this option prevents access to an application program's data by a station that dials the controller over this line at the moment the existing connection to a station is lost. Continuous monitoring of the data carrier detect signal from the modem gives positive assurance that the switched line connection is still established.

The CHECK operand is valid only if the LINE macro specifies DUPLEX=FULL and the GROUP macro specifies LNCTL=SS and DIAL=YES.

```
[CHNPRI={NORMAL}]
         {HIGH  }
```

(BSC lines in emulation mode only)

Specifies the priority of the emulation subchannel associated with this line with respect to the other emulation subchannels. This operand is valid only for a BSC line in emulation mode.

CHNPRI=NORMAL is ordinarily appropriate unless the data rate of the line is 19 200 bps or more and the majority of the remaining lines are slow-speed lines (2400 bps or less).

Note: Additional buffering is provided for lines in a type 2 communication scanner that require a high channel priority if CA=TYPE4 and OPCSB2=YES are specified in the BUILD macro. (See the description of the OPCSB2 operand.)

```
[CLOCKNG=( {INT} [,{INT}] )]
           {EXT}   {EXT}
```

Specifies whether the modem (data set) or the communication scanner for the line is to provide clocking.

CLOCKNG=INT specifies that the scanner provides clocking (business machine clocking). CLOCKNG=EXT specifies that the modem provides clocking.

If the ADDRESS operand in this LINE macro specifies one line interface address, specify only the first suboperand (CLOCKNG=INT or CLOCKNG=EXT).

If the ADDRESS operand specifies two line interface addresses (SDLC links only), specify only the first suboperand if both sides of the link use internal clocking or both use external clocking. If one uses internal clocking and the other uses external, specify CLOCKNG=(INT,EXT) or CLOCKNG=(EXT,INT), as appropriate. (The first and second suboperands apply respectively to the first and second addresses specified in the ADDRESS operand.)

If this LINE macro represents a BSC or SDLC line and the CLOCKNG operand is omitted, CLOCKNG=EXT is assumed. If the LINE macro represents a start-stop line, CLOCKNG=INT is assumed.

Note: Notice that the letter I is omitted from the CLOCKNG operand.

```
       {EBCDIC  }
       {USASCII }
       {BCD     }
       {EBCD    }
       {COR     }
[CODE={KATAKANA}]
       {ITA2    }
       {ZSC3    }
       {DIC1    }
       {BCD2    }
       {COR2    }
       {DIC3    }
       {ASCII   }
```

(start-stop and BSC lines only)

Specifies the transmission code the NCP uses to communicate with stations where a choice of codes is available. Determine from the system designer the code used. (The *ACF/NCP Program Reference Summary,* contains the translation tables associated with each transmission code.) This operand applies to all BSC lines and to start-stop lines in emulation mode only.

The NCP does not perform code translation for a line operating in emulation mode. The program transmits data in the same transmission code in which it was received. However, for BSC lines operating in emulation mode, the program must be aware of the transmission code employed (EBCDIC or USASCII). For start-stop lines operating in emulation mode, the program does not need to know the code being used.

Figure 5-14 lists the codes that may be specified for each type of station. ("WT TTY" refers to various European teletypewriters using either the ITA2 or ZSC2 transmission codes.)

The CODE operand should not be specified for any type of station not shown in Figure 5-14. (If a code is specified, it is ignored.) Also omit the CODE operand if the line is a multiple-terminal-access line or if the LINE macro is for an SDLC link.

		Type of Station					
Transmission Code	CODE=	Any BSC Station	IBM 1050	IBM 2740	IBM 2741	TWX	WT TTY
Extended BCD Interchange Code	EBCDIC	•[1]					
USA Standard Code for Information Interchange	USASCII	•					
USASCII (transparent)	USASCII-T	•[2]					
Extended Binary Coded Decimal	EBCD		•[1]	•[1]	•		
Binary Coded Decimal 1	BCD		•	•	•		
Binary Coded Decimal 2	BCD2		•		•		
Correspondence 1	COR			•	•[1]		
Correspondence 2	COR 2				•		
Katakana	KATAKANA		•	•			
Data Interchange Code 1	DIC1					•[1]	
Data Interchange Code 3	DIC3					•	
ASCII	ASCII					•	
International Telegraph Alphabet No. 2	ITA2						•[1]
Figure Protected Code	ZSC3						•

[1]Code assumed if CODE operand is omitted.
[2]Valid only for a line serviced by a type 3 scanner and operated in emulation mode.

Figure 5-14. CODE Operand Values

```
[CONFIG={SW     } ]
        {NONSW}
```

(remote NCP only)

Specifies whether the physical communication path making up the local-remote SDLC link is switched or nonswitched. If the path is switched (CONFIG=SW), the remote NCP can monitor the link for ring indicator signals (if RING=YES) and present an answer tone (if ANSTONE=YES) when the remote controller is called by the local NCP.

This operand is valid only in a remote NCP (TYPGEN=NCP-R) and only if the link to the local controller is specified as LNCTL=SDLC, DIAL=NO, and POLLED=NO in the LINE macro.

```
[CRRATE=integer]
```

(start-stop lines only)

Specifies the number of print positions the carriage returns for each idle character the NCP sends to a printer on this line. The minimum is 1; the maximum is 255.

If this operand is omitted, 40 print positions per idle character is assumed for 83B3, 115A, and TWX teletypewriter terminals; 13 is assumed for IBM 1050 terminals equipped with the accelerated carrier return feature. Ten positions per idle character is assumed for all other terminals.

This operand is valid only if LNCTL=SS is specified in the GROUP macro; it applies only to line operation in network control mode.

```
    {2701}
[CU={2702}]
    {2703}
```

(emulation mode only)

Specifies which transmission control unit functions the network control program is emulating for this line.

Code the value corresponding to the type of TCU that was attached to the line prior to installation of the communications controller.

```
[CUTOFF={count}]
        {NO   }
```

(start-stop and BSC lines only)

Specifies a maximum number of subblocks the NCP will accept from a station during line operation in network control mode. If the NCP receives this number of subblocks before receiving an end-of-block character from the station, it breaks off the transmission. (A subblock is the sequence of message text occupying the number of buffers specified by the TRANSFR operand of this macro.)

To limit the number of subblocks received, code CUTOFF=*count*, where *count* is from 1 to 255. If no limit is to be established, code CUTOFF=NO or omit the operand, and the NCP will continue to accept message text from the station until it receives an end-of-block character.

```
        {2972}
[CUTYPE={3271}]
        {3275}
```

(BSC lines only)

Specifies the type of control unit used for cluster type stations in emulation mode only. If the line defined by this macro operates in network control mode, specify the control unit in the CUTYPE operand in the CLUSTER macro and omit the operand from the LINE macro. If the line operates in both emulation mode and network control mode, specify the same value in the CUTYPE operands of both this LINE macro and the CLUSTER macro, or code one CUTYPE operand in the GROUP macro associated with this line.

```
[DATRATE=( {HIGH} [,{HIGH}] )]
           {LOW }   {LOW }
```

Specifies which of two data rates will be used for a dual-rate modem.

If the ADDRESS operand in this LINE macro specifies one line interface address, specify only the first suboperand. Code DATRATE=HIGH if the higher rate is to be used. Code DATRATE=LOW (or omit the operand) if the lower rate is to be used.

If the ADDRESS operand in this macro specifies two line interface addresses (SDLC links only), specify only the first suboperand if both sides of the link use the high rate or both use the low rate. If the rates differ, specify DATRATE=(HIGH,LOW) or DATRATE=(LOW,HIGH), as appropriate. The first and second suboperands apply respectively to the first and second addresses specified in the ADDRESS operand.

This operand should not be coded if the modem has only one data rate.

Note: DATRATE=HIGH is invalid for modems attached to line sets 1A, 1B, 1C, 2A, 3A, 4A, 4B, and 4C and, if specified, may cause a feedback-check error condition.

```
[DIALALT={dial set name}]
         {NONE          }
```

(nonswitched point-to-point or BSC multipoint lines only)

Specifies the name of a dial set (DIALSET macro) the NCP can use to select an alternate line to communicate with a station when unable to reach that station via the regular nonswitched point-to-point or BSC multipoint line. (This is called the switched network backup function and applies only to line operation in network control mode.) Upon request from the host processor, the NCP calls the station over a line from the dial set specified.

For the alternate line to be used successfully, the lines in the dial set must have the same characteristics as the nonswitched line.

Code a dial set name in this operand only if:

- the line represented by this LINE macro is a nonswitched point-to-point or BSC multipoint line
- You wish the NCP to reach the station over an alternate line when communication with the station over the regular nonswitched line is impossible
- You have specified switched network backup among the options included in the NCP (see the SYSCNTRL macro).

This operand is not valid for an SDLC link (LNCTL=SDLC is specified in the GROUP macro).

```
[DIALSET={dial set name}]
         {NONE          }
```

(call-out BSC or start-stop switched lines only)

Specifies the name of the DIALSET macro for the dial set for this line. The LINES operand of the DIALSET macro must contain the name of this LINE macro.

This operand must specify the name of a dial set if this LINE macro represents a switched point-to-point line and the network control program will call the stations. (CALL=OUT or CALL=INOUT must also be coded in this LINE macro.)

Code DIALSET=NONE (or omit the operand) if the line is used only for incoming calls (CALL=IN).

```
[DISABLE={YES}]
         {NO }
```

(emulation mode only)

Specifies whether the modem for this line requires a long disable time-out when disconnecting from the line. DISABLE=YES provides a 25.6 second time-out.

Most modems do not require the long time-out, and DISABLE=NO is recommended.

```
[DUALCOM={(line address,{A})}]
         {              {B} }
         { NONE             }
```

(emulation mode only)

Specifies that the NCP is to emulate the dual communications interface function for the line when operating in emulation mode. This operand applies only to a line emulating an IBM 2701 with the dual communications interface features.

line address

Specifies the *alternate* line interface address formerly attached to the 2701 dual communications interface.

A

Specifies that *this* line (address specified in the ADDRESS operand of this LINE macro) corresponds to dual communications interface *A*.

B

Specifies that *this* line (address specified in the ADDRESS operand of this LINE macro) corresponds to dual communications interface *B*.

NONE

Specifies that the 2701 being emulated was not equipped with the dual communications interface for this line.

Example: Assume that two communication lines that were formerly attached to an IBM 2701 via the dual communications interface and corresponded to dual communications interface *A* and *B* respectively, are now attached to the communications controller line addresses 020 and 021. Assume that the access method in the host processor is to communicate with these two lines alternately via emulation subchannel 017.

You would specify the respective LINE macros for the two lines such that (1) the ADDRESS operands of both specify the same emulation subchannel, 017; (2) each DUALCOM operand specifies the line address designated in the ADDRESS operand of the *other* LINE macro; and (3) the DUALCOM operand of each macro specifies the interface—*A* or *B*—to which the lines formerly corresponded when attached to the 2701:

```
LINE1 LINE ADDRESS=(020,17),   LINE2 LINE ADDRESS=(021,17),
           DUALCOM=(021,A),               DUALCOM=(020,B),
                .                              .
                .                              .
                .                              .
```

```
[DUPLEX=( {HALF} [, {HALF} ] )]
          {FULL}    {FULL}
```

Specifies whether the communication line and modem constitute a half-duplex or (full-)duplex facility.

Note: This should not be confused with half-duplex or duplex data transfer. This operand specifies only the physical characteristic of the communication facility (lines and modems).

If the ADDRESS operand in this LINE macro specifies one line interface address, code only the first suboperand (DUPLEX=HALF or DUPLEX=FULL).

If the ADDRESS operand specifies two line interface addresses, specify only the first suboperand if both sides of the link are half-duplex or both are duplex; if one is half-duplex and the other is duplex, specify DUPLEX=(HALF,FULL) or DUPLEX=(FULL,HALF), as appropriate. (The first and second suboperands apply respectively to the first and second addresses specified in the ADDRESS operand.)

Note: For SDLC, and BSC lines serviced by a type 3 communication scanner, coding DUPLEX=FULL causes the request-to-send signal to be active when either sending or receiving data. Coding DUPLEX=HALF causes the request-to-send signal to be active only when sending data. For SDLC links serviced by a type 2 scanner, this operand has no effect on the request-to-send signal. It is always active if the ADDRESS operand specifies two addresses and is active only when sending data if the ADDRESS operand specifies a single address.

```
[FEATURE=( ... )]
```

(emulation mode only)

Specifies features that certain BSC and start-stop terminals may have that must be specifically included in emulation mode. The features are specified by a list of keywords separated by commas and enclosed in parentheses. If the feature is not specified, the default value is assumed.

Note: The features in this operand are for emulation mode only. If the terminals are to operate in network control mode, the features must be specified in the FEATURE operand of the TERMINAL macro. If the same features apply to both emulation mode and network control mode, they may be specified in the GROUP macro.

```
[{DUALCODE}]
 {NODUALCD}
```

(emulation of 2701 BSC line only)

Specifies whether the program is to emulate the dual code feature of an IBM 2701. The dual code feature allows message transmission over the line in either EBCDIC or USASCII. Specify DUALCODE if either code can be used and the access method is capable of changing the code. Specify NODUALCD if the dual code function is not required.

```
[{IMEND   }]
 {NOIMEND}
```

(specially equipped start-stop terminals only)

Specifies whether the NCP immediately ends a receive operation from a start-stop terminal when the EOT is received. If IMEND is coded, the operation is ended immediately. If NOIMEND is coded, the program delays ending the receive operation until the line becomes electrically "quiet" following receipt of the EOT. The absence of further characters indicates that the EOT is valid.

```
[{LRC   }]
 {NOLRC}
```

(start-stop terminals only)

Specifies whether the start-stop terminals on this line are equipped with record-checking capability (either as an inherent function or as a feature).

LRC is appropriate for the following stations:

Station:	*TERM operand specifies:*
IBM 1050	1050
IBM 1060	1060
IBM 2740 Model 1	2740-1
IBM System/7	SYS7

```
[{SPACE   }]
 {NOSPACE}
```

(teletypewriter terminals only)

Specifies whether the NCP is to react to space characters received from 83B3, 115A, or World Trade teletypewriter terminals as downshift characters when the line is in emulation mode. If you specify SPACE, each space character received from a terminal causes the program to send all subsequent text characters to the host processor in their downshifted form.

If you specify NOSPACE (or omit the parameter), the program does not convert the characters to their downshifted form but sends them as received from the terminal.

```
[FGSLTRS={IN }]
         {OUT}
```

(teletypewriter lines in network control mode only)

Specifies whether the NCP is to delete FIGS and LTRS characters from message text received from 83B3, 115A, or World Trade teletypewriters terminals. This operand is valid for network control mode only.

```
[HDXSP={YES}]
       {NO }
```

(SDLC only)

Specifies whether sending data to a terminal has priority over polling. This applies only to SDLC links that have one line interface address (half-duplex) and more than one station active.

HDXSP=YES causes outgoing data that is ready for transmission to be sent at the first opportunity without having to wait until it is the destination terminal's time to be polled.

HDXSP=NO causes the outgoing data to be sent when it is the destination terminal's turn to be polled. In this case, the data is sent just before sending the polling sequence unless the link is in the poll-wait state. (See the PAUSE operand of the LINE macro for a description of poll wait.)

This operand is valid only if LNCTL=SDLC is specified in the GROUP macro; POLLED=YES and PAUSE=0, in the lLINE macro; and a single line interface address is specified in the ADDRESS operand of the LINE macro. This oeprand is invalid for a type 4 physical unit.

```
          {0}   {0}
[INTPRI=( {1} [,{1}] )]
          {2}   {2}
          {3}   {3}
```

Specifies the interrupt priority for this line relative to other lines attached to the controller. Priority 3 is highest and 0, lowest. Lines with high data rates should be assigned the higher priorities.

If the ADDRESS operand of this LINE macro specifies one line interface address, specify only the first suboperand.

If the ADDRESS operand specifies two line interface addresses (SDLC links only), specify only the first suboperand if the priority is the same for both sides of the link. Specify both suboperands if the priority differs. (The first and second suboperands are the priorities for the first and second addresses, respectively.)

Appendix E gives a method for determining the interrupt priorities for each line in the network.

[LINECB=symbol]

(user-written code only)

Specifies the name of a user-defined control block associated with a virtual link. This operand is valid only when VIRTUAL=YES is specified in the GROUP macro.

Note: The user control block must be assembled with the NCP in Stage 2 of the generation. (See the SRCHI operand in the GENEND macro.)

[LINEFVT=symbol]

(user-written code only)

Specifies the name of the function vector table associated with a virtual link. This operand is required if VIRTUAL=YES is specified in the GROUP macro. It is invalid if VIRTUAL=NO.

An EXTRN statement for each unique FVT symbol must be included in the user source code that is copied into the tables during assembly. (See the SRCHI operand in the GENEND macro.)

Note: The function vector table must be preassembled and included in the user object modules. (See the INCHI and INCLO operands in the GENEND macro.)

[LINESIZ=integer]

(start-stop lines in network control mode only)

Specifies the length of the print line for printer-type devices on this line. The minimum value for *integer* is 1; the maximum is 255. This operand applies only to line operation in network control mode.

If this operand is omitted, a line length of 72 print positions is assumed for TWX, 83B3, 115A, and World Trade teletypewriter terminals; 130 print positions is assumed for all other types of terminals.

```
[LNQTCNT={count}]
         {1    }
```

(start-stop lines in network control mode only)

Specifies the number of times the NCP tests a start-stop line after receiving data to ensure that the line has electrically "quiesced." The line must become electrically quiet before the next data transmission begins, or loss of message data may occur.

The default value of 1 is appropriate for most start-stop lines under 1200 bps. For lines operating at 1200 bps or more, several tests in succession may be necessary to ensure that the line is quiet. The recommended value for 1200 bps start-stop lines is 5.

The minimum value is 1; the maximum is 255.

```
[,MAXPU=count]
       {OPTION1}
```

(SDLC only)

Specifies the maximum number of physical units that may be associated with the link. If this operand is omitted, *count* is assumed to equal the number of PU macros following the LINE macro. If this link is a switched link or the PU is a type 4, MAXPU must equal 1.

The minimum value for this operand is the number of PU macros that follow this LINE macro. The maximum value is 255. If dynamic reconfiguration is not supported and MAXPU is coded, *count* must be equal to the number of PUs defined for this line.

Physical units may be added to a nonswitched SDLC link if dynamic reconfiguration is supported, *count* is greater than the number of PU macros, and the service order table contains an extension. The service order table extension is defined within the MAXLIST operand in the SERVICE macro. The number of entries in this extension should be equal to or greater than the number of physical units that may be added to the link.

This operand is not valid if VIRTUAL=YES is coded in the GROUP macro for this line.

```
        {OPTION1}
[MODEM= {OPTION2} ]
        {NTT     }
```

(emulation mode only)

Specifies whether the communication line (1) is enabled immediately after the communications controller has been loaded (IPL) or the System Reset key has been pressed (MODEM=OPTION1); or (2) is disabled after the controller has been loaded or the System Reset key has been pressed, and must subsequently be enabled by command from the access method (MODEM=OPTION2). Until the line is enabled, most commands issued to the line by the access method will result in an intervention required indication to the access method. Which of these two procedures is appropriate depends on the data-set-ready lead in the modem that attaches the line to the controller.

If the data-set-ready lead of the modem is continuously energized, specify MODEM=OPTION1. This is valid only for lines with IBM 1030, 1050, 1060, 2740 Model 1 or 2, 2741, or System/7 stations attached.

If the data-set-ready lead is *not* continuously energized (and the access method must issue an enable command to enable the line), specify MODEM=OPTION2.

Specify MODEM=NTT if the modems on the line are from the Nippon Telegraph and Telephone Company.

```
[MONITOR={YES} ]
         {NO }
```

(IBM 1050, 2741, TWX, and multiple-terminal-access lines in network control mode only)

Specifies whether the NCP is to monitor the line for an attention signal or a disconnect condition at the terminal. If MONITOR=YES, the program monitors the line whether or not it is currently executing a command for that line.

The access method is notified when either an attention signal or a disconnect condition is detected.

Note: If the logical keyboard lock for TWX terminals is coded, (KBDLOCK operand of the GROUP macro), MONITOR=YES must be specified.

```
[MPTALT={YES}]
        {NO }
```

(switched call-out BSC line only)

Specifies whether a switched BSC line can be used as an alternate line for communicating with BSC stations normally reached over a nonswitched multipoint line. When failure of the multipoint line prevents the NCP from communicating with a station over that line, the switched network backup facility can be used to reach the station over any switched call-out BSC line that has MPTALT=YES specified.

This operand is valid only if the line is a switched call-out BSC line (CALL=OUT or CALL=INOUT are specified in this LINE macro). If these operands are specified and you omit the MPTALT operand, MPTALT=NO is assumed.

```
[MTALIST=entry]
```

(call-in multiple-terminal-access lines only)

Specifies that this line is used for call-in multiple terminal access. It also identifies the list the network control program will use to identify the type of terminal calling the controller over this line. *entry* is the name of an MTALIST macro that defines the list that identifies the terminal type.

This operand is valid only if you specify DIAL=YES and LNCTL=SS in the GROUP macro and CALL=IN or CALL=INOUT in this LINE macro (or in the GROUP macro).

```
[NEGPOLP={n   }]
         {NONE}
```

(multipoint BSC lines in network control mode only)

Specifies a pause before polling is resumed after receiving a negative response to polling by a multipoint BSC station. The pause may be specified as seconds or as seconds and tenths. The minimum is .1 second; the maximum is 23.5 seconds.

If NEGPOLP=NONE is specified, there is no pause between receipt of the negative response and continuation of polling.

A negative polling pause limits the amount of nonproductive polling on a BSC multipoint line and therefore reduces the amount of processing overhead. However, too large a negative polling pause can increase the response time experienced by the operators of terminals on the line.

```
[NEWSYNC=({YES}[,{YES}])]
          {NO }  {NO }
```

(BSC and SDLC lines only)

Specifies whether the communications controller supplies the new sync signal to the modem (data set) on this line.

NEWSYNC=YES is valid only if:

- The modem (at the controller) serving the line has the new sync feature
- If the communications controller is the multipoint primary station for a duplex line
- The modem at the remote station does not continuously send carrier signal to the modem at the controller.

The modem at the remote station may send a continuous carrier signal or the station may send a continuous request-to-send to the modem. Consult your IBM representative or the installer of the modem for further information about these signals.

NEWSYNC=NO is required if there is only one remote station on the communication line.

If the NEWSYNC operand is omitted, NEWSYNC=YES is assumed if POLLED=YES, DUPLEX=FULL, CLOCKNG=EXT, and DIAL=NO are specified for this line. If any of these operands are not coded in this manner, NEWSYNC=NO is assumed.

If the ADDRESS operand in this LINE macro specifies one line interface address, specify only the first suboperand (NEWSYNC=YES or NEWSYNC=NO).

If the ADDRESS operand specifies two line interface addresses, specify only the first suboperand if the new sync option is to be used for both sides of the link (NEWSYNC=YES) or for neither (NEWSYNC=NO). If the option is required for only one of the two, specify NEWSYNC=(YES,NO) or NEWSYNC=(NO,YES), as appropriate. (The first and second suboperands apply respectively to the first and second addresses specified in the ADDRESS operand.)

```
[NRZI={YES}]
      {NO }
```

(SDLC lines only)

Specifies whether the data terminal equipment at the ends of the SDLC link must operate in non-return-to-zero-inverted (NRZI) mode (NRZI=YES) or in non-return-to-zero (NRZ) mode (NRZI=NO).

If internal (business machine) clocking is used on the link, specify NRZI=YES or omit the operand.

If external (modem) clocking is used on the link, specify NRZI=YES or omit the operand *unless* the modem is sensitive to repeated binary 10 bit patterns (that is, 10101010...). Sensitivity to this pattern can cause the modem to lose synchronism when it encounters sufficiently long sequences of this pattern in message data. If the modem is sensitive to this bit pattern, specify NRZI=NO. To determine whether your modems are sensitive to repeated binary 10 bit patterns consult your IBM representative (for IBM modems) or the supplier or installer of the modem (for non-IBM modems).

Note: NRZI=NO is the proper choice for most non-IBM modems; NRZI=YES is appropriate for IBM modems.

CAUTION
All business machine equipment (terminal equipment) on the same SDLC link must use the same encoding scheme; that is, all use NRZI mode (NRZI=YES) or all use NRZ mode (NRZI=NO). Mixing of modes on the same SDLC link will result in total lack of communication between stations on the link. In the case of an SDLC link between a local and a remote communications controller, the eqivalent choice (NRZI=YES or

NRZI=NO) is required for the LINE macros in each program (local and remote) that represent the SDLC link and must correspond to the setting of the NRZI bit in the IPL configuration data set of the remote program loader. (The IBM customer engineer sets the NRZI bit in the communications controller.) Some terminals require the user to set the NRZI bit.

This operand is valid only if LNCTL=SDLC is specified in the GROUP macro.

```
[PAD={YES}]
     {NO }
```

(BSC lines in emulation mode only)

Specifies whether a communications controller, emulating an IBM 2703, verifies the first 4 bits of trailing pad characters received from the lines.

If you specify PAD=YES or omit the operand, the controller checks each pad character received and indicates a data check error if the first 4 bits are not all 1's.

If you specify PAD=NO, the controller does not check the pad characters.

```
         {ODD    }
[PARCHK={EVEN   }]
         {NOCHECK}
```

(TWX lines in network control mode only)

Specifies whether the NCP checks for odd or even parity or does no parity checking for TWX terminals on the line. (Refer also to the PECHAR of the GROUP macro.)

```
         {EVEN}
[PARGEN={ODD }]
         {MARK}
```

(TWX lines in network control mode only)

Specifies that the network control program is to generate even, odd, or mark parity for transmitted data on the line. Mark parity generation always sets the parity bit to a 1.

```
[PAUSE={t}]
       {0}
```

For BSC and start-stop lines: Specifies the number of seconds of delay between successive service cycles when no sessions currently exist. *t* may be from 0 to 255; if you omit the operand, 0 is assumed, and there is no delay between successive cycles. This operand applies only to line operation in network control mode.

For SDLC links: Specifies the average duration of the polling cycle in seconds or seconds and tenths of seconds. The polling cycle extends from the moment the NCP examines the first entry in the service order table to the moment polling next begins at the same entry. It includes time for polling, reading, and writing to the terminals on the line. If the time expended in a complete polling cycle (servicing all active entries in the service order table) equals or exceeds the time specified as *t*, the next polling cycle begins immediately. If the time expended in a complete polling cycle is less than *t*, the beginning of the next polling cycle is deferred until *t* seconds have elapsed since the beginning of the cycle just completed. During the pause, the SDLC link is in poll-wait state; any outgoing data ready for transmission to the SDLC stations on the link is sent during this pause.

Allowing a pause to elapse when activity on the link is relatively low can reduce the amount of processing time consumed by unproductive polling.

Note: The larger the number of active entries in the service order table, the greater the likelihood that polling cycles will proceed continuously, without intervening pauses.

t may be from 0 to 25.5. If this operand is omitted, a pause of 0 seconds is assumed.

This operand is valid only if POLLED=YES in this LINE macro.

```
                  {WAIT  }
[POLIMIT=( [{n}] [,{NOWAIT}] )]
           {1}    {QUEUE }
```

(nonswitched multipoint start-stop or BSC lines in network control mode only)

Specifies the number of consecutive negative responses to polling the NCP will accept from a station before breaking the logical connection. The maximum value of *n* is 255.

This limit applies only to line operation in network control mode and only to polling performed after the NCP has received at least one message block from the station. It does not apply to initial polling.

This operand is valid only if the line is a nonswitched multipoint line.

Note: For a switched line used to communicate with IBM 1050 terminals, the NCP accepts up to 16 consecutive negative responses to polling and then executes the WAIT option.

```
 {WAIT  }
[{NOWAIT}]
 {QUEUE }
```

Specifies the action the network control program performs if the maximum number of negative responses, *n*, is reached.

WAIT

Specifies that the logical connection between the NCP and the station is to be maintained. The network control program notifies the host processor that the negative response limit has been reached and then waits for another request from the host processor before performing any further action on the line.

NOWAIT

Specifies that the NCP is to break the logical connection with the station, notify the host processor, and terminate the current read request.

Note: The default of (1,NOWAIT) cannot be used with VTAM.

QUEUE

Specifies that the NCP is to break the logical connection, notify the host processor, and queue the current read request onto the beginning of the queue for the station.

```
[POLLED={YES} ]
        {NO }
```

(lines in network control mode only)

Specifies whether the stations attached to the line must be polled and addressed.

Note: All lines in a line group must be specified the same (POLLED=YES or POLLED=NO). Both options cannot be included in the same line group.

Code POLLED=YES if:

- The line is a start-stop or BSC multipoint line (except when the controller is a tributary station on a BSC multipoint line).
- The line is a nonswitched or switched point-to-point line (including a multiple-terminal-access line) used to communicate with an IBM 1050 terminal.
- The line is an SDLC link, if the controller is the *primary* station on the link.

Code POLLED=NO if:

- The line is a start-stop nonswitched or switched point-to-point line or a BSC point-to-point line (unless the terminal is an IBM 1050).
- The line is a multiple-terminal-access line (MTALIST operand of this LINE macro is coded) over which no IBM 1050 terminals will communicate with the controller.
- The line is an SDLC link, if the controller in which the program is executed is the *secondary* station on the link. The physical unit on the link specified by this LINE macro must be type 4 (PUTYPE=4 is specified in the PU macro for the unit).

If this LINE macro represents a local-local SDLC link, the station at one end of the link must be a primary station, and the station at the other end must be a secondary station. That is, POLLED=YES and POLLED=NO must be specified, respectively, in the network control programs executed in the primary and secondary stations.

Note: If you specify that the controller in which the present NCP is executed is to be the primary station on the principal link, you must also specify it as the primary station for each backup SDLC link, if any. Similarly, you must specify the controller as the secondary station for any backup link if you specify it as the secondary station on the principal link.

If this LINE macro represents a local-remote link, the local NCP must specify that its 3705 is the primary station on the link (POLLED=YES), and the remote NCP must specify that its 3705 is the secondary station (POLLED=NO).

Note: POLLED=NO is invalid if the physical unit on the link represented by this LINE macro is a type 1 or type 2 physical unit.

If you omit this operand and if the line is a start-stop or BSC line (LNCTL=SS or LNCTL=BSC), POLLED=NO is assumed.

If you omit this operand, the line is a nonswitched SDLC link (LNCTL=SDLC and DIAL=NO), and:

- This is the *first* LINE macro following the GROUP macro, then POLLED=NO is assumed if the program is to be executed in a remote controller (TYPGEN=NCP-R). Otherwise, POLLED=YES is assumed.
- This is *not* the first LINE macro following the GROUP macro, then the value assumed is the same as that specified in the first LINE macro.

```
[POLLTO=(ERROR   )]
        (NEGRESP)
```

(start-stop lines in network control mode only)

Specifies whether a time-out occurring during polling is to be treated as an error condition or a negative response. If the time-out is treated as an error condition, the value specified in the CRETRY operand of the GROUP macro determines the number of subsequent attempts at polling. If the time-out is treated as a negative response, the value specified in the POLIMIT operand of this LINE macro (or the GROUP macro) determines the number of times the station is repolled.

```
[QUIET={YES}]
       {NO }
```

(start-stop lines in emulation mode only)

Specifies whether the program is to observe a long line quiet time-out of 25.6 seconds when receiving from the line. If you specify QUIET=YES, the program observes the long time-out. If you specify QUIET=NO (or omit the operand), the normal time-out of 3.0 seconds is observed.

QUIET=YES should not be specified if FEATURE=IMEND is specified for this line.

```
[REDIAL={(,[t1][,n[,t2]])    }]
        {([m][,t1][,n[,t2]])}
```

(switched SDLC links)
(switched BSC and start-stop lines in network control mode)

Specifies the number of dialing operations the network control program is to perform in attempting to reach a station over a switched line, and the pause desired between dialing operations.

- The number of dialing operations within each sequence is provided by the access method; this value (referred to as *m*) is therefore not specified in this operand. Code a comma directly following the (in this operand.)
- t_1 specifies (in part) the pause in seconds between successive dialing operations within each sequence. t_1 must be a multiple of three seconds or 0; the maximum is 765 seconds. (If you specify a value that is not a multiple of three, the next lower multiple of three is the value assumed by the generation procedure.)
- *n* specifies the desired number of sequences of $m+1$ dialing operations. The maximum is 254 sequences (n=254). (If you specify *n* as 255, an indefinite number of sequences of dialing operations will be performed.)
- t_2 specifies (in part) the pause in seconds between successive sequences of dialing operations. t_2 must be a multiple of three seconds or 0; the maximum is 765 seconds. (If you specify a value that is not a multiple of three, the next lower multiple of three is the value assumed by the generation procedure.)

The actual pause between dialing operations and between sequences of dialing operations is only partly represented by the t_1 and t_2 values, for the following reasons. (1) There is an inherent delay in the automatic calling unit (ACU) from the moment the unit transmits the last dial digit to the moment it abandons the call upon failure of the called station to answer. This delay is called the ACR (abandon call and retry) delay. (2) Each dialing operation is preceded by a Disable operation for the line, and the line remains disabled for the duration of the disable timeout specified by the DSABLTO operand of the BUILD macro.

The maximum number of dialing operations the network control program will perform is $(m+1)$ (n+1).

Example: Assume that you specify REDIAL=(,6,2,18), when the value of m (3) is provided by the access method. The maximum number of dialing operations $(m+1)$ $(n+1)$ will therefore be 12. The action of the network control program may be represented thus:

Dial (pause$_1$) Dial (pause$_1$) Dial (pause$_1$) Dial (pause$_2$)
Dial (pause$_1$) Dial (pause$_1$) Dial (pause$_1$) Dial (pause$_2$)
Dial (pause$_1$) Dial (pause$_1$) Dial (pause$_1$) Dial

pause $_1$ represents the pause whose duration is the sum of (1) the t_1 value (6 seconds), and (2) the ACR and disable timeouts. *pause* $_2$ represents the pause whose duration is the sum of (1) the t_2 value (18 seconds), and (2) the ACR and disable timeouts.

If you omit one or more parameters, the assumed values are:

t_1=0
n=0
t_2=0

```
[RETRIES={NONE         }]
         {(m[,t[,n]])}
```

(network control mode only)

Specifies the number of attempts to recover from errors occurring during transmission over the link specified by this LINE macro (with the exception of errors of the kinds mentioned in the note under "Maximum, Minimum, and Omitted Values").

If this LINE macro represents an SDLC link:

An error condition exists when the NCP does not receive a positive indication that a frame it has sent was successfully received, and the NCP retransmits that frame. If the error recurs, retransmission is repeated. The NCP persists in retransmitting until it successfully sends the frame or until a maximum number of retries (retransmissions) have been attempted.

When an error occurs while the network control program is receiving from a station, the NCP sends the station a command that causes the station to retransmit the frame in error. If the error recurs, the NCP repeats this command. The program persists in this way until (1) it has successfully received the frame; (2) it has retried the receive operation a user-specified maximum number of times; (3) the station sends an abnormal response (for example, a request for initialization).

The repetitive retries (retransmissions of data or of control commands) are called a retry sequence. The maximum number of retries in the sequence is specified as m, which may be from 0 to 128. (RETRIES=0 is equivalent to RETRIES=NONE, resulting in no retry attempts at all.)

Optionally, you may also specify that the NCP pause after completing the retry sequence, then begin a new retry sequence. This second sequence continues until the frame is successfully transmitted or received or the maximum, m, is again reached. Alternation of retry sequence and pause continues until the error is cleared or the maximum number of retry sequences is reached. The pause, specified by the t parameter, may be from 1 to 255 seconds. The maximum number of retry sequences, specified by n, may be from 1 to 127.

The maximum number of retries (retransmissions of data or of control commands)—$(n+1)m$—is 128.

The maximum number of retries per sequence, *m*, is always specified in the RETRIES operand of the LINE macro.

You may specify the *t* and *n* parameters individually for each station connected to the SDLC link represented by this LINE macro. (Specify these values in the RETRIES operand of the PU macro.)

The *t* and *n* parameters can be specified also in a PU macro associated with a LINE macro for a backup SDLC local-local or local-remote link (that is, a PU macro from which the SUBAREA operand is omitted).

Maximum, Minimum, and Omitted Values (SDLC Stations):

If *m* is 0 (or is omitted), *t* must be 0 (or must be omitted).
If *t* is 0 (or is omitted), *n* must be 0 (or must be omitted).

If *n* is omitted and *t* is *not* 0, $n=1$ is assumed if $(n+1)m$ is less than or equal to 128, and $n=0$ is assumed if $(n+1)m$ is greater than 128.

If *n* is omitted and *t* is 0, $n=0$ is assumed.

If you code RETRIES=NONE, no error recovery is attempted.

Note: Retries of the following kinds of error conditions are governed by fixed parameters within the NCP, not by the parameters you specify in the RETRIES operand: information frames received out of sequence, information frames having data check errors, and unexpected responses (such as a request for initialization).

If you omit this operand entirely, the assumed values will be $m=7$, $t=0$, $n=0$.

Figure 5-15 summarizes the maximum, minimum, and default values.

If this LINE macro represents a BSC or start-stop line:

Text-write errors: When an error occurs while the NCP is sending to a station (text-write error), the NCP retransmits the block on which the error occurred. If the error recurs, retransmission is repeated. The NCP persists in retransmitting until it successfully sends the block or until a maximum number of retries (retransmissions) has been attempted.

The repetitive retries (retransmissions of data) are called a retry sequence. The maximum number of retries in the sequence is specified as *m*, which may be from 0 to 255. (RETRIES=0 is equivalent to RETRIES=NONE, resulting in no retry attempts at all. RETRIES=255 specifies unlimited retries; that is, retransmissions are made without limit.)

Optionally, you may also specify that the NCP pause after completing the retry sequence and then begin a new retry sequence. This second sequence continues until the block is successfully transmitted or the maximum, *m*, is again reached. Alternation of retry sequence and pause continues until the error is cleared or the maximum number of retry sequences is reached. The pause, specified by the t parameter, may be from 0 to 255 seconds. The maximum number of retry sequences, specified by n, may be from 1 to 255.

Text-read errors: When an error occurs while the NCP is receiving from a station (text-read error), the NCP sends a negative response to the station, causing the station to retransmit the block in error. If the error recurs, the NCP again sends a negative response. The NCP persists in this way until (1) the block is successfully received; (2) the station sends an end-of-transmission (EOT) character or sequence instead of retransmitting the block; (3) the network control program has sent a user-specified maximum number of negative responses.

The repetitive retries (retransmissions of negative responses) are called a retry sequence. The maximum number of retries in the sequence is specified as m, which may be from 0 to 255. (RETRIES=0 is equivalent to RETRIES=NONE, resulting in no retry attempts at all. RETRIES=255 specifies unlimited retries; that is, retransmissions are made without limit.)

The parameters t and n do not apply to text-read errors.

Maximum, Minimum, and Default Values (BSC and Start-Stop Lines):

If $m=0$ or 255 (or is omitted), t must be 0 (or must be omitted).
If t is 0 (or is omitted), n must be 0 (or must be omitted).

If n is omitted and t is *not* 0, $n=1$ is assumed if $(n+1)m$ is less than or equal to 128, and $n=0$ is assumed if $(n+1)m$ is greater than 128.

If n is omitted and t is 0, $n=0$ is assumed.

If you code RETRIES=NONE, no error recovery is attempted.

If you omit this operand entirely, the assumed values will be $m=2$ (for start-stop stations) or $m=7$ (for BSC stations), $t=0$, $n=0$ if the type of station on the line is capable of retransmitting. That is, there will be one sequence of two retries for start-stop stations or seven retries for BSC stations. If the station is not of a type that is capable of retransmitting, m is assumed to be 0; that is, no retries at all are attempted.

Figure 5-15 summarizes the maximum, minimum, and default values.

		SDLC Links:	BSC and Start-Stop Lines:
m	Maximum:	128	255[4]
	Minimum:	0[1,2]	0[1]
	Default:	7	7 (BSC) 2 (start-stop) 0 (if station incapable of retransmitting)
t	Maximum:	255	255
	Minimum:	1	0
	Default:	0[3]	0
		(Note: t may be specified in the RETRIES operand of the PU macro.)	(Note: t applies only to text-write errors)
n	Maximum:	127	255
	Minimum:	1	1
	Default:	0, if $t=0$ 0, if $t\neq0$ and $(n+1)m>128$ 1, if $t\neq0$ and $(n+1)m\leq128$	0, if $t=0$ 0, if $t\neq0$ and $(n+1)m>128$ 1, if $t\neq0$ and $(n+1)m\leq128$
		(Note: n may be specified in the RETRIES operand of the PU macro.)	(Note: n applies only to text-write errors.)

[1] RETRIES=0 is equivalent to RETRIES=NONE.
[2] If $m=0$, specify $t=0$ (or omit t).
[3] If $t=0$, specify $n=0$ (or omit n).
[4] RETRIES=255 specified unlimited attempts (retransmission without limit).

Figure 5-15. Summary of Parameters for RETRIES Operand

```
[RING={YES}]
      {NO }
```

(switched links only; not applicable to U.S. and Canada)

Specifies whether the ring indicator mode of an automatic answer operation is used. This decision depends upon the type of modem (data set) that connects the link to the controller. Determine from the modem supplier or installer whether the modem has a ring-indicator interface lead.

If the modem has the ring-indicator interface, code RING=YES. If it does not, code RING=NO (or omit the operand).

The RING operand is valid only for a switched line.

```
[SCLSET={YES}]
        {NO }
```

(telegraph lines in network control mode only)

Specifies whether the communication line is attached to a type 2A line set (telegraph single current line set). SCLSET=YES is valid only for a start-stop line. If you omit this operand, SCLSET=YES is assumed for AT & T 83B3, WU 115A, and World Trade teletypewriter terminals; SCLSET=NO is assumed for all other terminals.

[SDLCST=(plcst,slcst)]

Specifies which SDLC selection table entry the NCP will use to operate the 3705 as a primary station on the link (*plcst* parameter) and which entry the program will use to operate the 3705 as a secondary station (*slcst* parameter). (Parameters specified in the GROUP macro associated with the link are used until the 3705 becomes the primary station on the link for the first time.)

The table entries, whose contents are defined by the following operands of the SDLCST macro, specify the parameters the network control program is to apply to the link. *plcst* and *slcst* are the *symbol* name of the SDLCST macro.

ADDR	RETRIES
IRETRY	SERVLIM
MAXOUT	TADDR
PASSLIM	TRANSFR

The SDLCST operand is valid only when the 3705 in which the NCP you are defining will be executed is to function alternately as a primary or as a secondary station on the link represented by this LINE macro.

See the description of the SDLCST macro for the conditions pertaining to the use of an SDLC selection table.

This operand is valid (1) only in a local network control program that is the secondary NCP for the link represented by this LINE macro (POLLED=NO is specified in this macro) and (2) only if this LINE macro represents an SDLC link (LNCTL=SDLC is specified in the GROUP macro) and the station on the link is a local communications controller (PUTYPE=(4,LOCAL) is specified in the PU operand representing the station).

[SECURE={YES}]
{NO}

(switched, duplex, start-stop lines in network control mode only)

Specifies whether the NCP is to use the secure option for the start-stop line represented by this LINE macro. Use of this option prevents access to an application program's data by a start-stop terminal that dials the communications controller over the line at the moment the existing connection to a different terminal is lost. If you specify SECURE=YES, the network control program continuously monitors the data carrier detect signal from the modem to give positive assurance that the switched connection is still established.

This operand is valid only if the LINE (or GROUP) macro specifies DUPLEX=FULL and the GROUP macro specifies LNCTL=SS and DIAL=YES.

[SERVLIM=count]

(multipoint lines and SDLC links in network control mode only)

For start-stop or BSC lines: Specifies the number of entries in the service order table for this line that the NCP is to check each time the program performs service seeking. The maximum you may specify is 255.

If you omit this operand and this LINE macro represents a start-stop or BSC line (LNCTL=SS, LNCTL=BSC), the program checks one-half of the table entries each time it performs service seeking.

For SDLC links: Specifies the maximum number of regular scans of the service order table that the network control program will make for normal servicing of physical and logical units on the link before it makes a special scan of the table. The regular scan of the table accommodates normal transmission of path information units between the access method or host application programs and physical or logical units on the link. In the special scan of the service order table, the NCP determines whether there are any outstanding commands from the access method to interrogate or alter the online status of any physical units on the link. If so, the program fulfills the first such command and then resumes regular scans of the table to perform normal servicing. If no status commands are outstanding, the program immediately resumes regular scans unless, in the previous regular scan, the program found no stations to be in the contacted state, that is, no station is presently active. In this case, resumption of regular scans occurs after a delay of 2.2 seconds. If more than one status command is outstanding, only one is honored each time the special scan is made; the remaining status commands are fulfilled one at a time, in turn, during subsequent special scans of the table.

Upon completing a regular scan, the program begins the special scan when (1) in the regular scan just completed, the program found no active stations or (2) the maximum number of regular scans specified by SERVLIM has been reached—whichever occurs first.

Specifying a low value in SERVLIM gives the NCP more frequent opportunities to fulfill accumulated status commands than does specifying a higher value. Such status commands can be fulfilled more promptly, but at the cost of frequent interruptions to normal servicing. Conversely, specifying a higher value in SERVLIM causes fewer interruptions to normal servicing of stations than does specifying a lower value, but delayed fulfillment of the status commands is more likely to result. The relative number of status commands the access method will issue for the link served by the service order table, the relative importance of the alternatives described above, and experience should influence your selection of a value for the SERVLIM operand.

CAUTION

The network control program will perform a time-out for any status command issued for the physical unit of an SDLC station whose power is off. The duration of this time-out interval is as specified in the REPLYTO operand of the GROUP macro.

Normal servicing of physical and logical units is interrupted during the time-out interval because regular scanning of the service order table is suspended. Only when the specified interval expires does regular scanning (and therefore normal servicing) resume. A low value for SERVLIM (which causes relatively frequent special scans) and/or a high value for REPLYTO (resulting in long time-out delays) will result in serious degradation of message throughput on the SDLC link if status commands are received for SDLC stations whose power is off. (The time-out interval recurs for each successive special scan.)

You may minimize the possibility of throughput degradation by specifying a high value in the SERVLIM operand and/or a low value in the REPLYTO operand. Alternatively, the risk may be averted by arranging network operating procedures to avoid sending commands to physical units whose power is off.

If you omit this operand and this LINE macro represents an SDLC link (LNCTL=SDLC), a value of 4 (four regular scans of the table) is assumed.

This operand is valid only for a nonswitched start-stop, BSC or SDLC link (operating in network control mode) for which DIAL=NO and POLLED=YES are specified in the GROUP and LINE macros, respectively.

```
[SERVPRI={OLD}]
         {NEW}
```

(multipoint start-stop and BSC lines in network control mode only)

Specifies whether the network control program is to give priority to servicing current sessions (SERVPRI=OLD) or to establishing new sessions (SERVPRI=NEW) on the line.

This operand is valid only for a nonswitched line in network control mode with multipoint line control.

```
[SESSION={count}]
         {1    }
```

(multipoint start-stop and BSC lines in network control mode only)

Specifies the number of sessions the NCP is to attempt to maintain concurrently on a nonswitched line in network control mode. The maximum number of sessions must not exceed the number of devices connected to the line. If this operand is omitted, no more than one session will be maintained on the line. This operand is valid only for a nonswitched line with multipoint line control.

This operand does not limit the number of sessions if clustered stations are attached to the line and general polling is used. The NCP cannot control the number of terminals that may respond to a general poll, and a separate session is established with each responding terminal.

Note: If this line has IBM 3270 terminals attached, SESSION should equal the sum of the number of cluster controllers and the number of terminals on the line.

```
[SPDSEL={YES}]
        {NO }
```

(lines in network control mode only)

Specifies whether the data rate of a dual-rate modem may be changed by request from the access method.

Specify SPDSEL=YES to allow the program to change the data rate of the modem. SPDSEL=YES is valid only if external clocking is used on the line. The data rate specified in the SPEED operand must be the higher of the two data rates.

Specify SPDSEL=NO (or omit the operand) if the modem has one data rate or if it has two data rates but the NCP is not to change the rate.

```
[SPSHIFT={YES}]
         {NO }
```

(teletypewriter lines in network control mode only)

Specifies whether the NCP is to react to space characters received from 83B3, 115A, or World Trade teletypewriter terminals as downshift characters when the line is in network control mode. If SPSHIFT=YES, each space character received from a terminal causes the program to send all subsequent text characters to the host processor in their downshifted form.

If you code SPSHIFT=NO (or omit the operand), the program does not convert the text characters and sends them as they are received from the terminal.

```
[TADDR={chars}]
      {NONE }
```

(tributary controller on BSC line or remote controller on SDLC link only)

Specifies, if this program is to be executed in a tributary controller on a BSC line or a 3705 operating as a secondary station on the SDLC link represented by this LINE macro, the 1-character address you wish to assign to the controller.

If this program is to be executed in a tributary controller on a BSC line: Code TADDR=*char*, where *char* is the 2-digit hexadecimal representation of a single character (in transmission code). The character specified must be the same as the polling character that is specified in the NCP for the controller that controls this line. If, for example, *A* is the polling character to which this tributary controller is to respond, you would specify *A* as the tributary address in this LINE macro; and in the other network control program, you would specify *A* in the POLL operand of the appropriate TERMINAL macro. (See also the ADDR operand of the TERMINAL macro.)

Although the same character—*A*—is required in the TADDR and POLL operands in the respective programs, the way in which you specify the character may differ. In the POLL operand, you code the hexadecimal representation of the *EBCDIC* letter A (POLL=C1). In the TADDR operand of this LINE macro, you code the hexadecimal representation of the *transmisssion code* bit pattern. If the transmission code used on the line is EBCDIC, the hexadecimal representation is the same (POLL=C1, TADDR=C1). If, however, the transmission code is USASCII, you would code the USASCII bit pattern for the letter A in the TADDR operand (TADDR=41).

Note: The polling character you assign to the tributary controller must have a bit pattern in which the third bit from the left (bit 2) is 0. Any uppercase alphabetic character in either EBCDIC or USASCII (except EBCDIC letters S-Z) meets this requirement. Any other bit pattern that meets this condition can also be assigned, whether or not that bit pattern represents a character. For example, all of the following are valid polling characters for a tributary controller:

EBCDIC Character	*Bit Pattern (Binary)*	*(Hex)*
A	1100 0001	C1
!	0101 1010	5A
(none)	0101 0100	54

USASCII Character	*Bit Pattern (Binary)*	*(Hex)*
A	0100 0001	41
[	0101 1011	5B

If this program is to operate the 3705 as a secondary station on the SDLC link:

Code TADDR=*char*, where *char* is the 2-digit hexadecimal representation of a single EBCDIC SDLC station address; that is, the address of the secondary station (3705 in which the present NCP will be executed). You may assign as the address any bit configuration except hexadecimal 00 or FF.

However, the address you specify in the PU macro within the NCP executing in the primary station (3705) must be identical to the address you specify in this TADDR operand. In addition, if the present NCP is a remote program (TYPGEN=NCP-R), these addresses must be identical to the physical address entered into the IPL configuration data set in the diskette contained within the remote controller. (This address is entered by the customer engineer.)

This operand is required (and is valid) only if POLLED=NO is specified in this LINE macro.

`[TERM=type]`

(start-stop and BSC lines only)

Specifies the type of station the network control program will communicate with over this line. It must be one of the types listed in Figure 5-16.

This operand is required if the line is in emulation mode. If the line is to be operated only in network control mode, the type of station may be specified in this operand or in the TERM operand of the TERMINAL macro.

Note that in Figure 5-16 some types of stations are restricted to operation in network control mode. Stations of these types (IBM 3780 and System/7; MTA terminals) must not be specified in this TERM operand if the line is to operate in both network control and emulation mode. Stations that cannot be specified for line operation in emulation mode do not appear in Figure 5-14.

Note 1: If the CUTYPE operand is coded in the LINE (or GROUP) macro, this TERM operand is ignored.

Note 2: If different types of BSC devices (for example, 3275s and 3277s) are attached to the line, specify only one of the device types.

If type of station is:	*Code TERM=*
IBM 1050 Data Communication System	1050
IBM 1130 Computing System	1130
IBM 1800 Data Acquisition and Control System	1800
IBM System/360 Model 20	2020
IBM System/360 Model 25	2025
IBM 2701 Data Adapter Unit	2701
IBM 2703 Transmission Control	2703
IBM 2715 Transmission Control Unit Model 2	2715
IBM 2740 Model 1 Communications Terminal	2740-1
IBM 2740 Model 2 Communications Terminal	2740-2
IBM 2741 Communications Terminal	2741
IBM 2770 Data Communications System	2770
IBM 2780 Data Transmission Terminal	2780
IBM 2972 General Banking Terminal System:	
IBM 2980 Models 1 and 4 Teller Station[1]	2980
IBM 2980 Model 2 Administrative Station[1]	2980
IBM 3270 Information Display System:	
IBM 3275 Display Station[3]	3275
IBM 3277 Display Station[2 3]	3277
IBM 3276 Control Unit Display Station [2 3]	3277
IBM 3278 Display Station [2 3]	3277
IBM 3284 Printers[2 3]	3284
IBM 3286 Printer[2 3]	3286
IBM 3287 Printer [2 3]	3284
IBM 3288 Line Printer [2,3]	3286
IBM 3289 Line Printer [2 3]	3286
IBM 3650 Retail Store System (in BSC mode)	SYS3
IBM 3660 Supermarket System (in BSC mode)	SYS3
IBM 3704 Communications Controller	3704
IBM 3705 Communications Controller	3705
IBM 3735 Programmable Buffered Terminal	3735
IBM 3740 Data Entry System:	
IBM 3741 Data Station	3741
IBM 3747 Data Converter	3747
IBM 3767 Communications Terminal (in start-stop mode):	
supported as 2740 Model 1	2740-1
supported as 2740 Model 2	2740-2
supported as 2741	2741
IBM 3770 Data Communic... ns System (in BSC mode)	2770
IBM 3780 Data Communications Terminal	3780
IBM System/370 Model 135	3135
IBM System/370 Model 125	3125
IBM System/3	SYS3
IBM System/7[4] (BSC version)	(see note 4)
IBM System/7 (start-stop version)	2740-1
IBM System/32 (BSC version)	SYS3
IBM Communicating Magnetic Card Selectric Typewriter	2741
AT & T 83B3 Selective Calling System	83B3
Western Union Plan 115A Outstations	115A
Western Union Teletypewriter Exchange Service (TWX)	TWX[5]
World Trade Teletypewriter Terminals[1]	WTTY
Multiple Terminal Access (IBM 1050, 2740, 2741; TWX)	MTA

[1] Attached to an IBM 2972 Model 8 or 11. Specify TERM=2980 only in a TERMINAL macro that follows a CLUSTER macro in which CUTYPE=2972 is coded.

[2] Valid only in a TERMINAL macro that follows a CLUSTER macro in which CUTYPE=3271 is coded.

[3] Not supported on switched lines.

[4] Specify a System/7 (BSC version) as follows: (1) Specify TERM=SYS3 in the TERMINAL (or LINE or GROUP) macro if the line is to operate only in network control mode. (2) Specify TERM=SYS3 in the LINE (or GROUP) macro if the line is to operate in both network control and emulation modes (alternately). (3) Specify TERM=SYS7 in the LINE (or GROUP) macro if the line is to operate only in emulation mode.

[5] The network control program provides support only for switched connection TWX terminals at 110 baud (or up to 300 baud with MTA) using business machine (internal) clocking. Even, odd, or no checking can be specified for received data, and even, odd, or mark parity generation can be specified for transmitted data.

Figure 5-16. Values for TERM Operand

[TRANSFR=count]

Specifies the number of NCP buffers corresponding to the maximum amount of data (the data transfer limit) that the NCP is to receive from the line during a single data transfer operation.

The path between the NCP being defined and the destination may comprise (1) a channel connection to an attached processor or (2) a combination of SDLC links (with or without intervening channel connections) between network control programs and a channel connection to a distant host processor.

You may use this operand to limit the amount of data accepted from a line, during a single data transfer, to no more than the access method in the destination host processor (and intermediate access methods, if any) can accept in a single channel operation (or to no more than the data transfer limit imposed by a subsequent network control program in the path). You may also use this operand to limit the number of buffers that the network control program can assign at one time to a station on the line thus, preventing that station from monopolizing the available NCP buffers.

The value you specify in the TRANSFR operand must lie between certain minimum and maximum values, which you may calculate as follows.

To calculate maximum value of count:

1. Determine which host processors in the network will receive the data.
2. Calculate the data transfer limit, in bytes, imposed by the access method in each destination host processor. To do so, multiply the values specified by the MAXBFRU and UNITSZ operands, then subtract the value of the BFRPAD operand, of the HOST macro that represents the access method to the network control program.
3. Determine which of the data transfer limits thus calculated is the smallest and divide that limit by the buffer size specified in the BFRS operand of the BUILD macro in the present network control program. The result (ignoring any fractional remainder) is the maximum value of *count* you should specify in this TRANSFR operand.

Example: Figure 5-17 shows three host processors, each of which may be a destination host processor for lines and stations attached to any of the three 3705s shown. As calculated in the manner described above, the data transfer limits are 5972, 7472, and 4972, respectively, for host processors 1, 2, and 3. You would therefore select the smallest limit, 4972, and divide it by the buffer size (BFRS operand), 232 bytes, in NCP *A*. The result, 21 (ignoring the fractional remainder), is the maximum value you would specify in the TRANSFR operand of the LINE macro for the line from which the data is received. You would proceed similarly to determine the value of TRANSFR for each line controlled by network control programs *B* and *C*.

It should be understood that the values thus calculated are only maximums; you may wish to limit the amount of data received from a line to much less than that which the access method will accept over the channel.

To calculate minimum value of count:

1. Establish the maximum amount of data to be received from the line represented by this LINE macro.
2. Add 23 bytes (representing control information) in addition to user data.
3. Divide the sum (maximum user data from line + 23 bytes) by the NCP buffer size (BFRS operand).
4. Round the result, if not an integer, to the next higher integer.

Example: Assume that a maximum of 700 bytes may be received in a single data transfer over the line represented by this LINE macro, and that the NCP buffer size is 232 bytes. Adding 23 to 700 and dividing by 232 yields 3.1. The minimum value you would specify in the TRANSFR operand would therefore be 4.

For a complex network that includes several network control programs and/or access methods, it is more convenient to establish the same data transfer limits (in bytes) for all lines between network control programs. Note, however, that the values you specify in the TRANSFR operands of the several network control programs will be the same only if the buffer size (as specified by the BFRS operands in these programs) is identical.

The minimum value accepted by the program generation procedure is one buffer (TRANSFR=1); the maximum is 255 buffers (TRANSFR=255).

If you omit this TRANSFR operand, the program generation procedure determines a value in one of the following ways:

- If the TRANSFR operand is specified in the BUILD macro (see the operand description under that macro), the value it specifies is assumed for the line represented by this LINE macro.
- If the TRANSFR operand is omitted from the BUILD macro but one or more HOST macros appear in the source statements for the present network control program, the generation procedure computes the data transfer limit(s) from the MAXBFRU, UNITSZ, and BFRPAD operands as described above. It then divides the result (lowest result, if there is more than one HOST macro) by the NCP buffer size (BFR), and uses the resulting value (or a maximum of 255) as the maximum buffer count for the line.
- If the program contains neither a TRANSFR operand in the BUILD macro nor HOST macro(s), the generation procedure assumes an arbitrary value of seven buffers for the line (TRANSFR=7).

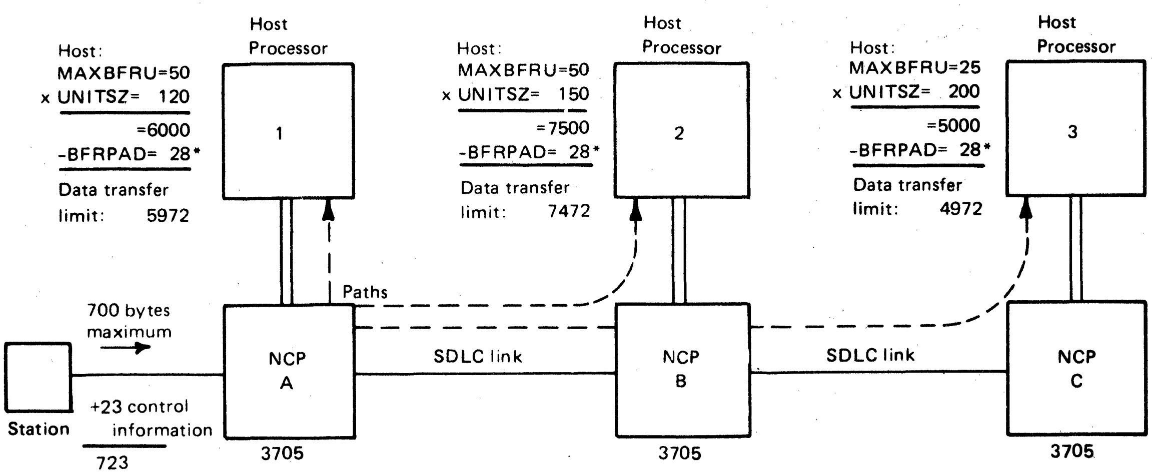
When the buffer limit for the line is reached, NCP procedures differ for SDLC and non-SDLC (BSC and start-stop) lines.

For an SDLC link, if the end of the path information unit (PIU) has not been received by the time the buffer limit is reached, the network control program discards all the data received and sends a negative acknowledgment to the sending station. (The discarded data is *not* sent to the host processor.)

For a BSC or start-stop line, the program sends the filled buffers to the host processor as a subblock and continues to receive message data from the station until the program receives an end-of-block or end-of-transmission character or until the subblock limit you have specified in the CUTOFF operand of this LINE macro (or the GROUP macro) is reached.

This operand applies only to line operation in network control mode.

BSC,SS,SDLC LINE NCP,PEP



BUILD: BFRS= 60

LINE: TRANSFR= 82 (maximum) $\left(\frac{4972}{60} = 82.9\right)$ TRANSFER= 13 (minimum) $\left(\frac{723}{60} = 12.1\right)$

*This value is only for OS/VS VTAM.

Since host processor can receive 82 and the minimum is 13, any value between 13 and 82 can be used.

Figure 5-17. Determining Maximum and Minimum Values of TRANSFR Operand

[TYPE={NCP}]
       PEP}

(start-stop and BSC lines only)

Specifies whether the line is to operate in network control mode only (TYPE=NCP) or alternately in either network control mode or emulation mode (TYPE=PEP). Refer to the following chart.

If...	*And...*	*Then...*
BUILD: TYPGEN=	**GROUP: TYPE=**	**LINE: TYPE=**
NCP (or NCP-LR) or NCP-R	NCP	NCP
PEP (or PEP-LR)	PEP[1]	PEP[1] NCP
	NCP	NCP

[1]Not valid if LINE macro represents an SDLC link.

If you omit the TYPE operand from the LINE macro, the value assumed is the same as specified in the TYPE operand of the GROUP macro.

If you specify TYPE=PEP, all operands of this LINE macro may be specified. If you specify TYPE=NCP, only the operands for network control mode should be coded.

[UACB=(symbol1 [,symbol2])]

(user-written line control only)

Specifies the names of the user adapter control blocks associated with this line. This operand is valid only when user-defined code is specified for levels 2 and 3 in the GROUP macro.

symbol1 is the name of the user adapter control block defined by the first address specified in the ADDRESS operand in this macro.

symbol2 is the name of the user adapter control block defined by the second address specified in the ADDRESS operand in this macro. If two addresses are specified in the ADDRESS operand, *symbol2* is required.

Note: The user adapter control blocks must be assembled with the NCP control blocks in stage 2 of the generation. See the SRCLO operand in the GENEND macro.)

```
[UNITXC={YES}
        {NO }
```

(emulation mode only)

Specifies whether the program is to signal unit-exception status to the host processor when the program receives an EOT from the line.

It is normally appropriate to specify UNITXC=YES (or omit the operand), which causes the program to signal unit-exception status upon receiving an EOT.

However, if read and write commands within the access method are command chained, UNITXC=NO may be appropriate. UNITXC=NO, by suppressing the Unit Exception indication, prevents the command chain from being broken. (Unit Exception status always breaks the command chain.)

```
[USE={NCP} ]
     {EP }
```

(lines operable alternately in network control and emulation modes)

Specifies whether the line operates initially in network control mode (USE=NCP) or emulation mode (USE=EP).

If TYPE=PEP is specified in this macro, either USE=NCP or USE=EP is valid; if you omit the USE operand, USE=NCP is assumed.

Switching a line from one mode to another is done by a request from the access method to the NCP.

This operand has no meaning unless TYPE=PEP is specified for this line.

```
[YIELD={YES} ]
       {NO }
```

(BSC lines in network control mode only)

Specifies whether the controller is to be the secondary station on a nonswitched point-to-point (contention) line. This operand applies only to line operation in network control mode.

Code YIELD=YES (or omit the operand) if the controller is to yield to the primary station when contention occurs. Code YIELD=NO if the controller is to be the primary station.

This operand is valid only for a BSC point-to-point line (GROUP macro specifies LNCTL=BSC and LINE [or GROUP] macro specifies POLLED=NO [or POLLED operand is omitted]).

Switched Virtual Link Support

To define a virtual link in a switched line group, specify a LINE macro followed by one PU macro. The SERVICE and LU macros may not be specified. Specify the MAXLU, LUCB, and LUFVT operands in either the PU, LINE, or GROUP macro.

The generation procedure creates the number of logical units specified in the MAXLU operand. Each of these logical units has the same user control blocks and function vector tables. A network address is also assigned to each LU.

Specifying Lower-Level Operands in a Higher-Level Macro

In addition to the preceding operands, most operands of the PU, LU, CLUSTER, TERMINAL, and COMP macros can be specified in the LINE macro (or the GROUP macro) instead of the individual macros mentioned. Similarly, most operands of the LINE macro can be specified in the GROUP macro. Figure 5-11 shows which of the lower-level operands you may specify at a higher level.

SERVICE Macro Instruction

The SERVICE macro generates a service order table for a nonswitched multipoint start-stop or BSC line, or for a nonswitched (multipoint or point-to-point) SDLC link. All lines are treated as a multipoint line, regardless of the number of stations on the line if they must be polled or addressed. Do not code a SERVICE macro for a switched link.

For a nonswitched start-stop or BSC line, a SERVICE macro is required directly following each LINE macro where POLLED=YES is specified. Each individually polled and addressed device represented by a TERMINAL or COMP macro (or by a CLUSTER macro in which the GPOLL operand is specified) must be represented by at least one entry in the service order table.

For an SDLC link, one SERVICE macro is required following the LINE macro that represents the link. Each SDLC station or communications controller attached to the link must be represented by an entry in the service order table. (Logical units associated with the SDLC station are not represented in the service order table.)

A nonswitched link between a local NCP and a remote NCP is always controlled by the local NCP. Therefore, a service order table is required only in the local (not the remote) NCP. For a link between two local network control programs, a service order table is required only within the controlling (primary) NCP. A SERVICE macro is required after the LINE macro representing the link only if POLLED=YES and DIAL=NO are specified in the LINE macro. A SERVICE macro is *not* required for a secondary station that might become a primary station.

If more than one SDLC link connects a local and a remote 3705, or two local 3705s, a SERVICE macro must appear in the primary NCP for the links, following each of the LINE macros representing the links.

The format of the SERVICE macro is:

```
[symbol]        SERVICE      operand[,operand]
```

Operands

```
ORDER=(entry,...)
[,MAXLIST=n]
```

`[symbol]`

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity. If more than one macro is required, specify a name on the first macro only.

`ORDER=(entry,...)`

Specifies the order in which the stations on the line are to be serviced.

entry is the name of a station, clustered terminal, component, or remote communications controller. These names appear on the TERMINAL, CLUSTER, COMP, or PU macros associated with the line for which this service order table is being generated. A station or component name may appear in the list more than once.

You may code a maximum of 255 characters in the ORDER operand, including the beginning and ending parentheses and all commas. This limit applies regardless of how many entries you code within the operand. If you need to specify more than 255 characters, code one or more additional SERVICE macros following the first. Specify the remaining entries in the ORDER operand and omit the MAXLIST operand in the additional macros. The maximum number of entries is 256.

Note: The generation procedure checks the first 35 entries of the service order table to determine if the devices represented by CLUSTER, TERMINAL, COMP, or PU macros are associated with this line. Also, each CLUSTER, TERMINAL, COMP, and PU macro associated with the LINE macro is checked to determine if it is represented in the service order table. Any service order table entries beyond 35 are not checked.

[MAXLIST=n]

Specifies the maximum *number* of entries in the service order table. The maximum value for *n* is 256. If the number you specify in MAXLIST exceeds the number of entries you code in the ORDER operand, you may add more entries (up to the MAXLIST limit) during network control program execution through the dynamic reconfiguration procedure.

If you omit this operand, *n* is assumed to equal the number of entries in the ORDER operand, and no further entries may be added during network control program execution. (Entries may be *changed*, however, with the dynamic reconfiguration function.)

PU Macro Instruction

The PU macro represents a physical unit of any type (1, 2, or 4) with which the network control program communicates over a nonswitched or switched SDLC link.

The PU macro specifies:

- The resource name for the physical unit.
- The station address of the physical unit.
- The physical unit type (1, 2, or 4).
- The maximum amount of data the physical unit can receive from the NCP in one segment.
- The maximum number of consecutive path information units (PIUs) or PIU segments the NCP will send to the physical unit before servicing other physical units on the link.
- Whether immediate polling retry is required.
- The number of error-recovery attempts the NCP will make when transmission errors occur.
- The maximum number of path information units or PIU segments the NCP will send to the physical unit before requesting a response from the unit.
- The subarea address of the physical unit (type 4 only).
- The maximum number of logical units associated with the physical unit.

  Other information, equivalent to that explicitly specified in this macro for a nonswitched physical unit, is passed to the NCP by the access method when the switched physical unit is called.

The sequence of PU macros (with intervening LU macros, if required) must appear directly following the SERVICE macro that defines the service order table for the SDLC link:

```
LINE POLLED=YES
SERVICE
PU
LU
LU
PU
LU
LU
.
.
.
```

If the PU macro appears in the secondary network control program for the SDLC link (and the PU macro thus represents the primary NCP for the link), no SERVICE macro is used. The sequence then is simply:

```
LINE
PU
```

If the primary and secondary network control programs can communicate over one or more backup (alternate) SDLC links in lieu of the principal link, a PU macro must appear after each LINE (or SERVICE) macro corresponding to an alternate link. Omit the SUBAREA operand from each of these alternate PU macros. Also, if the alternate PU macro is coded in a primary network control program (and thus represents the secondary NCP), omit all other PU macro operands except PUTYPE, IRETRY, and RETRIES; or, if the alternate PU macro is coded in a secondary network control program (and thus represents the primary NCP), omit all other PU macro operands except MAXOUT and PASSLIM, which should specify the same values as these operands specify in the *principal* PU macro.

If the NCP being defined is for a remote communication controller, the SUBAREA operand is omitted from any PU macros that define type 4 physical units. The SUBAREA operand is omitted regardless of whether this PU macro and its associated LINE macro request the principal SDLC link or a backup (alternate) SDLC link.

Example:

Local communications controller:
Primary NCP (subarea 4)

```
L1       LINE     POLLED=YES,
                  RETRIES=5

         SERVICE  ORDER=(RC1)

RC1      PU       SUBAREA=5,
                  PUTYPE=4,
                  DATMODE=FULL,
                  MAXOUT=7,
                  PASSLIM=200,
                  RETRIES=(10,5)

L1ALT    LINE     POLLED=YES,
                  RETRIES=5

         SERVICE  ORDER=(RC1ALT)

RC1ALT   PU       PUTYPE=4,
                  RETRIES=(10,5)
```

Remote communications controller:
Secondary NCP (subarea 5)

```
LA       LINE     POLLED=NO

LC1      PU
                  PUTYPE=4,
                  DATMODE=FULL,
                  MAXOUT=7,
                  PASSLIM=200

LAALT    LINE     POLLED=NO

LC1ALT   PU       PUTYPE=4,
                  MAXOUT=7,
                  PASSLIM=200
```

(VTAM Users Only): Appearing at the end of the following list of operands are the VTAM-only operands that may be coded in this macro instruction. The VTAM operands provide information only for the VTAM initialization procedure and are not required (though are permissible) as input for the NCP generation procedure. See the *ACF/VTAM Installation* manual for descriptions of these operands and for information on the VTAM initialization process. (See the *NOSP Installation Manual* for the description of the SPAN operand.)

The format of the PU macro is:

```
symbol        PU          [operands]
```

Operands

```
[ADDR=chars]
       {STOP     }
[,ANS={CONTINUE}]
       {CONT     }
[,BNNSUP=3270]
[,DATMODE={HALF}]
           {FULL}
[,IRETRY={YES}]
          {NO }
[,MAXDATA=size]
[,MAXLU=count]
[,MAXOUT={n}]
          {1}
[,PASSLIM={n}]
           {1}
[,PUCB=symbol]
[,PUDR={NO }]
        {YES}
[,PUFVT=symbol]
          {1                 }
          {2                 }
[,PUTYPE={(1,2)              }
          {(4[,{LOCAL }])}
                {REMOTE}
[,RETRIES=(,[t][,n])]
[,SUBAREA=n]
```

VTAM-only operands:

```
BUFLIM=
DISCNT=
DLOGMOD=
FEATUR2=
ISTATUS=
LOGAPPL=
LOGTAB=
MODETAB=
SPAN=
SSCPFM=
TERM=
USSTAB=
VPACING=
```

symbol

Provides a resource name for the physical unit and is required. *symbol* may be any valid assembler-language symbol; the first character may not be $.

[ADDR=chars]

(nonswitched link only)

Specifies the hexadecimal representation of the 8-bit address of the physical unit represented by this PU macro. Any bit configuration except hexadecimal 00 or FF is valid. This operand is invalid if (1) POLLED=NO is specified in the LINE macro; (2) POLLED=YES and PUTYPE=4 are specified but the SUBAREA operand is omitted; or (3) DIAL=YES is specified in the GROUP macro.

This operand is required if the communications facility is nonswitched (DIAL=NO), the physical units on the link must be polled (POLLED=YES), and the physical unit type is 1, or 2, or 4 (and the SUBAREA operand is specified.)

Note: For a type 4 physical unit, the address specified in the ADDR operand in the *primary* NCP and the address specified in the TADDR operands of the LINE and SDLCST macros within the secondary NCP must be identical. In addition, if the secondary NCP is executed in a remote controller, this address must be identical to the physical address of the remote controller. (The customer engineer enters the physical address into the IPL configuration data set in the diskette contained within the remote controller.)

```
     {STOP     }
[ANS={CONTINUE}]
     {CONT     }
```

Specifies whether operation of the station represented by this PU macro is to continue or to stop if the NCP you are defining enters automatic network shutdown.

If you specify ANS=STOP (valid only for type 1 and 2 physical units), the program stops all communication with the station upon entering automatic shutdown. The station, detecting that it can no longer communicate with the NCP, goes offline regardless of whether it is currently participating in sessions with nonadjacent subareas. This is the only valid choice for a station on a switched SDLC link.

If you specify ANS=CONTINUE (or ANS=CONT), the NCP, though in shutdown mode with respect to the access method with which it can no longer communicate, continues to service the link on which the station is located as long as the link and station remain operational. The station can then continue to participate in sessions with access methods and application programs other than those affected by the shutdown.

If this PU macro represents a 3705 (that is, a type 4 physical unit), only ANS=CONTINUE (or ANS=CONT) is valud; it also is the value assumed if you omit this operand. For type 1 and 2 physical units, ANS=STOP is assumed if you omit this operand.

[BNNSUP=3270]

(type 1 physical unit and nonswitched link only)

Specifies that the physical unit represented by this PU macro is a 3270 (operating in SDLC mode.)

This operand is invalid if the line is switched (DIAL=YES) or if the 3270 is a type 2 physical unit (PUTYPE=2).

[DATMODE={HALF}]
{FULL}

Specifies, for a communications controller (type 4 physical unit) on a nonswitched SDLC link that allows simultaneous transmission in both directions, whether the network control program is to communicate with the distant controller in half-duplex or duplex data mode. This operand is valid only if the ADDRESS operand of the corresponding LINE macro specifies both a receive and a transmit address and the LINE macro represents the principal (not backup) SDLC link.

If you (1) omit this operand, (2) code the SUBAREA operand, and (3) specify, in the LINE macro, both a receive and a transmit address, then DATMODE=FULL is assumed.

This operand is invalid if:

- the program you are defining is a remote NCP (TYPGEN=NCP-R)
- the program is a local NCP (TYPGEN=NCP or NCP-LR) and the SUBAREA operand is omitted from this macro
- the physical unit type is not 4.

DATMODE=FULL is invalid if DIAL=YES is specified in the GROUP macro.

Minimizing Line-Turnaround Delay:

Each reversal in direction of message transmission over an SDLC link causes a line-turnaround delay of several milliseconds if the carrier signal transmitted by the modem is interrupted. Efficiency of the link is thus lower than if the carrier signal is transmitted continuously, because no message data can be transmitted during these intervals.

Interruption of the carrier signal, and hence line turnaround delays, are inherent and unavoidable if the communication facility (including lines and modems) is half-duplex. They may also occur, however, in a duplex facility; and, in that case, they may be prevented by ensuring that the carrier signal is transmitted continuously. Continuous carrier transmission results from either (1) continuous activation of the request-to-send signal sent from the communications controller to the modem or (2) "strapping" (internally connecting) the modem to transmit a carrier signal continuously independent of whether the request-to-send signal is activated.

The request-to-send signal (and hence the carrier sent by the modem) is activated continuously if:

- the communication facility is duplex
- the line set to which the SDLC link is attached is duplex
- separate receive and transmit addresses are specified in the ADDRESS operand of the LINE macro

- DATMODE=FULL is specified in the LINE macro

The request-to-send signal is not activated continuously if the line set is half-duplex *or* if DATMODE=HALF is specified in the LINE macro. The line-turnaround delay can still be minimized or eliminated, however, if the modem is strapped for continuous carrier operation. Consult the supplier or installer of the modem to determine if it is capable of continuous carrier operation and, if so, have the modem so connected.

[IRETRY={YES}]
{NO }

(nonswitched link only)

Specifies whether the NCP, when an 'idle detect time-out' condition follows a polling operation, is to immediately retry the operation. If you specify IRETRY=YES, the program repolls; otherwise, the program services the station represented by the next entry in the service order table.

This operand is valid only if POLLED=YES is specified in the LINE (or GROUP) macro and DIAL=NO is specified in the GROUP macro. If you omit this operand and the LINE macro specifies POLLED=YES, IRETRY=NO is assumed.

Note: You may specify IRETRY in a PU macro associated with either the principal or the backup SDLC link.

[MAXDATA=size]

(type 1 and 2 physical units only; nonswitched link only)

Specifies the maximum amount of data, in bytes (including the transmission header and request/response header), that the physical unit can receive in one data transfer; that is, one entire PIU or a PIU segment.

To determine the amount of data that the physical unit can receive, consult the publications for the specific type of SDLC station represented by this PU macro.

The maximum amount of user data the NCP sends to the physical unit in one data transfer is the value of MAXDATA minus 5 bytes (for a type 1 physical unit) or minus 9 bytes (for a type 2 physical unit). These values represent the lengths of the request/response header (3 bytes) and the transmission header (2 bytes for a type 1 physical unit; 6 bytes for a type 2 physical unit.

The value you specify for MAXDATA, divided by the NCP buffer size (specified in the BFRS operand of the BUILD macro), determines the number of NCP buffers sent in one data transfer to the physical unit represented by this PU macro. If the result is not an integer, it is rounded down to the next lower integer. For example, if MAXDATA=265 and BFRS=64, then 265 divided by 64 is 4.14, which is rounded down 4.

This operand is valid only for a nonswitched type 1 or type 2 SDLC physical unit. It is invalid if DIAL=YES is specified in the GROUP macro. If you omit this operand, BFRS plus 2 is assumed as the maximum amount of data for PUTYPE=1 and BFRS plus 6 is assumed for PUTYPE=2.

[MAXLU=count]

Spccifies the maximum number of logical units that may be associated with this physical unit. If this operand is omitted, the count defaults to the value of MAXLU in the PUDRPOOL macro or the number of LUs defined for this PU, whichever is larger. The count is increased by 1 if this PU is a type 1.

If PUDR=YES is specified in this PU macro, the value specified in this MAXLU operand must be equal to, or greater than the MAXLU value in the PUDRPOOL macro. The maximum value is 255. If dynamic reconfiguration is supported, additional logical units may be added to this physical unit up to *count* minus 1 for a type 1 PU and up to *count* for a type 2 PU.

[MAXOUT={n}]
{1}

(nonswitched link only)

Specifies the maximum number of path information units (PIUs) (or PIU segments if the program divides PIUs into segments) the NCP will send to the physical unit represented by this PU macro before requesting a response from the physical unit. The minimum is 1; the maximum is 7.

If you omit this operand and specify DIAL=NO, MAXOUT=1 is assumed if you specify POLLED=YES, or MAXOUT=7 is assumed if you specify POLLED=NO.

This operand is invalid for a physical unit on a switched link (DIAL=YES is specified in the GROUP macro).

[PASSLIM={n}]
{1}

(nonswitched link only)

Specifies the maximum number of consecutive path information units (PIUs) or PIU segments the NCP will send at one time to the physical unit represented by this PU macro.

If this physical unit is associated with a LINE macro in which POLLED=YES is specified, the program services the station represented by the next entry in the service order table when the pass limit value is reached.

If this PU macro is coded in a remote NCP and represents a local communication controller, the remote program stops sending to the local controller when the pass limit is reached. Transmission to the local controller resumes when the local network control program again polls the remote controller.

The minimum is 1 PIU or segment. The maximum is 254. The default value assumed if you omit this operand is (PASSLIM=1).

If this PU macro appears in a remote NCP and represents a local communications controller, the maximum value of 254 is the recommended value and the default value for *n*.

This operand is invalid for a physical unit on a switched link (DIAL=YES is specified in the GROUP macro).

```
[PUCB=symbol]
```

(user-written line control only)

Specifies the name of a user-defined control block associated with a virtual physical unit. This operand is valid only when VIRTUAL=YES is specified in the GROUP macro.

Note: The user control block must be assembled with the NCP in stage 2 of the generation. (See the SRCHI operand in the GENEND macro.)

```
[PUDR={NO }]
      {YES}
```

(nonswitched link only)

Specifies whether the physical unit can be deleted from the network by using the dynamic reconfiguration function. This operand is invalid if dynamic reconfiguration is not supported, the PU is a type 4, or the PU is on a switched link.

```
[PUFVT=symbol]
```

(user-written line control only)

Specifies the name of the function vector table associated with a virtual physical unit. This operand is required if VIRTUAL=YES is specified in the GROUP macro. It is invalid if VIRTUAL=NO.

An EXTRN statement for each unique FVT symbol must be included in the user source code that is copied into the tables during assembly. (See the SRCHI operand in the GENEND macro.)

Note: The function vector table must be preassembled and included in the user object modules. (See the INCHI operand in the GENEND macro.)

```
        {1                     }
        {2                     }
[PUTYPE={(1,2)                 }]
        {(4,[,{LOCAL }] )}
              {REMOTE}
```

Specifies the physical unit type of the SDLC station represented by this PU macro. PUTYPE=(1,2) specifies, only for a group of switched SDLC links, that physical units of either type 1 or type 2 can communicate with the communications controller over the same link.

PUTYPE=1 and PUTYPE=2 are valid only for polled links (POLLED=YES).

PUTYPE=1 and PUTYPE=2 are valid for links that use either nonswitched or switched line control procedures (DIAL=NO or DIAL=YES).

PUTYPE=(4,LOCAL) specifies that the program in the controller represented by this PU macro is a local network control program (in which the TYPGEN operand of the BUILD macro specifies NCP or PEP.

PUTYPE=(4,REMOTE) specifies that the program in the controller represented by this PU macro is a remote NCP (in which TYPGEN=NCP-R is specified).

PUTYPE=4 is not valid if VIRTUAL=YES is specified in the GROUP macro for this PU.

If you omit this operand, the default values assumed are as follows:

- If you specify DIAL=YES in the GROUP macro, PUTYPE=(1,2) is assumed.
- If you specify DIAL=NO and POLLED=YES in the LINE (or GROUP) macro, PUTYPE=2 is assumed.
- If you specify DIAL=NO and POLLED=NO, PUTYPE=4 is assumed.

Note: If the program in the controller represented by this macro may be a local NCP at some times and a remote NCP at other times, specify PUTYPE=(4,LOCAL).

If you specify PUTYPE=4 (omitting the LOCAL or REMOTE parameter) and POLLED=NO in this PU macro (or in the LINE or GROUP macro), the program in the distant controller is assumed to be a local NCP [PUTYPE=(4,LOCAL)].

If you specify PUTYPE=4 and POLLED=YES in this PU macro (or the LINE or GROUP macro), the program in the distant controller is assumed to be a remote NCP [PUTYPE=(4,REMOTE)].

If the network control program you are defining is a remote NCP (TYPGEN=NCP-R), PUTYPE=(4,REMOTE) is invalid; and PUTYPE=(4,LOCAL) is invalid if you specify POLLED=YES in the LINE macro associated with this PU macro.

`[RETRIES=( [,t] [,n] )]`

(nonswitched link only)

Specifies, with the RETRIES operand of the LINE macro, the number of attempts to recover from errors occurring during transmission to or from the physical unit represented by this PU macro.

Note: (This note applies only to a local controller.) To avoid losing contact with a remote controller while it accesses its disk (for example, when the remote controller storage is being dumped), specify RETRIES and REPLYTO (GROUP macro) for a minimum retry time of 30 seconds.

The meaning of the *t* and *n* parameters is the same as for those parameters in the RETRIES operand of the LINE macro.

This operand is invalid if DIAL=YES is specified in the GROUP macro.

`[SUBAREA=n]`

(nonswitched link only)

Specifies the subarea address assigned to the network control program in a distant communications controller (type 4 physical unit) represented by this PU macro. (The NCP in each controller in the network must have a unique subarea address.) The value of *n* specified in this SUBAREA operand and the

SUBAREA operand of the BUILD macro for the NCP in the distant controller must be identical.

This operand identifies the principal SDLC link to a distant controller.

This operand is invalid if DIAL=YES is specified in the GROUP macro.

The minimum valid subarea address is 1; that is, SUBAREA=1. The maximum address is the same value you specify in the MAXSUBA operand of the BUILD macro.

If this PU macro and its associated LINE macros represent the principal SDLC link to a physical unit type 4 and the present NCP is not a remote NCP, SUBAREA must be specified. Omit this operand if you are defining a remote NCP.

If you are defining a local NCP, you must specify SUBAREA for the principal SDLC link to a distant controller; and you must omit SUBAREA (and all other PU operands except PUTYPE, IRETRY, and RETRIES) for alternate backup links.

If you are defining a remote NCP, you omit SUBAREA for both the principal and alternate backup links. In the remote controller, the principal link is determined at program execution time and is the link over which commands are first received.

When communicating with the distant controller over a backup link, the present network control program uses the same values as specified in the PU macro for the principal link except for the values of PUTYPE, IRETRY, and RETRIES.) The PUTYPE operand must be coded the same in the PU macros associated with the principal and the backup links.

Specifying Lower-Level Operands in a Higher-Level Macro

In addition to the preceding operands, some operands of the LU macro can be specified in the PU macro instead of the LU macro. Similarly, most operands of the PU macro can be specified in the LINE or GROUP macro. Figure 5-11 shows which of the lower-level operands you may specify at a higher level.

LU Macro Instruction

The LU macro instruction represents a logical unit associated with an SDLC station (type 1 or 2 physical unit) attached to a nonswitched SDLC link; it specifies:

- The resource name for the logical unit
- The local address of the logical unit
- Use of the pacing option
- Whether data transfer from the access method to the logical unit is in batch mode

Each logical unit associated with a type 1 or 2 physical unit on a nonswitched SDLC link must be represented by a separate LU macro instruction. The sequence of LU macros must immediately follow the PU macro representing the physical unit. The sequence must be in ascending order of local addresses assigned to logical units (as specified in the LOCADDR operand of this macro). That is, the LU macro specifying the lowest address must appear first, following the PU macro; and the LU macro specifying the highest address must appear last. (LU macros are not required for any local addresses with which no logical unit are associated; however, the generation procedure generates logical unit control blocks for each local address not defined, up to the highest address for which there is an LU macro).

(VTAM Users Only): Appearing at the end of the following list of operands are the VTAM-only operands that may be coded in this macro instruction. The VTAM operands provide information for the VTAM initialization procedure and are not required (though are permissible) as input for the NCP generation procedure. See the *ACF/VTAM Installation* manual for descriptions of these operands and for information on the VTAM initialization process. (See the *NOSP Installation Manual* for the description of the SPAN operand.)

The format of the LU macro is:

```
symbol          LU              operand[,operands]
```

Operands

```
LOCADDR=n
[,BATCH={YES} ]
        {NO }
[,DATASW=termname]
[,LUCB=(symbol1[,symbol2]...[,symbol9]))]
[,LUDR={NO }]
       {YES}
[,LUFVT=(symbol1[,symbol2]...[,symbol9]))]
[,PACING={(n[,m])} ]
         {(1,1)   }
```

VTAM-only operands:

```
BUFLIM=
DLOGMOD=
FEATUR2=
ISTATUS=
LOGAPPL=
LOGTAB=
MODETAB=
SPAN=
SSCPFM=
TERM=
USSTAB=
VPACING=
```

`symbol`

Provides a resource name for the logical unit and is required. *symbol* may be any valid assembler-language symbol; the first character may not be $. (This name must not be specified in the service order table.)

`LOCADDR=n`

Specifies the local address (in decimal, without leading zeros) of the logical unit. The valid range of addresses for a type 1 physical unit is 0-63; for a type 2 physical unit, 1-255. The addresses must be specified in ascending order. That is, the LU macro specifying the lowest address must appear first (following the PU macro), and the LU macro specifying the highest address must appear last.

```
[BATCH={YES}]
       {NO }
```

(VTAM only)

Specifies the processing priority that the network control program is to use for the logical unit. BATCH=NO indicates a high priority (suitable for interactive applications). BATCH=YES indicates a low priority.

`[DATASW=termname]`

(SDLC/BSC path function; VTAM only)

Specifies the name of the TERMINAL macro representing the BSC station that the NCP is to send data PIUs originated by the logical unit. Specifying this operand invokes the SDLC/BSC path function by which data PIUs (other than control and error messages) originating at the logical unit are sent directly to a BSC station rather than to VTAM. (Control and error message PIUs originating at the logical unit are sent to VTAM.)

Note: If you specify this operand, you must also specify the SPAFPT 3 block-handling macro instruction for the BSC station (or assign to the BSC station a user block-handling routine that performs the equivalent function). See the description of the SPAFPT3 macro. Also see the *SDLC/BSC Path Function System Programmer's Guide,* GC30-3029.

`[LUCB=(symbol1[,symbol2]...[,symbol9]))]`

(user-written line control only)

Specifies the names of user-defined control blocks associated with a virtual logical unit. This operand is valid only when VIRTUAL=YES is specified in the GROUP macro.

The symbols in the LUCB operand are positionally related to the symbols in the LUFVT operand. If a symbol is specified in LUCB, a corresponding symbol *must* be specified in the same symbol position in the LUFVT operand.

Note: The user control blocks must be assembled with the NCP in stage 2 of the generation. (See the SRCHI operand in the GENEND macro.)

```
[LUDR={NO }]
      {YES}
```

Specifies whether the logical unit can be deleted from the network by using the dynamic reconfiguration function. If this operand is omitted, the default assumes the same value as the PUDR operand of the PU macro. LUDR=YES is valid only if dynamic reconfiguration is supported. LUDR=NO is invalid if PUDR=YES.

```
[LUFVT=(symbol1[,symbol2]...[,symbol9])]
```

(user-written line control only)

Specifies the names of the functional vector tables associated with a virtual logical unit. This operand is valid only when VIRTUAL=YES is specified in the GROUP macro.

If virtual resources are included in this NCP, *symbol1* is required. That FVT is for the SSCP-LU session; the remaining FVTs are for LU-LU sessions that may occur for this resource. The symbols in this operand are positionally related to the symbols in the LUCB operand.

An EXTRN statement for each unique FVT symbol must be included in the user source code that is copied into the tables during assembly. (See the SRCHI and SRCLO operands in the GENEND macro.)

Note: Function vector tables must be preassembled and included in the user object modules. (See the INCHI operand in the GENEND macro.)

```
[,PACING={(n[,m])}]
         {(1,1)  }
```

Specifies whether the NCP is to pace data sent to the logical unit; that is, require the logical unit to acknowledge, at intervals, receipt of the message data and its ability to accept more data.

n

Specifies the number of requests the NCP is to send to the logical unit before stopping transmission to await a pacing response from that unit. The minimum is 1; the maximum is 255.

m

Specifies in which of the *n* requests the NCP is to turn on the pacing bit. The minimum is 1 (that is, the first request); the maximum is the value specified for *n*.

If you omit *m*, the NCP turns on the pacing bit in the last (*n*th) request sent.

0

Specifies that the network control program is not to pace data sent to the logical unit.

Note: Pacing applies only to normal flow (synchronous request).

Specifying LU Operands in a Higher-Level Macro

Some operands of the LU macro can be specified in the PU, LINE, or GROUP macro instead of in the LU macro. Figure 5-11 shows which of the operands you may code at a higher level.

CLUSTER Macro Instruction

The CLUSTER macro represents a "clustered" type BSC station (IBM 3270 or 2972).

The CLUSTER macro specifies:

- The resource name for the clustered station if general polling characters are specified
- The type of station
- Whether the station has the batched message input feature (2972 only)
- The general-polling characters of the station, if required
- The block handler set, if any, associated with the station and the points of execution of block handlers within the set
- Certain procedural options the NCP is to use when communicating with the station

Use a CLUSTER macro to represent an IBM 3270 series or 2972 only if the line is a nonswitched multipoint line (DIAL=NO in the GROUP macro and POLLED=YES in the LINE or GROUP macro).

Do not code CLUSTER macros following a LINE macro if:

- The line is a start-stop line (LNCTL=SS in the GROUP macro).
- The line is an SDLC link (LNCTL=SDLC).
- The line operates only in emulation mode.
- The line is a switched line (DIAL=YES in the GROUP macro).
- The line does not use multipoint line control (POLLED=NO in the LINE macro).

Each CLUSTER macro coded causes a resource name to be generated only if the GPOLL operand is coded. (Omit the GPOLL operand if the general polling function is not required.)

The CLUSTER macro applies only to lines operating in network control mode.

(VTAM Users Only): Appearing at the end of the following list of operands are the VTAM-only operands that may be coded in this macro instruction. The VTAM operands provide information only to the VTAM initialization process and are not required (though are permissible) as input to the NCP generation procedure. See the *ACF/VTAM Installation* manual for descriptions of these operands and for information on the VTAM initialization process. (See the *NOSP Installation Manual* for the description of the SPAN operand.)

The format of the CLUSTER macro is:

```
symbol          CLUSTER      [operands]
```

Operands

```
[BHEXEC={( [PT1] [,PT2] [,PT3] )}]
        {ALL                     }
        {NONE                     }
[,BHSET={DYNAMIC                  }]
        {setname[,EXEC={YES}]}
                       {NO }
[,CDATA={YES}]
        {NO }
         {2972}
[,CUTYPE={3271}]
         {3275}
[,FEATURE={BATCH   }]
          {NOBATCH}
[,GPOLL=chars]
[,INHIBIT={( [WACKCNT] [,SUBBLOCK] [,ERPR] [,ERPW] )}]
          {NONE                                      }
[,ITBMODE=( [{YES}] [,{YES}] )]
             {NO }    {NO }
[,LGRAPHS=( [{REJECT}] [,{REJECT}] )]
             {ACCEPT}    {ACCEPT}
[,PT3EXEC={YES}]
          {NO }
[,XMITLIM={count}]
          {NO    }
```

VTAM-only operands:

```
DLOGMOD=
FEATUR2=
ISTATUS=
LOGAPPL=
LOGTAB=
PU=
SPAN=
VPACING=
```

symbol

Specifies the name of the station and is required. *symbol* may be any valid assembler-language symbol. The first character may not be $. This name is a resource name for the cluster only if the GPOLL operand is coded.

```
[BHEXEC={( [PT1] [,PT2] [,PT3] )}]
        {ALL                     }
```

Specifies which block-handling routines from the block handler set defined by the BHSET operand are to be executed. If you specify BHSET=*setname*, you must specify at least one execution point. Do not code this operand if you specify BHSET=NONE or BHSET=DYNAMIC.

[PT1]

Specifies that the network control program is to execute the PT1 block handler. The NCP will execute this block handler upon receiving a request from the host processor but before determining whether the line is available to contact the station.

[PT2]

Specifies that the network control program is to execute the PT2 block handler. The NCP will execute this block handler upon receiving a request from the host processor *and* after determining that the line is available.

[PT3]

Specifies that the network control program is to execute the PT3 block handler. The NCP will execute this block handler after receiving a block, message, or transmission from the station.

ALL

Specifies that each block handler in the set is to be executed at the appropriate time.

If this operand is omitted and BHSET is specified, ALL is assumed.

```
        {NONE                  }
[BHSET={DYNAMIC                }]
        {setname[,EXEC={YES}]}
                       {NO }
```

Specifies the name of a set of block handlers associated with this station.

NONE

Specifies that no block handler set is assigned to this station.

DYNAMIC

Specifies that no block handler set is initially assigned to this station, but one may be assigned dynamically from the host processor.

setname

Specifies the block handler set for this station. This parameter must be the *setname* appearing on a BHSET macro.

```
[EXEC={YES}]
      {NO }
```

Specifies whether the block handler set can be executed without being activated by the host. If EXEC=YES, the block handler is activated during initialization of the NCP. If EXEC=NO, it must be activated by a request from the host.

```
[CDATA=(YES)]
       (NO )
```

Specifies whether the data sent to and received from the station represented by this CLUSTER macro is "critical" (for security reasons). If you code CDATA=YES, the network control program automatically clears all buffers containing data associated with the station before returning them to the buffer pool. If you code CDATA=NO, the program does not clear the buffers. CDATA=YES is valid only if you specify ERASE=YES in the BUILD macro.

```
         {2972}
[CUTYPE={3271}]
         {3275}
```

Specifies whether the station's control unit is a 2972, 3271, or 3275. Code CUTYPE=3271 for a 3274 or 3276 control unit.)

Note: If the line has 2972 or 3270 stations connected and is to operate in emulation mode as well as in network control mode, specify the CUTYPE operand in both the LINE macro and the CLUSTER macro.

```
[FEATURE={BATCH  }]
         {NOBATCH}
```

(IBM 2972 only)

Specifies whether an IBM 2972 has the batched message input feature. If you specify FEATURE=BATCH, the network control program does not deblock messages from the 2972 and therefore does not identify the individual 2980 terminals that each message was received from.

```
[GPOLL=chars]
```

Specifies that the general-polling procedure is used for the station and specifies the general-polling characters assigned to the control unit of the station. If you omit this operand, terminals must be individually polled. (The GPOLL operand must be coded if the control unit is a 2972 (CUTYPE=2972), since individual polling of terminals attached to a 2972 is not possible.)

This operand is required if this CLUSTER macro represents an IBM 3271.

Specifying Polling and Addressing Characters:

IBM 2972: Code the general-polling characters in the GPOLL operand of this CLUSTER macro. In addition, code a TERMINAL macro following this macro for each terminal address on the control unit. Code all of the addresses from the lowest address to the highest address regardless of whether some intermediate addresses are unused. For example, if terminals were attached to the third, fourth, eighth, and ninth addresses, you would code nine TERMINAL macros, the first representing the first (lowest) address and the last representing the ninth address.

Every attached terminal must be represented by a TERMINAL macro. Entering data from any terminal not so represented will cause the general-poll operation to be aborted for the control unit.

In the ADDR operand of each TERMINAL macro representing a 2980, code the addressing character assigned to that 2980. Since 2980s cannot be individually polled, the POLL operand is invalid.

IBM 3270 series except 3275: If general polling is required, code the general-polling character in the GPOLL operand of the CLUSTER macro. In addition, code a TERMINAL macro following this macro for each terminal address on the control unit. Code all of the addresses from the lowest address to the highest address regardless of whether some intermediate addresses are unused. For example, if terminals were attached to the third, fourth, eighth, and ninth addresses, you would code nine TERMINAL macros, the first representing the first (lowest) address and the last representing the ninth address.

Every attached terminal must be represented by a TERMINAL macro. Entering data from any terminal not so represented will cause the general-poll operation to be aborted for the control unit.

In the POLL and ADDR operands of each TERMINAL macro, code the polling and addressing characters assigned to the device.

If a 3277 is used as an input-only device and you specify general polling characters in the CLUSTER macro, you may omit the ADDR and POLL operands from the TERMINAL macro for that 3277.

IBM 3275: If general polling is required, code the general-polling characters in the GPOLL operand of this macro. In addition, directly follwing this CLUSTER macro, code a single TERMINAL macro with TERM=3275 specified. If the 3275 is to be individually polled and addressed, specify the polling and addressing characters in the POLL and ADDR operands of the TERMINAL macro. If you specify addressing characters in the ADDR operand, you must also code polling characters in the POLL operand. (The reverse is not true.)

```
[INHIBIT={( [WACKCNT] [,SUBBLOCK] [,ERPR] [,ERPW] )}]
         {NONE                                   }
```

Specifies which, if any, of the network control program facilities are to be inhibited from functioning initially; that is, when the program begins execution after being loaded into the controller.

[WACKCNT]

Specifies that the WACK limit specified by the WACKCNT operand of the GROUP macro is to be inhibited. The first WACK received from the station causes the NCP to return the request to the host processor rather than to respond to that and subsequent WACKs with an ENQ character.

[SUBBLOCK]

Specifies that the subblocking indicated by the TRANSFR operand of the LINE (or GROUP) macro is to be inhibited. If the number of buffers specified by the TRANSFR operand is filled by received text, the NCP terminates the receiving operation just as if the cutoff limit (specified by the CUTOFF operand) were reached.

[ERPR]

Specifies that recovery procedures for text-read errors are to be inhibited.

[ERPW]

Specifies that recovery procedures for text-write errors are to be inhibited.

NONE

Specifies that none of the network control program facilities are to be inhibited.

```
[ITBMODE=( [{YES}] [,{YES}] )]
            {NO }    {NO }
```

Specifies how the network control program should handle ITB characters in text received from the station.

The first suboperand specifies whether the program is to insert an EIB (error information block) character following each ITB character received from the station.

The second suboperand specifies whether each ITB character received from the access method is followed by an EIB character. If you specify YES, the NCP removes the first character following each ITB character from the message data before sending it to the station.

```
[LGRAPHS=( [{REJECT}] [{REJECT}] )]
            {ACCEPT}   {ACCEPT}
```

Specifies whether leading graphics received from the station are accepted or rejected by the NCP.

The first parameter specifies acceptance or rejection of leading graphics for read operations. The second parameter specifies acceptance or rejection for write operations.

```
[PT3EXEC={YES}]
         {NO }
```

Specifies whether a block handler set that is executed at point 3 (see the BHSET macro) is to be associated with the station represented by this macro.

This operand is ignored if you omit the BHSET macro, or if you code BHSET=NONE, BHEXEC=PT3, or BHEXEC=ALL in this CLUSTER macro.

```
[XMITLIM={count}]
         {NO   }
```

Specifies the maximum number of transmissions the NCP will receive from or send to this station. If this limit is reached before the host processor explicitly requests that the NCP disconnect the station from the controller, the program will automatically suspend the session.

The maximum value you may specify is 255; the minimum is 1.

XMITLIM=NO means that the NCP will send to or receive from the station indefinitely.

XMITLIM=*count* is valid only for stations on a line with a service order table. A SERVICE macro must follow the LINE macro.

In most applications, the default value, XMITLIM=NO, is *not* appropriate because it allows the station represented by this CLUSTER macro to monopolize the communication line indefinitely. You should normally specify a count in this operand. The lower the count you specify for each station attached to a line, the greater the degree of line sharing. It is recommended that XMITLIM=1 be specified for BSC 3270 clusters.

Note: On nonswitched lines this operand is host dependent.

Specifying Lower-Level Operands in a Higher-Level Macro

In addition to the preceding operands, most operands of the TERMINAL macro can be specified in the CLUSTER macro (for cluster-type stations) or in the LINE or GROUP macro, instead of the TERMINAL macro. Figure 5-11 shows which of the TERMINAL macro operands you may specify at a higher level.

TERMINAL Macro Instruction

The TERMINAL macro represents a start-stop or BSC station and specifies:

- The resource name for the station
- The type of station
- The features with which the station is equipped
- The polling and addressing characters by which the NCP will contact the station (multipoint line control)
- The telephone number by which the NCP can reach the station (switched line)
- The length of the controller ID sequence the NCP will send to the station when contact is established
- The block handler set, if any, associated with the station, and the points of execution of block handlers within the set
- Certain procedural options the NCP is to use when communicating with the station

Each start-stop and BSC station attached to a nonswitched point-to-point or multipoint line must be represented in the NCP by a separate TERMINAL macro. All terminals attached to an IBM 2972 or 3270 control unit must be represented by TERMINAL macros.

Note: TERMINAL macros are also required for unused terminal addresses on certain BSC control units. See the CUTYPE operand of the CLUSTER macro.

Stations that call the communications controller over the switched telephone network are not individually represented by TERMINAL macros. Instead, a TERMINAL macro is required for each of the controller's connections (ports) that are used for receiving calls. The control blocks generated by each TERMINAL macro are used to represent whichever station is connected to the controller for the duration of any given call.

Stations that will be called by the controller may be represented either by individual TERMINAL macros or by a common TERMINAL macro that represents whichever station the controller is connected to during any given call.

Code an individual TERMINAL macro for each station whose telephone number that is to be maintained within the controller. Specify the telephone number of the station in the DIALNO operand.

Code a single TERMINAL macro to represent all stations for which VTAM will supply the telephone number.

Each TERMINAL macro coded causes a resource name to be generated and applies only to lines operating in network control mode.

If the station specified in this TERMINAL macro is an IBM 3275 or a device attached to an IBM 2972 or 3270 series (except 3275) control unit, the control unit must be represented by a CLUSTER macro. (See the CUTYPE operand of the CLUSTER macro for requirements regarding use of TERMINAL macros.) Do not code a TERMINAL macro for a printer attached to an IBM 3275 (TERM=3275).

(VTAM Users Only): Appearing at the end of the following list of operands are the VTAM-only operands that may be coded in this macro instruction. The VTAM operands provide information only to the VTAM initialization process and are not required (though are permissible) as input to the NCP generation procedure. See the *ACF/VTAM Installation* manual for descriptions of these operands and for information on the VTAM initialization process. (See the *NOSP Installation Manual* for the description of the SPAN operand.)

The format of the TERMINAL macro is:

```
symbol          TERMINAL      operand[,operands]
```

Operands

```
TERM=type
        {addr chars                 }
[,ADDR={selection chars             }]
        {([normal addr chars]
             [,alt addr chars])}
[,ATTN={ENABLED }]
        {DISABLED}
[,BFRDLAY=count]
[,BHEXEC={([PT1][,PT2][,PT3])}]
          {ALL                 }
         {NONE                  }
[,BHSET={DYNAMIC                }]
         {setname [,EXEC={YES}]}
                         {NO }
[,CDATA={YES}]
         {NO }
[,CONV={YES}]
        {NO }
[,CRDLAY={YES}]
          {NO }
[,CRITSIT={YES}]
           {NO }
[,CTERM={YES}]
         {NO }
           {n  }
[,CUIDLEN={TWX}]
           {ALL}
           {0  }
[,DIALNO=([chars][,count])]
[,DIALSET={dialset name}]
           {NONE        }
           {IN   }
[,DIRECTN={OUT  }]
           {INOUT}
[,ENDTRNS={EOT}]
           {EOB}
```

Operands

```
[,FANOUT=terminal name]
[,FEATURE=([{ACR   }][,{ATTN   }]
            {NOACR}     {NOATTN}
           [,{BREAK  }][,{CHECK  }]
             {NOBREAK}   {NOCHECK}
           [,{SCTL  }][,{TOSUPPR}]
             {NOSCTL}   {NOTOSUP}
           [,{XCTL  }])]
             {NOXCTL}
         {symbol}
[,IDSEQ={NONE  }]
         {PASS  }
         {IGNORE}
[,INHIBIT={([TEXTTO][,TIMEFILL][,WACKCNT]}]
                [,SUBBLOCK][,ERPR][,ERPW])}
          {NONE                           }
[,ITBMODE=([{YES}][,{YES}])]
            {NO }   {NO }
[,LCST={mtalcst name}]
       {NONE        }
[,LGRAPHS=([{REJECT}][,{REJECT})]
            {ACCEPT}    {ACCEPT}
[,POLL=chars]
[,PT3EXEC={YES}]
          {NO }
[,XMITLIM={count}]
          {NO   }
```

VTAM-only operands:

```
BUFLIM=
DEVICE=
DLOGMOD=
FEATUR2=
ISTATUS=
LOGAPPL=
LOGTAB=
PU=
SPAN=
UTERM=
```

`symbol`

Provides a resource name for the station and is required. *symbol* may be any valid assembler-language symbol. The first character may not be $.

`TERM=type`

Specifies the type of station represented by this TERMINAL macro. It must be one of the types listed in Figure 5-16.

If the line the station is attached to will operate in emulation mode as well as in network control mode, specify the TERM operand in the LINE (or GROUP) macro instead of in the TERMINAL macro. (Operands specified in the LINE macro apply to line operation in network control mode and emulation mode; operands specified in the TERMINAL macro apply only to line operation in network control mode.)

If this TERMINAL macro defines a call-in logical-connection station (CTERM=YES) representing more than one type of BSC station, specify any of the types in this operand. For example, if this logical-connection station represents IBM 1130, 2780, and System/370 Model 135 stations, you may specify TERM=1130, TERM=2780, or TERM=3135.

This operand is required. It may be coded in this TERMINAL macro or in a higher-level (LINE or GROUP) macro.

If Type of Station is:	*Code TERM=*
IBM 1050 Data Communication System	1050
IBM 1130 Computing System	1130
IBM 1800 Data Acquisition and Control System	1800
IBM System/360 Model 20	2020
IBM System/360 Model 25	2025
IBM 2701 Data Adapter Unit	2701
IBM 2703 Transmission Control	2703
IBM 2715 Model 1 Communications Terminal	2740-1
IBM 2740 Transmission Control Unit Model 2	2715
IBM 2740 Model 2 Communications Terminal	2740-2
IBM 2741 Communications Terminal	2741
IBM 2770 Data Communications System	2770
IBM 2780 Data Transmission Terminal	2780
IBM 2972 General Banking Terminal System:	
IBM 2980 Models 1 and 4 Teller Station	2980
IBM 2980 Model 2 Administrative Station	2980
IBM 3270 Information Display System:	
IBM 3275 Display Station	3275
IBM 3276 Control Unit Display Station	3277
IBM 3277 Display Station	3277
IBM 3278 Display Station	3277
IBM 3284 Printer	3284
IBM 3286 Printer	3286
IBM 3287 Printer	3284
IBM 3288 Line Printer	3286
IBM 3289 Line Printer	3286
IBM 3650 Retail Store System (in BSC mode)	SYS3
IBM 3660 Supermarket System (in BSC mode)	SYS3
IBM 3704 Communications Controller	3704
IBM 3705 Communications Controller	3705
IBM 3735 Programmable Buffered Terminal	3735
IBM 3740 Data Entry System:	
IBM 3741 Data Station	3741
IBM 3747 Data Converter	3747
IBM 3767 Communications Terminal (in start-stop mode):	
supported as 2740 Model 1	2740-1
supported as 2740 Model 2	2740-2
supported as 2741	2741
IBM 3770 Data Communications System (in BSC mode)	2770
IBM 3780 Data Communications Terminal[1]	3780
IBM System/370 Model 125	3125
IBM System/370 Model 135	3135
IBM System/3	SYS3
IBM System/7[2] (BSC version)	(see note 2)
IBM System/7 (start-stop version)	2740-1
IBM System/32 (BSC version)	SYS3
IBM Communicating Magnetic Card Selectric Typewriter	2741
AT&T 83B3 Selective Calling Station	83B3
Western Union Plan 115A Outstations	115A
Western Union Teletypewriter Exchange Service	TWX
World Trade Teletypewriter Terminals	WTTY
Multiple Terminal Access (IBM 1050,2740,2741;TWX)[1]	MTA

[1] Specify this type only if the line is to operate only in network control mode.
[2] Specify a System/7 (BSC version) as follows: (1) Specify TERM=SYS3 in the TERMINAL (or LINE or GROUP) macro if the line is to operate only in network control mode. (2) Specify TERM=SYS3 in the LINE (or GROUP) macro if the line is to operate in both network control and emulation modes (alternately). (3) Specify TERM=SYS7 in the LINE (or GROUP) macro if the line is to operate only in emulation mode.

Figure 5-18. Values for TERM Operand of TERMINAL Macro Instruction

```
      {addr chars                           }
{ADDR={selection chars                      }]
      {([normal addr chars][,alt addr chars])}
```

Specifies the hexadecimal addressing or selection characters assigned to the station represented by this TERMINAL macro.

`addr chars`

Specifies the addressing characters for a start-stop or BSC station on a multipoint line or an IBM 1050 on a switched point-to-point line (POLLED=YES in the LINE or GROUP macro).

This operand is invalid (1) if this TERMINAL macro represents a call-in multiple-terminal-access, logical-connection station (TERM=MTA and CTERM=YES), or (2) if POLLED=NO is specified in the LINE macro and LNCTL=SS is specified in the GROUP macro for the line this station is attached to.

If this macro represents a *call-out* MTA terminal (TERM=MTA, CTERM=NO) and if any of the MTALCST macros named in the LCST operand represent IBM 1050 terminals, specify the 1050 polling and/or addressing characters in the POLL and ADDR operands.

Code only the alphanumeric addressing characters, omitting any control characters. For example, for an IBM 2740 with station control whose address is A, you would code the hexadecimal representation of the EBCDIC character *A* (ADDR=C1). Do not include the start-of-address character that is transmitted before the addressing character or the space character transmitted following it (S A SP).

Note: The character specified for a tributary controller on a multipoint line must have the same bit pattern as the corresponding polling character specified in the POLL operand, except that the third bit from the left (bit 2) must be 1. (Bit 2 in the polling character is always 0.)

`selection chars`

Specifies (if component selection is required) the component selection characters for one of the output components attached to the station on a BSC point-to-point line. For an IBM 2780, specify the alphanumeric component selection character but not the escape (ESC) character that precedes it. For an IBM 2770, specify the device control character—DC1, DC2, DC3, or —that serves as the component selection character. For all BSC stations, specify all characters except ENQ, ESC, and ETB.

`([normal addr chars][,alt addr chars])`

Specifies the normal and/or alternate addressing characters to be used for a 2980 terminal.

The addressing character (escape sequence) assigned to the station should be determined from the system designer.

```
[ATTN={ENABLED }]
      {DISABLED}
```

(IBM 1050, 2741; AT&T 83B3; WU 115A, TWX; World Trade teletypewriters terminals)

Specifies whether the NCP is to assume that the attention feature of the terminal is enabled or disabled. When the feature is enabled, an attention signal received from the terminal causes the NCP to stop sending to the terminal and to notify the host processor. If the feature is disabled, the program ignores it.

ATTN=ENABLED is valid only if FEATURE=ATTN is coded in this TERMINAL macro.

```
[BFRDLAY=count]
```

(buffered terminals only)

Specifies the delay, in seconds, between successive transmissions to the device represented by this TERMINAL macro. The maximum value for *count* is 255; the minimum is 0 (no delay). This operand is valid only for the following buffered devices:

2740-2 (Model 2 with the buffered receive feature)
2770
2980
3275
3277
3284
3286
3780

If TERM=3275 and a printer is attached or if TERM=3284 or TERM=3286, the buffer delay should be 13 seconds or more. The 13 seconds are required to print a message that fills half the device's print buffer. Average messages that are shorter or longer require either a shorter or longer delay. BFRDLAY=0 is invalid for TERM=3284 or TERM=3286.

The terminal types listed will return a busy response if addressed while printing. There is no loss of data, but additional control program overhead is required to handle the busy response and then resend the message.

If you omit this operand, the NCP assumes a 13-second delay for TERM=3284 or TERM=3286 and no delay for any other device.

This operand is valid only for a nonswitched multipoint line.

```
[BHEXEC={([PT1][,PT2][,PT3])}]
        {ALL                }
```

Specifies which block-handling routines from the block handler set defined by the BHSET operand are to be executed. Code this operand only if you specify *setname* in the BHSET operand of this macro. BHEXEC will be ignored if you have coded BHSET=NONE or BHSET=DYNAMIC or if the BHSET operand is omitted.

[PT1]

Specifies that the network control program is to execute the PT1 block handler. The NCP will execute this block handler upon receiving a contact or write request from the host processor but before it determines whether the line is available to contact the station.

[PT2]

Specifies that the network control program is to execute the PT2 block handler. The NCP will execute this block handler upon receiving a contact or write request from the host processor or after message data has been received from the line. The NCP will execute the block handler while the line is available for sending to or receiving from the station.

[PT3]

Specifies that the network control program is to execute the PT3 block handler. The NCP will execute this block handler after receiving a block, message, or transmission from the station.

ALL

Specifies that each block handler in the set is to be executed at the appropriate time.

If this operand is omitted and BHSET=setname is specified, ALL is assumed.

At least one execution point must be specified if you code BHSET=*setname*.

```
        (NONE                        )
[BHSET=(DYNAMIC                      )]
        (setname[,EXEC=(YES)]        )
                        (NO )
```

Specifies the name of a set of block handlers associated with this station.

NONE

Specifies that no block handler set is assigned to this station.

DYNAMIC

Specifies that no block handler set is initially assigned to this station, but one may be assigned dynamically from the host processor.

setname

Specifies a block handler set to be assigned to this station. *setname* must be the name of a BHSET macro.

```
[EXEC=(YES)]
      (NO )
```

Specifies whether the block handler set can be executed without being activated by the host. If EXEC=YES, the block handler is activated during initialization of the NCP. If EXEC=NO, it must be activated by a request from the host.

[CDATA={YES}]
{NO }

Specifies whether the data sent to and received from the station represented by this TERMINAL macro is "critical" (for security reasons). If you code CDATA=YES, the network control program automatically clears all buffers containing data associated with the station before returning them to the buffer pool. If you code CDATA=NO, the program does not clear the buffers. CDATA=YES is valid only if you specify ERASE=YES in the BUILD macro.

[CONV={YES}]
{NO }

Specifies whether the NCP, upon receiving a message block from a station, sends the station a message block (instead of a positive acknowledgment) in response. Sending a message block in response (conversational response) is possible only if the program currently holds a request to send to the station. If it has no request, the program sends the usual positive acknowledgment.

Stations capable of accepting conversational responses are:

IBM 1050
IBM 2740 with record checking feature
IBM 2770 with conversational mode feature
All BSC stations except IBM 2715 and 2780

[CRDLAY={YES}]
{NO }

Specifies whether the NCP should delay executing a write operation that follows a read operation for the terminal. The delay allows the carriage of the terminal printer time enough to return to the left margin.

CRDLAY=YES is valid only if this macro represents an IBM 1050, 2740 Model 1, 2741, or TWX terminal or a multiple-terminal-access (MTA) terminal.

[CRITSIT={YES}]
{NO }

Specifies whether this station, if logically active, is to be notified when the NCP is about to close down the network because of a critical situation.

Note: A critical situation message cannot be sent from a tributary controller to its control station. Therefore, if the NCP is to be executed in a tributary controller and this macro represents the control station where the tributary controller is connected, CRITSIT=YES is invalid. (If the TADDR operand is coded in the LINE macro preceding this TERMINAL macro, do not code CRITSIT=YES.)

If you code CRITSIT=YES, also specify the text of the message in the CSMSG operand of the BUILD macro. If the CSMSG operand is omitted, all CRITSIT operands are ignored.

The NCP sends the notification message to the output component represented by the TERMINAL macro, not to any output component represented by a COMP macro.

If this TERMINAL macro represents a station on a multipoint line or a 1050 on a switched point-to-point line, you must specify the addressing characters of the station in the ADDR operand in order for the critical situation message to be sent.

[CTERM={YES}]
{NO }

(switched lines only)

Specifies that this TERMINAL macro represents stations that call the controller over the switched line associated with this macro. Control fields are generated by the TERMINAL macro to hold information about the stations. The NCP uses these control fields sucessively for various stations that call over the line represented by the LINE macro preceding this TERMINAL macro.

Only one TERMINAL macro with CTERM=YES is required following each LINE macro in which CALL=IN or CALL=INOUT is coded for a switched line. The CTERM operand is not required and should be omitted for a LINE macro in which CALL=OUT is coded.

TERMINAL macros that represent stations called over switched call-out lines have no required positional relationship to the LINE macros for such lines. The association of stations and call-out lines in this case is made via DIALSET operands that specify the names of logical groups. The DIALSET macro defines dial sets. The TERMINAL macros may appear following any LINE macro representing a switched line whose line control matches that used by the terminals to be called. For example: the following statements specify two switched lines to be used for call-out only, one line to be used for call-in and call-out use, and four stations. All three lines are included in one dial set. Five TERMINAL macros are required. One, in which CTERM=YES is coded, represents the logical-connection station. The remaining macros represent each of the four stations to be called over lines in the dial set.

```
DSET     DIALSET      LINES=(A,B,C)
G        GROUP        DIAL=YES,...
A        LINE         CALL=OUT,DIALSET=DSET,...
T1       TERMINAL     DIALSET=DSET,DIALNO=(5142),...
T2       TERMINAL     DIALSET=DSET,DIALNO=(7615),...
T3       TERMINAL     DIALSET=DSET,DIALNO=(8204),...
T4       TERMINAL     DIALSET=DSET,DIALNO=(7382),...
B        LINE         CALL=OUT,DIALSET=DSET,...
C        LINE         CALL=INOUT,DIALSET=DSET,...
LOGCON   TERMINAL     CTERM=YES,...
```

(In this example, one or more of the TERMINAL macros could follow LINE macro B or LINE macro C. The resulting configuration would be identical.)

Note: If you specify CTERM=YES, no operand in any macro can refer to the name of this TERMINAL macro. For example, this TERMINAL macro cannot be named in the IDSEQ operand of the IDLIST macro.

```
          {n  }
[CUILDEN={TWX}]
          {ALL}
          {0  }
```

(switched BSC or TWX station only)

Specifies the length of the ID sequence the NCP sends to the station represented by this macro.

n

Specifies the length of the ID sequence the program will send as EBCDIC characters.

If this TERMINAL macro represents a BSC station, *n* must be less than or equal to the length of the EBCDIC characters specified on the *chars* parameter of the CUID operand of the BUILD macro.

If this TERMINAL macro represents a switched TWX station, *n* must equal the length of the *inchars* parameters specified by the TWXID operand of the BUILD macro.

The maximum value for *n* is 20.

TWX

Specifies that the program is to send the TWX ID to a TWX terminal represented by this macro. If CTERM=NO is specified, the entire sequence specified by the *outchars* parameter of the TWXID operand is sent; if CTERM=YES is specified, the entire sequence specified by the *inchars* parameter of the TWXID operand is sent.

ALL

Specifies that the program is to send all of the appropriate ID sequence (TWXID for TWX terminals, CUID for BSC stations) to the station represented by this TERMINAL macro.

0

Specifies that no ID sequence is to be sent to the station represented by this TERMINAL macro.

Note: This operand is valid only if the GROUP macro specifies DIAL=YES and the LINE macro specifies POLLED=NO; and either (1) the GROUP macro specifies LNCTL=SS and this TERMINAL (or higher-level) macro specifies TERM=TWX or (2) the GROUP macro specifies LNCTL=BSC.

[DIALNO=(chars[,count])]

Specifies the numerals and the length of a telephone number the NCP uses to call the station represented by this TERMINAL macro.

[chars]

Specifies the telephone number of the station. *chars* represents dial digits (EBCDIC numeric characters) the NCP will use to contact the station over a switched line. You may specify this parameter either if the station is reached by a switched connection or if the switched connection is an alternate to the primary nonswitched point-to-point line.

[count]

Specifies the number of bytes to be reserved to hold the telephone number. The minimum value is 1; the maximum, 32.

Use these two suboperands as follows:

- If the station will always be called using the dial digits specified in this operand, code *chars* and omit *count*. For example, DIALNO=19195678888.
- Code *chars* if the host processor will update the dial digits using the dynamic control facility. Also code *count* if the number of replacement digits may exceed the number of digits originally specified. For example, DIALNO=(19195678888,12). Otherwise, you may omit the *count* suboperand.

This operand is invalid if CTERM=YES or DIALSET=NONE.

Separator and End-of-Number Characters: The sequence of dial digits may include separator and/or end-of-number characters if the auto-call unit at the communications controller is designed to accept these characters. A separator character causes the auto-call unit to delay sending the next dial digit on the communication line until the auto-call unit receives a secondary dial tone. An end-of-number character sent to the auto-call unit after the last dial digit signals the auto-call unit that it has received the last digit of the telephone number and causes it to start monitoring the line for an answer tone from the distant station. Use of the end-of-number character can reduce the time needed to establish a connection, thus increasing line utilization.

The separator character can be coded in the dial-digit sequence wherever a pause for a secondary dial tone is needed. The end-of-number character must be coded at the end of the sequence.

The communications controller sends to the auto-call unit only the 4 low-order bits of the digits specified in the sequence. The end-of-number bit pattern is 1100 (hexadecimal C); the separator bit pattern is 1101 (hexadecimal D). Any EBCDIC characters whose 4 low-order bits equal these patterns may be used; for example, *(hex 5C) for end-of-number and ' (hex 7D) for a separator would be coded DIALNO=(8'5799*).

Programmed dialing pause: If the auto-call unit is not equipped to use separator characters, you may introduce a programmed dialing pause into the dialing action to allow time to receive a secondary dial tone. At the point in the sequence at which the pause is required, code one vertical bar character (hex FA) for each second of the pause. For instance, code 3 such characters if a 3-second pause is required. For example, DIALNO=(8------5799).

```
[DIALSET={dialset name}]
         {NONE        }
```

(call-out switched line only)

Specifies the name of the DIALSET macro that specifies the primary dial set associated with the station. This operand is required if the NCP will call the station; it is not required if the program will not call the station.

Each time the NCP receives a call-out request for the station, it attempts to call that station over one of the switched lines in the specified dial set.

Rules for use of this operand are as follows:

- If you want the NCP to call the station via the dial set containing the line whose LINE macro precedes this TERMINAL macro, omit the DIALSET operand in this macro. The DIALSET operand of the LINE macro specifies the dial set used.
- If you want the NCP to call the station via a dial set different from the dial set containing the line whose LINE macro precedes this TERMINAL macro, specify the desired dial set in the DIALSET operand of this TERMINAL macro.
- If no dial set needs to be associated with this station, code DIALSET=NONE. This is appropriate when the station can call the communications controller but the NCP cannot call the station.

Note: The lines in the dial set specified by this operand must have the same operating characteristics as those specified by the LINE macro that precedes this TERMINAL macro.

```
         {IN   }
[DIRECTN={OUT  }]
         {INOUT}
```

Specifies, for the station represented by this TERMINAL macro, whether the NCP will only receive text from the station (DIRECTN=IN), only send text (DIRECTN=OUT), or both send and receive (DIRECTN=INOUT [or DIRECTN operand omitted]).

If you omit this operand and you code POLLED=YES in the LINE macro, IN is assumed if you specify polling characters (POLL operand) in this macro; OUT is assumed if you specify addressing characters (ADDR operand) in this macro; and INOUT is assumed if you specify both polling and addressing characters.

If you omit this operand and you code POLLED=NO in (or omit the POLLED operand from) the LINE (or GROUP) macro, DIRECTN=INOUT is assumed.

Note: For IBM 3284 and 3286 terminals, DIRECTN=OUT is assumed because these devices transmit only status information (not message data) when polled.

```
[ENDTRNS={EOT}]
         {EOB}
```

Specifies whether the NCP is to consider a transmission terminated by an end-of-transmission character (EOT) or by an end-of-block character (EOB).

Note: This operand is valid only for start-stop terminals that can transmit an EOB EOT ending sequence; it is, however, invalid for an IBM 2740 Model 2.

[FANOUT=terminal name] *(BSC terminals only)*

Specifies that this TERMINAL macro is the first of a sequence of TERMINAL macros representing stations attached to the same modem (called a *fanout* modem because several stations can be attached to it) and names the last such TERMINAL macro. (The modem referred to is the modem at the distant end of the communication line, not the modem attached to the communications controller.)

For example, assume that four terminals represented by four TERMINAL macros named T1, T2, T3, T4, are attached to a fanout modem. In the FANOUT operand of the first TERMINAL macro, you would specify the name of the last TERMINAL macro representing a terminal attached to the same modem, thus:

```
T1   TERMINAL  FANOUT=(T4),...
T2   TERMINAL  ...
T3   TERMINAL  ...
T4   TERMINAL  ...
```

The FANOUT operand appears only in the first TERMINAL macro, as shown. If only one terminal is attached to the fanout modem, omit the FANOUT operand.

All TERMINAL macros representing stations attached to the same modem must appear in a single sequence, with no other intervening TERMINAL macros for other stations.

This operand is valid only for stations attached to a BSC line (LNCTL=BSC is specified in the GROUP macro).

[FEATURE=...]

Specifies the machine features with which certain types of terminals may be equipped.

[{ACF }] *(IBM 1050 only)*
{NOACR}

Specifies whether the station is equipped with the accelerated carrier return feature. (The network control program makes use of the accelerated carrier return feature only if you specify FEATURE=ACR in *all* TERMINAL macros following the LINE macro. If you omit FEATURE=ACR in any TERMINAL macro, the network control program ignores the presence of the feature when sending to any station on the line.)

[{ATTN }] *(IBM 1050, 2741, 3767 (in 2741 mode); AT&T 83B3; WU 115A,*
{NOATTN} *TWX; World Trade teletypewriters; and MTA lines only)*

Specifies whether the terminal is able to send attention signals to the controller. For IBM 1050 or 2741 terminals, specify FEATURE=ATTN only if the terminal is equipped with the receive interrupt (1050) or interrupt (2741) feature.

[{BREAK }] (*IBM 1050, 2741, 3767 (in 2741 mode); AT&T 83B3; WU 115A,*
{NOBREAK} *TWX; World Trade teletypewriters; and MTA lines only)*

Specifies whether the station is capable of interrupting its transmission to the controller upon receiving a break signal from the network control program. For IBM 1050 or 2741 terminals, specify FEATURE=BREAK only if the terminal is equipped with the transmit interrupt feature. FEATURE=BREAK is valid only if DUPLEX=FULL or SCLSET=YES is specified in the LINE macro for the line over which this station communicates with the controller. *Exception:* For a 3767 specified as a 2741 (in TERM operand of the TERMINAL macro), FEATURE=BREAK is also valid if DUPLEX=HALF is specified.

[{CHECK }] *(IBM 2740 only)*
{NOCHECK}

Specifies whether the station is equipped with the checking feature.

[{SCTL }] *(IBM 2740 Model 1 only)*
{NOSCTL}

Specifies whether the station is equipped with the station control feature. (This feature is optional for the IBM 2740 Model 1 but standard for the Model 2.)

[{TOSUPPR}]
{NOTOSUP}

Specifies whether the 1050 terminal represented by this TERMINAL macro is equipped with the time-out suppression feature.

[{XCTL }] *(IBM 2740 Model 1 only)*
{NOXCTL}

Specifies whether the terminal is equipped with the transmit control feature.

```
        {symbol}
[IDSEQ={NONE  }]
        {PASS  }
        {IGNORE}
```

(BSC and TWX stations only)

Specifies whether the station or stations represented by this TERMINAL macro will transmit identification (ID) sequences when calling or being called by the controller and, if so, how the NCP is to react to the sequence.

symbol

Specifies that the NCP is to expect and to verify ID sequences received from stations; it also names the identification list (IDLIST macro) with which the program is to check the sequences it receives. *symbol* is valid whether this TERMINAL macro represents a call-in logical-connection station (CTERM=YES) or a call-out station (CTERM=NO), but *symbol* is valid only if the type of station is not "multiple-terminal-access" (TERM operand does not specify MTA).

NONE

Specifies that no ID sequences are expected from stations that call the controller or are called by the controller.

PASS

Specifies that the NCP is to expect ID sequences from stations and is to pass to VTAM all ID sequences it receives. IDSEQ=PASS is valid only if this TERMINAL macro represents a call-in logical-connection station (CTERM=YES is specified).

IGNORE

Specifies that the NCP is to ignore—that is, neither check nor pass to the host processor—any ID sequences it receives. IDSEQ=IGNORE is valid only if the controller calls the station represented by this TERMINAL macro; if stations are to call the controller (CTERM=YES is specified), IDSEQ=IGNORE is invalid.

The IDSEQ operand is valid only for BSC or TWX stations on a switched line (GROUP macro specifies LNCTL=BSC or LNCTL=SS, DIAL=YES; LINE macro specifies POLLED=NO [or POLLED operand is omitted]; this TERMINAL macro specifies a BSC or TWX station in TERM); and if controller is to expect an ID sequence from a station it calls, this TERMINAL macro specifies a telephone number in DIALNO).

```
[INHIBIT={[TEXTTO][,TIMEFILL][,WACKCNT][,SUBBLOCK][,ERPR][,ERPW])}]
         {NONE                                                   }
```

Specifies which of the NCP facilities are to be inhibited from functioning initially; that is, when the program begins execution after being loaded into the controller.

[TEXTTO]

Specifies that the NCP is to use the interval specified in the ITEXTTO operand of the BUILD macro, rather than the value specified by the TEXTTO operand of the GROUP macro, as the time limit between receipt of successive text characters.

[TIMEFILL] *(start-stop lines only)*

Specifies that automatic insertion of idle characters following carriage return and horizontal tab characters is to be inhibited; that is, no idle characters are to be sent. This parameter is valid only if LNCTL=SS is specified in (or LNCTL operand omitted from) the GROUP macro.

[WACKCNT] *(BSC lines only)*

Specifies that the WACK limit specified by the WACKCNT operand of the GROUP macro is to be inhibited; that is, the first WACK received from the station causes the NCP to return the request to the host processor rather than to respond to that and subsequent WACKs with an ENQ character. This parameter is valid only if LNCTL=BSC is specified in the GROUP macro.

[SUBBLOCK]

Specifies that the subblocking indicated by the TRANSFR operand of the LINE (or GROUP) macro is to be inhibited; that is, if the number of buffers specified by the TRANSFR operand are filled by received text, the network control program terminates the receiving operation just as if the cutoff limit (specified by the CUTOFF operand) had been reached.

[ERPR]

Specifies that recovery procedures for text-read errors (as specified by the RETRIES operand of the LINE macro) are to be inhibited. That is, the network control program will not attempt recovery for text-read errors.

[ERPW]

Specifies that recovery procedures for text-write errors (as specified by the RETRIES operand of the LINE macro) are to be inhibited. That is, the NCP will not attempt recovery for text-write errors.

NONE

Specifies that none of the NCP facilities are to be inhibited; that is, all will become effective when execution of the program begins.

```
[ITBMODE=( [{YES}] [,{YES}] )]
            {NO }    {NO }
```

(BSC lines only)

Specifies, for a BSC station, how the NCP is initially to handle ITB characters in text received from the station or the host processor.

The first suboperand specifies whether the program is to insert a EIB (error information block) character following each ITB character received from the station.

The second suboperand specifies whether each ITB character received from the access method is followed by an EIB character. If you specify YES, the NCP removes from the data it sends to the station the first character following each ITB character received from the access method.

This operand is valid only if you code LNCTL=BSC in the GROUP macro.

```
[LCST={mtalcst name}]
      {NONE        }
```

Specifies the name of the MTALCST macro representing the set of device characteristics to be used by the NCP when calling the device represented by this TERMINAL macro.

Code LCST=*mtalcst name* in the TERMINAL macro for each station that the NCP is to call via a multiple-terminal-access line.

Code LCST=*mtalcst name* only if you specify DIAL=YES and LNCTL=SS (or you omit the LNCTL operand) in the GROUP macro; TERM=MTA in the TERMINAL macro (or a higher-level macro); CTERM=NO in (or omit CTERM operand from) the TERMINAL macro; and if you specify a telephone number in the DIALNO operand of the TERMINAL macro.

Code LCST=NONE (or omit the operand) if no MTALCST macros need to be specified.

```
[LGRAPHS=( [{REJECT}] [,{REJECT}] )]
           {ACCEPT}     {ACCEPT}
```

Specifies whether leading graphics received from the station are to be accepted by the NCP or rejected (treated as an error condition).

The first parameter specifies acceptance or rejection of leading graphics for read operations; the second parameter specifies acceptance or rejection for write operations.

Note: For a 2740 Model 2, the second parameter must be ACCEPT.

```
[POLL=chars]
```

Specifies, in hexadecimal representation, the EBCDIC polling characters assigned to this station.

Code only the alphanumeric polling character, omitting any control characters. For example, for an IBM 2740 with station control whose polling character is A you would code the hexadecimal representation of the EBCDIC character *A*, thus POLL=C1. Do not code the space character that is transmitted after the polling character.

Note: If TERM=3275, 3277, 3284, or 3286 is coded and the ADDR operand specifies the addressing character, the POLL operand must specify the polling character.

If this TERMINAL macro represents a call-in logical-connection 1050 station (CTERM=YES, TERM=1050), you must specify polling characters in at least one of the macros representing this station (that is, in the TERMINAL macro or in one of the COMP macros, if any, representing station components).

This operand is invalid if the TERMINAL macro represents an IBM @ & $) WITHOUT Station Control, a 2741, or a 2980, as these types of terminals cannot be individually polled.

This operand is also invalid if this TERMINAL macro represents a call-in multiple-terminal-access, logical-connection station (TERM=MTA, CTERM=YES).

If this TERMINAL macro represents an MTA station (TERM=MTA) that is not a call-in MTA station (CTERM=NO), the POLL and/or ADDR operands may be specified or omitted. *Exception:* If any of the MTALCST macros named in the LCST operand of this TERMINAL macro represents IBM 1050 terminals, either POLL, or ADDR, or both, must be specified in this TERMINAL macro.

Note: The character you specify in this POLL operand for a tributary controller must conform to the requirements described under the TADDR operand of the LINE macro.

```
[PT3EXEC={YES}]
        {NO }
```

Specifies whether a block handler set that is executed at point 3 (see the BHSET macro) is to be associated with the device represented by this TERMINAL macro.

This operand is ignored if you omit the BHSET operand or if you code BHSET=NONE, BHSET=PT3, or BHSET=ALL in this TERMINAL macro.

```
[XMITLIM={count}]
        {NO   }
```

Specifies the maximum number, if any, of transmissions (ended by EOT unless you have specified ENDTRNS=EOB) the NCP will receive from or send to this station or both. If this limit is reached before the host processor explicitly requests that the network control program disconnect the station from the controller, the NCP will automatically suspend the session.

The maximum value you may specify is 255; the minimum is 1.

XMITLIM=NO means that the NCP will send to, or receive from the station indefinitely.

> **CAUTION**
> In most applications, the default value, XMITLIM=NO, is *not* appropriate because it allows the station represented by this TERMINAL macro to monopolize the communication line indefinitely. You should therefore normally specify a count in this operand. The lower the count you specify for each station attached to a line, the greater will be the degree of line sharing.

XMITLIM=count is valid only for stations associated with a line for which you have specified a service order table (that is, a SERVICE macro follows the LINE macro).

Specifying Lower-Level Operands in a Higher-Level Macro

In addition to the previous operands, most operands of the COMP macro can be specified in the TERMINAL (or LINE or GROUP) macro instead of in the COMP macro. Figure 5-11 shows which of the COMP operands you may specify in the TERMINAL macro.

VTERM Macro Instruction (VTAM Only)

The VTERM macro instruction specifies a name for each type of terminal that may call the controller on a multiple-terminal-access line. This is needed so that VTAM can associate the calling terminal with a specific application program. The macro also provides information to VTAM about the VTAM buffers required by the terminal and specifies logon information associated with the terminal operating parameters specified by an MTALCST macro.

The logon information includes (1) the name of the application program to which the calling terminal is logged on and (2) the name of the VTAM interpret table that determines the application program.

This macro also includes the NOSP-only SPAN operand.

The VTERM macro, if used, must appear directly following the TERMINAL macro representing the call-in MTA terminal. For example:

```
GROUP
LINE
TERMINAL     TERM=MTA, CTERM=YES,...
VTERM
VTERM
TERMINAL     TERM=MTA, CTERM=NO,...
TERMINAL     TERM=MTA, CTERM=NO,...
     .
     .
     .
```

See the *ACF/VTAM Installation* manual for a complete description of this macro and its operands and for the conditions of its use.

The format of the VTERM macro is:

```
symbol        VTERM        operands
```

Operands

```
BUFLIM=
LCST=
LOGAPPL=
LOGTAB=
SPAN=
```

COMP Macro Instruction

The COMP macro represents one input component or one output component, or both, of a start-stop or BSC station in the network. COMP also specifies:

- The resource name for the component
- The polling and addressing characters by which the NCP will contact the components
- The block handler set, if any, associated with the components, and the times of execution of block handlers within the set
- Certain procedural options the NCP is to use when communicating with the component

All COMP macros for a station must be grouped together immediately following the associated TERMINAL macro.

COMP macros are required only for components the NCP must individually poll or address with specific component polling or addressing characters. A COMP macro is not required for a station having only one input component and one output component because the polling and addressing characters for these are specified in the TERMINAL macro. *Exception:* Each of two or more components with which separate sessions will be established concurrently must be represented by an individual (TERMINAL or COMP) macro. For example, to establish concurrent sessions with a keyboard and a card punch would require that a separate TERMINAL (or COMP) macro be used to represent each device.

Each COMP macro generates a resource name. All operands of the COMP macro apply only to line operation in network control mode.

(VTAM Users Only): Appearing at the end of the following list of operands are the VTAM-only operands that may be coded in this macro instruction. The VTAM operands provide information only to the VTAM initialization process and are not required (though are permissible) as input to the NCP generation procedure. See the *ACF/VTAM Installation* manual for descriptions of these operands and for information on the VTAM initialization process. (See the *NOSP Installation Manual* for the description of the SPAN operand.)

The format of the COMP macro is:

```
symbol          COMP          [operands]
```

Operands

```
[ADDR={addr chars      }]
      {selection chars}
[,ATTN={ENABLED }]
       {DISABLED}
[,BHEXEC={([PT1][,PT2][,PT3])}]
         {ALL                }
[,BHSET={NONE                    }]
        {setname[,EXEC={YES}]}
                       {NO }
[,CDATA={YES}]
        {NO }
[,CONV={YES}]
       {NO }
[,ENDTRNS={EOT}]
          {EOB}
[,INHIBIT={([TEXTTO][,TIMEFILL][,WACKCNT]}]
          {     [,SUBBLOCK][,ERPR][,ERPW])}
          {NONE                          }
[,ITBMODE=([{YES}][,{YES}])]
            {NO }   {NO }
[,LGRAPHS=([{REJECT}][,{REJECT}])]
            {ACCEPT}   {ACCEPT}
[,POLL=chars]
[,PT3EXEC={YES}]
          {NO }
[,XMITLIM={count}]
          {NO   }
```

VTAM-only operands:

```
BUFLIM=
DEVICE=
ISTATUS=
LOGAPPL=
LOGTAB=
PU=
SPAN=
```

symbol

Provides a name for the component and is required. *symbol* may be any valid assembler-language symbol. The first character may not be $.

The operands in this macro are equivalent to, and are to be used similarly to, the corresponding operands of the TERMINAL macro. See the TERMINAL macro for descriptions of the operands.

Specifying COMP Operands in a Higher-Level Macro

Except for the POLL and ADDR operands, the operands of the COMP macro can be specified in the TERMINAL, LINE, or GROUP macro instead of the COMP macro. Figure 5-11 shows which of the COMP macro operands you may specify at a higher level.

Block Handler Definition Macro Instructions

With these macro instructions, you specify what processing operations the NCP is to perform on blocks of message data received from the access method over the network control subchannel or from stations on lines operating in network control mode. Each function is performed by a routine called a block-handling routine. Several block-handling routines may be grouped into a block handler; the routines process the data in the same sequence as the block handling routine macros appear within the block-handler.

You may establish a set of block handlers for each BSC and start-stop station, with each set including any of the block handling routines appropriate for the station. You may also specify separate block handler sets for individual components of a station. Block handling routines cannot be specified for processing message data originating from or destined for SDLC stations.

The two macros STARTBH and ENDBH define the beginning and end of a block handler. Only one STARTBH macro and one ENDBH macro may appear in a block handler.

STARTBH Macro Instruction

The STARTBH macro:

- Establishes the beginning of a block handler
- Provides a name for the block handler

The format of the STARTBH macro is:

```
bhname          STARTBH        [operand]
```

Operands

```
         {PT1}
[BHEXEC={PT2}]
         {PT3}
```

bhname

Specifies the name of the block handler set and is required; *bhname* may be any symbol valid in the assembler language; the first character must not be $. (*bhname* is referred to by the BHSET macro.)

```
         {PT1}
[BHEXEC={PT2}]
         {PT3}
```

Specifies the point at which this block handler will be executed.

Data sent to a station

If you wish the data received from the access method to be processed *after* the network control program has contacted the station, code BHEXEC=PT2.

Note: If you specify BHEXEC=PT2, a logical connection between the NCP and the station must exist before the block handler is executed. The interval during which the line is unavailable for communication with other stations is extended by the execution time of the block handler.

The block-handling routines you can include in a block handler that processes outgoing data are date and time or any user-provided block-handling routines. Therefore, a STARTBH macro that specifies BHEXEC=PT1 or PT2 may be followed by DATETIME or UBHR macros.

If the block handler is to process incoming data, as well as outgoing data (possible only if you code BHEXEC=PT2), you may also include the EDIT macro. However, the block-handling routine invoked by the EDIT macro will process only the *incoming* data.

Block handlers process outgoing data only if the data transfer command is not in error.

Data received from a station

If you wish to allow communication over the line to continue while the block handler is processing the received data, code BHEXEC=PT3.

If you wish the NCP to suspend further communication over the line until the block handler has processed the received message data, code BHEXEC=PT2.

Note: If you specify BHEXEC=PT2, the line is unavailable for further communication with stations during the time occupied by block-handler execution.

The block-handling routines you can include in a block handler that processes incoming data are date and time, edit, or any user-provided block-handling routines. Therefore, a STARTBH macro that specifies BHEXEC=PT2 or PT3 may be followed by a DATETIME, EDIT, or UBHR macro.

The IBM-supplied routine that converts path information units (PIU) from BSC to SDLC format can be included only in a block handler that is executed at point 3. The macro that calls this routine, SPAFPT3, must appear after a STARTBH macro in which you specify BHEXEC=PT3.

Except for the date and time routine and any user block-handling routines, block handlers process incoming data only if the data block was correctly received (that is, a data-check error did not occur).

The ACCESS operand of the UBHR macro determines whether a user-written routine processes only error-free blocks or only blocks containing errors.

If you omit the BHEXEC operand, the block handler will be executed as if BHEXEC=PT1 were specified.

BHSET Macro Instruction

BHSET defines a set of block handlers which may be statically or dynamically assigned to a device. A maximum of 255 block handler sets may be created.

The format of the BHSET macro is:

```
setname          BHSET          [operands]
```

Operands

```
[EXEC={YES} ]
      {NO }
[,PT1=bhname]
[,PT2=bhname]
[,PT3=bhname]
```

`setname`

Provides a name for the set of block handlers and is required. *setname* may be any valid assembler-language symbol. The first character may not be $. (*setname* is referred to by the BHSET operand of the CLUSTER, TERMINAL, and COMP macros.)

```
[EXEC={YES} ]
      {NO }
```

Specifies whether the block handler set may be executed when it is associated with a device by command from the host processor.

Do not confuse this operand with the EXEC operand of the TERMINAL macro, which specifies whether the block handler set is to be *initially* executable.

`[PT1=bhname]`

Specifies the name of the block handler to be executed when a request has been received from the host processor for the device but before it has been determined that the line is available.

`[PT2=bhname]`

Specifies the name of the block handler to be executed when a request has been received from the host processor for the device but after the line has been found to be available.

`[PT3=bhname]`

Specifies the name of the block handler to be executed at PT3. It will be executed when an input operation on a communication line ends and after the line is released for use with another station.

DATETIME Macro Instruction

The DATETIME macro specifies whether the NCP is to insert the date, the time of day, or both, in a data block.
The macro also specifies whether to insert this information in only the first block of a message or in all blocks.
It also specifies one of four formats in which the date is to appear.

The network control program inserts the date and/or time as the first data in a block.
The date and time immediately precede the first text character in a block.
The date precedes the time.
The format of the DATETIME macro is:

```
[symbol]        DATETIME      [operands]
```

Operands

```
[DATE={YES}]
      {NO }
[,DATEFMT={YY.DDD  }]
          {MM/DD/YY}
          {YY/MM/DD}
          {DD/MM/YY}
[,INSERT={FIRST}]
         {ALL  }
[,PT2EXEC={BEFORE}]
          {AFTER }
[,TIME={YES}]
       {NO }
```

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

```
[DATE={YES}]
      {NO }
```

Specifies whether the current date is to be inserted.

```
          {YY.DDD  }
[DATEFMT={MM/DD/YY}]
          {YY/MM/DD}
          {DD/MM/YY}
```

Specifies the format for the date.

Code DATEFMT=YY.DDD or omit the operand if you wish the date to appear as the year followed by the day of year; for example, 78.264 (October 21,1978).

Code DATEFMT=MM/DD/YY if you wish the date to appear in the month/day/year format; for example, 10/21/78.

Code DATEFMT=YY/MM/DD if you wish the date to appear in the year/month/day format; for example,78/10/21.

Code DATEFMT=DD/MM/YY if you wish the date to appear in the day/month/year format; for example, 21/10/78.

```
[INSERT={FIRST}]
        {ALL  }
```

Specifies whether the date and/or time is to be inserted in the first block of each message or in all blocks.

```
[PT2EXEC={BEFORE}]
         {AFTER }
```

Specifies, for a routine executed at point 2, whether the routine is to insert the date and/or time *before* the I/O operation (PT2EXEC=BEFORE) or after the I/O operation (PT2EXEC=AFTER).

```
[TIME={YES}]
      {NO }
```

Specifies whether the current time is to be inserted. The time is always in the format *hh.mm.ss*, using the continental (24-hour) form. For example, 07.42.18 represents 7:42:18 a.m.; 19.42.18 represents 7:42:18 p.m.

Do not code both DATE=NO and TIME=NO.

If you specify both EDIT and DATETIME in the same block handler, the EDIT macro must precede DATETIME.

EDIT Macro Instruction

The EDIT macro causes the NCP to edit data originally entered from a keyboard. Erroneously entered characters followed by a text-canceling character are deleted from the message data. For example, the NCP will correct PENNSLYV///YLVANIA to PENNSYLVANIA if the / character the keyboard operator used as the text canceling (""backup') character is specified in the EDIT macro.

The EDIT macro may be specified only in a block handler that processes data received from a station. It cannot be specified following a STARTBH macro in which BHEXEC=PT1 is coded. (A block handler executed at point 1 acts only upon data being sent to a station.) If you specify both EDIT and DATETIME in the same block handler, the EDIT macro must precede DATETIME.

The format of the EDIT macro is:

```
[symbol]      EDIT       [operand]
```

Operands

```
[BKSP={char}]
      {16   }
```

`[symbol]`

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The first character may not be $.

```
[BKSP={char}]
      {16   }
```

Specifies the character acting as the text canceling (backspace) character*char* is the hexadecimal representation of the text canceling character. If this operand is omitted, 16 (the hexadecimal representation of the EBCDIC backspace [BS] character) is assumed to be the text-canceling character. The value of this operand must be specified without framing characters (X" ').

SPAFPT3 Macro Instruction

The SPAFPT3 macro causes the IBM-supplied PIU format conversion routine for the SDLC/BSC path function to be included in a block handler. A block handler that includes this macro (or includes a UBHR macro that specifies a user-written conversion routine) must be specified for any BSC station that is to communicate with an SDLC logical unit via the SDLC/BSC path function. The block handler must be specified as executable at point 3 (that is, message data is processed after it is received from the BSC station).

Do not specify both the SPAFPT3 macro and a UBHR macro (calling a user-written conversion routine) in the same block handler.

See the publication *Network Control Program/VS SDLC/BSC Path Function System Programmer's Guide,* GC30-3029, for information on the path function.

The format of the SPAFPT3 macro is:

```
[symbol]      SPAFPT3
```

Operands
This macro has no operands

[symbol]

Provides a name for the macro. *symbol* must be any valid assembly language symbol. The first character may not be $.

This macro has no operands.

UBHR Macro Instruction

The UBHR macro allows you to include a user-written block-handling routine in a block handler. It specifies the name of the module and its entry point, and specifies under what conditions it is to be executed.

A maximum of 65 uniquely named user modules may be specified by UBHR macros. However, the generation procedure does not limit the number of entry points (ENTRY operand) you may specify in a user block-handling routine.

Note: Use of the UBHR macro forces storage boundary alignment to the next 2K boundary because of the storage-protect feature of the communications controller. Therefore, up to 2K bytes of storage may be unused when the network control program is loaded into the controller.

See ""'Guidelines for Writing User Block Handling Routines' in Chapter 3 for more information on these routines.

The format of the UBHR macro is:

```
[symbol]        UBHR          operand[,operands]
```

Operands

```
NAME=member name
[,ACCESS={GOOD }]
         {ERROR}
          {READ  }
[,COMMAND={INVITE}]
          {BOTH  }
[,ENTRY=entry point name]
[,PT2EXEC={BEFORE}]
          {AFTER }
```

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The first character may not be $.

NAME=member name

Specifies the name of a user-written module. The module name must be contained in the data set specified by the USERLIB operand of the BUILD macro.

[ACCESS={GOOD }]
{ERROR}

Specifies whether the user block-handling routine is to process only good (error-free) blocks or only blocks containing errors.

The ACCESS operand may be coded only if the UBHR macro follows a STARTBH macro in which BHEXEC=PT2 or BHEXEC=PT3 is specified. If

BHEXEC=PT2 is specified, PT2EXEC=AFTER must also be coded in the UBHR macro. If this UBHR macro follows a STARTBH macro that specifies BHEXEC=PT1, this operand is invalid and must be omitted.

```
          {READ   }
[COMMAND={INVITE}]
          {BOTH   }
```

Specifies whether the user-written routine is to process data received in response to a Read request, an Invite request, or both. This applies only for incoming data when the UBHR macro follows a STARTBH macro that specifies BHEXEC=PT2 or BHEXEC=PT3.

This operand is valid only if ACCESS=GOOD is specified in this macro.

If this UBHR macro follows a STARTBH macro that specifies BHEXEC=PT1, this operand is invalid and must be omitted.

```
[ENTRY=entry point name]
```

Specifies whether the name of the entry point within the user-written module named by the NAME operand. If you omit the ENTRY operand, the entry-point name is assumed to be the the same as the module name.

```
[PT2EXEC={BEFORE}]
          {AFTER }
```

Specifies the user routine is to process outgoing message data *before* the I/O operation (PT2EXEC=BEFORE) or *after* the I/O operation (PT2EXEC=AFTER).

This operand is valid only in a UBHR macro that follows a STARTBH macro that specifies BHEXEC=PT2.

ENDBH Macro Instruction

The ENDBH macro:

- Specifies the end of a block handler.
- Must be the last macro of each block handler.

The format of the ENDBH macro is:

```
[symbol]        ENDBH
```

Operands
This macro has no operands

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The generation procedure does not check the symbol for validity.

This macro has no operands.

Generation Delimiter Macro Instruction

GENEND Macro Instruction

The GENEND macro indicates the end of the network control program generation input deck. It must be the last network control program generation macro instruction coded.

This macro defines the entry points and library member names of user-written modules you may want to include in your NCP.

The GENEND macro also specifies the scan limits and address substitution mask for each type 2 communication scanner installed in the communications controller and the scan limits and high speed select mask for each type 3 scanner installed. These parameters are for use only if any communication lines in the network operate at 4800 or more bits per second. Specifying these parameters causes the scanner to scan line interfaces to which high speed lines are attached more frequently than those for lower speed lines; the more frequent scanning is done at the expense of not scanning other line interface addresses. The addresses not scanned are therefore rendered unusable.

Use of scan limits, address substitution mask, and high speed select masks are described in more detail in Appendix F.

The format of the GENEND macro is:

```
[symbol]        GENEND        [operands]
```

Operands

```
[HSPDSEL=( [mask1] [,mask2] , [mask3] , [mask4] )]
[,SCANCTL=( [limit1] , [limit2] , [limit3] ,
            [limit4] [,asmask] )]
[,INIT=( symbol1 [,symbol2] . . . [,symboln] )]
[,SRCHI=( symbol1 [,symbol2] . . . [,symboln] )]
[,SRCLO=( symbol1 [,symbol2] . . . [,symboln] )]
[,TMRTICK=symbol]
[,INCHI=( symbol1 [,symbol2] . . . [,symboln] )]
[,INCLO=( symbol1 [,symbol2] . . . [,symboln] )]
[,INCINIT=( symbol1 [,symbol2] . . . [,symboln] )]
[,INCL2HI=( symbol1 [,symbol2] . . . [,symboln] )]
[,INCL2LO=( symbol1 [,symbol2] . . . [,symboln] )]
[,ORDHI=( symbol1 [,symbol2] . . . [,symboln] )]
[,ORDLO=( symbol1 [,symbol2] . . . [,symboln] )]
[,ORDINIT=( symbol1 [,symbol2] . . . [,symboln] )]
[,ORDL2HI=( symbol1 [,symbol2] . . . [,symboln] )]
[,ORDL2LO=( symbol1 [,symbol2] . . . [,symboln] )]
```

[symbol]

Provides a name for the macro. *symbol* may be any valid assembler-language symbol. The first character may not be $.

[HSPDSEL=([mask1] , [mask2] , [mask3] , [mask4]) *(type 3 scanner only)*

Specifies the high-speed select masks for each type 3 communication scanner installed in the communications controller. The masks are used to cause high-speed line interfaces to be scanned more frequently than interfaces for lower-speed lines (under 4800 bps).

mask1..mask4

Specifies 8-bit binary sequences (for example, 00101000) constituting the masks. For scanning purposes, the line interface base (LIB) serviced by a type 3 scanner is divided into eight portions. The eight bit positions of a mask correspond to the eight portions (0-7) within all LIBs serviced by the scanner. See Appendix F for an illustration.

A mask bit of 0 specifies that all line interface addresses in the corresponding portion of the LIB are scanned equally often. A mask bit of 1 specifies that only the line interface with the lowest address within that LIB portion is scanned; all other addresses within that LIB portion are not scanned. The scans that would otherwise be applied to these addresses are instead applied to the lowest address, thus increasing the scan frequency of that address. See the following table for addresses scanned and not scanned for each high speed select mask bit position.

LIB Portion and HSS Mask Bit Position	*Bit Value*	*Scanner Position*	*Address Scanned*	*Addresses Not Scanned*
0	1	First	020	021,030,031,040,041
		Second	0A0	0A1,0B0,0B1,0C0,0C1,0D0,0D1
		Third	120	121,130,131,140,141,150,151
		Fourth	1A0	1A1,1B0,1B1,1C0,1C1,1D0,1D1
1	1	First	022	023,032,033,042,043
		Second	0A2	0A3,0B2,0B3,0C2,0C3,0D2,0D3
		Third	122	123,132,133,142,143,152,153
		Fourth	1A2	1A3,1B2,1B3,1C2,1C3,1D2,1D3
2	1	First	024	025,034,035,044,045
		Second	0A4	0A5,0B4,0B5,0C4,0C5,0D4,0D5
		Third	124	125,134,135,144,145,154,155
		Fourth	1A4	1A5,1B4,1B5,1C4,1C5,1D4,1D5
3	1	First	026	027,036,037,046,047
		Second	0A6	0A7,0B6,0B7,0C6,0C7,0D6,0D7
		Third	126	127,136,137,146,147,156,157
		Fourth	1A6	1A7,1B6,1B7,1C6,1C7,1D6,1D7
4	1	First	028	029,038,039,048,049
		Second	0A8	0A9,0B8,0B9,0C8,0C9,0D8,0D9
		Third	128	129,138,139,148,149,158,159
		Fourth	1A8	1A9,1B8,1B9,1C8,1C9,1D8,1D9
5	1	First	02A	02B,03A,03B,04A,04B
		Second	0AA	0AB,0BA,0BB,0CA,0CB,0DA,0DB
		Third	12A	12B,13A,13B,14A,14B,15A,15B
		Fourth	1AA	1AB,1BA,1BB,1CA,1CB,1DA,1DB
6	1	First	02C	02D,03C,03D,04C,04D
		Second	0AC	0AD,0BC,0BD,0CC,0CD,0DC,0DD
		Third	12C	12D,13C,13D,14C,14D,15C,15D
		Fourth	1AC	1AD,1BC,1BD,1CC,1CD,1DC,1DD
7	1	First	02E	02F,03E,03F,04E,04F
		Second	0AE	0AF,0BE,0BF,0CE,0CF,0DE,0DF
		Third	12E	12F,13E,13F,14E,14F,15E,15F
		Fourth	1AE	1AF,1BE,1BF,1CE,1CF,1DE,1DF
any	0	All addresses in corresponding scanner position are scanned.		

mask1 applies to a type 3 scanner installed in the first scanner position (base module), *mask2* to a type 3 scanner installed in the second scanner position (first expansion module), etc. If a scanner position does not contain a type 3 scanner, code a comma to represent the missing mask if succeeding positions are occupied by a type 3 scanner.

The bit settings you specify should correspond to the high-speed lines requiring increased scanning. For each-such line interface installed in the controller, a high speed select feature is present that blocks the attachment of lines to all but the lowest address in the corresponding LIB portion.

Example: Assume that a 3705 having three modules is equipped with type 3 scanners in the first and second expansion modules but not in the base module. If high-speed select features are present in the second scanner for LIB portions 3 and 7 (thus allowing high speed lines to be attached to addresses 0A6 and 0AE), you would specify HSPDSEL=(,00010001,00000000). The first comma signifies that no type 3 scanner is installed in the base module; the first 8-bit mask indicates that increased scanning frequency is required for addresses 0A6 and 0AE in LIB portions 3 and 7, respectively; and the second mask indicates that no addresses in the second expansion module (scanner position 3) require increased scanning frequency.

If you omit the HSPDSEL operand but the program generation procedure determines that the high-speed select function is required, the procedure determines the appropriate mask and assumes that the appropriate high-speed select features are installed.

CAUTION
If you specify the HSPDSEL operand and omit the SCANCTL operand, 0 will be assumed for the SCANCTL operand. If the network includes lines having speeds between 4800 and 19 200 bps, a scan limit of 0 is invalid; specify a scan limit of 2 for the scanners servicing lines within the range of speeds mentioned.

`[SCANCTL=([limit1],[limit2],[limit3],[limit4][,asmask])]`

Specifies the scan limits for each type 2 and 3 communication scanner installed in the controller and specifies the address-substitution mask, if used.

This operand is valid only if one or more type 2 or type 3 scanners are installed in a 3705. (An address substitution mask must not be specified if a type 3 scanner is installed.)

`limit1...limit4`

Specifies the scan limits for each installed type 2 or type 3 scanner. Each limit can be from 0 to 3; these values have the meanings shown below. *limit1* specifies the scan limit for the first scanner position (base module), *limit2* for the second position (first expansion module), etc. All addresses associated with a scanner are scanned if the scan limit for that scanner is 0. Scan limits of 1, 2, and 3 reduce the number of addresses scanned to 8, 48, and 16, respectively. If a scanner position does not contain a type 2 or type 3 scanner, code a comma for the corresponding limit; for example, SCANCTL=(limit1,,limit3,,asmask). If a type 2 or type 3 scanner is installed but you specify no limit, the generation procedure assigns the appropriate limit based on the range of actual installed addresses and line speeds as specified in the LINE macros.

Note: See the Caution note under the HSPDSEL operand description.

The scan limits have the following meanings:

Scan Limit	*Addresses Scanned*	*Addresses Not Scanned*	*Maximum Line Speed*
0	020-05F 0A0-0FF 120-17F 1A0-1FF	(all addresses scanned)	4 800 bps
1	020-027 0A0-0A7 120-127 1A0-1A7	028-05F 0A8-0FF 128-17F 1A8-1FF	57 600 bps
2	020-04F 0A0-0CF 120-14F 1A0-1CF	050-05F 0D0-0FF 150-17F 1D0-1FF	9 600 bps
3	020-02F 0A0-0AF 120-12F 1A0-1AF	030-05F 0B0-0FF 130-17F 1B0-1FF	19 200 bps

asmask

Specifies the address-substitution mask to be used if the communications controller is equipped with the address substitution feature. Specify the mask as a binary sequence of four bits (omitting frame characters, B''), as follows:

Bit	*Value*	*Meaning*
0	1	Address substitution is to be performed for address 0 in LIB position 1. Addresses E and F in all LIB positions are disabled.
0	0	No address substitution; all addresses enabled.
1	1	Address substitution is to be performed for address 2 in LIB position 1. Addresses C and D in all LIB positions are disabled.
1	0	No address substitution; all addresses enabled.
2	1	Address substitution is to be performed for address 4 in LIB position 1. Address A and B in all LIB positions are disabled.
2	0	No address substitution; all addresses enabled.
3	1	Address substitution is to be performed for address 6 in LIB position 1. Addresses 8 and 9 in all LIB positions are disabled.
3	0	No address substitution; all addresses enabled.

CAUTION

The address-substitution mask should not be specified if one or more type 3 scanners are installed in the communications controller because address substitution inhibits scanning of corresponding addresses in *all* LIBs regardless of whether serviced by type 2 or type 3 scanners. Instead of address substitution, use upper scan limits or high-speed select masks to provide increased scanning frequency for high-speed lines.

If you omit the SCANCTL operand, the generation procedure automatically calculates the appropriate scan limits and, if the network configuration requires the use of address-substitution, calculates the address substitution mask. The procedure assumes that the appropriate address-substitution feature is installed. A message is printed in the assembly listing when the feature is required. Determine from the system designer whether the feature is installed. If not, a discrepancy exists; either respecify the network configuration or have the address-substitution feature installed.

Note: See the Caution note under the HSPDSEL operand description.

Including User-Written Code

The following operands provide the NCP generation procedure with the entry points and library member names of user-written modules.

With the exception of the TMRTICK operand, as many symbols as necessary may be specified for each operand. However, the assembler has a limit of 255 characters per operand, which includes commas and framing parentheses. If more than 255 characters are required, additional operands may be specified.

`[INIT=(symbol1[,symbol2]...[,symboln])]`

Specifies the entry point(s) of user-written initialization routines. These routines must be object modules cataloged in the OS/VS object library or the DOS/VS relocatable library. (See ""Including User-Written Modules' in Chapter 4.)

If this operand is specified, either INCINIT, INCHI, or INCLO must be specified.

`[SRCHI=(symbol1[,symbol2]...[,symboln])]`
`[SRCLO=(symbol1[,symbol2]...[,symboln])]`

These two operands specify the source code for user-defined control blocks and tables that must be assembled with the NCP control blocks and tables. The source code must be in macro form using 3705 assembler instructions.

symbol is the member names of the macros as they are cataloged in the user macro library or the NCP stage 2 macro library (SYS1.MAC3705).

SRCHI defines source code that may reside anywhere in the 3705 controller storage.

SRCLO defines source code that must reside in the low 64K of storage. If user-written line control is specified (LEVEL2 operand in a GROUP macro), SRCLO is required.

[TMRTICK=symbol]

Specifies the entry point of a user-written timer-tick service routine. This routine must be an object module cataloged in the OS/VS object library or the DOS/VS relocatable library. (See ""Including User-Written Modules' in Chapter 4.)

To include the timer-tick routine in the NCP, either the INCL2HI or the INCL2LO operand must be specified.

Linkage Editor Operands to Include User Code

The following operands specify the names of macro library members that contain the linkage editor INCLUDE and ORDER statements for user-written object modules.

On an OS/VS system, the object modules must be members of the user object library specified in the USERLIB operand of the BUILD macro or the NCP object library (SYS1.OBJ3705). Every CSECT in the user object modules must be specified on a linkage editor ORDER statement. Any CSECT not specified on an ORDER statement will not be in the correct position in the load module and will be overlaid during initialization of the NCP.

In a DOS/VS system, the object modules must be members of the relocatable library. Only the INCLUDE operands (INC*xx*) of this macro should be specified for a DOS/VS system; the ORDER operands (ORD*xx*) are invalid.

[INCHI=(symbol1[,symbol2]...[,symbol***n***])]
[INCLO=(symbol1[,symbol2]...[,symbol***n***])]

These operands specify the macro library members that contain linkage editor INCLUDE statements for object modules that are *not* level 2 or level 3 code. At least one of these operands must be specified if VIRTUAL=YES is specified in a GROUP macro or if an NCPNAU macro is included in the generation.

INCHI names the object modules that may reside anywhere in the 3705 controller storage. These modules will execute with a storage protect key of 1.

INCLO names those object modules that must reside in the low 64K of controller storage. These modules will execute with a storage protect key of 0.

Note: If user block-handling routines (UBHR macros) are included in the generation and INCHI is specified, the generation procedure assumes that the block-handling routine linkage editor control statements are included in the members specified in the INCHI operand.

[INCINIT=(symbol1[,symbol2]...[,symbol***n***])]

Specifies the macro library members that contain linkage editor INCLUDE statements for user-written initialization code. The modules included in the NCP generation by this operand are overlaid once the initialization has been completed. If the user initialization code is not to be overlaid, specify the library members containing the INCLUDE statments in the INCHI or INCLO operands.

```
[INCL2HI=( symbol1 [,symbol2] ... [,symboln] )]
[INCL2LO=( symbol1 [,symbol2] ... [,symboln] )]
```

These operands specify the macro library members that contain linkage editor INCLUDE statements for object modules that are level 2 or level 3 code. These modules will execute with a storage protect key of 0.

INCL2HI names the object modules that may reside anywhere in the 3705 controller storage.

INCL2LO names those object modules that must reside in the low 64K of controller storage. This operand is required if user-written line control is included in the system (LEVEL2 operand specified in a GROUP macro).

```
[ORDHI=( symbol1 [,symbol2] ... [,symboln] )]
[ORDLO=( symbol1 [,symbol2] ... [,symboln] )]
```

These operands specify the macro library members that contain linkage editor ORDER statements for object modules that are *not* level 2 or level 3 code. If this generation is on a DOS/VS system, these operands are invalid and must be omitted.

ORDHI names the object modules that may reside anywhere in the 3705 controller storage. This operand is required if INCHI is specified and TYPESYS=OS is specified in the BUILD macro.

ORDLO names those object modules that must reside in the lower 64K of controller storage. This operand is required if INCLO is specified and TYPESYS=OS is specified in the BUILD macro.

Note: If user block-handling routines (UBHR macros) are included in the generation and INCHI is specified, the generation procedure assumes that the block-handling routine linkage editor control statements are included in the members specified in the INCHI and ORDHI operands.

```
[ORDINIT=( symbol1 [,symbol2] ... [,symboln] )]
```

Specifies the macro library members that contain linkage editor ORDER statements for user-written initialization code. This operand is valid only if INCINIT is specified and TYPESYS=OS is specified in the BUILD macro. This operand must be omitted if TYPSYS=DOS.

The members specified in this operand are overlaid once the initialization has been completed. If the user initialization code is not to be overlaid, specify the library members containing the ORDER statements in the ORDHI or ORDLO operands.

```
[ORDL2HI=( symbol1 [,symbol2] ... [,symboln] )]
[ORDL2LO=( symbol1 [,symbol2] ... [,symboln] )]
```

These operands specify the macro library members that contain linkage editor ORDER statements for object modules that are level 2 or level 3 code. If this generation is on a DOS/VS system, these operands are invalid and must be omitted.

ORDL2HI names the object modules that may reside anywhere in the 3705 controller storage. This operand is required if INCL2HI is specified and TYPSYS=OS is specified in the BUILD macro.

ORDL2LO names the object modules that must reside in the lower 64K of controller storage. This operand is required if INCL2LO is specified and TYPSYS=OS is specified in the BUILD macro.

Chapter 6. NCP Storage Estimates

This chapter is structured to aid in calculating storage estimates for an NCP operating in network control mode (TYPGEN=NCP) or both network control mode and emulation mode (TYPGEN=PEP).

The total storage estimate for the NCP is the sum of storage for the following individual categories:

- Base code
- User code
- Buffers
- Optional code
- Emulation program estimates (TYPGEN=PEP)

The NCP has many optional system functions and line/device support capabilities. There are also many optional tables and variable-length resource control blocks that require storage in the controller. All of these influence the total amount of controller storage required for your network control program. Select the options and other categories and place the appropriate storage requirement in the space provided. This storage estimate (minus buffer requirements) can then be used to determine the auxiliary storage needed for the OS/VS load module or the DOS/VS phase.

Network Control Mode Storage Estimates

This section describes the procedures for calculating storage estimates for the network control mode of an NCP (TYPGEN=NCP).

NCP Base Code

Determine the total estimates for base code from Figure 6-1 by selecting the basic components that make up your system. For example, a network consisting of a local controller with BSC or start-stop lines, a type 3 communication scanner base, synchronous data link control (SDLC) links, and IBM 3600 SDLC terminals with a type 4 channel adapter and without auto network shutdown would require:

- Base code of 36,842
- BSC or start-stop code of 7,274
- SDLC link support code of 43,408
- SDLC terminal support code of 6,626
- Type 4 channel adapter code of 3,073

for a total base code estimate of 67,223.

Enter the total from Figure 6-1 on line 1 of Figure 6-6.

	*Without Auto Network Shutdown**	*With Auto Network Shutdown**	
Base code - local controller			
Type 2 communication scanner only	34,318	36,228	________
Type 2 and 3 scanners	37,452	39,362	________
Type 3 scanner only	36,842	38,554	________
BSC/SS lines attached			
Type 2 scanner only	10,298	12,492	________
Types 2 and 3 scanners	13,052	15,246	________
Type 3 scanner only	7,274	9,468	________
SDLC lines attached			
Type 2 scanner only	14,024	14,902	________
Types 2 and 3 scanners	14,952	15,832	________
Type 3 scanner only	13,408	14,287	________
PU type 4 remote or cross-domain link (POLLED=YES)** *or* backup SSCP (SDLCST macro)	798	798	________
PU type 4 for secondary** cross-domain link (POLLED=NO)	N/A	2,578	________
Backup SSCP (Primary-secondary switch) (SDLCST macro)	N/A	534	________
Type 1 or type 4 channel adapter	3,073	3,754	________
Concurrent type 2/3 channel adapters	N/A	320	________
Type 2 or type 3 channel adapter	2,574	2,988	________
Second active type 4 channel adapter	N/A	357	________
Each additional active type 4 channel adapter	N/A	304	________

Figure 6-1. Base Code Estimates (Part 1 of 2)

	Without Auto Network Shutdown*	With Auto Network Shutdown*	
Base code - remote controller			
Type 2 communication scanner only	39,776	41,664	________
Types 2 and 3 scanners	43,542	45,430	________
Type 3 scanner only	42,944	44,622	________
BSC/SS lines attached			
Type 2 scanner only	10,298	12,492	________
Types 2 and 3 scanners	13,052	15,246	________
Type 3 scanner only	7,274	9,468	________
SDLC lines attached			
Type 2 scanner only	8,618	9,498	________
Types 2 and 3 scanners	9,988	10,894	________
Type 3 scanner only	8,468	9,352	________
Multiple links to local controller	0	515	________
Activity time-out	143	143	________
SDLC terminals	6,626	6,626	________
Type 2 physical unit			
Type 1 physical unit	536	536	________
3270 Models 11 and 12 SDLC	1,491	1,491	________
SDLC switched lines	5,200	5,300	________
PUDRPOOL or	6,054	6,054	________
LUDRPOOL	1,378	1,378	________
SDLC dial lines	3,724	3,798	________
Total base code requirement:			________

Note: Add in second-level (indented) items only if first-level item applies.

* Auto network shutdown is an option (ANS) on the BUILD macro. It includes the configuration restart facility.

** Must also include "SDLC lines attached.
ANS is required for cross-domain links.
ANS is required for concurrent channel adapters.

Figure 6-1. Base Code Estimates (Part 2 of 2)

User Code

This is the user-written code supporting control blocks and BHR code that you generate. Your level 5 code will be placed on a 2K boundary; therefore, you have a potential storage loss of (2K minus 1) bytes.

Calculate the number of bytes that will be added as a result of user-written code, plus 2,238 and enter the total on line 2 of Figure 6-6.

Buffers

After the NCP is generated and loaded into the controller, the NCP determines the total amount of available storage in the controller (the smaller of either the size specified in the MEMSIZE operand of the BUILD macro or the size of the installed storage). The NCP then determines the storage already used, computes the amount of storage left in the controller, and uses that remaining space for buffers. For example, an 80K-byte controller containing a 64K NCP will have 16K of buffer space built by the NCP.

Use the following formulas to determine if you have enough storage left to build the necessary buffers. The final result of these formulas indicates the amount of buffer storage you will need to handle your network's requirements 95 percent of the time.

The estimate for buffers will tend to be high in most cases, especially as the number of attached lines increases.

Calculating Buffer Storage Estimates for Start-Stop and BSC Lines

1. Buffers per block = (average block size for each line + 30)/(buffer size - 4). Round the total up to the next integer. (The buffer size should be the value specified in the BFRs operand on the BUILD macro rounded up to the nearest multiple of 4, plus 4.)

2. Calculate the number of buffers per line, *N*, as follows:

   For start-stop and BSC point-to-point lines,

   N = result of step 1.

   For start-stop and BSC multipoint lines,

   N = (result of step 1 x number of terminals on the line).

3. (N x buffer size) = approximate number of bytes of buffer storage needed for this line.

Repeat steps 1, 2, and 3 until all start-stop and BSC lines are accounted for. Enter the total below.

Calculating Buffer Storage Estimates for SDLC Links

The procedure for calculating buffer storage estimates for SDLC links (other than the communication link between a local and a remote communications controller) is found in Appendix C. If your NCP supports SDLC links, turn to that appendix to perform your calculations and then enter the total below.

Calculating Buffer Storage Estimates for Local/Remote and Local/Local Communication Links

The procedure for calculating the buffer storage estimates for local-remote and local-local communication links is found in Appendix D. If your NCP supports a local-remote or local-local communication link, turn to that appendix to perform your calculations and then enter the total below.

Calculating Total Buffer Storage Estimates

To get the total number of bytes of buffer storage required for your network, first add together the following:

Total for all start-stop and BSC lines	________
Total for all SDLC links	________
Total for all local-remote and local-local communication links	________
Size of one buffer (value specified in BFRS operand of BUILD macro)	________
Buffer storage required to contain the largest average block size in the network. (Average block size + 30)/(buffer size - 4); round up to next integer and multiply by buffer size	________
Total	________

After the NCP is loaded into the controller, any remaining storage is reserved as buffer space. However, a certain percentage of that buffer space, the percentage specified for the SLODOWN operand of the BUILD macro, is available only when the NCP is operating in slowdown mode. To assure that the NCP's responsiveness is not degraded when operating in slowdown mode, you should increase the amount of buffer storage. Calculate the total number of buffers needed (total buffer storage calculated above divided by buffer size). Multiply the result by (100/(100 - R)), where R is the slowdown percentage; then round the product up to the nearest integer and multiply by the buffer size.

If you are calculating the estimates for a remote communications controller, you must allow at least enough storage for the load program to be loaded into the controller. (This space is used for buffers after the NCP is loaded.) Use whichever value is larger: (1) the result of your buffer storage calculations or (2) the amount of storage needed for the load program, which is 8,192 bytes.

Enter total buffer storage, including slowdown contingency on line 3 of Figure 6-6.

Example of Calculating Buffer Storage Estimates

Calculate the amount of buffer storage needed for the following network. The buffer size is 64 (BFRS specified as 60) and SLOWDOWN is specified as 12.

Line 1 Start-stop, point-to-point line with average block size of 200.
Line 2 Start-stop, multipoint line with 10 terminals and average block size of 80
Line 3 4800 bps, half-duplex, multipoint SDLC line

Interactive portion of line load:
- 1 cluster node with 3 logical units (75% of traffic)
- 1 terminal node (25% of traffic)

Batch portion of line load:

2 cluster nodes with 4 logical units each (80% of traffic)
1 terminal node (20% of traffic)

Line 4 7200 bps, half-duplex, SDLC local-remote communication link

For line 1:

1. (200 + 30)/(64 - 4) = 3.8 (round up to 4)
2. N = 4
3. (4 x 64) = 256 bytes of buffer storage

For line 2:

1. (80 + 30)/(64 - 4) = 1.8 (round up to 2)
2. (N = (2 x 10) = 20
3. (20 x 64) = 1280 bytes of buffer storage

For line 3:

See Appendix C for example of calculations.

For line 4:

See Appendix D for example of calculations.

Total for lines 1, 2, 3, and 4:

Assume that all lines are attached to the localy communications controller. 256 + 1280 + 13,376 + 1920 = 16,832 bytes

Storage needed for largest block size:

Largest average block size is 400 (for batch portion of line 3).
(400 + 23)/(64 - 4) = 7.05 (round up to 8)
8 x 64 = 512 bytes

Total buffer storage required:

16,832 + 64 + 512 = 17,408 bytes

Number of buffers required:

17,408/64 = 272 buffers

Number of buffers needed with SLODOWN specified as 12:

272 x (100/(100 - 12)) = 272 x 1.1363 = 309.1 (round up to 310)

Total buffer storage required with slowdown contingency:

310 x 64 = 19,840 bytes

Optional System Functions

This category is composed of:

- Control command support (dynamic control facilities)
- Block-handling options (local controller)
- Address trace
- Critical situation notification
- Online test
- Supervisor abend facility

- Time sharing options
- BSC/SDLC path function
- Channel support
- Programmed resource service functions

The total storage requirement for this category is the total, in bytes, for all optional system functions included in your program.

Select each function that you have included and enter the indicated amount in the space provided. The total for this category is the sum of the individual estimates.

Control Command Support (SYSCNTRL Macro)

See Figure 6-2 to determine storage estimates for SS/BSC device control command support.

Reset Immediate (RIMM) TCAM/VTAM	*					
Reset Conditional (RCOND) TCAM/VTAM	*	*				
Reset at End of Command (RECMD) VTAM	*					
Change Session Limit (SESSION) VTAM			*	*		
Change Service-Seeking Pause (SSPAUSE) VTAM			*	*		
Change Negative Poll Limit (NAKLIM) VTAM			*	*		
Set Destination Mode (MODE) TCAM/VTAM				*	*	
Change Device Transmission Limit (XMTLMT) TCAM				*		*
Insert ✓ in any column containing an asterisk if it is adjacent to a command you employ						
TOTAL	1843	236	78	52	164	120

Figure 6-2. Control Command Storage for OS/VS and DOS/VS TCAM Users

To use Figure 6-2, select the commands used in your NCP. For example, *Change Session Limit* has an asterisk in two boxes. Place a check mark in the box at the bottom of each asterisked column. (There should be a check mark above 78 and 52.) Now select another command, for example, *Change Negative Poll Limit.* This command has an asterisk in the same columns as *Change Session Limit.* Ignore this command and proceed to the next command.

After placing check marks in the appropriate boxes, write the byte count into the box below the checked column and add the numbers.

Enter total from Figure 6-2 here: __________

Additionally, if your network control program includes any of the following options, enter the appropriate number in the space provided.

Display line status (LNSTAT):	62	__________
Change speed (SPDSEL):	240	__________
Physical disconnect (ENDCALL):	76	__________

Change BH set association (BHSASSC):	182	________
Replace line session initiation information (SESINIT):	1,428	________
Replace device session initiation information (DVSINIT):	398	________
If SESINIT or DVSINIT and BSC lines, add:	480	________
If no BHRs, add:	126	________
If no BHRs, and ABEND=YES, add:	82	________
Switched network backup (BACKUP):	2,592	________
Total-control command support:		________

Block-Handling Routine (BHR) Support (SS/BSC only)

If your NCP has any of the following, enter the appropriate number in the space provided.

BHR base support:	1,320	________	
ABEND=YES (BUILD macro):	128	________	
Backspace edit (EDIT macro):	258	________	
Date/time insertion (DATETIME macro):	554	________	
Number of BHSET macros times 18:		________	
Number of EDIT macros times 14:		________	
Number of DATETIME macros times 10:		________	
Number of UBHR macros times 10:		________	
Total BHR support:			________

Select the following applicable support functions and place the appropriate figure in the space provided.

Address Trace (BUILD Macro)

Address trace (TRACE):	378	________

Critical Situation Notification (BUILD Macro)

For critical situation notification, enter 272 plus the number of characters in the critical situation message and header (CRITSIT, CSMSG, CSMSGC, CSMHDR, CSMHDRC):	________

Online Test (BUILD Macro)

For online test (OLT) for a system:

SDLC only		
Type 2 communication scanner only	5,231	________
Types 2 and 3 scanners	6,188	________
Type 3 scanner only	5,751	________
BSC or SS with or without SDLC		
Type 2 communication scanner only	8,182	________
Types 2 and 3 scanners	9,139	________
Type 3 scanner only	8,702	________
With auto network shutdown	454	________
Total for online test:		________

Supervisor Abend Facility (BUILD Macro)

Supervisor abend facility (ABEND):	1,020	________

Time Sharing Option for start-stop (LINE Macro)

Carriage delay and monitor:	1,190	________

BSC/SDLC Path Function (LU Macro)

BSC/SDLC path support:	4,140	________

Channel Adapter Optional-Features Support (HOST Macro)

Channel adapter optional-features support:

Type 1 channel adapter:

Attention time-out (TIMEOUT):	26	________
Attention delay (DELAY):	68	________
Status modifier (STATMOD):	60	________
Erase (BUILD/ERASE):	60	________
PEP extension:	126	________
CA trace - 34 bytes per entry, plus 207:		________

Type 2 or 3 channel adapter optional features:

Attention time-out (TIMEOUT):	126	________
Status modifier (STATMOD):	24	________
Erase (BUILD/ERASE):	68	________
Attention delay (DELAY):	68	________
CA trace - 34 bytes per entry, plus 207:		________

Control Word (CW) area:

If *padding* (BFRPAD) is used in transfers to the host, CW area equals:

4A(B + 3)+4C ________

where

A = Host buffer units (MAXBFRU)

B = Host unit size (UNITSZ) divided by controller buffer size (BFRS) rounded down to the nearest whole number

C = Controller buffer allocation for input (INBFRS)

If *no padding* (BFRPAD) is used in transfers to the host, CW area equals·

4A(B + 2)+4C ________

Note: If there are concurrent type 2 or type 3 CAs, calculate the CW area for each and then multiply the larger number by 2. This calculation will give you the actual CW area required.

Total channel adapter optional features: ________

Programmed Resource Service Functions

If one or more programmed resoruce is defined, add 1400 bytes ________

Enter total optional system functions on line 4 of Figure 6-6.

Optional Start-Stop and BSC Line/Device Support

The options described in this section apply to:

- Multipoint lines
- Point-to-point lines
- Switched lines
- Nonswitched lines
- Miscellaneous line/device support

If your system has any of the following support, enter the appropriate number in the space provided.

BSC Line Control (GROUP Macro)

Binary synchronous (LNCTL=BSC):	2,928	________	
Point-to-point (POLLED=NO):	410	________	
Multipoint tributary (TADDR):	700	________	
Online test in system (OLT):	216	________	
Transparent ITB mode (XITB):	520	________	
BSC ASCII code (CODE)	570	________	
Multipoint tributary (TADDR):	32	________	
Online test (OLT):	32	________	
Total BSC:			________

Start-Stop Line Control (GROUP Macro)

Start-stop (LNCTL=SS)	2,160	________	
Point-to-point (POLLED=NO):	782	________	
Longitudinal redundancy checking (FEATURE=LRC):	76	________	
Multipoint (POLLED=YES):	128	________	
Total Start-Stop:			________

BSC and Start-Stop Multipoint Lines (LINE Macro)

Multipoint lines:	1,420	________	
Type 2 communication scanner:	678	________	
Total multipoint:			________

BSC and Start-Stop Point-to-Point Lines (LINE Macro)

Switched (DIAL=YES):	88	________	
Nonswitched (DIAL=NO):	68	________	
Total point-to-point:			________

BSC and Start-Stop Switched Lines (GROUP Macro)

Switched lines (includes manual dial):		
Type 2 communication scanner ony	1,550	________
Types 2 and 3 scanners	1,709	________
Type 3 scanner only	1,631	________
CALL=INOUT or CALL=OUT:	1,560	________
Ring indicator mode (RING):	38	________
IDLIST macro		
If IDSEQ=(chars):	800	________
If IDSEQ=(chars, term name):	1,110	________
Multiple Terminal Access (TERMINAL/TERM=MTA)		
MTA call-out: (LINE/CALL)	42	________

If MTA call-out only (no call in):	146	______	
MTA call-in:	1,540	______	
MTA and block handler support (BHSET on TERM, COMP, or CLUSTER macro):	52	______	
Total switched lines:			______

Nonswitched Lines (GROUP Macro)

Nonswitched lines:	182	______	

Miscellaneous Line/Device Support

General poll (excluding 2972 with batch message input feature) (GPOLL=chars):	1,530	______	
2972 terminals (without batch message input feature):	56	______	
Both 3270 and 3740 BSC terminals:	460	______	
3270 BSC terminals only:	134	______	
2740-II terminals:	224	______	
83B3 or 115A terminals:	600	______	
TWX terminals:	472	______	
WTTY terminals:	676	______	
Both TWX and WTTY terminals:	138	______	
Buffered receive feature (BFRDLAY):	162	______	
ENDTRNS=EOB (TERMINAL or COMP):	52	______	
2741/1050 break feature:	110	______	
Total miscellaneous line/device support:			______

Enter total optional line/device support on line 5 of Figure 6-6.

Tables

If your NCP uses any of the following tables generated by the FEATURE operand, the CODE operand, or the TADDR operand, enter the appropriate number in the space provided. Usage is determined by whether code translation or command decoding is necessary.

For translate tables (CODE operand on the LINE macro), enter the number from Figure 6-3.

SS	BSC	Table	Bytes	
•		1050 with KATAKANA	564	________
•		1050/2740/2741 BCD[1] (normal)	564	________
	•	1050/2740/2741 BCS (modified)	564	________
•		1050/2740/2741 EBCD	564	________
•		2740/41 correspondence (normal)	564	________
	•	2740/41 correspondence (modified)	565	________
•		WTTY with ITA2	564	________
•		WTTY with ZSC3	564	________
•		TWX (normal)	564	________
	•	TWX (modified)	282	________
•		83B3/115A	564	________
	•	EBCDIC	142	________
	•	ASCII	564	________
			Total	________

[1] BCD is required if call-in MTA is used in your system, even if none of your terminals require BCD.

Figure 6-3. Translate Decode Table

For state address tables (FEATURE operand on TERMINAL macro), enter the number from Figure 6-4: ________

SS	BSC	SDLC	Table	Bytes	
•			S-S (but not 2740-I) with checking	72	________
•			S-S (but not 2740-I without checking	72	________
•			2740-I with checking	72	________
•			2740-I without checking	72	________
•			WTTY	72	________
•			TWX	72	________
•			83B3/115A	72	________
	•		EBCDIC	72	________
	•		ASCII	72	________
		•	Primary station	18	________
		•	Secondary station	18	________
				Total	________

Figure 6-4. State Address Table

For command decode tables, enter the number from Figure 6-5: ________

SS	BSC	Table		
•		2741	18	________
•		2740-II with checking	18	________
•		Multipoint 2740-II without checking or 2740-I with station control and without checking	18	________
•		2740-I with transmit control	18	________
•		2740-I with transmit control and checking	18	________
•		2740-I contention with checking	18	________
•		TWX 33/35	18	________
•		WTTY	18	________
•		2740-I basic	18	________
•		2740-I multipoint	18	________
•		Multipoint 83B3/115A	18	________
	•	Point-to-point contention (BSC or SS)	18	________
	•	Multipoint control station (BSC or SS)	18	________
	•	Multipoint tributary (BSC or SS)	18	________
			Total	________

Figure 6-5. Command Decode Table

For MTA tables, calculate for the following macros:

MTALCST - 100 bytes for the first macro, plus ________
18 bytes for each additional macro ________
MTALIST - 1 byte per specified terminal type ________
MTAPOLL - 2 bytes per set of polling characters ________
MTATABL - 22 bytes per macro ________

Total for MTA Tables: ________

For line types (one or both, if applicable):

Switched lines: 450 ________
Nonswitched lines: 330 ________

Total line types: ________

For each group of switched dial-out lines (DIALSET):

18 + 4 (number of line entries) + 4 (if DIALALT is specified) + 4 (if the preceding DIALSET specified this DIALSET as its DIALALT) ________

Total for dial sets. ________

For each IDLIST macro enter,

Number of ID characters ________ + 6: ________
If device association, add 2 per entry: ________
If IDSEQ is required, add the number of bytes for CUID and TWXID characters: ________

Total for IDLIST: ________

For multiple host support:

Number specifed on MAXSSCP ________
x32 (without SNA) or
x60 (with ANS)= ________

Number of subareas specified
on HOST macro __________ x 14 = __________

For address trace tables (TRACE):
Number of units __________ x 18 = __________ + 34 = __________

For Control command lookup tables:
Number of optional Control commands __________ x 4= __________
(from "Control-Command Support" above)

For online test (OLT):

Switched lines:	52	__________
Nonswitched lines:	32	__________
Total online test:		__________

If TYPE=TYPE2, select the number of CSB macros coded and enter the appropriate figure:

1 CSB macro	192 bytes	__________
2 CSB macros	512 bytes	__________
3 CSB macros	768 bytes	__________
4 CSB macros	1,024 bytes	__________
Total for CSB macros:		__________

For traced lines:

For each line traced:	5	__________
Type 2 scanner only:		
If all lines are half-duplex,	41	__________
but if at least one line is duplex:	78	__________
Type 3 scanner (with or without a type 2 scanner):		
If all lines are half-duplex,	87	__________
but if at least one line is duplex:	170	__________
Total for traced lines:		__________

Enter total for optional tables on line 6 of Figure 6-6.

Resource Control Blocks

This category of optional code is composed of:

- Physical line groups (GROUP macro)
- Lines (LINE macro)
- Stations (TERMINAL, COMP, CLUSTER, LU, PU, LUPOOL, LUDRPOOL, and PUDRPOOL macros)

The total storage estimate for this category is the total, in bytes, for all resource control blocks included in your program.

Physical Line Groups (GROUP Macro)

Calculate the following for each physical line group:

Number of 83B3/115A groups x 41:	__________
Number of WTTY groups x 45:	__________
Number of start-stop and BSC groups x 58:	__________
Number of SDLC groups x 20:	__________
Total physical line groups:	__________

Lines (LINE Macro)

Multiply the number of point-to-point BSC or S/S nonswitched lines by 200 (DIAL=NO, POLLED=NO) ________

Multiply the number of point-to-point BSC nonswitched lines used for the BSC/SDLC path function by 206 (DIAL=NO, POLLED=NO, BSC/SDLC path conversion BHR associated with the terminal) ________

Multiply the number of multipoint BSC or SS lines by 232 (DIAL=NO, POLLED=YES) ________

For each switched BSC or SS line, add 208 (+ 4 bytes if the line is designated CALL=IN or CALL=INOUT). Add the per-line estimates together and enter the total: ________

Half-duplex SDLC line, 164 for each line plus 4 bytes per line if POLLED=NO: ________

Duplex SDLC line, 272 for each line plus 8 bytes per line if POLLED=NO: ________

Total lines: ________

Stations (TERMINAL, COMP, CLUSTER, LU, PU, and LUPOOL Macros)

SDLC Stations:

For PU type 1 or PU type 2 macros:

On nonswitched lines, multiply the number of PU macros times 148 bytes: ________

On switched lines, multiply the number of PU macros times 148 bytes: ________

For LU macros: ________

Associated with PU type 1, multiply the number of LU macros times 112 bytes: ________

Associated with PU type 2, multiply the number of LU macros times 96 bytes: ________

Associated with BSC/SDLC path function PU type 2, multiply the number of PU macros times 136 bytes: ________

For LUPOOL macro:

Multiply the number of logical units specified times 102 bytes: ________

For PU type 4 (PU macro):

Multiply the number of PU type 4 macros;
with POLLED=YES times 120 bytes: ________
with POLLED=NO times 132 bytes: ________

For each SDLCST macro add 16 bytes: ________

For dynamic reconfiguration:

Multiply the number of RVT extensions times 8:
(RESOEXT operand on the BUILD macro) ________

For LUDRPOOL macro:

Multiply 102 times the number of type 1 PUs: ________

Multiply 88 times the number of type 2 PUs: ______

For PUDRPOOL macro:
Multiply the value in the NUMBER operand times 148: ______

For programmed resources add:
52 bytes for each line: ______
56 bytes for each physical unit: ______
52 bytes for each logical unit: ______
44 bytes for each additional LU-LU session relative to this secondary logical unit: ______

BSC/Start-Stop Stations:
For each TERMINAL, COMP, or CLUSTER macro, enter 82 bytes: ______
Plus the following amounts as applicable:

If DIRECTN=IN or INOUT:	12	______
If switched backup line exists:	4	______
If switched CTERM=YES terminal:	4	______
If IDSEQ is not equal to NONE:	4	______
If switched dial-out terminal, add the number of dial digits:	7	______
If BHEXEC=PT3 (TERMINAL macro):	22	______
If GPOLL=chars (CLUSTER macro):	18	______
If the station is on a multpoint line:	10	______
add the number of polling characters and the number of addressing characters:		______

Total for all stations: ______

Enter total for resource control blocks on line 7 of Figure 6-6.

Total Storage Estimates for NCP Mode

Add the individual totals in Figure 6-6 to arrive at the total network control program storage estimates. If you are operating in network control mode only (TYPGEN=NCP), this is your total estimated storage.

Total network control program storage estimate: ______

If you are operating in a partitioned emulation programming environment (TYPGEN=PEP), carry the total for NCP mode forward and add it to the subtotal for the emulation lines to arrive at a total control program storage estimate.

Line	*Code Description*	*Storage*
1	Base NCP code (from Figure 6-1)	______
2	User-written code	______
3	Buffer storage requirement	______
4	Optional system functions	______
5	Optional line/device support	______
6	Optional tables	______
7	Resource control blocks	______
	Total NCP Storage Estimate	______

Figure 6-6. Total Storage Estimate for Network Control Mode

Emulation Mode Storage Estimates

This section describes the procedures for calculating storage estimates for the emulation mode portion of the NCP (TYPGEN=PEP). The amount of storage required depends on:

- The number and types of terminals
- The type of communication scanner used
- The type of communication line control used

Determine the total amount of storage for the emulation portion of ACF/NCP/VS by adding the individual storage estimates for:

- Base code
- Code needed to support specific configurations and options
- Emulation program control blocks and tables

Emulation Base Code

	Type 1 CA	Type 4 CA	
Emulation base code (including code for the loader). Enter on line 1 of Figure 6-8.	5,738	11,560	__________

Type 1 Communication Scanner Support

Type 1 scanner support for (select one):

	Type 1 CA	Type 4 CA	
Binary synchronous lines only:	1,432	N/A	__________
Start-stop lines only:	1,720	N/A	__________
Binary synchronous and start-stop:	1,858	N/A	__________
Total type 1 scanner support and enter on line 2 of Figure 6-8:			__________

Note: There are no type 2 or type 3 scanner storage estimates similar to the type 1 scanner support.

Start-Stop Terminal Support

This code is required to support the operation of start-stop terminal types (choose the applicable options). For example, for a network consisting of IBM 2260 and 1050 terminals, select IBM type I, II and III for a total of 2,368 bytes.

IBM type I and II	- IBM 1030, 1050, 1060, 2740, 2741
IBM type III	- IBM 2260/2848, 2265/2845
TTY type I	- AT & T 83B2, 83B3, WU 115A line control
TTY type II	- TWX 33/35 line control

	Type 1 CA	Type 4 CA	
IBM type I and II:	2,032	2,136	__________
IBM type III:	2,128	2,176	__________
IBM type I, II, and III:	2,368	2,472	__________
IBM and TTY type I and II:	2,632	2,736	__________
TTY type I and II:	2,408	2,464	__________

TTY I and II and IBM type III:	2,744	2,800	________
TTY I and II and IBM I, II and III:	2,968	3,072	________
If DELAY, add:	88	88	________

Total start-stop terminal support and enter on line 3 of Figure 6-8: ________

Binary Synchronous Terminal Support

This code is required to support synchronous line control (choose the applicable option):

	Type 1 CA	Type 4 CA	
Type 2 scanner only			
EBCDIC only:	2,672	3,664	________
USASCII only:	2,320	3,304	________
Both:	2,928	3,920	________
Types 2 and 3 scanners			
EBCDIC only:	N/A	5,008	________
USASCII only:	N/A	4,648	________
Both:	N/A	5,264	________
Type 3 scanner only			
EBCDIC only:	N/A	2,704	________
USASCII only:	N/A	2,704	________
Both:	N/A	2,704	________

Total binary synchronous support and enter on line 4 of Figure 6-8: ________

Service Aid Options

Three service aid options (line trace, dynamic dump, and panel test) that require controller storage may be included for emulation mode. Calculate the appropriate requirements and enter the total on line 5 of Figure 6-8.

Line Trace

Choose the applicable option:

	Type 1 CA	Type 4 CA	
Type 1 scanner:	1,664	N/A	________
Type 2 scanner:	1,704	N/A	________
Type 3 scanner with or without type 2 scanner:	N/A	1,688	________

If you include the line trace option, you should provide enough additional storage to trace one line entry for every line being traced. An entry is either a level 1, level 2, or level 3 interrupt, and each entry requires one or more of the following:

For a type 3 scanner,
  (8 x (3 + n) x number of level 2 entries): ________
    where n is 0 if no data transfer, or *n* is the number of 8-byte entries
    sufficient to contain the type 3 scanner buffer
    (1 ID byte plus 7 data bytes per entry)

Plus, for type 1 or 2 scanners,

(16 x number of level 2 entries): ________
(8 x number of other type entries): ________

Dynamic Dump

Dynamic Dump (DYNADMP):

	Type 1 CA	Type 4 CA	
Using native subchannel only:	1,008	1,080	________
Using any other subchannel:	1,040	1,222	________

Panel Test

Panel Test (TEST):

	Type 1 CA	Type 4 CA	
Type 1 scanner without auto-call:	1,184	N/A	
Type 1 scanner with auto-call:	1,488	N/A	
Type 2 scanner without auto-call:	1,128	3,344	
Type 2 scanner with auto-call:	1,416	3,344	
Types 2 and 3 scanners with or without auto-call:	N/A	3,824	________
Type 3 scanner only with or without auto-call:	N/A	2,288	________

Total test option: ________

Total Service Aid Options

Enter the total service aid requirements on line 5 of Figure 6-8: ________

Character Control Block

Emulation mode requires a character control block for each line in your configuration. (Storage for the character control block is required regardless of the communication scanner type used.) The character control block contains current information, primarily on the physical operation of the line. Figure 6-7 shows the character control block storage estimates for each line and its possible options. Select the size applicable for each line.

Every character control block is associated with a particular subchannel address; however, every subchannel address within the range of the high and low subchannel addresses configured does not necessarily apply to a line. Any subchannel address within the range of the high and the low subchannel address not associated with a line is a *skipped subchannel address*. Note, however, that each MSLA subchannel is associated with a line, and its dummy character control block's size is 12 bytes (instead of the 10 bytes for a skipped subchannel address that is not associated with any line.) Skipped subchannel addresses require dummy character control blocks, each of which occupies 10 or 12 bytes of storage.

Calculate storage estimates for the character control blocks as follows:

1. Determine the storage estimate for each line in emulation mode from Figure 6-7:
2. Add all individual line estimates: ________

	BSC lines Bytes		Start-Stop lines Bytes	
Type 4 channel adapter and Type 3 communication scanner	42 + (2 x CS3 buffer size) [CS3 buffer size may be 4, 8, 16, 32, 64, 96, 128, 160, 192, or 224]		N/A	
	Type 1 CA	Type 4 CA	Type 1 CA	Type 4 CA
Type 1 or type 4 channel adapter and type 2 communication scanner high priority (CHNPRI=HIGH on the LINE macro) and additional buffer space (OPCSB2=YES on the BUILD macro).				
High priority without dual communications interface	58	60	N/A	N/A
High priority with dual communications interface	60	62	N/A	N/A
Type 1 and type 4 channel adapter and type 2 communication scanner				
Without dual communications (DUALCOM) interface, without station select[1]	42	44	38	40
With dual communications interface, with or without station select	44	46	N/A	N/A
With station select, with or without dual communications interface	46	48	N/A	N/A

[1]Station select is indicated when you specify TADDR on the LINE macro.

Figure 6-7. Character Control Block Storage Values

3. Compute the dummy character control blocks:
   For a type 1 channel adapter, multiply the number of skipped subchannels by 10 bytes: __________
   For each type 4 channel adapter,
   multiply the number of skipped subchannels by 10 bytes: __________
   multiply the number of unassigned multiple subchannel line access (MSLA) subchannels by 12 bytes: __________

Total character control block estimates and enter on line 6 of Figure 6-8: ________

Example:

A communications controller with a grouping of subchannel addresses between X'00' and X'3F' has 64 subchannel addresses for lines. Assume these lines are start-stop, and 54 of the addresses are associated with lines and 10 are not. The storage for the character control blocks is calculated as follows:

1. The storage estimate for start-stop lines is 38 bytes.
2. Since all lines are the same, 54 subchannels x 38 bytes = 2,052.
3. There are 10 skipped subchannels. 10 subchannels x 10 bytes = 100.
4. 2.052 (from step 2) + 100 (from step 3) = 2,152 bytes.

Tables

Emulation mode requires the following three tables:

- Channel vector table
- Line vector table
- Line group table

Calculate the requirements for these tables and enter the total on line 7 of Figure 6-8.

Channel Vector Table

The channel vector table (CHVT) translates the multiplexer subchannel address to the corresponding communication scanner line interface address.

The CHVT is a variable-length table. Its size depends on the highest subchannel address assigned to the program.

Determine storage estimates as follows:

For a type 1 channel adapter,
10 bytes + ((high-low subchannel address) x 2)
+(number of scans x 2) __________

For each type 4 channel adapter,
112 bytes + ((high-low subchannel address) x 2) __________

Example:

Assume that you have a type 1 channel adapter where the high subchannel address is X'54' and the low subchannel address is X'22' and that you have two communication scanners installed. (Converting to decimal: X'54' = 84; X'22' = 34).

Using the formula:

10 bytes + ((84-34) x 2 bytes) + (2 x 2 bytes) = 114 bytes of storage required.

Line Vector Table

The line vector table is used to index to the corresponding character control block once a line interface address is known. It also contains the fields and pointers required by the scanner portion of the control program. The length of the line vector table depends on the highest line interface address specified. Calculate storage for the line vector table as follows:

For type 2 scanner with type 1 channel adapter:
(Highest line address in decimal + 1) x 2: __________

For a type 2 and/or a type 3 scanner with a type 4 channel adapter:

((Highest line address in decimal + 1) x 2) + 16 ________

Total for line vector table ________

Assume you are using the highest possible address on a type 1 communication scanner—X'03F' (63 in decimal notation).

Using the formula:

(63 + 1) x 16 = 1,024 bytes of storage.

Assume you are using the highest possible address on a type 2 communication scanner—X'05F' (95 in decimal notation) and that you have a type 1 channel adapter.

Using the formula:

(95 + 1) x 2 = 192 bytes of storage.

Line Group Table

The line group table (created by GROUP macro) contains the parameters a group of lines have in common.

Allow 8 bytes for every GROUP macro. ________

Total Table Requirements

Enter the storage estimate for all emulation mode tables on line 7 of Figure 6-8. ______

Line	*Code Description*	*Storage*
1	Base emulation code	________
2	Type 1 communication scanner	________
3	Start-stop terminal support	________
4	BSC terminal support	________
5	Service aid options	________
6	Character control blocks	________
7	Tables	________
	Total Emulation Storage Estimate	________

Figure 6-8. Total Storage Estimate for Emulation Mode

Total Storage Estimates for Emulation Mode

Add the individual totals in Figure 6-8 to arrive at the total emulation storage estimates. This total must be added to the NCP mode estimate to determine the estimated storage requirement for an NCP with the PEP extension.

Total emulation storage estimate: ______

Total NCP Storage Estimates (TYPGEN=PEP)

Subtotal network control program: ________

Subtotal emulation program: ________

For type 1 channel adapter with a type 2 scanner, *subtract*; 670 ________

For a type 4 channel adapter with a type

2 and/or a type 3 scanner, *subtract*	856	______	
If NCP shares the type 1 or type 4 channel adapter, *add*;	100	______	
Subtotal adjustments:			______
Total adjusted NCP storage:			______
For NCP line mode switching, add:	544	______	
Line trace for emulation lines, add:	160	______	
Panel-initiated line test, add:	144	______	
Total NCP options:			______

Total storage estimate for a network control program with the partitioned emulation programming extension: ______

Chapter 7. Planning for Performance for NCP

This chapter describes the NCP generation operands that affect performance of the communications controller. Performance in the controller is largely determined by the proper choice of operands that control the following system functions:

- Data transfer between the host processor and the controller over the channel
- Processing of data within the controller
- Data transfer between the controller and terminals over communication facilities

Data Transfer over the Channel

The number of times the network control program has to be interrupted to transmit or receive data is an important consideration when specifying the buffers in both the communications controller and the host processor.

The network control program must know:

- How much data the teleprocessing access method can accept in one continuous transmission
- How many buffers to allocate in the controller for each data transfer from the host

The host access method's buffer quantity and size determines the maximum amount of data that can be transferred across the channel from the controller on a single host Read command. The network control program operands concerned with this facility are MAXBFRU, UNITSZ, and BFRPAD on the HOST macro. MAXBFRU is the number of buffer units that the access method will allocate for receiving a single data transfer from the NCP, and UNITSZ is the size of each buffer unit (in bytes) used by the access method. BFRPAD is the number of bytes included for use by TCAM or VTAM. Data can be transferred to the host as it becomes available, or it can be blocked and transferred after a time interval (specified in the DELAY operand) has expired.

Channel Attention Delay (Local Only)

The DELAY operand on the HOST macro specifies the interval, to the nearest tenth of a second, that the network control program will delay sending an Attention signal to the host processor after data has become available. This delay ensures, if specified correctly for your network, that the interrupts to the host processor are kept to a minimum.

As the network control program begins to fill its buffers, the delay feature waits for the specified period before raising an Attention to the host. If the amount of data is sufficient to fill the buffers allocated by the host processor, the Attention interrupt will be presented before the delay count has been reached.

Data Collection Applications

In data collection applications, terminal operators usually supply data to the host processor without waiting for a text reply.

For example, the transactions that produce records in a bank can be gathered into channel transfer units to be shipped to the host for processing later.

Because controller delays are not usually critical in this application, a high channel attention delay would probably be sufficient. This delay allows the network control program buffers to fill, but it does not allow excessive time to expire before their contents are transferred to the host processor. The delay is overridden if the buffers fill (as a result of message traffic intensity) before the specified time runs out.

Conversational Applications

A conversational (inquiry/response) application usually involves terminals interacting with an application program running in the host processor. Response time at the terminal is important; therefore, the delay feature should be short, for example, approximately 0.1 to 0.5 second.

There is a need to balance the load (as defined by percent utilization), the buffer size, and the attention delay for each application. For example, a heavily loaded NCP with a small host access method buffer size and an insufficient number of host access method buffers will be continuously interrupting the host with Attention signals. Even a long attention delay will constantly be overridden in this situation.

Specifying DELAY, UNITSX, BFRPAD, and MAXBFRU at Generation Time (Local Only)

Choose the most appropriate value for DELAY, UNITSZ, BFRPAD, and MAXBFRU by performing an analysis of the characteristics of your network and application to determine the average number of bytes per message and the average number of messages per second.

For example, assume an average message size of 25 bytes. Multiply the average by 1.5 to obtain a buffer unit size that will handle approximately 75 percent of the traffic without wasting buffer space. Add the required buffer pad to the product; the value is now 37 plus 30 for the buffer pad, or 67 bytes. Specify UNITSZ (the size of a TCAM buffer) as 67.

Assume also, an attention delay of 0.8 seconds (800 milliseconds) is acceptable for your application; specify DELAY as 0.8.

Assume an average of 21 messages per second.

Delay times messages per second = MAXBFRU (the number of host buffers); that is, 0.8 X 21 = 16.8 (rounded up to 17).

The above calculations yield the following operand values:

UNITSZ=67
MAXBFRU=17
DELAY=0.8

The values of UNITSZ and MAXBFRU operands, once determined, must also be used in the buffer specifications for the access method to permit the most efficient utilization of the host and transfer of data over the channel.

Subchannel Service Priorities for TYPGEN=PEP

Subchannel service priorities are those priorities the communications controller services internally in emulation mode. They are assigned at generation time, for TYPGEN=PEP, by coding CHNPRI=NORMAL or HIGH on the LINE macro.

When the highest speed line is less than or equal to 9,600 bps, all lines may be assigned to the NORMAL priority; however, performance is about 1.0% better if they are all assigned to the HIGH priority.

When the highest speed line is greater than 9,600 bps, all lines of speed less than or equal to 9,600 bps should be assigned to the NORMAL priority, and all lines greater than 9,600 bps must be assigned to the HIGH priority, with the following exception. When the highest speed line is greater than 19,200 bps (20,400 bps for non-U.S.A.), the 19,200 or 20,400 bps line and lower speed lines should be assigned to the NORMAL priority, and the higher speed lines should be assigned to the HIGH priority.

Device Priority on the Byte Multiplexer Channel

The communications controller is an "overrunnable" device while operating in the partitioned emulation programming extension mode. It should have the highest priority on the byte multiplexer channel; that is, the controller should be the first device to secure the *select out* signal.

When multiple communications controllers are placed on the same channel, the controller with the highest speed lines and the most heavily used lines should be positioned to secure the *select out* signal first on the channel.

Communications Controller Performance

Local communications controller performance is affected by the size and frequency of data transfers from the host processor. The controller needs sufficient buffer space to support these transfers. The INBFRS operand of the HOST macro specifies the number of controller buffers initially allocated for each data transfer to be received from the host processor.

When estimating a value for INBFRS, consider two factors:

1. If the size of a data transfer consistently exceeds the allocated buffer space, the network control program supervisory routine is frequently interrupted to provide more buffers for the excess data in the block. As the proportion of time the network control program spends in allocating buffers increases, supervisory service requirements of the network control program increase and performance may suffer.
2. If the amount of data received is consistently less than the allocated buffer space, many buffers are not used. Although the unused buffers are eventually used for receiving the next data transfer, their absence from the buffer pool lowers the overall efficiency of buffer utilization.

In choosing a value for INBFRS, you should strike a balance between possible degraded network control program performance due to excessive demands on the supervisory routine, and unnecessary over-allocation of buffers.

The Performance Measurement Facility which generates statistics on buffer utilization can aid you in determining the most efficient values for buffer related operands. This facility is a combination of NCP and the 3705 hardware cycle utilization counter (CUC) that generates and stores statistic on the number of NCP free buffers. See the *IBM 3704 and 3705 Communications Controllers Principles of Operation* for detail on the operation of the CUC and how it relates to performance measurement.

The statistics accumulated by the performance measurement facility are stored in a dummy data buffer and available only through the 3705 control panel. See the *Guide to Using the IBM 3705 Communications Controller Control Panel* for information about how to use this facility.

Preventing a Monopoly of Buffers for BSC and Start-Stop Stations

All buffer requests in both the local and remote controllers are filled from a single pool, and no stations should monopolize the supply to the exclusion of other stations.

You can prevent buffer monopolization with the TRANSFR and CUTOFF operands on the LINE macro. A terminal may have a slow data rate or large messages; if so, the network control program can control reception of data from the station by limiting the number of buffers filled before sending the data to the host processor. The TRANSFR operand limits the number of buffers that a station on a line can fill to send to the host in a single transfer; this limited number of buffers is called a sub-block. (A sub-block is a logical group of buffers that does not contain a complete message.)

The CUTOFF operand limits the total number of sub-blocks a station on a line can send as the result of a single host Read command.

The network control program determines the maximum value for the TRANSFR operand based on the maximum size of the channel transfer unit. (A channel transfer unit is the amount of data transferred to or from the host processor by a single start I/O.) To improve performance you may have to change the buffer size of the access method. For example, the access method may not have enough buffers to allow the network control program to perform at peak efficiency.

Segmentation of Data Transfers for SDLC Stations

The maximum amount of data that the NCP can send and a cluster controller can receive is specified in the MAXDATA operand on the PU macro. The buffer size of the cluster controller and MAXDATA in the NCP should be equal to the most common message size received by the cluster controller. If messages larger than the cluster controller's buffer size are transmitted, the NCP must send them in smaller segments. This segmentation of messages requires more processing; therefore, the most efficient operation of the link is achieved when MAXDATA is equal to the buffer size of the cluster controller.

Network Slowdown

The network control program in both the local and remote communications controllers must have buffers available to receive data. Overloads can occur in which the program receives more data than it can send. If the overload is protracted, the network control program will exhaust its supply of buffers. The supply of buffers in the local communications controller supporting a remote communications controller can be exhausted quickly if the local controller does not have sufficient buffer space available to process the data coming from the remote controller and the stations attached to the local controller.

The network program continuously monitors its supply of buffers, and when the supply reaches a specified level, the network control program automatically enters *slowdown mode*. In slowdown mode, the program reduces the amount of data it receives from the network and the access method, but it continues to perform those functions, such as sending, which result in buffers being released. In this way, a net gain in the number of available buffers is achieved. When the buffer supply is replenished, the program automatically resumes normal operation.

The SLODOWN operand in the BUILD macro determines at which stage of buffer pool exhaustion the network control program will enter and exit slowdown mode. Most systems, having a sufficiently large buffer pool and a variety of terminals transmitting and receiving messages in a random fashion, should use the 12 percent value for SLODOWN.

If the system going into slowdown mode too often, then more buffer space may be necessary or, where applicable, pacing requirements may have to be modified. See the discussion on pacing under "Pacing for SDLC Stations". The amount of buffer space available is the difference between the size of the network control program load module (plus user-written code) and the size of storage in the controller.

Data Security Option

Unless data security is required in controller storage, do not choose the security option for controller storage (by the CDATA operand on the TERMINAL macro). The option creates performance overhead in the network control program because each buffer released to the free buffer pool must be cleared of residual data.

Processing within the Communications Controller

The network control program optionally processes data passing through the controller with block handling routines. Block handling refers to the optional processing of message data received from the host processor for retransmission to a station, or of message data received from stations for forwarding to the host processor. Typical block handling options are automatic text correction and insertion of date and time. Because these options require machine cycles and storage, they should be specified only when necessary.

Block Handling Routine Considerations for Local BSC and Start-Stop Lines

The block handling routine can be specified at various logical points (PTs) in the network control program's processing. You can specify PT1, PT2, PT3, or ALL on the BHEXEC operand of the appropriate TERMINAL, COMP, or CLUSTER macro.

If you specify PT1, the block handling routine will be processed for host Write commands before the line has been allocated. If you specify PT2, the block handling routine will be processed for host Read and Write commands, and the line will remain allocated during the processing. If you specify PT3, the block handling routine will be processed for host Read commands after the line has been released. Unless there is a requirement to have this line allocated during block handling routine execution, processing a PT1 or PT3 will allow for more effective use of the line. However, if you specify PT3, you can slow processing down in a heavily loaded system because PT3 occurs just before the data is sent to the host processor. The data will not be transferred until all of the processing on that block is completed; therefore it is possible that queuing will occur, and buffer space for the queued data will be required. PT3 also requires additional storage because another control block is required for each affected device.

Therefore, where a choice exists between processing at PT2 or PT3 and where line utilization is not a factor, PT2 is the better choice.

Data Transfer over Communications Facilities

Optimum performance in the communications controller depends on the efficient use of communications facilities connecting the controller to outlying stations. Line utilization depends on the amount of information passing over these facilities.

Nonswitched BSC and Start-Stop Multipoint Lines

Service order table entries are those that determine the order in which the network control program attempts to communicate with stations on a line. This table is defined by the SERVICE macro coded immediately after the LINE macro representing the multipoint line. Nonswitched multipoint lines may be used more efficiently by not grouping multiple service order table (SOT) entries for the same device. For example, terminals A, B, and C, should be specified as ABC ABC ABC rather than AAA BBB CCC. A station may also be assigned more than one entry to achieve a higher degree of priority for the station and also to achieve load balancing on an SDLC link; load balancing reduces the number of unproductive polls and improves overall efficiency.

The network control program can be kept responsive to application requirements by including the *change session initiation information* control option (using the SESINIT dynamic control facility on the SYSCNTRL macro). This option allows the contents of the service order table to be changed during program execution by an operator control request from the host processor. Control requests can cause the program to add or delete devices or change the order or frequency in which the devices are serviced.

In general, two multipoint lines with two terminals each are more efficient than a single line with four terminals because concurrent input and output operations can be scheduled on two lines; however, the correct use of the SESSION operand on the LINE macro will greatly improve the efficiency of a line with more than two terminals.

The Session Limit for BSC and Start-Stop Multipoint Lines

The number of concurrent sessions to be conducted on a multipoint line is called the *session limit* (specified by the SESSION operand on the LINE macro). This limit depends on several factors. Among these factors are:

1. The relative amount of time that a terminal in use does not need the communication line.
2. The permissible delay between readiness to use the terminal and the availability of the communication line.

The capability of the network control program to conduct multiple sessions on the same multipoint line depends on the possibility of data transfers not occurring continuously during the session.

While message data is being entered into the terminal's buffer or the terminal is printing the contents of the buffer, the terminal has no need for the communication line. The terminal, therefore, needs the line for relatively small portions of the session period. The line can be used for servicing other terminals in the interim.

Interleaving transmissions with several stations gives maximum use of a multipoint communication line.

In general, if the rate of message transfers between the network control program and terminals on the line is low, then the session limit should be set high. However, care should be taken not to set it so high that the number of sessions on the line lengthens response time at the terminal. If the message transfer rate is high, the session limit should be set low.

Delay from BSC Terminals

Delays due to various conditions at BSC terminals may cause excessive use of a multipoint line by a single terminal. You should consider:

1. The maximum number of times that the BSC temporary-text-delay (TTD) sequence is to be received from a station before the operation is to be aborted.
2. The maximum number of times that the BSC wait-before-transmit (WACK) sequence is to be received before the operation is to be aborted.

In general, lower TTD limit and WACK limits are preferred because this reduces the time that one terminal controls the line. These options are specified, respectively, in the TTDCNT operand and the WACHCNT operand on the GROUP macro.

Transmission Limit for BSC and Start-Stop Multipoint Lines

Transmission limits (specified on the XMITLIM operand of the TERMINAL and COMP macro) depend more on the application than on any other factor. A card reader sharing a line with other devices might monopolize the line with a high transmission limit; therefore a low transmission limit would promote greater line sharing. An inquiry/response application with a transmission limit of one would cause the terminal operator to have to wait for the response to an inquiry until the next session was established. A data collection application, on the other hand, is well suited to a transmission limit of one as no response is necessary in most cases. As a general rule, transmission limit should be specified according to the requirements of the types of devices sharing a line or according to the type of application sharing the line.

Pass Limit for SDLC Stations

The pass limit for duplex stations (PASSLIM on the PU macro) specifies the maximum number of blocks which can be sent to an SDLC station for a given entry in the Service Order Table. The pass limit is analogous to the transmission limit for BSC terminals on multipoint lines.

A large pass limit causes the line to be dedicated to a station for a long time before the NCP attempts to service another station. A large value should not be specified in applications where response time is critical.

A small pass limit causes each station on the line to receive frequent service from the NCP. A small value should be specified in applications where response time is critical.

Maximum Unacknowledged Transmissions

The number of blocks that can be outstanding, without being acknowledged, is specified by the MAXOUT operand on the PU macro. A low value should be specified if the quality of the line is not high to avoid having excessively large blocks retransmitted due to error conditions.

Buffer Delay for BSC and Start-Stop Buffered Stations

Some types of IBM BSC and Start-Stop stations receive incoming data into buffers at high speed, then print or otherwise display the data at a much slower rate. If the network control program has multiple data blocks to send to the same terminal, it must wait after sending each block for the terminal to print the contents of its buffer before it is able to send the next block. If the line is a multipoint line, the network control program can use the time the line would otherwise be idle for communicating with other terminals. That is, at any given moment the program can be sending to one of several terminals while the others are printing data received earlier.

If your network includes stations of this kind, code the delay (in seconds) on the BFRDLAY operand of the TERMINAL macro. The value you specify should equal the length of time the terminal needs to print the contents of its buffer.

Control of Nonproductive Polling

BSC and Start-Stop Lines

Two LINE macro operands, PAUSE and NEGPOLP, are available to control nonproductive polling on BSC and start-stop lines. The PAUSE operand specifies the amount of time that may elapse after the service limit value (specified by the SERVLIM operand) is reached, if no session is active on the line. The NEGPOLP operand, which is used only for BSC multipoint lines, specifies the amount of time that may elapse after the NCP receives a negative response to polling, before polling is resumed. Both of these operands limit nonproductive polling and reduce the processing overhead associated with such polling. You should specify values for the PAUSE and NEGPOLP operands to correspond with the expected negative responses to polling. If you expect most polls to receive negative responses, set the pause interval relatively high; this will reduce the processing overhead associated with such responses. However, too large a pause value can increase the response time experienced by operators of terminals on the line. If the line is so busy that terminals on the line will usually be ready when they are polled, there should be little or no pause. If you specify an integer for NEGPOLP, then you should also specify PAUSE=0 (or let it default to zero) to avoid increasing response times at the terminals. For lines with many terminals, responses can be slowed significantly with even small values specified for NEGPOLP.

SDLC Lines

The PAUSE operand of the LINE macro is available to control nonproductive polling on SDLC lines. PAUSE (for SDLC) specifies the minimum duration of the polling cycle. The polling cycle extends from the moment polling begins with the first active entry in the service order table to the moment polling next begins at the same entry. If the time expended in servicing all the active entries in the service order table equals or exceeds the PAUSE value, the next polling cycle begins immediately. On the other hand, if the time expended in servicing all the active entries in the service order table is less than the PAUSE value, the next polling cycle is deferred until the time defined by PAUSE has expired. Allowing a pause to elapse when activity on the line is relatively low reduces the amount of processing time consumed by nonproductive polling. However, too large a PAUSE value can increase the response time experienced by operators of terminals on the line.

Pacing for SDLC Stations

The pacing option is a means of regulating the flow of data between the host logical unit (primary LU) and the cluster controller's logical unit (secondary LU). The VPACING operand controls the flow of path information units (PIUs) from the primary LU to the communications controller. The PACING operand controls the flow of PIUs from the communications controller to the secondary LU. Both VPACING and PACING and specified on the LU macro during NCP generation. (See the appropriate VTAM *System Programmer's Guide* for inform; ion on how to code the VPACING operand.)

Pacing is used to prevent situations that can occur when, for example, the primary LU sends PIUs into the network faster than the NCP cn transmit them to the secondary LU or faster than the secondary LU can process them. Eventually the primary LU may flood the network with data, either forcing the NCP into slowdown mode, or exhausting the buffer supply in the cluster controller. Proper specification of the VPACING and PACING operands can prevent such situations occurring in your system.

Both the VPACING and the PACING operands have two suboperands, *n* and *m*. For VPACING, *n* denotes the number of PIUs the primary LU is to send to the NCP before stopping to await a pacing response from the NCP. *m* specifies in which of the *n* PIUs the primary LU is to turn on the pacing bit in the block header, indicating that it expects a pacing response when the NCP is able to accept more blocks.

Similarly, for the PACING operand, *n* denotes the number of PIUs the NCP is to send to the secondary LU before stopping to await a pacing response. *m* specifies in which of the *n* PIUs the NCP is to turn on the pacing bit.

Specific recommendations cannot be made for specifying particular values for *m* and *n* because the effect of pacing depends on many factors, such as:

- The speed at which the primary LU can generate data for a particular secondary LU.
- The speed of the communications line between the NCP and the cluster controller that owns the secondary LU.
- The number of cluster controllers on the communications line.
- The structure of the service order table for the communications line.
- The PASSLIM and MAXOUT attributes of the cluster controller. (See descriptions of these attributes in this chapter.)
- The number of logical units at the cluster controller.
- The speed at which the secondary LU can process data.

All of these factors need to be considered in selecting values for PACING and VPACING. Figure 7-1 will aid in this selection. See Appendix G for a detailed example of how pacing works.

Figure 7-1 lists some advantages and disadvantages of specifying various relative values of *m* and *n*. As a general rule, *n* should be as small as possible to prevent typing up buffers unnecessarily in the NCP or the cluster controller receiving the requests; *m* should be equal to or relatively close to the value specified for *n* to prevent buffer usage build-up in the receiving NCP or cluster controller. The most efficient values for *m* and *n* would allow the NCP to stay one request ahead of the requirements of each logical unit.

CONDITION	*ADVANTAGE*	*DISADVANTAGE*
1. **VPACING M & N**		
N=small value	Ties up relatively little NCP buffer space.	Less likely that NCP will be able to stay one request ahead of LU.
N=large value	NCP is more likely to have a request on hand when the LU is ready to process one.	Requests in NCP awaiting processing by the LU ties up storage. Since *N* is specified by LU, the total impact on NCP buffer usage will be the sum of the buffers tied up for all the LUs at that NCP. This total may deplete buffers to a point where NCP enters slowdown. This condition should be avoided, and can be remedied by increasing NCP storage or lowering *N* for the LUs.
M<N	Promotes relatively steady flow of requests for LU from Host, since the time between transmission of nth request by the Host and receipt of pacing response by the Host is lessened.	More likely to tie up NCP buffer space; if pacing response is received before the nth request is generated, no real pacing occurs.
M = N	Less likely to tie up NCP buffer space, since the time between transmission of nth request by the Host and the receipt of pacing response by the Host is widened.	Flow of requests from Host less likely to be steady, since the likelihood of host having to wait for pacing response is increased.
2. **PACING M & N[1]**		
N=small value	More likely that the NCP will be able to stay one request ahead of the LU.	Requests are more likely to linger in NCP, tying up storage.
N=large value	Ties up relatively little NCP buffer space, since requests are sent to the LU more rapidly.	Less likely that NCP will be able to stay one request ahead of the LU.
M<N	Less likely to tie up NCP buffer space, since the time between transmission of nth request by the NCP and the receipt of the pacing response is lessened. Also, flow of requests from NCP to LU is relatively steady.	More LU buffer space will be required.
M = N	Less LU buffer space may be required.	More likely to tie up NCP buffer space; flow of requests from NCP to LU is less steady.
M = 0 N = 0 (no pacing)	Will provide data to the secondary LU as fast as it can receive it.	More likely to tie up NCP buffers because the NCP has no control over the rate at which the host sends blocks of data to the NCP.

[1]The values specified here are in some cases dictated by the secondary LU. For this reason some of the advantages listed are unattainable.

Figure 7-1. Advantages and Disadvantages of Various Relative Pacing Values of M and N

Appendix A. Types of Stations Supported by the IBM 3705

Under control of ACF/NCP/VS, the IBM 3705 Communications Controller can communicate with any of the terminals, transmission control units, and computers listed in this appendix. This appendix is complete up to the time that the manual was published. If the device you want to communicate with is not listed, consult your IBM marketing representative for additional information.

Note: Terminals that are functionally equivalent to those listed may also operate satisfactorily with ACF/NCP/VS; the customer is responsible for establishing equivalency. Attachment of non-IBM terminals is under the provisions of the IBM Multiple Supplier Systems Policy.

Terminals

SDLC

- IBM 3270 Information Display System
  - IBM 3275 Display Station
  - IBM 3277 Display Station (via IBM 3271 Control Unit Models 11,12)
  - IBM 3284 Printer (via IBM 3271 Control Unit Models 11,12)
  - IBM 3286 Printer (via IBM 3271 Control Unit Models 11,12)
- IBM 3600 Finance Communication System (network control mode only) (via IBM 3601 or 3602 Finance Communications Controller)
- IBM 3614 Consumer Transaction Facility
- IBM 3650 Retail Store System (via IBM 3651 Store Controller Models A50, B50)
- IBM 3660 Supermarket System (via IBM 3651 Store Controller Models A60, B60 or IBM 3661 Store Controller)
- IBM 3767 Communication Terminal (Models 1, 2, 3)
- IBM 3770 Data Communications System comprising the following Communication Terminals:
  - IBM 3771 (Models 1, 2, 3)
  - IBM 3773 (Models 1, 2, P1, P2, P3)
  - IBM 3774 (Models 1, 2, P1, P2)
  - IBM 3775 (Models 1, P1)
  - IBM 3776 (Models 1, 2)
  - IBM 3777 (Model 1)
- IBM 3790 Communications System (via IBM 3791 Communcations Controller)

BSC

- IBM 2770 Data Communications System (via IBM 2772 Multi-purpose Control Unit)
- IBM 2780 Data Transmission Terminal
- IBM 2972 General Banking Terminal System
  - IBM 2980 Teller Station (Models 1 and 4)
  - IBM 2980 Administrative Station (Model 2)
- IBM 3270 Information Display System (nonswitched lines only)
  - IBM 3275 Display Station
  - IBM 3277 Display Station (via IBM 3271 Control Unit Models 1,2)
  - IBM 3284 Printer (via IBM 3271 Control Unit Models 1, 2)
  - IBM 3286 Printer (via IBM 3271 Control Unit Models 1, 2)
- IBM 3650 Retail Store System (via IBM 3651 Store Controller Models A50, B50)
- IBM 3660 Supermarket System (via IBM 3651 Store Controller Models A60, B60, or IBM 3661 Store Controller)
- IBM 3671 Shared Terminal Control Unit
- IBM 3735 Programmable Buffered Terminal
- IBM 3740 Data Entry System
  - IBM 3741 Data Station (Model 2)
  - IBM 3741 Programmable Work Station (Model 4)
  - IBM 3747 Data Converter
- IBM 3770 Data Communications System

comprising the Communication Terminals listed above under SDLC (supported as an IBM 2772 Multi-purpose Control Unit)

- IBM 3780 Data Communication Terminal (supported as an IBM 2772 Multi-purpose Control Unit)
- IBM 5275 Direct Numerical Control Station (supported as an IBM 3275 Model 1 or 2)
- IBM 5937 Industrial Terminal (supported as an IBM 3270 Model 1 or 2)

Start-Stop

- IBM 1030 Data Collection System (emulation mode only)
- IBM 1050 Data Communication System
- IBM 1060 Data Communication System (emulation mode only)
- IBM 2260 Display Station (via IBM 2848 Display Control) (emulation mode only)
- IBM 2265 Display Station (via IBM 2845 Display Control) (emulation mode only)
- IBM 2740 Communications Terminal (Models 1 and 2)
- IBM 2741 Communications Terminal
- IBM 2760 Optical Image Unit (via IBM 2740) (emulation mode only)
- IBM 3767 Communications Terminal (Models 1,2) (supported as IBM 2740 Model 1 or 2 or IBM 2741)
- IBM 3767 Communications Terminal (Model 3) (supported as IBM 2740 Model 2)
- IBM 5100 Portable Computer (supported as IBM 2741)
- IBM Communicating Magnetic Card Selectric Typewriter (supported as IBM 2741 [switched line] only)

World Trade teleprinters that use CCITT No. 2 or No. 5 code on nonswitched point-to-point lines.

Terminals using the following line control disciplines: AT & T 83B3 or WU 115A start-stop code, over nonswitched telegraph lines using a multipoint discipline; WU CPT-TWX (Model 33/35) start-stop code, over switched lines.

Communications Control Units

- IBM 2701 Data Adapter Unit
- IBM 2703 Transmission Control
- IBM 2715 Transmission Control (Model 2)
- IBM 3704 Communications Controller
- IBM 3705-I Communications Controller
- IBM 3705-II Communications Controller

Computers

SDLC

- IBM System/32 (supported as IBM 3770)

BSC

- IBM System/3
- IBM System/7 (supported as IBM System/3)
- IBM System/32 (supported as IBM System/3)
- IBM System/360 Model 20 (with BSC Adapter)
- IBM System/360 Models 25, 30, 40, 50, 65, 65MP, 67 (in 65 mode), 75, 85, 91, 195)
- IBM System/370 Models 115, 125, 135, 138, 145, 148, 158, 165, 168, 195
- IBM 1130 Computing System (via IBM 1131 Central Processing Unit with Synchronous Communications Adapter)
- IBM 1800 Data Acquisition and Control System (via IBM 1826 Data Adapter Unit with Communication Adapter)
- IBM 3750 Switching System

Start-Stop

- IBM System/7 (supported as IBM 2741)

Appendix B: Partial Program Generation

As explained in Chapter 2, partial program generation permits you to modify an existing network control program by re-assembling selected modules and link editing them with object modules not requiring changes. Significant processing time can be saved by not performing a complete generation when only some modules require changes.

To perform a partial program generation, specify PARTIAL=YES in the BUILD macro, and list the modules to be reassembled in the CONDASM operand.

A partial generation is not allowed if it involves changing the program type (NCP to PEP or PEP to NCP), the communication scanner type, or if any modules affected by the change are not at the same release level as the original program.

Note: Under OS/VS, only the object modules produced by the previous generation are available to the partial generation procedure. The library for these modules is named in the OBJLIB operand of the BUILD macro.

Modules Requiring Reassembly

This appendix lists the modules requiring reassembly for changes to the controller hardware, or for changes to the network configuration or program options. Specify the last two digits of the module names in the CONDASM operand of the BUILD macro. For changes to program tables, specify TABLE in the operand. For example, to change the channel timeout value or the text of a critical situation message, you would specify CONDASM=TABLES.

References to "adding support" refer to adding the indicated type of line, station, etc., to a network control program that does not currently have *any* such lines or stations. For example, if you were adding the first start-stop line to the network, you would reassemble modules SYSCG00A, SYSCG00B, SYSCG00C, and SYSCG00D (and SYSCG003, if controller has a type 1 scanner). Similarly, you would reassemble these modules if you were deleting *all* start-stop lines from the network.

Change the controller storage size (MEMSIZE of BUILD macro)

SYSCG000

Add or delete Abend option (ABEND of BUILD macro)

SYSCG000
SYSCG001
SYSCG002
SYSCG006
SYSCG007
SYSCG010

Change buffer limit for slowdown mode (SLODOWN of BUILD macro)

SYSCG000

Add or delete automatic network shutdown (ANS of BUILD macro)

SYSCG000
SYSCG001
SYSCG002
SYSCG006 *(If local NCP)*
SYSCG007
SYSCG00E *(If SDLC support)*

Change the maximum subarea (MAXSUBA of BUILD macro)

SYSCG000
SYSCG002
SYSCG010

Change the NCP subarea (SUBAREA of BUILD macro)

SYSCG000
SYSCG010

Change the system text timeout (ITEXTTO of BUILD macro)

SYSCG000

Change the NCP buffer size (BFRS of BUILD macro)

SYSCG000

Change the system disable timeout (DSABLTO of BUILD macro)

SYSCG000

Change the system enable timeout (ENABLTO of BUILD macro)

SYSCG000

Change the system dial timeout (DIALTO of BUILD macro)

SYSCG000

Add or delete online test option (OLT of BUILD macro)

SYSCG000
SYSCG009 *(If BSC support)*

SYSCG00A (*If start-stop support)*
SYSCG00B
SYSCG00C
SYSCG00D
SYSCG011

Change the number of lines specified (LINE)

SYSCG000

Add or delete status modifier support (STATMOD of HOST macro)

SYSCG006

Add or delete partitioned emulation line support (TYPE of LINE macro)

SYSCG000

Change the number of break characters (XBREAK of BUILD macro)

SYSCG00B

Add or delete answer tone support (ANSTONE of LINE macro)

SYSCG00B

Change the number of concurrently traced lines (LTRACE of BUILD macro)

SYSCG011

Add or delete the address trace option (TRACE of BUILD macro)

SYSCG007
SYSCG010

Add or delete block handler support

SYSCG000
SYSCG001
SYSCG002

Add or delete date/time block handling routine (DATETIME)

SYSCG001
SYSCG002

Add or delete critical situation notification message
(CSMSG, CSMSGC, CSMHDR, CSMHDRC of BUILD macro)

SYSCG000

Add or delete channel adapter trace support (CATRACE of BUILD macro)

SYSCG006
SYSCG007

Change the number of entries in the channel adapter trace table (CATRACE of BUILD macro)

SYSCG006
SYSCG007

Add or delete automatic call support (AUTO of LINE macro)

SYSCG009
SYSCG00A (*If start-stop support)*
SYSCG00B
SYSCG00C
SYSCG00D

Add or delete ring indicator mode support (RING of LINE macro)

SYSCG00B
SYSCG00C
SYSCG00D

Add or delete primary line and link support (POLLED of LINE macro)

SYSCG00B
SYSCG00C
SYSCG00D

Add or delete secondary line and link support (POLLED of LINE macro)

SYSCG00B
SYSCG00C
SYSCG00D

Add or delete tributary support (TADDR of LINE macro)

SYSCG009 (*BSC tributary only)*
SYSCG00B
SYSCG00C
SYSCG00D

Change from no channel timeout value(s) to timeout value(s) or vice versa (TIMEOUT of HOST mode)

SYSCG006
SYSCG007

Change from no channel attention delay(s) to channel attention delay(s) or vice versa (DELAY of HOST mode)

SYSCG006
SYSCG007

Change the data transfer size in BUILD macro (TRANSFR of BUILD mode)

SYSCG000

Add or delete the data transfer size in BUILD macro (TRANSFR of BUILD macro)

SYSCG000

If the TRANSFR operand is omitted from the BUILD macro and a HOST macro with a smaller transfer size is added or a HOST macro is changed and the transfer size is smaller than existed in the original generation (UNITSZ, MAXBFRU, BFRPAD of HOST macro)

SYSCG000

If the TRANSFR operand is omitted from the BUILD macro and all HOST macro(s) are replaced by PATH macro(s) (remote NCP only)

SYSCG000

Add or delete type 3 communication scanner support (TYPE of CSB macro)

SYSCG000
SYSCG007
SYSCG008 (*If SDLC support*)
SYSCG009 (*If BSC support*)
SYSCG00B
SYSCG00C
SYSCG00D
SYSCG00E (*If SDLC support*)
SYSCG010
SYSCG011

Add or delete a scanner of the same type as specified in the full generation.

SYSCG007

Change from a single type of scanner to a mixture of types or vice versa (TYPE of CSB macro)

SYSCG007
SYSCG008 (*If SDLC support*)
SYSCG009 (*If BSC support*)
SYSCG00B
SYSCG00C
SYSCG00D
SYSCG00E (*If SDLC support*)
SYSCG011

Change the hardware channel adapter configuation (CA of BUILD macro and TYPEGEN not NCP-R)

SYSCG000
SYSCG001
SYSCG002
SYSCG006
SYSCG007
SYSCG010

Change the NCP support of the channel adapters (CHANTYP, NCPCA of BUILD macro)

SYSCG000
SYSCG001
SYSCG002
SYSCG006
SYSCG007
SYSCG010

Add or delete support for SESINIT, DVSINIT, or the EDIT block handler (OPTIONS, EDIT of SYSCNTRL macro)

SYSCG001
SYSCG002

Add or delete NCP callout numbers and count (DIALNO or TERMINAL macro)

SYSCG001
SYSCG002

Add or delete manual dial support (AUTO, CALL of LINE macro)

SYSCG001
SYSCG002
SYSCG00B

Add or delete switched support (DIAL of GROUP macro or CONFIG of LINE macro)

SYSCG00B

Add start-stop support (LNCTL of GROUP macro)

SYSCG00A
SYSCG00B
SYSCG00C
SYSCG00D

Delete start-stop support (LNCTL of GROUP macro)

SYSCG00B
SYSCG00C
SYSCG00D

Add BSC support (LNCTL of GROUP macro)

SYSCG009
SYSCG00A (*If start-stop support*)
SYSCG00B
SYSCG00C
SYSCG00D

Delete BSC support (LINCTL of GROUP macro)

SYSCG00A (*If start-stop support*)
SYSCG00B
SYSCG00C
SYSCG00D

If program already includes BSC support:

Add or delete USASCII code (CODE of LINE macro)

SYSCG009

Add or delete transmit ITB support (XITB of BUILD macro)

SYSCG009
SYSCG00C

Add or delete BSC secondary/point-to-point support (POLLED of LINE macro or DIAL of GROUP macro)

SYSCG00B (*If no secondary/point-to-point start-stop, or SDLC support*)
SYSCG00C
SYSCG00D

Add or delete BSC primary-multipoint support
(POLLED of LINE macro)

SYSCG00B (*If no start-stop or SDLC primary/multipoint support*)
SYSCG00C (*If no start-stop or SDLC primary/multipoint support*)
SYSCG00D (*If no start-stop or SDLC primary/multipoint support*)

Add or delete SDLC/BSC path function
(DATASW of LU macro)

SYSCG001
SYSCG002
SYSCG010

If program already includes start-stop support:

Add or delete devices using LRC checking
(TERM, FEATURE of TERMINAL macro)

SYSCG00A

Add or delete support for 83B3/115A terminals
(TERM of TERMINAL macro)

SYSCG00A
SYSCG00C
SYSCG00D

Add or delete support for TWX terminals
(TERM of TERMINAL macro or LCTYPE of MTALCST macro)

SYSCG00A
SYSCG00B
SYSCG00C
SYSCG00D

Add or delete TWX parity support (PARCHK of LINE macro)

SYSCG00A

Add or delete support for World Trade
teletypewriter (WTTY) terminals (TERM of TERMINAL macro)

SYSCG00A
SYSCG00B
SYSCG00C
SYSCG00D

Add or delete multiple terminal access support
(MTALCST, MTATABL, TERMINAL macros)

SYSCG00B
SYSCG00C
SYSCG00D

Add or delete multipoint start-stop support
(POLLED of LINE macro)

SYSCG00A
SYSCG00B (*If no BSC or SDLC multipoint support)*
SYSCG00C (*If no BSC or SDLC multipoint support)*
SYSCG00D (*If no BSC or SDLC multipoint support)*

Add or delete secondary or point-to-point start-stop
support (POLLED of LINE macro or DIAL or GROUP macro)

SYSCG00B (*If no secondary or point-to-point BSC or SDLC support)*
SYSCG00C
SYSCG00D

Add or delete carriage return delay support
(CRDLAY of TERMINAL macro)

SYSCG00B
SYSCG00C

Add or delete support for 2740 Model 1 or change
the type of 2740 Model 1 supported (TERMINAL: TERM,
FEATURE of TERMINAL macro or LCTYPE of MTALCST macro))

SYSCG00C
SYSCG00D

Add or delete 2740 Model 2 support (TERM, FEATURE
of TERMINAL macros)

SYSCG00A
SYSCG00C
SYSCG00D

Add or delete support for 2741 terminals
(TERM of TERMINAL macro)

SYSCG00A
SYSCG00C
SYSCG00D

Add SDLC support (LNCTL of GROUP macro)

SYSCG000
SYSCG007
SYSCG008
SYSCG00B
SYSCG00D
SYSCG00E

Delete SDLC support (LNCTL of GROUP macro)

SYSCG000
SYSCG007
SYSCG00B
SYSCG00D

If program already includes SDLC support (LNCTL of GROUP macro)

Add or delete physical unit type 1 support
(PUTYPE of PU macro or NUMTYP1 of LUDRPOOL macro)

SYSCG000
SYSCG00E

Add or delete link activity timeout (ACTIVTO of GROUP macro)

SYSCG007
SYSCG00E

Add or delete full duplex path support (ADDRESS of LINE macro)

SYSCG011

Add or delete local-local link support
(POLLED, SDLCST of LINE macro or PUTYPE of PU macro)

SYSCG008
SYSCG00E

Add or delete primary local-local link support
(POLLED, SDLCST of LINE macro or PUTYPE of PU macro)

SYSCG008
SYSCG00E

Add or delete secondary local-local link support
(POLLED, SDLCST of LINE macro or PUTYPE of PU macro)

SYSCG008
SYSCG00E

Add or delete primary SDLC support
(POLLED, SDLCST of LINE macro or PUTYPE of PU macro)

SYSCG008
SYSCG00E

If no start-stop or BSC secondary support:

SYSCG00B
SYSCG00C
SYSCG00D

Add or delete secondary SDLC support
(POLLED of LINE macro)

SYSCG008
SYSCG00E

If no start-stop secondary support:

SYSCG00B
SYSCG00C
SYSCG00D

Add or delete support for switching from secondary to primary SDLC support (SDLCST macro or SDLCST of LINE macro)

SYSCG00B

Add or delete security support (SECURE of LINE macro)

SYSCG00A
SYSCG00B
SYSCG00C

Add or delete ALC support (LNCTL of GROUP macro)

SYSCG000
SYSCG007

Add or delete dynamic reconfiguration support (PUDRPOOL macro)

SYSCG000

Add or delete SDLC 3270 support (DR3270 of BUILD macro)

SYSCG000

Add or delete user-written code (NCPNAU macro or VIRTUAL, LEVEL2 of GROUP macro)

SYSCG000
SYSCG001
SYSCG002
SYSCG007
SYSCG010

Change entry point(s) of user-written initialization routine (INIT of GENEND macro)

SYSCG000

Appendix C. Calculating Buffer Storage Estimates for SDLC Lines

The amount of buffer storage needed for each SDLC communication line in a network control program (NCP) is a function of many variables. The effect of the variables on buffer estimates depends on the application, the configuration, and the type of network synchronization (interactive or batch).

Interactive (Inquiry/Response) or Immediate Control Mode Network Synchronization

In an interactive environment, the issuer (primary or secondary logical unit) sends a single request unit (RU) and waits for a response. The response may be data (that is, another request), or it may be simply an acknowledgment. The data exchanged in this manner is hereafter referred to as *interactive data.*

The most important variables affecting buffer storage requirements in an interactive environment are:

- Block (request/response) rate
- Number of clusters and/or terminals per line
- Line speed
- Average block size
- NCP buffer size
- Number of NCP buffers required for an average block

Batch or Delayed Control Mode Network Synchronization

In a batch environment, the issuing logical unit may send many request units into the network before waiting for a response. The data exchanged in this manner is hereafter referred to as *batch data.* In the formulas that follow, only data that is outbound from the NCP into the network is considered batch data.

In the batch environment, requests will accumulate in the NCP if the host node and the network can present requests to the NCP faster than the line can handle them. In such a case, the number of requests that accumulate in the NCP will be the number the primary logical unit sends unless the number is limited by the effect of PACING and VPACING operands specified in the LU macro instruction during NCP generation.

The most important variables affecting buffer storage requirements in a batch environment are:

- The number of requests a primary logical unit will send before waiting for a response
- PACING and VPACING values specified in the LU macro during NCP generation
- Number of LU-to-LU sessions concurrently active on one line
- Average block size
- NCP buffer size
- Number of NCP buffers required for an average block

Using the Buffer Storage Formulas

The following formulas allow you to calculate, by line, the buffer storage estimates for either a batch or an interactive environment or a mixture of both on one line.

The formulas are arranged in a series of 13 steps (A-M). The results of the earlier steps are used in subsequent steps until the final buffer storage requirement for the line is computed in step M.
Within each step you are asked first to list the variables that are needed to find a certain value. Then you are given a formula in which you use these variables to calculate the value. Each variable is given a two-digit designator identifying the step in which is it explained and the number of the variable within that step (for example, A5 is the fifth variable in step A). Sometimes, a variable is used in more than one step. In such cases, the same designator is used each time the variable recurs, and you should refer to the definition of the variable in the step where it was first used.

If the line for which you are calculating the storage estimates handles only batch data, you can skip A, E, H, and J. If the line handles only interactive data, you can skip steps F and K. If the line handles a mixture of both interactive and batch data, you may not skip any of the steps.

As the number of SDLC lines in the network increases, the results of your buffer storage calculations tend to be more conservative. However, this tendency should allow for any additional storage required as a result of queuing at the channel, which is not included in the calculations.

Buffer Storage Estimates for SDLC Lines

Step A - Inbound Utilization (Interactive Data Only)

Is the line FDX multipoint? —Yes→ Value A = $\boxed{0}$

No

Go to Value B

Average number of blocks per second during peak load ______ A1

Header length (enter appropriate value from below) ______ A2
FID2 - 9
FID3 - 5

If there will be a mixture of FID2s and FID3s on the line, use a weighted average for the header length. This weighted average must be a value between 5 and 9.

Average length of input data block in characters ______ A3

Responses per second during peak load (+FME/PACING) ______ A4

Line speed (bps)/8 ______ A5

Calculate Value A:

Value A

((A1 x (6 + A2 + A3)) + (A4 x (6 + A2)))/ A5 = $\boxed{}$

Step B - Outbound Utilization

Average number of blocks per second of batch data during peak load	_____	B1
Header length (enter appropriate value from below) FID2 - 9 FID3 - 5	_____	B2

If there will be a mixture of FID2s and FID3s on the line, use a weighted average for the header length. This weighted average must be a value between 5 and 9.

Average length of batch data output block in characters	_____	B3
Line speed (bps)/8	_____	B4
Average number of blocks per second of interactive data during peak load	_____	B5
Average length of interactive data output block in characters	_____	B6
Responses per second during peak load	_____	B7

Calculate Value B:

((B1 x (6 + B2 + B3))
+ (B5 x (6 + B2 + B6))
+(B7 x (6 + B2)))/ B4 =

Value B $\square$

Step C - Total Line Utilization

Calculate Value C:

Value A + Value B =

Value C $\square$

Note: When line utilization is in excess of 65 to 75 percent, buffer estimates may be significantly high. This is an indication that response time may be degraded because blocks have to be queued for the line.

Step D - Ratio of Interactive and Response Data to Total Outbound Data

Calculate Value D:

(B5 + B7)/(B1 + B5 + B7) =

Value D $\square$

Step E - Estimated Number of Interactive Blocks Queued for This Line

Number of clusters and/or terminals over which the interactive sessions are distributed. _____ E1

Calculate Value E:

((E1 + 2) x Value C x (1 - (0.5 x Value C)) /(1 - Value C)) x Value D (round up to next integer) = **Value E** ☐

Step F - Estimated Number of Batch Blocks Queued for This Line

Average values specified for the following operands on the LU macros for only those logical units expected to be in session in batch mode during peak load period (use integral values for M and N):

VPACING N _____ F1

VPACING M _____ F2

PACING N _____ F3

PACING M _____ F4

Number of batch sessions expected to be active concurrently during peak load period which are limited by pacing. They must satisfy the following conditions:

- Both VPACING and PACING have been specified as other than (0,0) for the affected logical units, *and*
- The average number of requests the primary logical unit sends before waiting for a response is equal to or greater than

((2 x F1) - F2) + F3. _____ F5

Number of batch sessions expected to be active concurrently during peak load period which are not limited by pacing. They must satisfy one or both of the following conditions:

- Either VPACING or PACING has been specified as (0,0) for the affected logical units, *or*
- The average number of requests the primary logical unit sends before waiting for a response is less than ((2 x F1) - F2) + F3. _____ F6

Average number of requests the primary logical units send before waiting for a response for those LU-to-LU sessions included in value F6. _____ F7

Number of cluster and/or terminals on which batch sessions will be active. _____ F8

Calculate Value F:

(F5 x (((2 x F1) - F2) + F3)])
+(F6 x F7) + (F8 x (F3 - F4)) = **Value F** □

Step G - Average Input Block Size

Calculate Value G: **Value G** □

(A1 x A3)/(A1 + A4) =

Step H - Average Output Block Size for Interactive Data

Calculate Value H: **Value H** □

(B5 x B6)/(B5 + B7) =

Step I - Average Number of Buffers per Input Block

Value specified in BFRS operand of the BUILD macro rounded up to the next multiple of 4. _____ I1

Calculate Value I: **Value I** □

(Value G + 23)/ I1 (round up to next integer) =

Step J - Average Number of Buffers per Interactive Output Block

Calculate Value J: **Value J** ☐

(Value H + 23)/ I1 (round up to next integer) =

Step K - Average Number of Buffers per Batch Output Block

Calculate Value K: **Value K** ☐

(B3 +23)/ I1 (round up to next integer)=

Step L - Number of Buffers for This Line

Calculate Value L: **Value L** ☐

Value I + (Value J x Value E) + (Value K x Value F) =

Step M - Number of Bytes of Buffer Storage Needed for This Line

Calculate Value M: **Value M** ☐

Value L x (I1 + 4) =

If all lines in a group have equal characteristics (that is, all variables used in calculating the buffer storage estimates are the same), calculate the storage estimate for one line and multiply by the number of lines in the group to get the total requirement for the group.

Repeat steps A through M for each SDLC line or line group in the network. Then add the results of each calculation together to get the total buffer storage estimate for SDLC lines. Enter the total in the space provided in Chapter 6.

Example of Buffer Storage Calculation for an SDLC Line

Calculate the buffer storage estimate for an SDLC communication line with the following characteristics:

- 4800 bps, half-duplex, multipoint line
- Interactive portion of line load:
  - 1 cluster node with 3 logical units (FID2) - 75% of interactive inbound and outbound
  - 1 terminal node (FID3) - 25% of interactive inbound and outbound
- Batch portion of line load:
  - 2 cluster nodes with 4 logical units each (FID2) - 80% of batch
  - 1 terminal node (FID3) - 20% of batch

Use the following values for the variables needed to calculate the buffer storage estimates:

A1 = 1 block per second

A2	=		Since there is a mixture of FID2s and FID3s flowing on the line, use a weighted average for the header length, as follows: $((9 \times 0.75) + (5 \times 0.25)) \times (1.0/1.4) + ((9 \times 0.8) + (5 \times 0.2)) \times (0.4/1.4) = 8.1$
A3	=	40	characters (average length of input data block)
A4	=	0.4	responses per second (assume that the only responses are isolated pacing responses solicited by the outbound batch flow; response rate = average output rate(B1)/PACING M(F4))
A5	=	600	(4800 bps/8)
B1	=	0.4	blocks per second (for batch data)
B2	=		Since there is a mixture of FID2s and FID3s flowing on the line, use a weighted average for the header length, as follows: $((9 \times 0.75) + (5 \times 0.25)) \times (1.0/1.4) + ((9 \times 0.8) + (5 \times 0.2)) \times (0.4/1.4) = 8.1$
B3	=	400	characters (average length of batch data output block)
B4	=	600	(4800 bps/8)
B5	=	1	block per second (for interactive data)
B6	=	160	characters (average length of interactive data output block)
B7	=	0	responses per second
E1	=	2	interactive clusters and terminals on the line
F1	=	2	VPACING N value
F2	=	1	VPACING M value
F3	=	1	PACING N value
F4	=	1	PACING M value
F5	=	5	sessions satisfying both given conditions
F6	=	2	sessions satisfying one of the given conditions
F7	=	2	requests before waiting for a response
F8	=	3	batch clusters and terminals on the line
I1	=	60	BFRS value rounded up to next multiple of 4

Using these values, calculate the buffer storage requirements as follows:

Step A:

((1 x (6 + 8.1 + 40)) + (0.4 x (6 + 8.1)))/600 **Value A**

=(54.1 + 5.7)/600 = $\boxed{.10}$

Step B: **Value B**

((0.4 x (6 + 8.1 + 400)) + (1 x (6 + 8.1 + 160)) + 0)/600= $\boxed{.57}$

Step C: **Value C**

0.10 + 0.57 = $\boxed{.67}$

Step D: **Value D**

(1 + 0)/(0.4 + 1 + 0) = 1/1.4 = $\boxed{.71}$

Step E:

((2 + 2) x 0.67 x (1 - (0.5 x 0.67))/(1 - 0.67))x 0.71

=((4 x 0.67 x 0.665)/0.33) x 0.71 = 5.4 x 0.71 = 3.83 **Value E**

Round up to next integer = $\boxed{4}$

Step F:

(5 x (((2 x 2) - 1) + 1))) + (2 x 2) + (3 x (1 - 1)) **Value F**

=(20 + 4 + 0) = $\boxed{24}$

Step G: **Value G**

(1 x 40)/(1 + 0.4) = 40/1.4 = $\boxed{28.6}$

Step H: **Value H**

(1 x 160)/(1 + 0) = $\boxed{160}$

Step I:

(28.6 + 23)/60 = 51.6 ÷ 60 = 0.86 **Value I**

Round up to next integer. = $\boxed{1}$

Step J:

(160 + 23) ÷ 60 = 3.05 **Value J**

Round up to next integer. = $\boxed{4}$

Step K:

$(400 + 23) \div 60 = 7.05$ Value K

Round up to next integer. $= \boxed{8}$

Step L: Value L

$1 + (4 \times 4) + (8 \times 24) = 1 + 16 + 192 = \boxed{209}$

Step M: Value M

$209 \times (60 + 4) = \boxed{13,376}$

Thus, 13,376 bytes of buffer storage are required for this SDLC line.

Appendix D. Calculating Buffer Storage Estimates for Local/Remote and Local/Local Communication Links

The amount of buffer storage needed in the network control program (NCP) for a local/remote or local/local communication link is a function of several variables. The most important of these are:

- Type of communication link (duplex or half-duplex)
- Line speed
- Average block size
- NCP buffer size

The following formulas allow you to calculate the buffer storage requirements for a local/remote or local/local communication link. The formulas are divided into four groups:

(1) Those for calculating the buffer estimates in a local communications controller when the link is half-duplex and POLLED=YES (primary) is specified.

(2) Those for calculating the buffer estimates in a local communications controller when the link is duplex and POLLED=YES (primary) is specified.

(3) Those for calculating the buffer estimates in a remote communications controller when the link is half-duplex or in a local communications controller when POLLED=NO (secondary) is specified and the link is half-duplex.

(4) Those for calculating the buffer estimates in a remote communications controller when the link is duplex or in a local communications controller when POLLED=NO (secondary) is specified and the link is duplex.

POLLED=NO is always specified for a remote controller.

Choose the group that applies to the NCP for which you are estimating storage. To calculate the buffer storage estimate for local/local links, use group (1) or (2) when POLLED=YES is specified on the LINE macro for the link; use group (3) or (4) when POLLED=NO is specified.

The formulas in each group are arranged in a series of steps. The results of the earlier steps are used in subsequent steps until the final buffer storage requirement for the communication link is computed in the last step.

Within each step you are asked first to list the variables that are needed to find a certain value. Then you are given a formula or a graph by which to find the value, using the variables. Each variable is given a two-digit designator identifying the step in which it is explained and the number of the variable within that step (for example, B5 is the fifth variable in step B). Sometimes, a variable is used in more than one step. In such cases, the same designator is used each time the variable recurs, and you should refer back to the definition of the variable in the step where it was first used.

Buffer Storage Estimates for Local/Remote and Local/Local Communication Links

(1) Local Communications Controller Estimates—Half-Duplex Link with POLLED=YES (Primary)

Step A - Number of Buffers per Block

Average block size for primary to secondary traffic on the local-remote or local-local communication link ______ A1

Value specified in the BFRS operand of the BUILD macro for the local NCP with POLLED=YES, rounded up the nearest multiple of 4, plus 4 _____ A2

Calculate Value A: **Value A**

(A1 +30)/(A2 - 4) (rounded up to next integer) = ☐

Step B - Total Utilization of Local/Remote or Local/Local Communication Link

Average output rate (primary to secondary) in blocks per second _____ B1

Average block size for primary to secondary traffic _____ B2

Average input rate (secondary to primary) in blocks per second _____ B3

Average block size for secondary to primary traffic _____ B4

Speed of communication link (bps) ÷ 8 _____ B5

Calculate Value B: **Value B**

((B1 x (B2 + 23)) + (B3 x (B4 + 23)))/ B5 = ☐

Step C - Ratio of Traffic Transmitted by the Communications Controller with POLLED=NO specified to Total Traffic on the Local/Remote or Local/Local Communication Link

Calculate Value C: **Value C**

(B3 x (B4 + 23))/((B1 x (B2 + 23))
+ (B3 x (B4 + 23)))= ☐

Step D - Average Number of Blocks Queued in the Communications Controller with POLLED=YES specified for this Local/Remote or Local/Local Communication Link

Use the graph in Figure D-1 to find the Value D.

If Value D lies between two curves, use the higher value. **Value D**

☐

Step E - Number of Bytes of Buffer Storage Needed for this Local/Remote or Local/Local Communication Link

Calculate Value E: Value E

Value A x Value D x A2 = ☐

(2) Local Communictions Controller Estimates—Duplex Link with POLLED=YES (Primary)

Step A - Number of Buffers per Block

Average block size for primary-to secondary traffic on the communication link ______ A1

Value specified in the BFRS operand of the BUILD macro for the local NCP with the primary end of the link, rounded up to the nearest multiple of 4, plus 4 ______ A2

Calculate Value A: Value A

(A1 + 30)/(A2 - 4) (round up to next integer) = ☐

Step B - Utilization for the primary-to-secondary Leg of the Communication Link

Average output rate (primary-to-secondary, in blocks per second ______ B1

Average block size for primary-to-secondary traffic ______ B2

Speed of local/remote or local/local communication link (bps) ÷ 8 ______ B3

Calculate Value B: Value B

(B1 x (B2 + 23))/ B3 = ☐

Step C - Average Number of Blocks Queued in the Local Communications Controller with the primary end of the link for this Local/Remote or Local/Local Communication Link

Use the graph in Figure D-1 to find the Value C. Value C

☐

Step D - Number of bytes of Buffer Storage Needed for this Local/Remote or Local/Local Communication Link

Calculate Value D: Value D

Value A x Value C x A2 = ☐

(3) Estimates for a Remote Communications Controller or a Local Communications Controller with POLLED=NO specified—Half-Duplex Link (Secondary)

Step A - Number of Buffers per Block

Average block size for secondary-to-primary traffic on the communication link	______	A1
Value specified in the BFRS operand of the BUILD macro for the NCP with the secondary end of the link, rounded up to the nearest multiple of 4, plus 4	______	A2

Calculate Value A: **Value A** ☐

(A1 + 30)/(A2 - 4) (round up to next integer) =

Step B - Total Utilization of Local/Remote or Local/Local Communication Link

Average output rate (primary-to-secondary) in blocks per second	______	B1
Average block size for primary-to-secondary traffic	______	B2
Average input rate (secondary-to-primary) in blocks per second	______	B3
Average block size for secondary-to-primary traffic	______	B4
Speed of local/remote or local/local communication link (bps) ÷ 8	______	B5

Calculate Value B: **Value B** ☐

((B1 x (B2 + 23)) + (B3 x (B4 + 23)))/ B5 =

Step C - Ratio of Traffic Transmitted by the Communication Controller with the secondary end of the link to Total Traffic on the Local/Remote or Local/Local Communication Link

Calculate Value C: **Value C** ☐

(B3 x (B4 + 23))/((B1 x (B2 + 23))
+ (B3 x (B4 + 23))) =

Step D - Average Number of Blocks Queued in the Communications Controller with the Secondary end of the Link for the Local/Remote or Local/Local Communication Link

Use the graph in Figure D-1 to find the Value D.

If Value D lies between two curves, use the higher value. Value D ☐

Step E - Number of Bytes of Buffer Storage Needed for the Local/Remote or Local/Local Communication Link

Calculate Value E: Value E ☐

Value A x Value D x A2 =

(4) Estimates for a Remote Communications Controller or a Local Communications Controller with POLLED=NO specified—Duplex Link (Secondary)

Step A - Number of Buffers per Block

Average block size for secondary-to-primary traffic on the local/remote or local/local communication link ______ A1

Value specified in the BFRS operand of the BUILD macro for the NCP with the secondary end of the link, rounded up to the nearest multiple of 4, plus 4 ______ A2

Calculate Value A: Value A ☐

(A1 + 30)/(A2 - 4) (round up to next integer) =

Step B - Utilization for the primary-to-secondary Leg of the Communication Link

Average input rate (secondary-to-primary) in blocks per second ______ B1

Average block size for secondary-to-primary traffic ______ B2

Speed of local/remote or local/local communication link (bps) ÷ 8 ______ B3

Calculate Value B: Value B ☐

(B1 x (B2 + 23))/ B3 =

Step C - Average Number of Blocks Queued in the Communication Controller with the secondary end of the link for the Local/Remote or Local/Local Communication Link

Use the graph in Figure D-1 to find the Value C.

Value C

☐

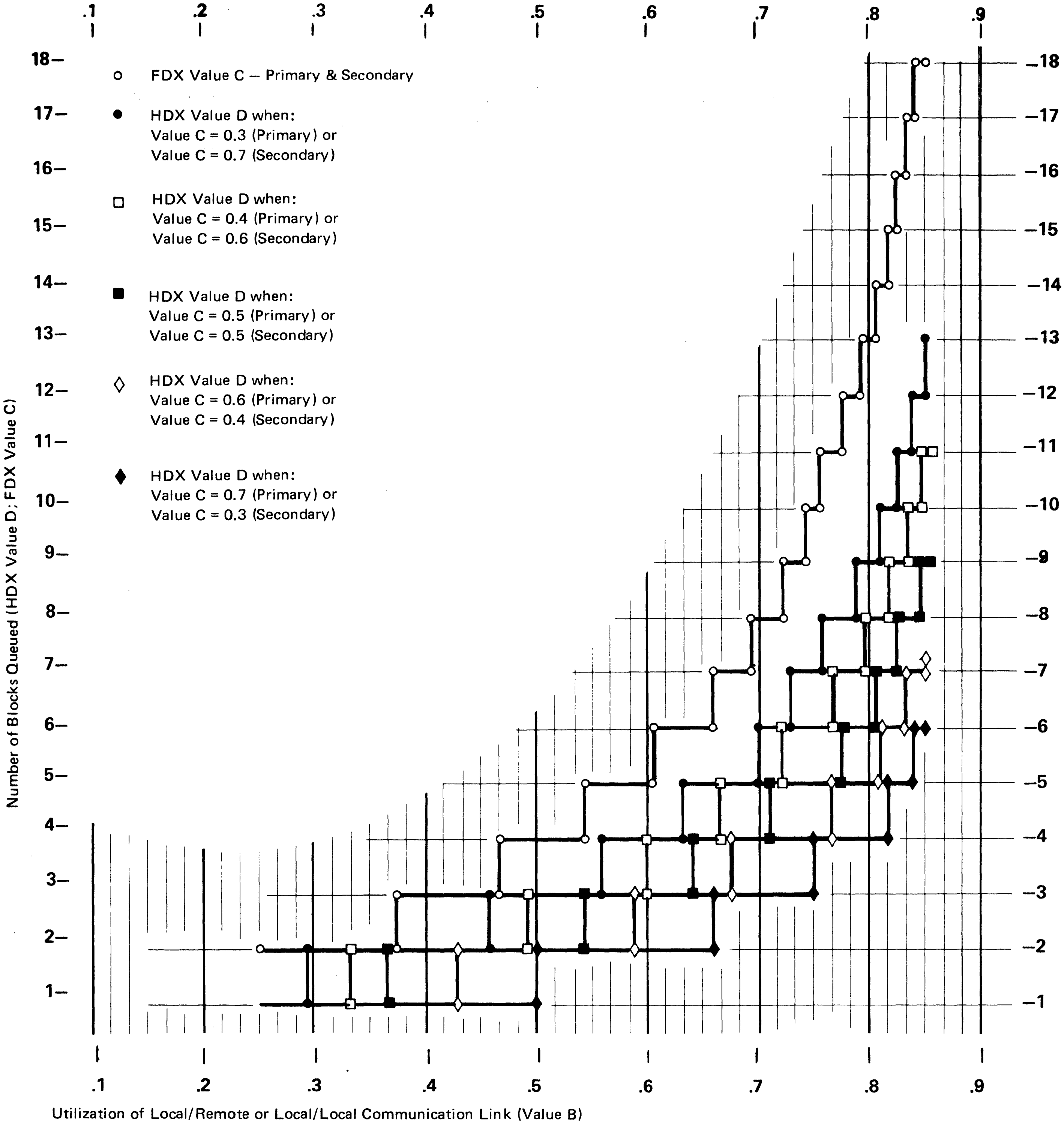
Step D - Number of Bytes of Buffer Storage Needed for the Local/Remote or Local/Local Communication Link

Calculate Value D:

Value D

☐

Value A x Value C x A2 =



Note: The following formulas were used to derive the curves in this graph. You may use these formulas instead of the graph if you desire more precise calculations.

For FDX Value C:

$[(\mathrm{Log}_{10} 0.05) / (\mathrm{Log}_{10} \text{Value B})] - 1$

For HDX Value D for the primary end:

$[(\mathrm{Log}_{10} 0.05) / (\mathrm{Log}_{10} [(\text{Value B} \times (1 - \text{Value C})) / (1 - (\text{Value B} \times \text{Value C}))])] - 1$

For HDX Value D for the secondary end:

$[(\mathrm{Log}_{10} 0.05) / (\mathrm{Log}_{10} [(\text{Value B} \times \text{Value C}) / (1 - (\text{Value B} \times (1 - \text{Value C})))])] - 1$

Figure D-1. Number of Blocks Queued for the Local/Remote or Local/Local Communication Link Versus Utilization of the Link

Example of Buffer Storage Calculation for a Local/Remote Communication Link

Calculate the buffer storage estimates for a 7200 bps, half-duplex local/remote communication link, using the following values for the variables.

For the local communications controller (primary end of the link):

A1 = 100 characters (average local-to-remote block size)

A2 = 64 buffer size (for BFRS value of 60)

B1 = 4 blocks per second

B2 = 100 characters (average local-to-remote block size)

B3 = 4 blocks per second

B4 = 40 characters (average remote-to-local block size)

B5 = 900 (7200 bps/8)

For the remote communications controller (secondary end of the link):

A1 = 40 characters (average remote-to-local block size)

A2 = 64 buffer size (for BFRS value of 60)

B1 = 4 blocks per second

B2 = 100 characters (average local-to-remote block size)

B3 = 4 blocks per second

B4 = 40 characters (average remote-to-local block size)

B5 = 900 (7200 bps/8)

Using these values, calculate the buffer storage estimates as follows.

For the local communications controller (using first group of formulas):

Step A:

(100 + 30)/(64 - 4) = 2.2 — Value A

Round up to next integer. = $\boxed{3}$

Step B:

((4 x (100 + 23)) + (4 x (40 + 23)))/900 — Value B

= (492 + 252)/900 = $\boxed{.826}$

Step C:

(4 x (40 + 23))/(4 x (100 + 23)) + (4 x (40 + 23))) — Value C

= 252/744 = $\boxed{.34}$

Step D:

Using the graph in Figure D-1, Value D lies between 9 (curve for Value C = 0.4) and 10 (curve for Value C = 0.3). Use the higher value. — Value D = $\boxed{10}$

Step E:

3 x 10 x 64 — Value E = $\boxed{1920}$

Thus, 1920 bytes of buffer storage are required in the local communications controller for this local/remote communication link. When you determine the storage requirement for the local communications controller, add in this figure.

For the remote communications controller (using third group of formulas):

Step A:

(40 + 30)/(64 - 4) = 1.2 — Value A

Round up to next integer. = $\boxed{2}$

Step B:

((4 x (100 + 23)) + (4 x (40 + 23)))/900 — Value B

= (492 + 252)/900 = $\boxed{.826}$

Step C:

(4 x (40 + 23))/(4 x (100 + 23)) + (4 x (40 + 23))) — Value C

= 252/744 = $\boxed{.34}$

Step D:

Using the graph in Figure D-1, Value D lies between 5 (curve for Value C = 0.3) and 6 (curve for Value C = 0.4). Use the higher value. — Value D = $\boxed{6}$

Step E:

2 x 6 x 64 — Value E = $\boxed{768}$

Thus, 768 bytes of buffer storage are required in the remote communications controller for this local/remote communication link. When you determine the storage requirement for the remote communications controller, add in this figure.

Appendix E. Procedure for Determining Line Interrupt Priorities

This appendix gives a recommended procedure for determining the interrupt priorities for lines serviced by type 2 and type 3 communication scanners.

You can achieve optimum servicing of communication lines and maximum protection from overruns (1) by using the priority registers associated with all four interrupt priorities and (2) by evenly distributing the total throughput of the network, in bits per second, among the four different priorities. This is the case whether lines in the network operate in network control mode or emulation mode.

The following procedure ensures that the four interrupt priorities are evenly distributed among lines in the network.

1. Divide by 8 the speed (as specified by the SPEED operand of the LINE macro) of each line serviced by a *type 3* communication scanner. The result is the adjusted line speed for such lines and is to be used in the remaining steps of the procedure. For lines serviced by a *type 2* scanner, the adjusted line speed is the same as the speed specified in the SPEED operand.
2. For each speed category, multiply the adjusted speed by the number of lines to which that speed applies.
3. Calculate the total throughput rate for all lines (in bits per second) by adding up the values calculated in step 2. Then divide the result by 4 to determine one-fourth of the throughput in bits per second.
4. List all lines in the network in the sequence of their adjusted line speeds. Those lines having the highest adjusted speed should appear at the top of the list and those having the lowest adjusted speed should appear at the bottom. Within any speed category, the sequence of lines does not matter.
5. Divide the list of lines into four sections such that the throughput for each section is roughly the same as one-fourth of the total throughput.
6. Assign priority 3 to each line in the first section of the list, priority 2 to the lines in the second section, priority 1 to those in the third section, and priority 0 to those lines in the last section.
7. Specify the assigned priority in the INTPRI operand of the LINE macro representing each line.

For purposes of the foregoing procedure, approximately the same proportion of lines in each section of the list are assumed to be active at any given moment. If the planned use of the network or experience shows that the proportions are markedly different, you may wish to adjust the distribution of lines to the sections of the list to compensate, then respecify the values in the INTPRI operands for the affected lines.

For instance, if experience shows that several of the lines in the last section of the list are relatively inactive compared to lines in the other sections, you could adjust each of the section boundaries upward so that more lines appear in the last group and fewer in each of the other sections.

The use of the procedure is illustrated by the following examples.

Example One:

The network has six lines rated at 9600 bps and serviced by a type 3 scanner, and thirteen lines—seven rated at 2400 bps and six rated at 600 bps—serviced by a type 2 scanner. Determine the total throughput and one-fourth of the total:

Line ID (Name of LINE Macro)	*Line Speed (LINE: SPEED)*	*Type of Scanner*	*Adjusted Line Speed*	*Number of Lines*	*Throughput*
			(Step 1)		(Step 2)
LH1-LH6	9,600	3	1,200	6	7,200
LM1-LM7	2,400	2	2,400	7	16,800
LL1-LL6	600	2	600	6	3,600
			Total throughput (Step 3)		27,600
			One-fourth of total		6,900

List lines in order of adjusted speed (step 4), divide list into four sections (step 5), and assign priorities to each section (step 6):

Line ID (Name of LINE Macro)	*Adjusted Line Speed*	*Total Bit Rate for Section*	*Interrupt Priority*
LM1	2,400	7,200	3
LM2	2,400		
LM3	2,400		
LM4	2,400		
LM5	2,400	7,200	2
LM6	2,400		
LM7	2,400		
LH1	1,200	6,000	1
LH2	1,200		
LH3	1,200		
LH4	1,200		
LH5	1,200		
LH6	1,200		
LL1	600		
LL2	600	7,200	0
LL3	600		
LL4	600		
LL5	600		
LL6	600		

As step 7, specify the priority values in the INTPRI operands of the LINE macros.

Example Two:

The network has eleven lines serviced by a type 3 scanner—three lines at 19,200 bps, three at 9,600 bps, and five at 2,400—and the following lines serviced by a type 2 scanner: ten lines at 1,200 bps, nine at 600 bps, seven at 150 bps, and eight at 134.5 bps.

As in example one, first determine the total throughput and one-fourth of that value, then list the lines in order of adjusted line speed, divide into four sections, and assign priorities.

Line ID (Name of LINE Macro)	*Line Speed (LINE: SPEED)*	*Type of Scanner*	*Adjusted Line Speed*	*Number of Lines*	*Throughput*
LA1-LA3	19,200	3	2,400	3	7,200
LB1-LB3	9,600	3	1,200	3	3,600
LC1-LC5	2,400	3	300	5	1,500
LD1-LD10	1,200	2	1,200	10	12,000
LE1-LE9	600	2	600	9	5,400
LF1-LF7	150	2	150	7	1,050
LG1-LG8	134.5	2	134*	8	1,072

*Decimal fraction dropped as insignificant

Total throughput	31,822
One-fourth of total	7,956

Line ID (Name of LINE Macro)	*Adjusted Line Speed*	*Total Bit Rate for Section*	*Interrupt Priority*
LA1	2,400		
LA2	2,400	7,200	3
LA3	2,400		
LB1	1,200		
LB2	1,200		
LB3	1,200		
LD1	1,200	8,400	2
LD2	1,200		
LD3	1,200		
LD4	1,200		
LD5	1,200		
LD6	1,200		
LD7	1,200		
LD8	1,200	8,400	1
LD9	1,200		
LD10	1,200		
LE1	600		
LE2	600		
LE3	600		
LE4	600		
LE5	600		
LE6	600		
LE7	600		
LE8	600		
LE9	600		
LC1	300		
LC2	300		
LC3	300		
LC4	300		
LC5	300		
LF1	150		
LF2	150	7,822	0
LF3	150		
LF4	150		
LF5	150		
LF6	150		
LF7	150		
LG1	134		
LG2	134		
LG3	134		
LG4	134		
LG5	134		
LG6	134		
LG7	134		
LG8	134		

Appendix F. Upper Scan Limits, Address Substitution, and High Speed Select Options

The maximum data rate, or speed, at which a communication line can operate is limited by the frequency at which that line's interface address is scanned by the communication scanner. In the absence of upper scan limits, address substitution, and high speed select options, each line interface address associated with a type 2 or type 3 communication scanner is scanned once per scanning cycle. The maximum line speed in this case is 4800 bits per second (bps). (The type of line set and oscillator or modem clocking rate determines the actual line speed.) To accommodate higher maximum line speeds requires the imposition of upper scan limits or the application of the address substitution or high speed select technique.

Upper Scan Limits

Imposing an upper scan limit is a means of increasing the frequency at which a selected range of line addresses is scanned, at the expense of not scanning the remaining addresses associated with the scanner. (No lines attached to the unscanned addresses can be active while the upper scan limit is in effect.) By not scanning some addresses, the communication scanner can scan the others more often within each scanning cycle, thus raising the maximum line speed. (The number of scans per cycle is constant regardless of how they are distributed to the line addresses.)

Upper scan limits can be specified individually for each of the type 2 and type 3 scanners in the communications controller. Figure F-1 shows for each scan limit value (from 0 to 3, as specified in the GENEND macro of the emulation program or network control program), the range of addresses that are scanned (light boxes) and those not scanned (dark boxes). Also given is the maximum line speed for the addresses scanned (disregarding any lower limit that may be imposed by choice of line set and oscillator speeds).

Address Substitution

A technique similar to the use of upper scan limits alters the scanning pattern so that a single, predetermined address is scanned several times per scanning cycle, again at the expense of not scanning other addresses. In this technique, however, only one address, of the range of addresses scanned, receives the benefit of increased scanning frequency. Further, address substitution applies uniformly to *all installed type 2 (and type 3) scanners,* rather than being specified separately for the individual scanners. Address substitution should not be specified if the controller contains one or more type 3 scanners.

Figure F-2 shows, for each bit position in the address substitution mask (specified in the GENEND macro of the emulation program), the selected addresses that will be scanned more often, and the group of addresses that will accordingly not be scanned. As is the case for upper scan limits, no lines attached to the unscanned addresses can be active while address substitution is in effect.

High Speed Select Option

The high speed select option is similar to address substitution in that bit settings within a mask alter the scanning pattern so that a predetermined address is scanned several times per scanning cycle, at the expense of not scanning other addresses. This option differs from address substitution, however, in that (1) up to eight addresses serviced by the scanner can receive the increased scanning frequency, and (2) masks are individually specified for each of the installed scanners, thus allowing more flexibility in selecting addresses to receive the increased scanning. Figure F-3 shows, for each bit position in the high speed select mask (specified in the GENEND macro) the selected addresses that will be scanned more often and the group of addresses that will not therefore be scanned. No lines attached to the unscanned addresses can be active while the high speed select mask is in effect.

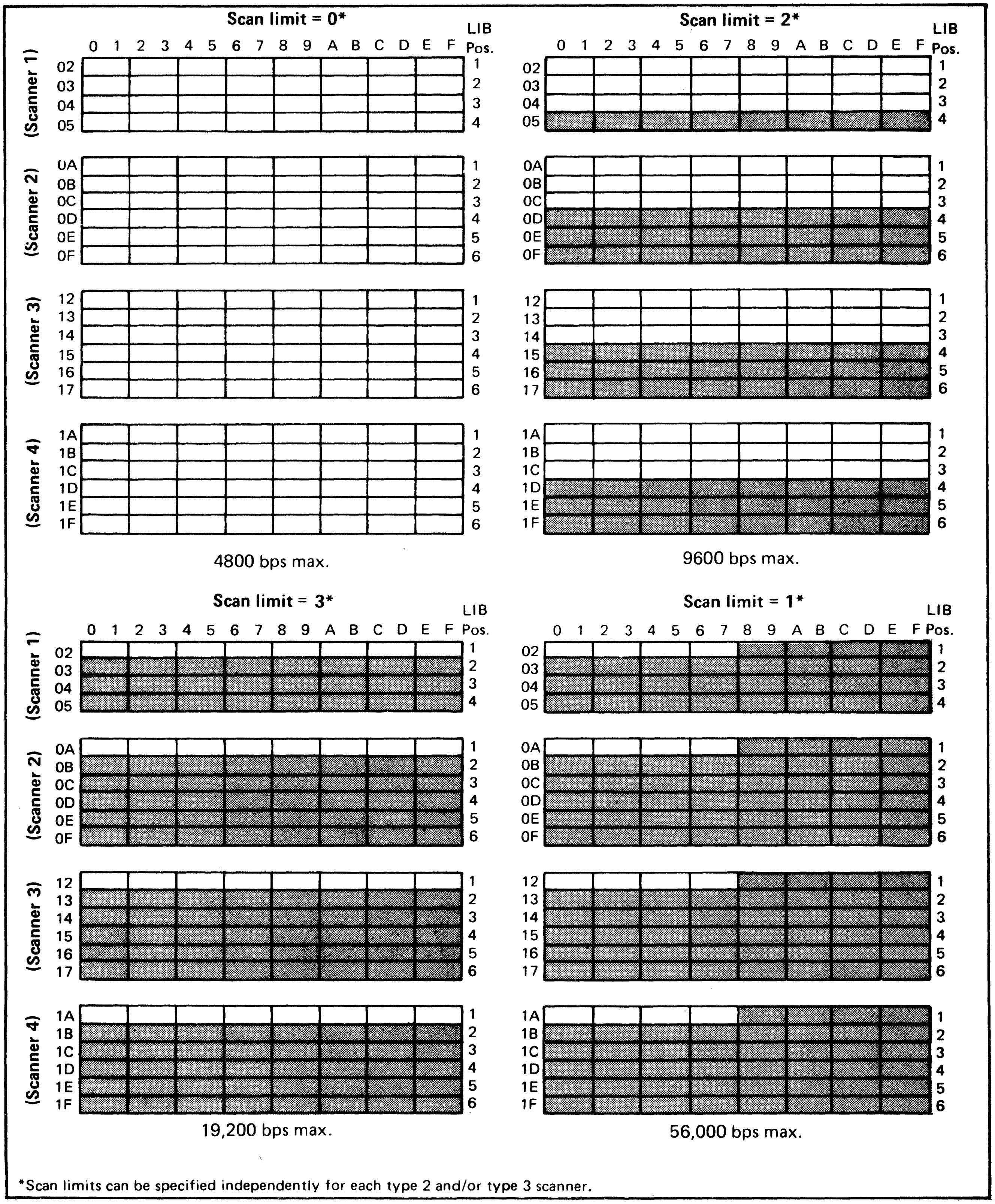


Figure F-1. Addresses Scanned and Not Scanned When Upper Scan Limits Are Used

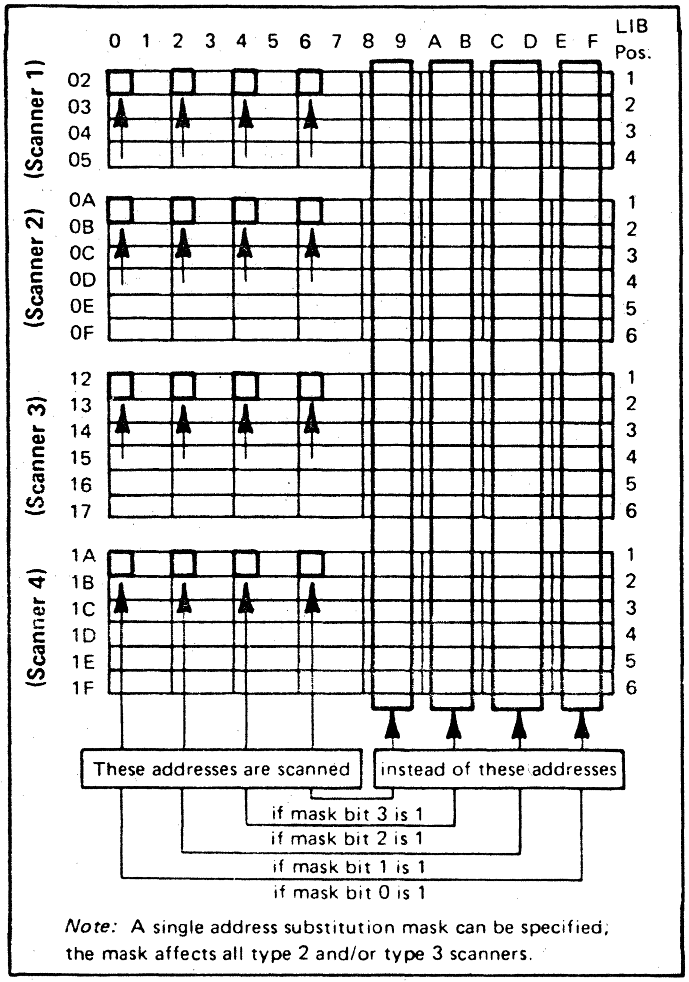


Figure F-2. Addresses Scanned and Not Scanned When Address Substitution Mask Is Used

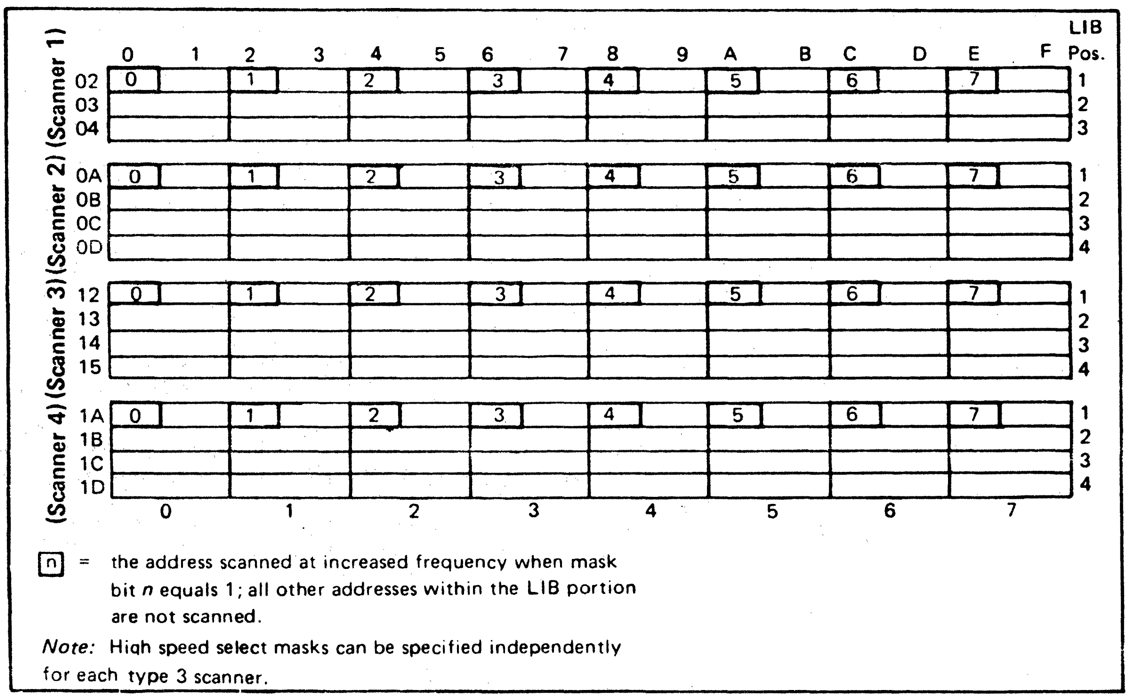


Figure F-3. Addresses Scanned and Not Scanned When High Speed Select Mask Is Used

Appendix G. Example of Pacing

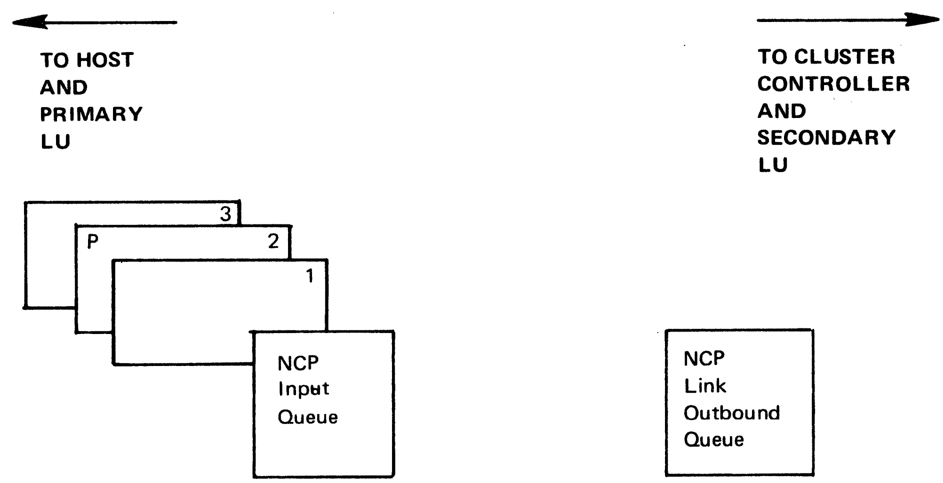
The following example illustrates the way in which the NCP participates in pacing. The example is for the case:

LU VPACING=(3,2),PACING=(2,1)

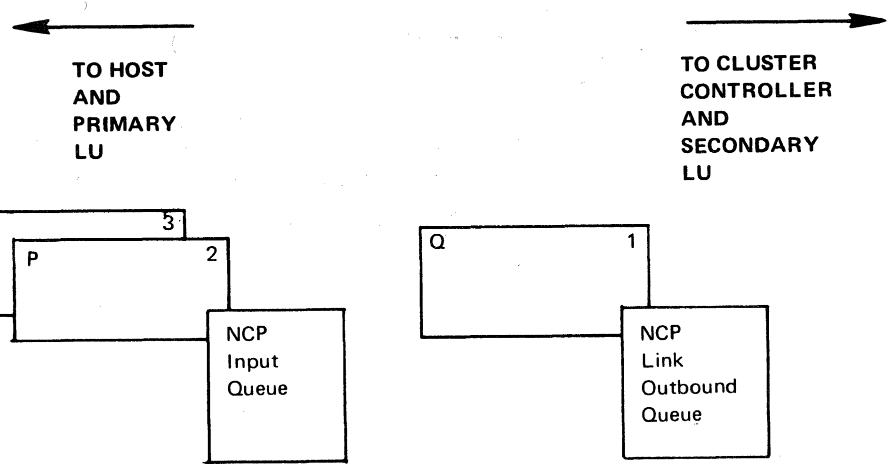
The example assumes that the primary LU has an unlimited supply of data for the secondary LU. For simplicity, this example excludes expedited blocks and response blocks and includes only exception responses.

Each block in the example is numbered as it is sent from the primary LU. *P* indicates the presence of the pacing indicator on a block as it passes from the primary LU to the NCP. (The example assumes that the pacing indicator is present on blocks that require exception responses only.) *Q* indicates the presence of the pacing indicator on a block as it passes from the NCP to the secondary LU.

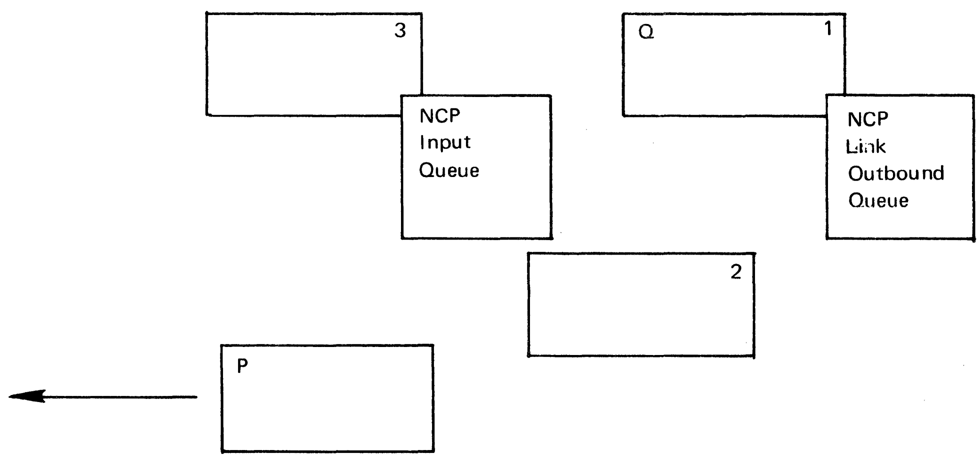
The NCP begins operation not in "awaiting pacing response" state for this secondary LU. The NCP keeps a pacing counter. At the beginning of the example this counter is initialized to 0.



1. **Since the primary LU is not currently waiting for a pacing response, it sends three (N=3) blocks to the NCP. The pacing indicator is present on the second block (M=2) (where VPACING=(3,2) applies to the primary LU).**

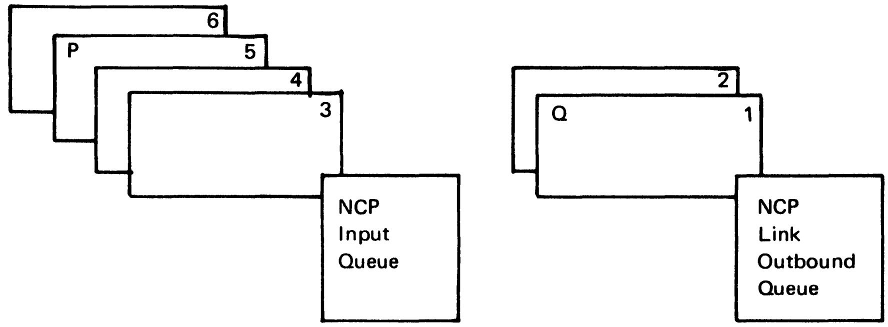


2. The NCP fetches the first block on the input queue for this secondary LU. The NCP increments its pacing counter by one. It notes that pacing block *M* has been reached, and it sets the pacing indicator *Q* in this block. The block is scheduled for transmission to the cluster controller.

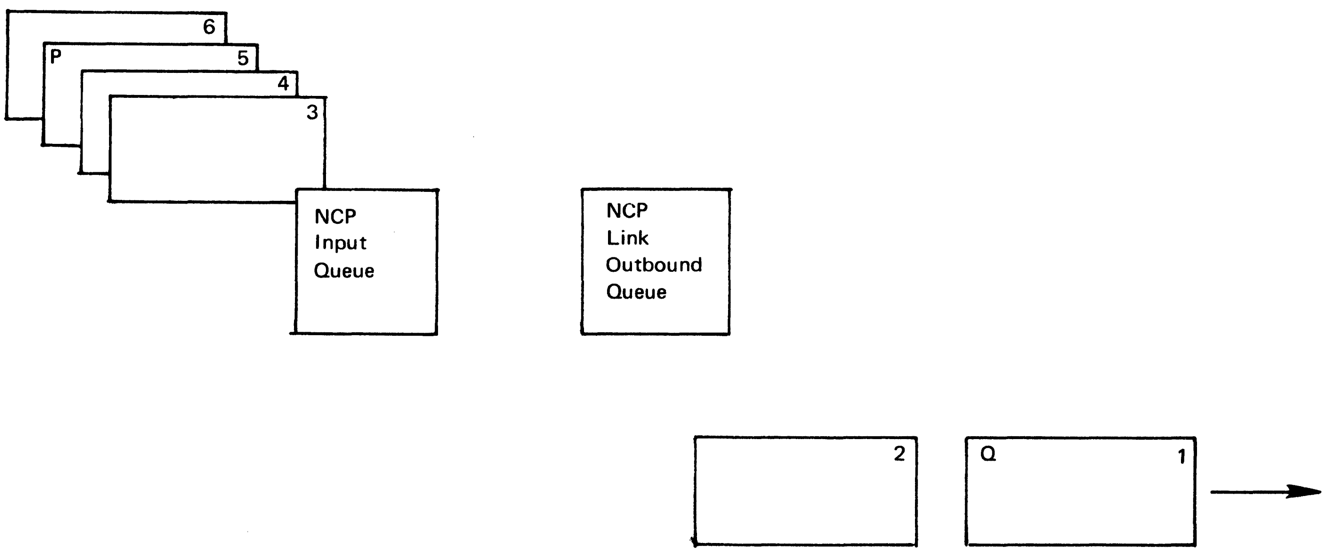


3. Since the NCP pacing counter is not equal to VPACING *M*, the NCP gets the second block from the input queue and processes it. The pacing counter is incremented by one. It notes that the pacing indicator *P* is on.

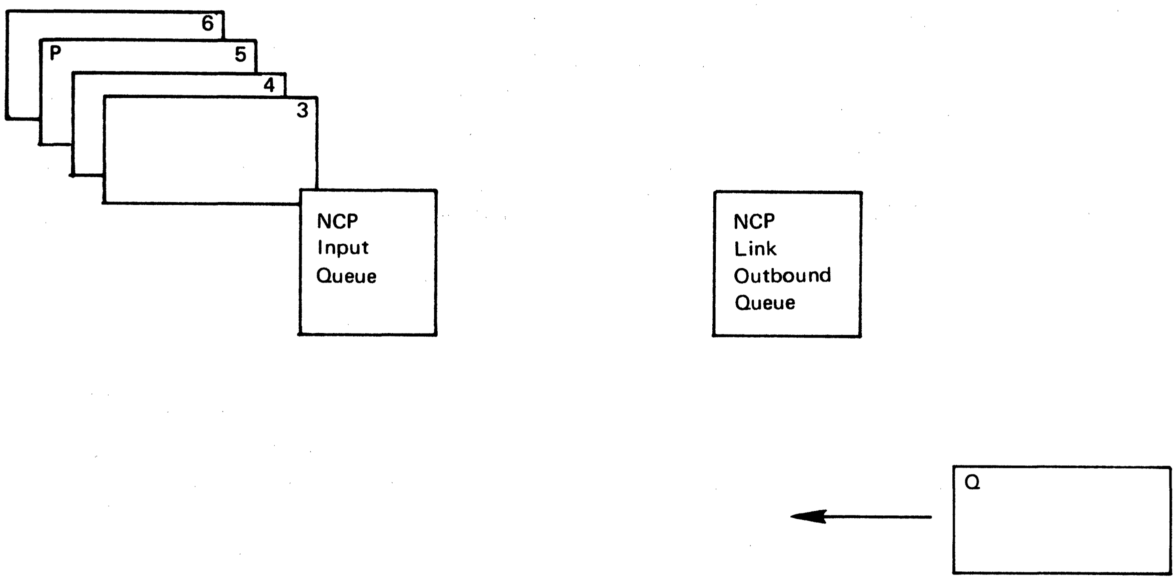
   The NCP generates an "isolated pacing response" and sends it to the primary LU.



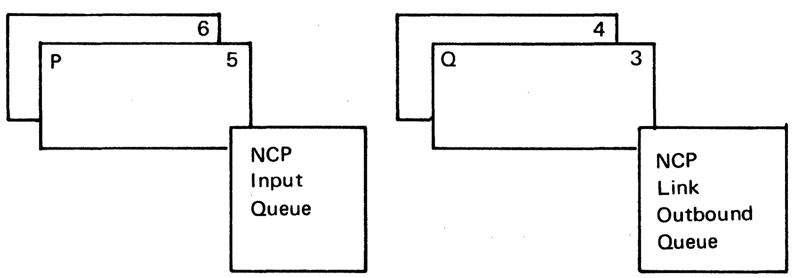
4. Since PACING *N* has now been satisfied, the NCP enters the "awaiting pacing response state." No other blocks will be removed from the input queue until a pacing response is received from the secondary LU. Concurrently, the NCP receives VPACING *N* additional blocks from the primary LU.



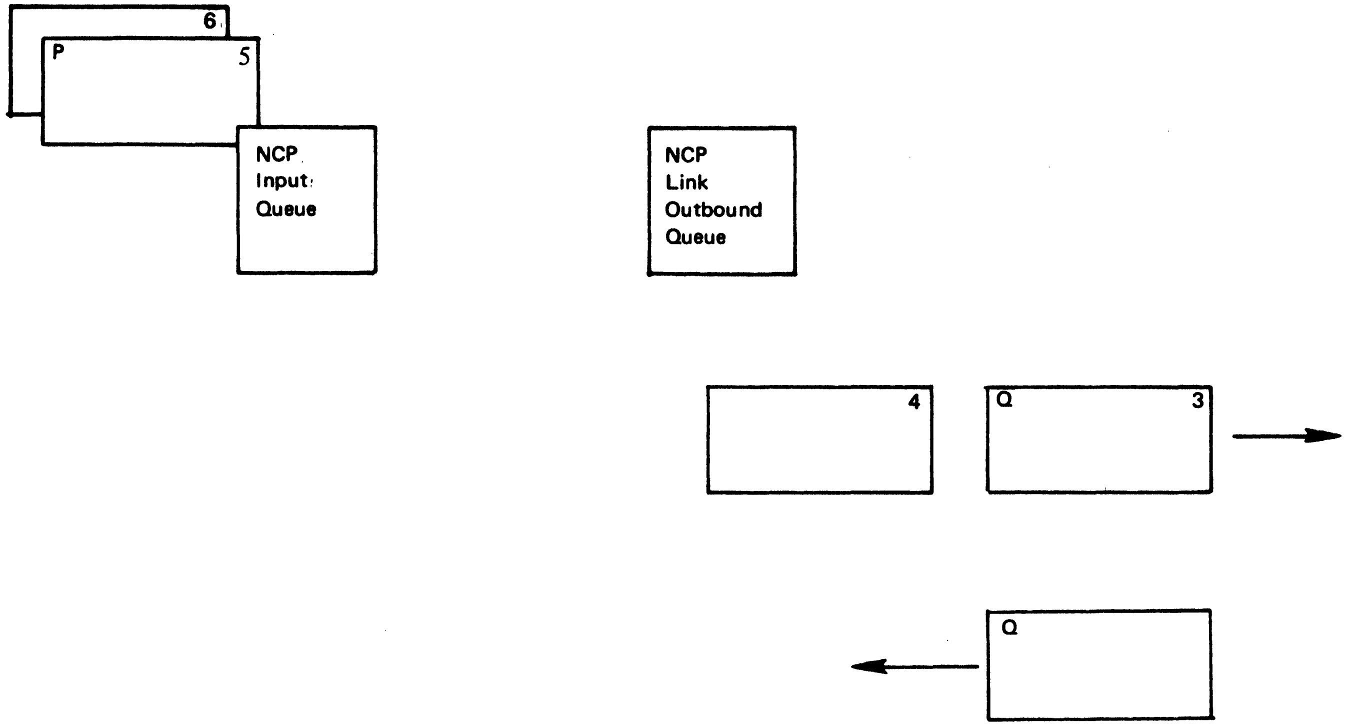
5. Blocks 1 and 2 are transmitted to the cluster controller. Eventually they are processed by the secondary LU.



6. The secondary LU notes that the pacing indicator *Q* is present in block 1, and it returns a pacing response *Q* to the NCP. This response authorizes the NCP to send PACING *N* more blocks to the secondary LU (where PACING=(2, 1) applies to the secondary LU).

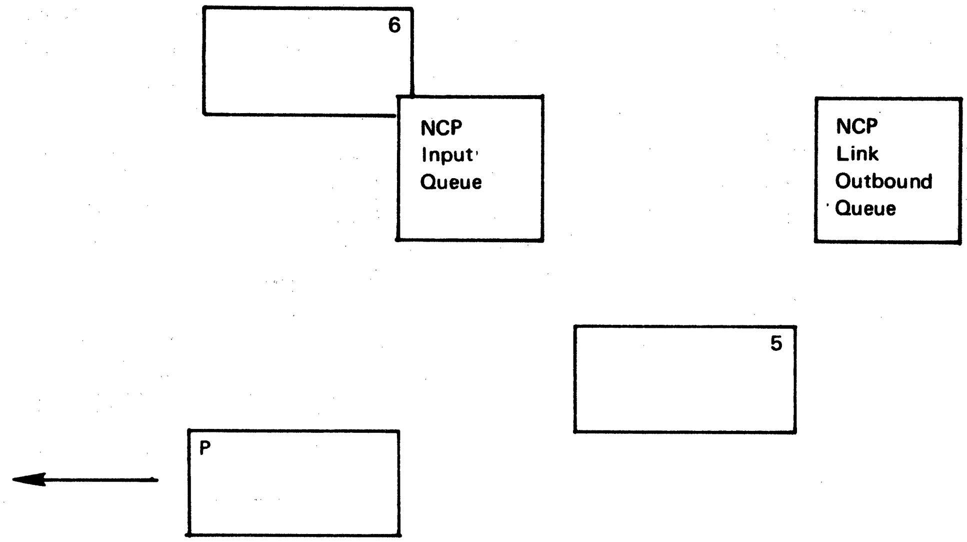


7. The NCP leaves the "awaiting pacing response" state and initializes its pacing counter to zero. Since the pacing counter is not equal to VPACING *M*, the NCP fetches the next block from the input queue and processes it, incrementing the pacing counter by one. The NCP notes that it has reached block *M* and sets the pacing indicator in this block. Since the pacing counter is not equal to VPACING *M*, NCP gets the next block from the input queue and processes it, incrementing the pacing counter by one.

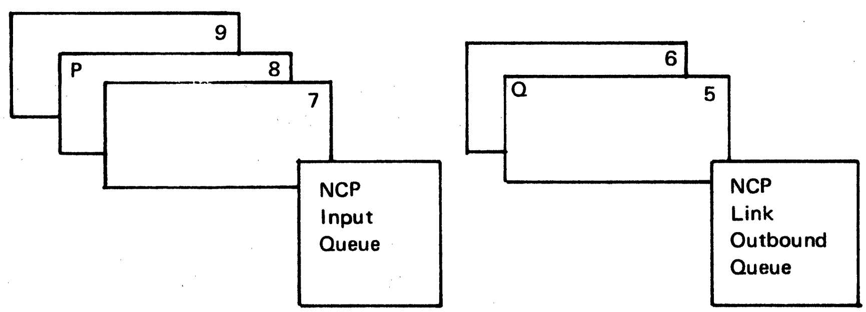


8. Since PACING *N* has now been satisfied, the NCP enters the "awaiting pacing response" state. No other blocks will be removed from the input queue until a pacing response is received from the secondary LU. Blocks 3 and 4 are transmitted to the cluster controller and are eventually processed by the secondary LU.

   The secondary LU notes that the pacing indicator *Q* is present in block three, and it returns a pacing response *Q* to the NCP. This response authorizes the NCP to send PACING *N* more blocks to the secondary LU.



9. The NCP leaves the "awaiting pacing response state" and initializes its pacing counter to zero. The NCP gets the next block from the input queue, notes that the pacing indicator *P* is on, and generates an isolated pacing response, which it sends to the primary LU.



10. The NCP increments its pacing counter by one. It notes that pacing block *M* has been reached, and it sets the pacing indicator *Q* in this block. The block is scheduled for transmission to the cluster controller. Since the NCP pacing counter is not equal to VPACING *M*, the NCP fetches the next block from the input queue and processes it, incrementing the pacing counter by one.

    Since PACING *N* has now been satisfied, the NCP enters the "awaiting pacing response" state. No other blocks will be removed from the input queue until a pacing response is received from the secondary LU. Concurrently, the NCP receives VPACING *N* additional blocks.

    Blocks 5 and 6 are transmitted to the cluster controller and are eventually processed by the secondary LU. The secondary LU notes that the pacing indicator *Q* is present in block 5, and it returns a pacing response *Q* to the NCP. This response authorizes the NCP to send PACING *N* more blocks to the secondary LU. The NCP exits the "awaiting pacing response" state, initializes the pacing counter value to zero, and the entire process is repeated for all subsequent blocks on the input queue.

To illustrate the effect of no pacing, consider the case where PACING=(0,0) is specified for an LU. At step 4 of the example, the NCP is not keeping a pacing counter and so would not enter the "awaiting pacing response" state. Processing would continue directly to step 7. Blocks 3, 4, and 5 would be processed and scheduled for transmission. The pacing indicator *P* would cause the NCP to send an isolated pacing response to the primary LU, and the primary LU would send VPACING *N* more blocks to the NCP. The NCP would have no control over the rate at which it moved blocks from the input queue to the link outbound queue. If the rate of transmission from the primary LU to the NCP were faster than the rate of transmission from the NCP to the secondary LU, blocks would accumulate on the link outbound queue in the NCP. If too many buffers were needed to hold these blocks, the NCP might eventually be forced to enter the slowdown state. In other words, if you specify PACING=(0,0), VPACING does not work; the values specified for PACING control the NCP's search and control of VPACING.

Appendix H. Transmission Codes for World Trade Teletypewriter Terminals

This appendix shows the transmission codes associated with the ITA2 and ZSC3 character sets used by World Trade teletypewriters. These transmission codes are used when defining the EOB and EOT character sequences in the WTTYEOB and WTTYEOT operands of the GROUP macro. Specify the hexadecimal value for each character of the sequence in the order they are to appear when transmitting an EOB or EOT.

Character Set ITA2 & ZSC3	*Transmission Code (hexadecimal)*	*Character Set ITA2*	*Character Set ZSC3*	*Transmission Code (hexadecimal)*
A	03	-	+	23
B	19	?	6	39
C	0E	:	8	2E
D	09	WRU	WRU	29
E	01	3	-	21
F	0D		4	2D
G	1A		0	3A
H	14		?	34
I	06	8	Bell	26
J	0B	Bell	2	2B
K	0F	(	(	2F
L	12	)	)	32
M	1C	.	7	3C
N	0C	,	,	2C
O	18	9	:	38
P	16	0	9	36
Q	17	1		37
R	0A	4	/	2A
S	05	'	'	25
T	10	5	.	30
U	07	7	1	27
V	1E	=	=	3E
W	13	2	3	33
X	1D	/		3D
Y	15	6	5	35
Z	11	+		31
CR	08	CR	CR	28
LF	02	LF	LF	22
LTRS	1F	LTRS	LTRS	-
FIGS	1B	FIGS	FIGS	-
Space	04	Space	Space	24

Appendix I: Sample Network Control Programs

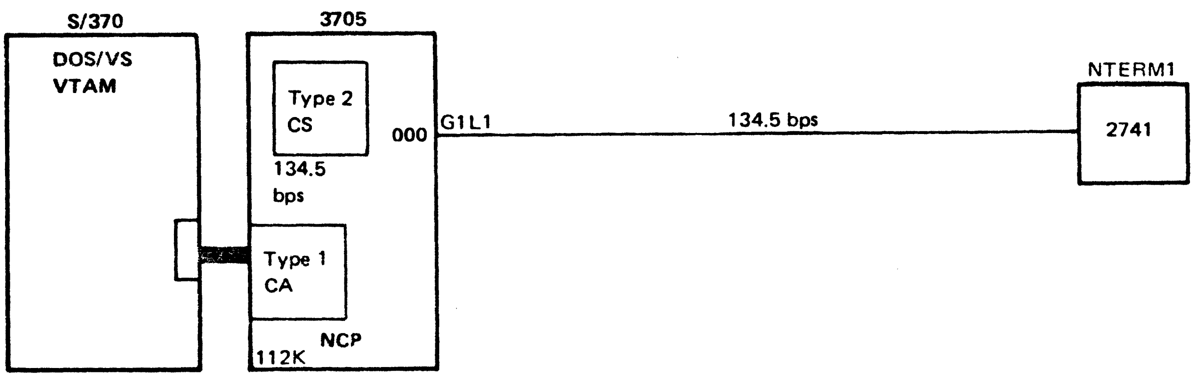
In this appendix are the source statements for sample communications controller control programs, representing a variety of programs: network control functions only, combinations of network control functions with the PEP extension, and one network control program for a remote communications controller. These samples progress from the simple, with one or two lines and few program options, to the complex, with many types of lines, stations, and program options. Preceding each sample program is the configuration of the network for which the program was coded, and accompanying remarks.

These programs are only representative samples showing in general how the source statements may be coded, and do not necessarily reflect the most appropriate choices of options for any particular application.

Sample 1

Network Control Program (TYPGEN=NCP)

Lines and Stations: One start-stop, nonswitched point-to-point, IBM 2741 with Interrupt feature.



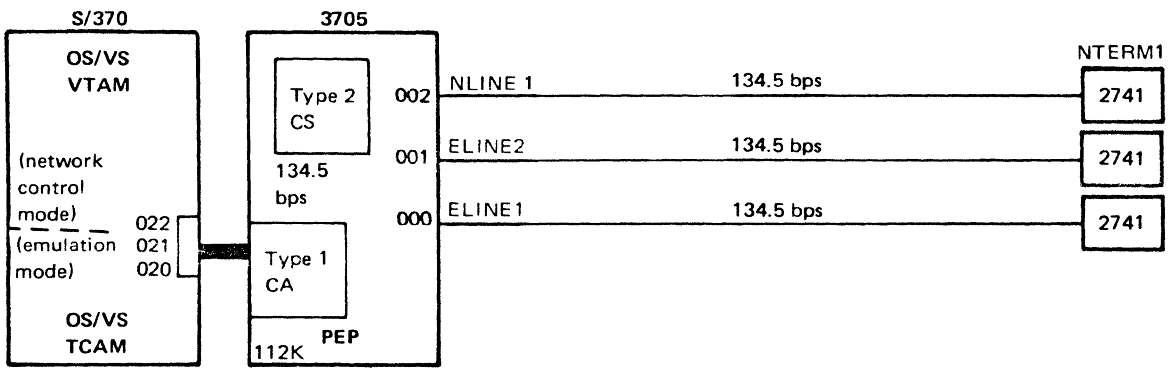
```
*  SAMPLE PROGRAM 1
*
*  NETWORK CONTROL PROGRAM (LOCAL), GENERATED UNDER DOS/VS, ACCESS
*  METHOD:  DOS/VS VTAM
*
SAMPLE2    BUILD     MODEL=3705,                                                  X
                     MEMSIZE=112,         STORAGE SIZE IS 112K BYTES              X
                     TYPGEN=NCP,          LOCAL NCP-NETWORK CTL FUNCTIONS ONLY    X
                     SUBAREA=3,           SUBAREA ADDR OF NCP IS 3                X
                     MAXSUBA=7,           HIGHEST SUBAREA ADDR POSSIBLE IS 7      X
                     CA=TYPE1,                                                    X
                     CHANTYP=TYPE1,                                               X
                     BFRS=80,             NCP BUFFER SIZE IS 80                   X
                     TYPSYS=DOS
           SYSCNTRL  OPTIONS=(MODE,RCNTRL,RCOND,       VTAM-REQUIRED DYNAMIC      X
                     RECMD,RIMM,ENDCALL,BHSASSC)   CONTROL OPTIONS
                                          3 BFRS INITIALLY ALLOCATED FOR
           HOST      INBFRS=3,            DATA TRANSFERS FROM VTAM                X
                     MAXBFRU=10,          MAX VTAM BFR UNITS FOR DATA FROM NCP    X
                     UNITSZ=88,           SIZE OF VTAM BFR UNITS                  X
                     BFRPAD=15            DOS/VS VTAM REQUIRES 15 BFR PADS
           CSB       TYPE=TYPE2,                                                  X
                     MOD=0,                                                       X
                     SPEED=134
G1         GROUP     TYPE=NCP             ALL DEFAULT VALUES ARE APPROPRIATE
NLINE1     LINE      ADDRESS=000,         3705 LINE INTERFACE ADDRESS             X
                     SPEED=134,                                                   X
                     CLOCKNG=INT,                                                 X
                     CODE=EBCD,           TERMINAL USES EXTENDED BCD CODE         X
                     DUPLEX=FULL          COMMUNICATION FACILITY IS DUPLEX
NTERM1     TERMINAL  TERM=2741,                                                   X
                     ATTN=ENABLED,        NCP STOPS SENDING ON 2741 ATTN          X
                     FEATURE=ATTN         2741 IS EQUIPPED WITH INTERRUPT FEATURE
           GENEND
           END
/*
/&
```

Sample 2

Network Control Program (local) with partitioned emulation programming extension (TYPGEN=PEP).

Lines and Stations: One nonswitched point-to-point start-stop, IBM 2741 (network control mode).

Two nonswitched point-to-point start-stop, IBM 2741 (emulation mode).



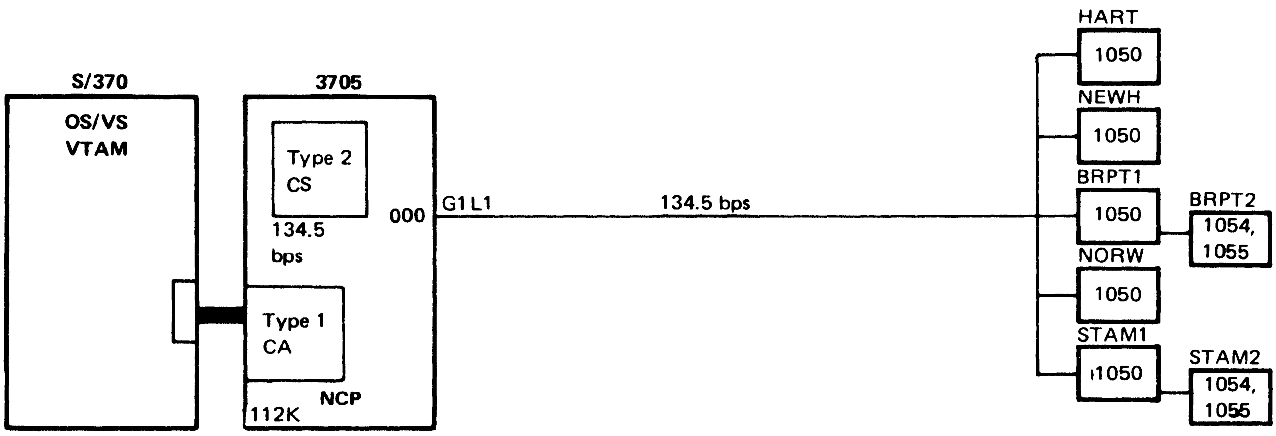
Remarks: In this example, one line operates only in network control mode and two lines operate only in emulation mode. Notice that only the terminal on the line in network control mode is represented by a TERMINAL macro; terminals on lines in emulation mode are not represented by TERMINAL macros.

```
*  SAMPLE PROGRAM 2
*
*  NETWORK CONTROL PROGRAM (LOCAL), GENERATED UNDER OS/VS, ACCESS
*    METHOD:  OS/VS VTAM FOR NETWORK CONTROL MODE LINE,
*    OS/VS TCAM FOR EMULATION MODE LINES
*
*  IN COMMENTS FIELD, E  INDICATES OPERAND IS APPLICABLE TO EMULATION MODE,
*                     N  INDICATES OPERAND IS APPLICABLE TO NETW CTL MODE
SAMPLE3    BUILD     MODEL=3705,                E  N                         X
                     MEMSIZE=112,                  N                         X
                     TYPGEN=PEP,                E  N                         X
                     TYPSYS=OS,                 E  N                         X
                     LOADLIB=NCPLIB,            E  N                         X
                     CA=TYPE1,                  E  N                         X
                     CHANTYP=TYPE1,             E  N                         X
                     HICHAN=023,                E                            X
                     LOCHAN=020,                E                            X
                     OBJLIB=NCPOBJLIB,          E  N                         X
                     SUBAREA=3,                    N                         X
                     MAXSUBA=7,                    N                         X
                     BFRS=80                       N
           SYSCNTRL  OPTIONS=(MODE,                N                         X
                     RNCTRL,RCOND,RECMD,           N                         X
                     RIMM,ENDCALL,BHSASSC)         N
           HOST      INBFRS=3,                     N                         X
                     NCPCHAN=023,                  N                         X
                     MAXBFRU=10,                   N                         X
                     UNITSZ=84,                    N                         X
                     STATMOD=YES,                  N                         X
                     BFRPAD=28                     N
           CSB       TYPE=TYPE2,                E  N                         X
                     MOD=0,                     E  N                         X
                     SPEED=134,                 E  N                         X
                     WRAPLN=001                 E
G1         GROUP     TYPE=PEP                   E  N
ELINE1     LINE      ADDRESS=(000,20),          E                            X
                     TYPE=EP,                                                X
                     SPEED=134,                 E                            X
                     CLOCKNG=INT,               E                            X
                     CU=2701,                   E                            X
                     MODEM=OPTION1,             E                            X
                     TERM=2741                  E
ELINE2     LINE      ADDRESS=(001,21),          E                            X
                     TYPE=EP,                                                X
                     SPEED=134,                 E                            X
                     CLOCKNG=INT,               E                            X
                     CU=2701,                   E                            X
                     MODEM=OPTION1,             E                            X
                     TERM=2741                  E
NLINE1     LINE      ADDRESS=002,                  N                         X
                     TYPE=NCP,                                               X
                     SPEED=134,                    N                         X
                     CLOCKNG=INT,                  N                         X
                     CODE=EBCD,                    N                         X
                     DUPLEX=FULL                   N
NTERM1     TERMINAL  TERM=2741,                    N                         X
                     ATTN=ENABLED,                 N                         X
                     FEATURE=ATTN                  N
           GENEND
           END
/*
//
```

Sample 3

Network Control Program (local) (TYPGEN=NCP)

Lines and Stations: One start-stop, nonswitched multipoint, IBM 1050.



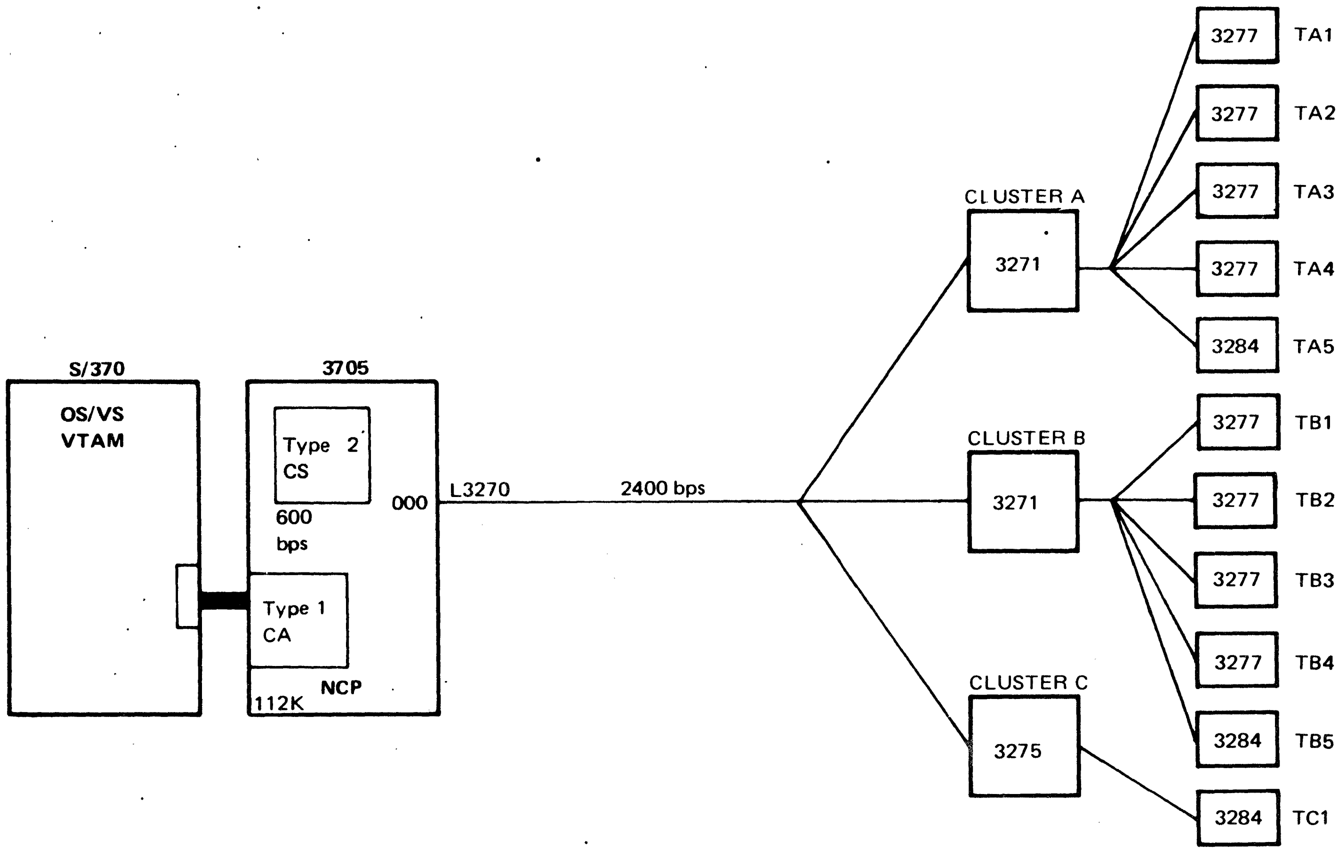
Remarks: This example illustrates the coding for a multipoint start-stop line on which multiple sessions are to be conducted. The use of COMP macros for subsidiary components of two of the terminals is shown. Notice that the GROUP macro is coded without operands; this reflects that all of the default values of this macro are appropriate.

```
*  SAMPLE PROGRAM 3
*
*  NETWORK CONTROL PROGRAM (LOCAL), GENERATED UNDER OS/VS, ACCESS
*     METHOD: OS/VS VTAM
*
SAMPLE4   BUILD     MODEL=3705,                                                X
                    MEMSIZE=112,                                               X
                    TYPGEN=NCP,                                                X
                    SUBAREA=3,                                                 X
                    MAXSUBA=7,                                                 X
                    CA=TYPE1,                                                  X
                    CHANTYP=TYPE1,                                             X
                    BFRS=80,                                                   X
                    TYPSYS=OS,                                                 X
                    LOADLIB=NCPLIB,                                            X
                    OBJLIB=STG2ASM
          SYSCNTRL  OPTIONS=(MODE,ENDCALL,RCOND,RECMD,RIMM,BHSASSC,RCNTRL)
          HOST      MAXBFRU=10,                                                X
                    UNITSZ=84,                                                 X
                    INBFRS=3,                                                  X
                    BFRPAD=28,                                                 X
                    STATMOD=YES,         STATUS MODIFIER OPTION AND CHANNEL    X
                    DELAY=.2             DELAY AID CHANNEL PERFORMANCE
          CSB       TYPE=TYPE2,                                                X
                    SPEED=134,                                                 X
                    MOD=0
G1        GROUP
G1L1      LINE      SPEED=134,                                                 X
                    CLOCKNG=INT,                                               X
                    ADDRESS=000,                                               X
                    TERM=1050,                                                 X
                    CODE=EBCD,                                                 X
                    POLLED=YES,          MULTIPOINT LINE CONTROL IS REQUIRED   X
                    SESSION=5,           MAX. CONCURRENT SESSIONS ON LINE      X
                    SERVLIM=3,           SERV LMT: 3 SERV ORDER TABLE ENTRIES  X
                    PAUSE=20,            SERVICE-SEEKING PAUSE: 20 SECONDS     X
                    TRANSFR=10,          TRANSFER LIMIT: 10 BUFFERS/SUB-BLOCK  X
                    CUTOFF=1             CUTOFF LIMIT: 1 SUB-BLOCK
                    POLIMIT=1,QUEUE
          SERVICE   ORDER=(HART,NEWH,HART,BRPT1,BRPT2,NORW,HART,               X
                    STAM1,STAM2)
HART      TERMINAL  ADDR=C1F9,           COMMON ADDR CHARACTERS: A9            X
                    POLL=C1F0,           COMMON POLLING CHARACTERS: A0         X
                    XMITLIM=1            TRANSMISSION LIMIT: 1
NEWH      TERMINAL  ADDR=C2F1,           PRINTER ADDR CHARACTERS: B1           X
                    POLL=C2F5,           KEYBOARD POLLING CHARACTERS: B5       X
                    XMITLIM=1
BRPT1     TERMINAL  ADDR=C3F1,           PRINTER ADDR CHARACTERS: C1           X
                    POLL=C3F5,           KEYBOARD POLLING CHARACTERS: C5       X
                    XMITLIM=1
BRPT2     COMP      ADDR=C3F3,           PAPER TAPE PUNCH ADDR CHARACTERS: C3  X
                    POLL=C3F6,           PAPER TAPE RDR POLLING CHARACTERS: C6 X
                    XMITLIM=1
NORW      TERMINAL  ADDR=C4F9,           COMMON ADDR CHARACTERS: D9            X
                    POLL=C4F0,           COMMON POLLING CHARACTERS: D0         X
                    XMITLIM=1
STAM1     TERMINAL  ADDR=C5F1,           PRINTER ADDR CHARACTERS: E1           X
                    POLL=C5F5,           KEYBOARD POLLING CHARACTERS: E5       X
                    XMITLIM=1
STAM2     COMP      ADDR=C5F3,           PAPER TAPE PUNCH ADDR CHARACTERS: E3  X
                    POLL=C5F6,           PAPER TAPE RDR POLLING CHARACTERS: E6 X
                    XMITLIM=1
          GENEND
          END
/*
//
```

Sample 4

Network Control Program (local) (TYPGEN=NCP)

Lines and Stations: One BSC, nonswitched multipoint, IBM 3270 terminals (3271, 3275, 3277, 3284).



Remarks: As in sample 4, the line is nonswitched multipoint; in this case, however, the terminals are binary synchronous. Several more options are included in this program than in the previous sample; for example, NCP slowdown, buffer erase, and use of critical situation notification messages and headers. Notice how the general type of terminal (3277) is specified in the CLUSTER macros for clusters A and B, with the exception (3284 printers) specified in their respective TERMINAL macros (terminals TA5 and TB5). This illustrates the use of the operand hierarchy to save coding effort. (The TERM=3277 operand could just as well have been coded in the LINE or GROUP macro as in the CLUSTER macro.)

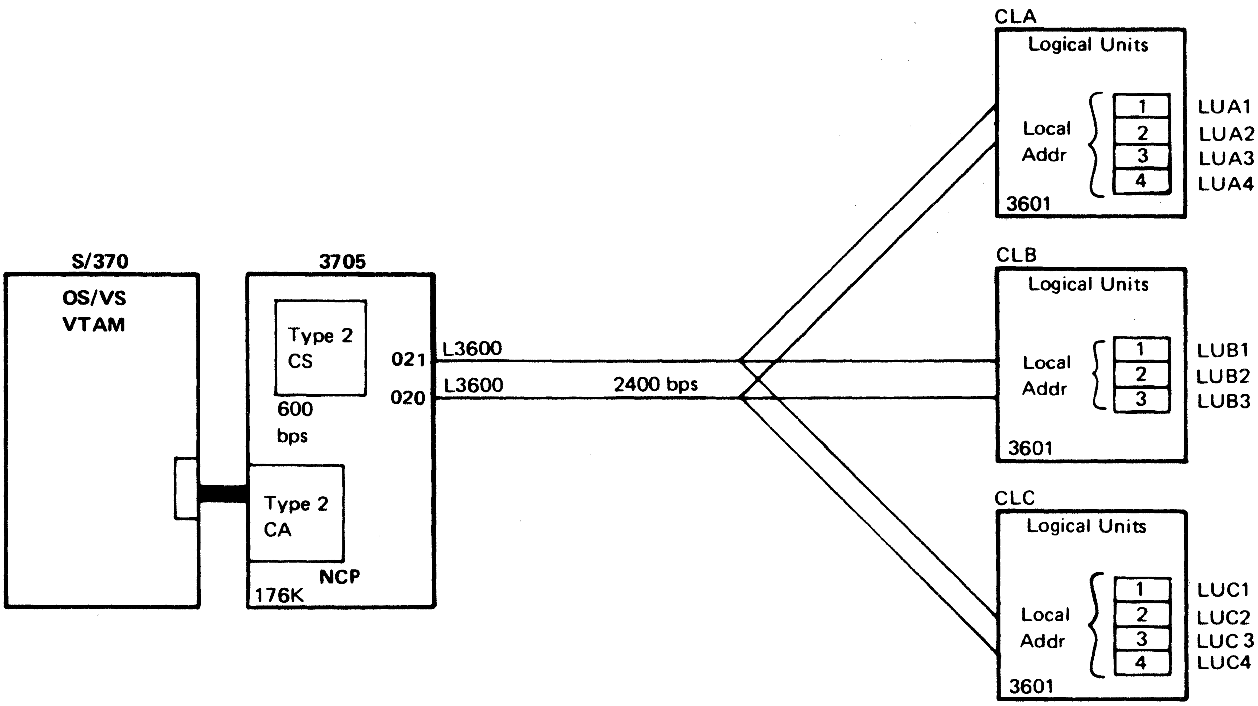
```
*  SAMPLE PROGRAM 4
*
*  NETWORK CONTROL PROGRAM (LOCAL), GENERATED UNDER OS/VS, ACCESS
*      METHOD: OS/VS VTAM
*
SAMPLE5    BUILD    MODEL=3705,                                                      X
                    MEMSIZE=112,                                                     X
                    TYPGEN=NCP,                                                      X
                    SUBAREA=3,                                                       X
                    MAXSUBA=7,                                                       X
                    BFRS=80,                                                         X
                    CA=TYPE1,                                                        X
                    CHANTYP=TYPE1,                                                   X
                    TYPSYS=OS,                                                       X
                    LOADLIB=NCPLIB,                                                  X
                    OBJLIB=STG2ASM,                                                  X
                    NEWNAME=NCP3270,                                                 X
                    ERASE=YES,           BUFFER ERASE OPTION REQUIRED                X
                    SLODOWN=25,          NCP SLOWDOWN REQD WHEN 1/4 BFRS LEFT        X
                    CSMHDR=27F5C8,       3270 CTL CHARS: ESC,ERASE/WRITE,WCC         X
                    CSMSG=5A5A5A40D5D640C6E4D9E3C8C5D940C9D5D7E4E340E4D5E3C9          X
                    D340D5D6E3C9C6C9C5C4405A5A5A  CS MSG TEXT: ! ! ! NO FURTHER      X
                                         INPUT UNTIL NOTIFIED ! ! !
           SYSCNTRL OPTIONS=(RCNTRL,MODE,RCOND,RECMD,RIMM,BHSASSC,                   X
                    ENDCALL)
           HOST     INBFRS=3,                                                        X
                    MAXBFRU=10,                                                      X
                    UNITSZ=84,                                                       X
                    BFRPAD=28,                                                       X
                    STATMOD=YES,                                                     X
                    DELAY=.2
           CSB      TYPE=TYPE2,                                                      X
                    MOD=0,
                    SPEED=600            OSCILLATOR RATE LT 1/2 MODEM RATE           X
GP3270     GROUP    LNCTL=BSC,                                                       X
                    TYPE=NCP
L3270      LINE     ADDRESS=000,                                                     X
                    SPEED=2400,                                                      X
                    CLOCKNG=EXT,         EXTERNAL (MODEM) CLOCKING USED              X
                    CODE=EBCDIC,                                                     X
                    CRITSIT=YES,                                                     X
                    POLLED=YES,                                                      X
                    SESSION=14,          EQ OR EXCEED NO. OF DEVICES IN S.O.T        X
                    SERVLIM=3,                                                       X
                    PAUSE=5,                                                         X
                    TRANSFR=3,                                                       X
                    CUTOFF=10,                                                       X
                    CDATA=YES            BUFFER ERASE REQUIRED                       X
                    POLIMIT=(1,QUEUE)    MAX. INPUT IS BFRS X TRANSFR X CUTOFF       X
                                         = 80 X 3 X 10 = 2400 BYTES
           SERVICE  ORDER=(CLUSTERA,TA1,TA2,TA3,TA4,TA5,CLUSTERB,TB1,                X
                    TB2,TB3,TB4,TB5,CLUSTERC,TC1)
CLUSTERA   CLUSTER  CUTYPE=3271,         CLUSTER CONTROL UNIT TYPE                   X
                    TERM=3277,           TERMINAL TYPE (EXCEPT TA5)                  X
                    GPOLL=40407F7F,      GENERAL POLLING CHARACTERS                  X
                    XMITLIM=1            VTAM REQUIRES XMITLIM=1
TA1        TERMINAL ADDR=60604040,       SPECIFIC ADDR CHARACTERS                    X
                    POLL=40404040        SPECIFIC POLLING CHARACTERS
TA2        TERMINAL ADDR=6060C1C1,                                                   X
                    POLL=4040C1C1
TA3        TERMINAL ADDR=6060C2C2,                                                   X
                    POLL=4040C2C2
TA4        TERMINAL ADDR=6060C3C3,                                                   X
                    POLL=4040C3C3
TA5        TERMINAL ADDR=6060C4C4,                                                   X
                    POLL=4040C4C4,                                                   X
                    TERM=3284,           THIS TERMINAL IS 3284 PRINTER               X
                    BFRDLAY=13           13 SEC. DELAY FOR BUFFERED PRINTER
```

```
CLUSTERB    CLUSTER  CUTYPE=3271,                                                    X
                     TERM=3277,          TERMINAL TYPE (EXCEPT TB5)                  X
                     GPOLL=C1C17F7F,                                                 X
                     XMITLIM=1
TB1         TERMINAL ADDR=62614040,                                                  X
                     POLL=C1C14040
TB2         TERMINAL ADDR=6161C1C1,                                                  X
                     POLL=C1C1C1C1
TB3         TERMINAL ADDR=6161C2C2,                                                  X
                     POLL=C1C1C2C2
TB4         TERMINAL ADDR=6161C3C3,                                                  X
                     POLL=C1C1C3C3
TB5         TERMINAL ADDR=6161C4C4,                                                  X
                     POLL=C1C1C4C4,                                                  X
                     TERM=3284,                                                      X
                     BFRDLAY=13
CLUSTERC    CLUSTER  CUTYPE=3275,                                                    X
                     GPOLL=C2C27F7F,                                                 X
                     XMITLIM=1
TC1         TERMINAL TERM=3275
            GENEND
            END
/*
//
```

Sample 5

Network Control Program (local) (TYPGEN=NCP)

Lines and Stations: One SDLC, nonswitched multipoint link, IBM 3600 terminals.



Remarks: In this configuration, the 3705 communicates with three IBM 3601 controllers over a duplex, nonswitched multipoint SDLC link comprising separate paths (lines) for transmitting and receiving. This sample program shows the use of the text retries option, as specified in the LINE and PU macros.

```
*  SAMPLE PROGRAM 5
*
*  NETWORK CONTROL PROGRAM (LOCAL), GENERATED UNDER OS/VS, ACCESS
*     METHOD: OS/VS VTAM
*
SAMPLE6    BUILD    MEMSIZE=176,                                                 X
                    TYPGEN=NCP,                                                  X
                    SUBAREA=3,                                                   X
                    MAXSUBA=7,                                                   X
                    CA=TYPE2,                                                    X
                    CHANTYP=TYPE2,                                               X
                    BFRS=80,                                                     X
                    TYPSYS=OS,                                                   X
                    LOADLIB=NCPLIB,                                              X
                    OBJLIB=STG2ASM,                                              X
                    NEWNAME=NCP3601,                                             X
                    SLODOWN=25
           SYSCNTRL OPTIONS=(MODE,RCNTRL,RCOND,RECMD,RIMM,BHSASSC,               X
                    ENDCALL)
           HOST     INBFRS=3,                                                    X
                    MAXBFRU=10,                                                  X
                    UNITSZ=84,                                                   X
                    BFRPAD=28,                                                   X
                    STATMOD=YES,                                                 X
                    DELAY=.2
           CSB      TYPE=TYPE2,                                                  X
                    MOD=0,                                                       X
                    SPEED=600
GP3601     GROUP    LNCTL=SDLC
L3600      LINE     ADDRESS=(020,021),     TRANSMIT AND RECEIVE ADDRESSES        X
                    DUPLEX=FULL,           SDLC LINK IS FULL DUPLEX              X
                    SPEED=2400,            3601'S OPERATE AT 2400 BPS            X
                    POLLED=YES,                                                  X
                    RETRIES=5,             5 RETRIES PER RECOVERY SEQUENCE       X
                    MAXDATA=265,                                                 X
                    PASSLIM=3,                                                   X
                    PACING=(1,1)
           SERVICE  ORDER=(CLA,CLB,CLC)
CLA        PU       ADDR=C1,               CLUSTER ADDRESS - A (EBCDIC)          X
                    PUTYPE=2,              TYPE 2 PHYSICAL UNIT                  X
                    MAXOUT=3,              MAX PATH INFO UNITS SENT BEFORE RESP  X
                    RETRIES=(,10,4)        4 RETRY SEQUENCES MAX, 10 SEC. BETWEEN
FALUA1XX   LU       LOCADDR=1,PACING=(3,1) (FA REQD 1ST 2 CHARS FOR 1ST
LUA2XXXX   LU       LOCADDR=2              LOCAL ADDRESS (3601 LOAD ADDR))
LUA3XXXX   LU       LOCADDR=3
LUA4XXXX   LU       LOCADDR=4
CLB        PU       ADDR=C2,               ADDRESS IS B                          X
                    PUTYPE=2,                                                    X
                    MAXOUT=3,                                                    X
                    RETRIES=(,10,4)
FALUB1XX   LU       LOCADDR=1,PACING=(3,1)
LUB2XXXX   LU       LOCADDR=2
LUB3XXXX   LU       LOCADDR=3
CLC        PU       ADDR=C3,               ADDRESS IS C                          X
                    PUTYPE=2,                                                    X
                    MAXOUT=3,                                                    X
                    RETRIES=(,10,4)
FALUC1XX   LU       LOCADDR=1,PACING=(3,1)
LUC2XXXX   LU       LOCADDR=2
LUC3XXXX   LU       LOCADDR=3
LUC4XXXX   LU       LOCADDR=4
           GENEND
           END
/*
//
```

Sample 6

Network Control Program (Local) with Partitioned Emulation Programming Extension (TYPGEN=PEP)

Lines and Stations: Five BSC switched call-in lines (network control mode only) for communicating with IBM 3735 terminals.

Four BSC switched call-in/call-out lines (network control mode only) for communicating with IBM System/3s (two lines reserved for call-in use).

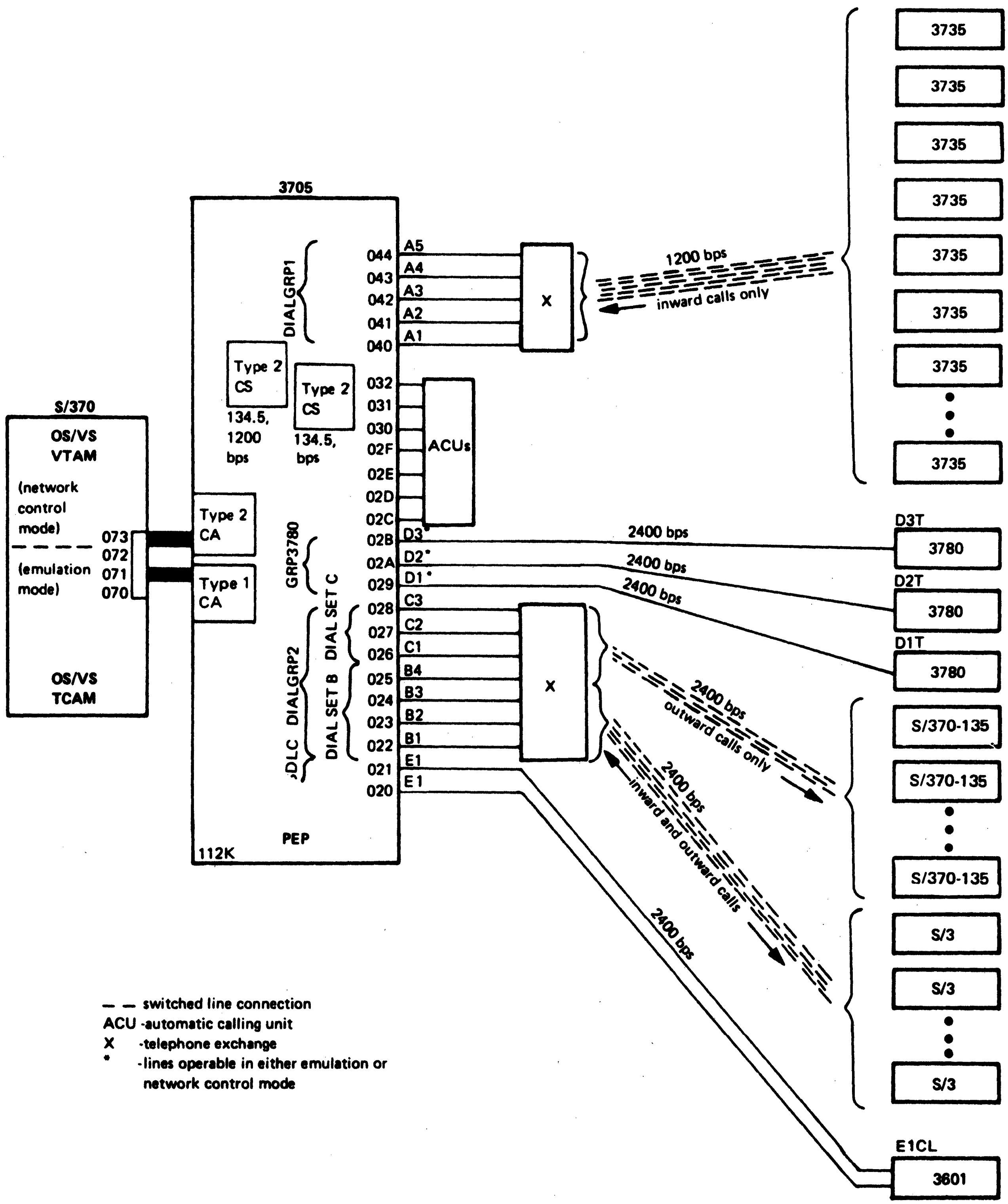
Three BSC switched call-out lines (network control only) for communicating with IBM System/370 Model 135s.

Three BSC nonswitched point-to-point lines (network control or emulation mode) for communicating with IBM 3780s.

One SDLC nonswitched point-to-point link (network control mode only) for communicating with an IBM 3601 controller.

Remarks: This sample program is for a network configuration considerably more complex than those shown in previous samples. Illustrated is the coding required to establish switched lines and dial sets. In this example, five lines are used for receiving calls only (call-in lines), three lines for originating calls only (call-out lines), and four lines for either originating or receiving calls (call-in/call-out lines). The lines for outgoing calls are grouped into two dial sets, with one of these serving as an alternate dial set to the other.

This example also includes a single SDLC link (network control mode only) and three point-to-point BSC lines operable alternately in emulation mode and network control mode.



— — switched line connection
ACU -automatic calling unit
X -telephone exchange
* -lines operable in either emulation or network control mode

```
*  SAMPLE PROGRAM 6
*
*  NETWORK CONTROL PROGRAM (LOCAL) WITH PARTITIONED EMULATION PROGRAM-
*     MING EXTENSION, GENERATED UNDER OS/VS, ACCESS METHOD:  OS/VS VTAM
*     FOR NETWORK CONTROL MODE LINES, OS/VS TCAM FOR EMULATION MODE LINES
*
SAMPLE7    BUILD     TYPGEN=PEP,                                                  X
                     MODEL=3705,                                                  X
                     MEMSIZE=176,                                                 X
                     SUBAREA=3,                                                   X
                     MAXSUBA=7,                                                   X
                     CA=(TYPE1,TYPE2),                                            X
                     CHANTYP=TYPE2,          TYPE 2: NC MODE, TYPE 1: EM MODE     X
                     HICHAN=073,             SUBCHANNEL RANGE FOR                 X
                     LOCHAN=070,                EMULATION MODE DATA TRANSFER      X
                     BFRS=80,                                                     X
                     ANS=NO,                 NO AUTO NETWORK SHUTDOWN             X
                     CUID=61C4C5D5E5D5D943C3D7E4F261,    ID:  /DENVER.CPU2/       X
                     ERASE=YES,              BUFFER ERASE FOR LN GPS DIALGRP1,2   X
                     TYPSYS=OS,                                                   X
                     LOADLIB=NCPLIB,                                              X
                     OBJLIB=STG2ASM,                                              X
                     NEWNAME=PEP001
           SYSCNTRL  OPTIONS=(MODE,RCNTRL,RCOND,RECMD,BHSASSC,                    X
                     RIMM,ENDCALL)
           HOST      INBFRS=3,                                                    X
                     UNITSZ=84,                                                   X
                     MAXBFRU=10,                                                  X
                     BFRPAD=28,                                                   X
                     STATMOD=YES,                                                 X
                     DELAY=.2
           CSB       TYPE=TYPE2,                                                  X
                     MOD=0,                                                       X
                     SPEED=134.
           CSB       TYPE=TYPE2,                                                  X
                     MOD=1,                                                       X
                     SPEED=(134,1200),                                            X
                     WRAPLN=023              LINE ADDRESS FOR WRAPLINE TEST
*                    ID LIST FOR 3735 TERMINALS
IDLISTA    IDLIST    IDSEQ=(61F2F0F061,61F2F0F1C161,            /200/,/201A/      X
                     61F2F0F1C261,61F2F0F561,61F2F0F761,        /201B/,/205/,/207/ X
                     61F5F0F961,61F5F1F261,61F5F1F7C161,        /509/,/512/,/517A/ X
                     61F5F1F7C261,61F5F1F7C361,61F5F3F861,      /517B/,/517C/,/538/ X
                     61F5F5F261,61F6F7F161,61F6F7F261,          /552/,/671/,/672/ X
                     61F7F0F1C161,61F7F0F1C261,                 /701A/,/701B/     X
                     NOMATCH=STOP
           IDLIST    IDSEQ=(61F7F0F661,61F7F0F761,              /706/,/707/       X
                     61F7F1F261,61F7F4F461)                     /712/,/744/
DIALSETB   DIALSET   LINES=(B1,B2,B3,B4),                                         X
                     DIALALT=DIALSETC,                                            X
                     RESERVE=2,              2 LINES ALWAYS RESERVED FOR          X
                     QLIMIT=3,                  INCOMING CALLS                    X
                     QLOAD=2
DIALSETC   DIALSET   LINES=(C1,C2,C3),                                            X
                     QLIMIT=3
DIALGRP1   GROUP     LNCTL=BSC,                                                   X
                     DIAL=YES,                                                    X
                     CALL=IN,                *                                    X
                     TYPE=NCP,                                                    X
                     CODE=EBCDIC,            *                                    X
                     SPEED=1200,             *                                    X
```

```
                  CLOCKNG=INT        *                                          X
                  TERM=3735,         *                                          X
                  CDATA=YES,         *                                          X
                  IDSEQ=IDLISTA      *                                          X
*                                    * --- OPERANDS OF LINE OR TERMINAL
*                                          MACROS CODED HERE TO SAVE
*                                          CODING EFFORT
A1        LINE     ADDRESS=040
CTA1      TERMINAL CTERM=YES
A2        LINE     ADDRESS=041
CTA2      TERMINAL CTERM=YES
A3        LINE     ADDRESS=042
CTA3      TERMINAL CTERM=YES
A4        LINE     ADDRESS=043
CTA4      TERMINAL CTERM=YES
A5        LINE     ADDRESS=044
CTA5      TERMINAL CTERM=YES
DIALGRP2  GROUP    LNCTL=BSC,                                                   X
                   DIAL=YES,                                                    X
                   REDIAL=5,         *     MAX. TIMES NCP WILL REDIAL           X
                   TYPE=NCP,                                                    X
                   CODE=EBCDIC,      *                                          X
                   SPEED=2400,       *                                          X
                   CLOCKNG=EXT,      *                                          X
                   TERM=SYS3,        *     (EITHER SYS3 OR 3135 ACCEPTABLE)     X
                   CDATA=YES         *
*                                    * --- LINE OR TERMINAL MACRO OPERANDS
B1        LINE     ADDRESS=022,                                                 X
                   AUTO=02C,                                                    X
                   CALL=INOUT,       LINE USED FOR CALLING IN AND OUT           X
                   DIALSET=DIALSETB
CTB1IN    TERMINAL CTERM=YES         LOGICAL TERMINAL FOR INCOMING CALLS
CTB1OUT   TERMINAL DIALNO=(,7),      LOG TERM FOR OUTGOING CALLS, VTAM          X
                   CUIDLEN=ALL          SUPPLIES TEL NO., NCP SENDS ENTIRE      X
                                        CU ID SEQUENCE
B2        LINE     ADDRESS=023,                                                 X
                   AUTO=02D,                                                    X
                   CALL=INOUT,                                                  X
                   DIALSET=DIALSETB
CTB2IN    TERMINAL CTERM=YES
CTB2OUT   TERMINAL DIALNO=(,7),                                                 X
                   CUIDLEN=ALL
B3        LINE     ADDRESS=024,                                                 X
                   AUTO=02E,                                                    X
                   CALL=INOUT,                                                  X
                   DIALSET=DIALSETB
CTB3IN    TERMINAL CTERM=YES
CTB3OUT   TERMINAL DIALNO=(,7),                                                 X
                   CUIDLEN=ALL
B4        LINE     ADDRESS=025,                                                 X
                   AUTO=02F,                                                    X
                   CALL=INOUT,                                                  X
                   DIALSET=DIALSETB
CTB4IN    TERMINAL CTERM=YES
CTB4OUT   TERMINAL DIALNO=(,7),                                                 X
                   CUIDLEN=ALL
C1        LINE     ADDRESS=026,                                                 X
                   AUTO=030,                                                    X
                   CALL=OUT,         LINE USED FOR CALLING OUT ONLY             X
                   DIALSET=DIALSETC  (NO ALTERNATE DIALSET FOR THIS LINE)
```

```
CTC1OUT    TERMINAL DIALNO=(,7),                                               X
                    CUIDLEN=ALL
C2         LINE     ADDRESS=027,                                               X
                    AUTO=031,                                                  X
                    CALL=OUT,                                                  X
                    DIALSET=DIALSETC
CTC2OUT    TERMINAL DIALNO=(,7),                                               X
                    CUIDLEN=ALL
C3         LINE     ADDRESS=028,                                               X
                    AUTO=032,                                                  X
                    CALL=OUT,                                                  X
                    DIALSET=DIALSETC
CTC3OUT    TERMINAL DIALNO=(,7),                                               X
                    CUIDLEN=ALL
GRP3780    GROUP    LNCTL=BSC,                                                 X
                    DIAL=NO,                                                   X
                    POLLED=NO,         *                                       X
                    TYPE=PEP,               GROUP OPERATES IN BOTH NETWORK     X
                    CODE=EBCDIC,       *       CONTROL AND EMULATION MODES     X
                    SPEED=2400,        *                                       X
                    CLOCKNG=EXT,       *                                       X
                    TERM=3780,         *                                       X
                    CU=2701,           *    TCU EMULATED IN EMULATION OPERN    X
                    YIELD=NO           *    (3705 IS PRIMARY STN ON LINE)
*                                      * -- LINE OR TERMINAL MACRO OPERANDS
D1         LINE     ADDRESS=(029,70),                                          X
                    TYPE=PEP,                                                  X
                    USE=EP             LINE OPERATES INITIALLY IN EMUL MODE
D1T        TERMINAL ,                  (SEE GROUP MACRO FOR OPERANDS)
D2         LINE     ADDRESS=(02A,71),                                          X
                    TYPE=PEP,                                                  X
                    USE=EP
D2T        TERMINAL
D3         LINE     ADDRESS=(02B,72),                                          X
                    TYPE=PEP,                                                  X
                    USE=EP
D3T        TERMINAL
GRPSDLC    GROUP    LNCTL=SDLC,                                                X
                    TYPE=NCP
E1         LINE     ADDRESS=(020,021),                                         X
                    SPEED=2400,                                                X
                    POLLED=YES
           SERVICE  ORDER=(E1CL)
E1CL       PU       ADDR=C1,           CLUSTER ADDRESS IS A (EBCDIC)           X
FAE1LU1X   LU       LOCADDR=1
E1LU2XXX   LU       LOCADDR=2
E1LU3XXX   LU       LOCADDR=3
           GENEND
           END
/*
//
```

Sample 7

The configuration below includes both local and remote communications controllers. Two sample programs are given—7-L and 7-R—for one of the local controllers and the remote controller. (The program for the other local controller [subarea5] attached to the BSC line is not included.)

Lines and Stations—Local:
Three start-stop lines (emulation mode only):

One nonswitched multipoint line, IBM 2848/2260
One nonswitched point-to-point line, IBM 2848/2260
One nonswitched multipoint line, IBM 2845/2265

One BSC nonswitched multipoint line, IBM 2701, 3705, and 1130 (network control mode only)

One SDLC nonswitched point-to-point (principal) link to a remote 3705.

One SDLC switched point-to-point (backup) link to the same remote 3705 (nonswitched line control used).

Lines and Stations—Remote:
Three BSC nonswitched point-to-point lines, IBM 3780.

Five BSC switched call-in lines, IBM 3735

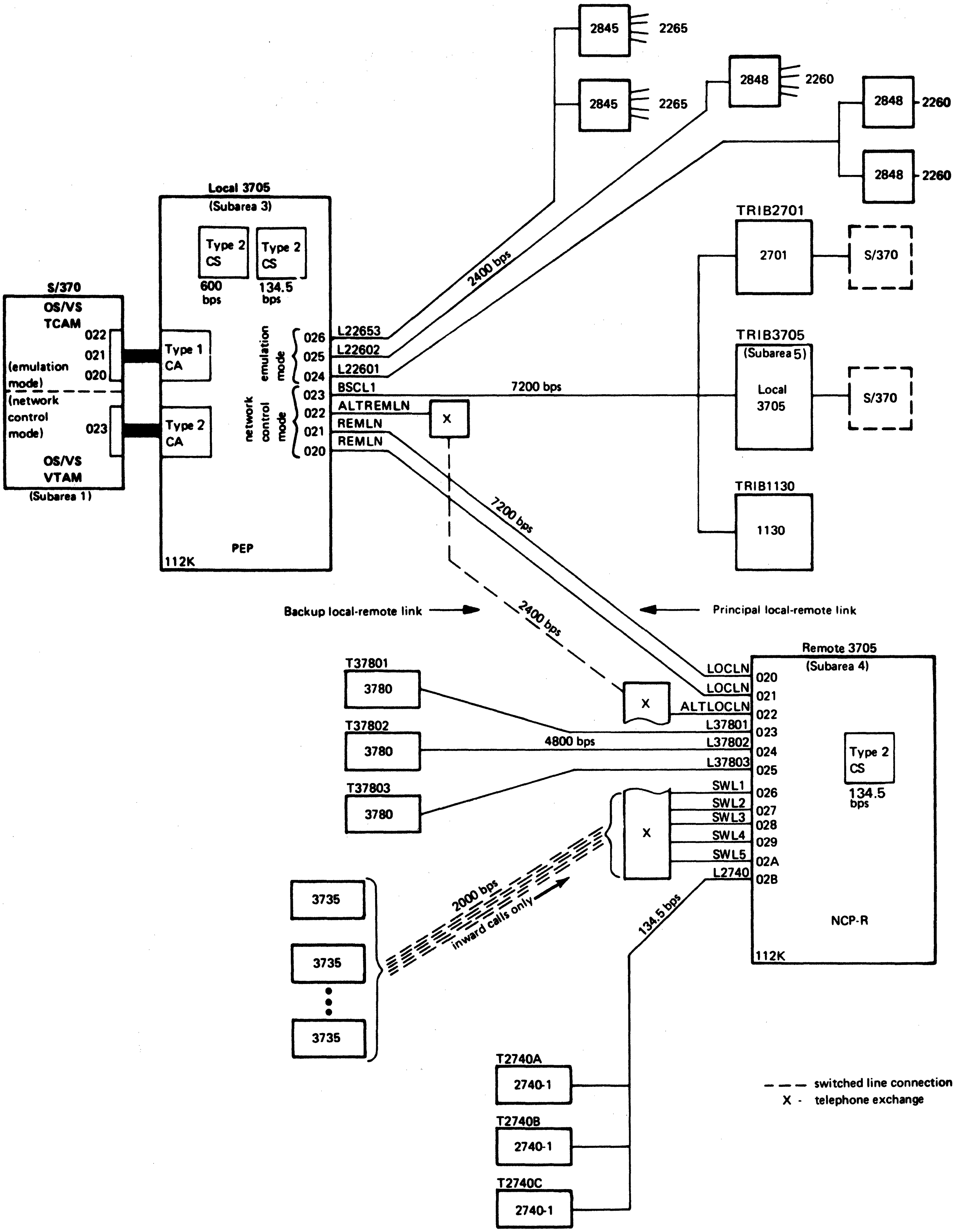
One start-stop nonswitched multipoint line, IBM 2740 Model 1.

One SDLC nonswitched point-to-point (principal) link to a local 3705.

One SDLC switched point-to-point (backup) link to the same local 3705 (nonswitched line control used).

Remarks—Sample 7-L: This sample program, executed in a local 3705, supports emulation mode lines, network control mode lines, and a remote communications controller; it is therefore specified as TYPGEN=PEP-LR. Notice that the 3705 in which the program will be executed is equipped with separate channel adapters for network control mode and emulation mode data transfer. Also notice the use of the address substitution option (GENEND macro), required because of the presence of lines operating at 7200 bps.

Remarks—Sample 7-R: Because the sample program is to be executed in a remote 3705, it is specified as TYPGEN=NCP-R. Notice the absence of channel information in the BUILD macro; the remote controller has no channel adapter. The operands and parameters of the HOST macro are specified identically to those in the HOST macro of the local NCP (sample 7-L). Notice further that because all BSC switched lines are used only for incoming calls, no dial sets are specified, as is the case in sample 6. Finally, observe that most operands of the LINE and TERMINAL macros specifiable at a higher level appear in the GROUP macro to save coding effort.


2845
2265
2845
2265
2848
2260
2848
2260
2848
2260
Local 3705
(Subarea 3)
Type 2 CS
Type 2 CS
600 bps
134.5 bps
S/370
OS/VS
TCAM
022
021
020
(emulation mode)
(network control mode)
023
OS/VS
VTAM
(Subarea 1)
Type 1 CA
Type 2 CA
emulation mode
network control mode
026
025
024
023
022
021
020
L22653
L22602
L22601
BSCL1
ALTREMLN
REMLN
REMLN
2400 bps
7200 bps
PEP
112K
TRIB2701
2701
S/370
TRIB3705
(Subarea 5)
Local 3705
S/370
TRIB1130
1130
X
7200 bps
2400 bps
Backup local-remote link
Principal local-remote link
Remote 3705
(Subarea 4)
T37801
3780
T37802
3780
T37803
3780
4800 bps
X
LOCLN
LOCLN
ALTLOCLN
L37801
L37802
L37803
SWL1
SWL2
SWL3
SWL4
SWL5
L2740
020
021
022
023
024
025
026
027
028
029
02A
02B
Type 2 CS
134.5 bps
NCP-R
112K
X
3735
3735
3735
2000 bps
inward calls only
134.5 bps
T2740A
2740-1
T2740B
2740-1
T2740C
2740-1
switched line connection
X - telephone exchange

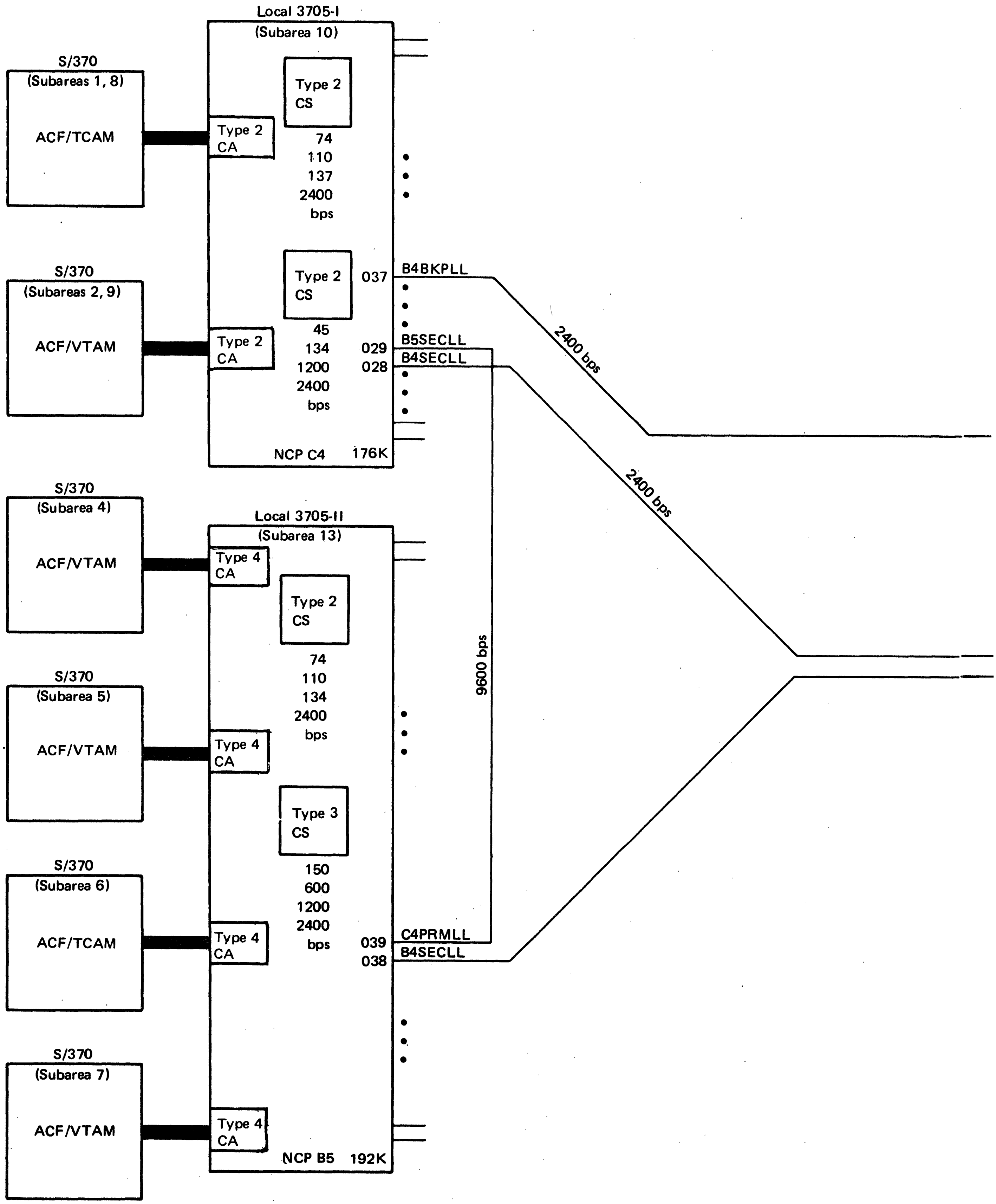
```
*  SAMPLE PROGRAM 7-L
*
*  NETWORK CONTROL PROGRAM WITH PARTITIONED EMULATION PROGRAMMING EXTEN-
*     SION (LOCAL), GENERATED UNDER OS/VS, ACCESS METHOD: OS/VS VTAM FOR NETWORK
*     CONTROL MODE LINES, OS/VS TCAM FOR EMULATION MODE LINES
*
SAMPLE8L    BUILD     MODEL=3705,                                              X
                      MEMSIZE=112,                                             X
                      TYPGEN=PEP,                                              X
                      SUBAREA=3,                                               X
                      MAXSUBA=7,                                               X
                      CA=(TYPE1,TYPE2),                                        X
                      CHANTYP=TYPE2,         TYPE 2 CA FOR NETW CTL MODE DATA  X
                      HICHAN=023,            TRANSFER, TYPE 1 FOR EMUL MODE    X
                      LOCHAN=020,                                              X
                      ANS=YES,               AUTO NETWORK SHUTDOWN             X
                      BFRS=80,                                                 X
                      TYPSYS=OS,                                               X
                      LOADLIB=NCPLIB,                                          X
                      OBJLIB=STG2ASM,                                          X
                      NEWNAME=LOCNCP
            SYSCNTRL  OPTIONS=(MODE,RCNTRL,RCOND,RECMD,RIMM,                   X
                      ENDCALL,BHSASSC)
            HOST      INBFRS=3,                                                X
                      MAXBFRU=10,                                              X
                      BFRPAD=28,                                               X
                      UNITSZ=84,                                               X
                      DELAY=.2,                                                X
                      STATMOD=YES,                                             X
                      SUBAREA=1
            CSB       TYPE=TYPE2,                                              X
                      MOD=0,                                                   X
                      SPEED=600,                                               X
                      WRAPLN=023
            CSB       TYPE=TYPE2,                                              X
                      MOD=1,                                                   X
                      SPEED=134,                                               X
                      WRAPLN=0A3
BSCGRP      GROUP     LNCTL=BSC,                                               X
                      TYPE=NCP
BSCL1       LINE      CODE=EBCDIC,                                             X
                      SPEED=7200,                                              X
                      CLOCKNG=EXT,                                             X
                      ADDRESS=023,                                             X
                      POLLED=YES
                      POLIMIT=(1,QUEUE)
            SERVICE   ORDER=(TRIB3705,TRIB2701,TRIB1130)
TRIB3705    TERMINAL  TERM=3705,                                               X
                      ADDR=E1,                                                 X
                      POLL=C1,                                                 X
                      XMITLIM=1
TRIB2701    TERMINAL  TERM=2701,                                               X
                      ADDR=E2,                                                 X
                      POLL=C2,                                                 X
                      XMITLIM=1
TRIB1130    TERMINAL  TERM=1130,                                               X
                      ADDR=E3,                                                 X
```

```
                      POLL=C3,                                                        X
                      XMITLIM=1
EM2260GP    GROUP     LNCTL=SS,                                                       X
                      TYPE=EP,                                                        X
                      SPEED=2400,                                                     X
                      CLOCKNG=EXT,                                                    X
                      CU=2701,                                                        X
                      DUPLEX=FULL
                      POLIMIT=(1,QUEUE)
L22601      LINE      TERM=2260,                                                      X
                      MULTI=YES,          MULTIPLE DISPLAYS ON LINE                   X
                      ADDRESS=(024,20)
L22602      LINE      TERM=2260,                                                      X
                      MULTI=NO,                                                       X
                      ADDRESS=(025,21)
L22653      LINE      TERM=2265,                                                      X
                      MULTI=YES,                                                      X
                      ADDRESS=(026,22)
REMOTEGP    GROUP     LNCTL=SDLC,                                                     X
                      TYPE=NCP
REMLN       LINE      ADDRESS=(020,021),  PRINCIPAL SDLC LINK TO REMOTE CTLR          X
                      DUPLEX=FULL,        COMMUNICATION FACILITY IS DUPLEX            X
                      SPEED=7200,                                                     X
                      CLOCKNG=EXT,                                                    X
                      POLLED=YES,                                                     X
                      RETRIES=(3,10,10)
            SERVICE   ORDER=REM3705A
REM3705A    PU        ADDR=C1,            PHYSICAL ADDR (A) OF REMOTE 3705            X
                      PUTYPE=4,           TYPE 4 PHYSICAL UNIT                        X
                      MAXOUT=4,           MAX OF 4 PIU'S SENT BEFORE RESPONSE         X
                      DATMODE=FULL,       DATA XFER SIMULTANEOUS IN BOTH DIREC-       X
                      SUBAREA=4           SUBA ADDR OF REMOTE 3705           TIONS
ALTREMLN    LINE      ADDRESS=022,        ALT (BACKUP) SDLC LINK TO REM CTLR          X
                      DUPLEX=HALF,                                                    X
                      SPEED=2400,                                                     X
                      CLOCKNG=EXT,                                                    X
                      POLLED=YES,                                                     X
                      RETRIES=(3,10,10)
            SERVICE   ORDER=REM3705B
REM3705B    PU        PUTYPE=4
            GENEND    SCANCTL=(,,,,1100)  ADDR SUBSTITUTION MASK --- ADDR SUBST       X
                                          REQUIRED TO ACCOMMODATE 7200 BPS LINES

            END
/*
//
```

```
*  SAMPLE PROGRAM 7-R
*
*  NETWORK CONTROL PROGRAM (REMOTE), GENERATED UNDER OS/VS, ACCESS
*     METHOD: OS/VS VTAM
*
SAMPLE8R   BUILD     MODEL=3705,                                                        X
                     MEMSIZE=112,                                                       X
                     TYPGEN=NCP-R,                                                      X
                     SUBAREA=4,              (NOTE ABSENCE OF CHANNEL                   X
                     MAXSUBA=7,              INFORMATION, NOT APPLICABLE                X
                     BFRS=80,                FOR A REMOTE NCP)                          X
                     TYPSYS=OS,                                                         X
                     ANS=YES,                                                           X
                     LOADLIB=NCPLIB,                                                    X
                     OBJLIB=STG2ASM,                                                    X
                     NEWNAME=REMNCP
           SYSCNTRL  OPTIONS=(MODE,RCNTRL,RCOND,RECMD,                                  X
                     RIMM,ENDCALL,BHSASSC)
           PATH      ADJSUB=3,DESTSUB=1      PATH FROM THIS NCP TO SUBAREA 1
           CSB       TYPE=TYPE2,                                                        X
                     SPEED=134
GP3780     GROUP     LNCTL=BSC,                                                         X
                     CODE=EBCDIC,                                                       X
                     SPEED=4800,                                                        X
                     CLOCKNG=EXT,                                                       X
                     TERM=3780
                     POLIMIT=(1,QUEUE
L37801     LINE      ADDRESS=023
T37801     TERMINAL
L37802     LINE      ADDRESS=024
T37802     TERMINAL
L37803     LINE      ADDRESS=025
T37803     TERMINAL
SW3735GP   GROUP     LNCTL=BSC,                                                         X
                     SPEED=2000,                                                        X
                     CLOCKNG=EXT,                                                       X
                     TERM=3735,                                                         X
                     CALL=IN,                                                           X
                     DIAL=YES,                                                          X
                     CODE=EBCDIC
                     POLIMIT=(1,QUEUE)
SWL1       LINE      ADDRESS=026
SWT1       TERMINAL  CTERM=YES
SWL2       LINE      ADDRESS=027
SWT2       TERMINAL  CTERM=YES
SWL3       LINE      ADDRESS=028
SWT3       TERMINAL  CTERM=YES
SWL4       LINE      ADDRESS=029
SWT4       TERMINAL  CTERM=YES
SWL5       LINE      ADDRESS=02A
SWT5       TERMINAL  CTERM=YES
GP2740     GROUP     LNCTL=SS,                                                          X
                     SPEED=134,                                                         X
                     CODE=COR,                                                          X
                     POLLED=YES,                                                        X
                     CLOCKNG=INT,                                                       X
                     TERM=2740-1,                                                       X
                     FEATURE=SCTL            TERMINALS HAVE STN CONTROL FEATURE         X
                     XMITLIM=1               TRANSMISSION LIMIT FOR TERMINALS
                     POLIMIT=(1,QUEUE)
L2740      LINE      ADDRESS=02B
           SERVICE   ORDER=(T2740A,T2740B,T2740C)
T2740A     TERMINAL  POLL=C1,                POLLING AND ADDRESSING CHARACTER:          X
                     ADDR=C1                 A  (EBCDIC)
T2740B     TERMINAL  POLL=C2,                B  (EBCDIC)                                X
                     ADDR=C2
T2740C     TERMINAL  POLL=C3,                C  (EBCDIC)                                X
                     ADDR=C3
```

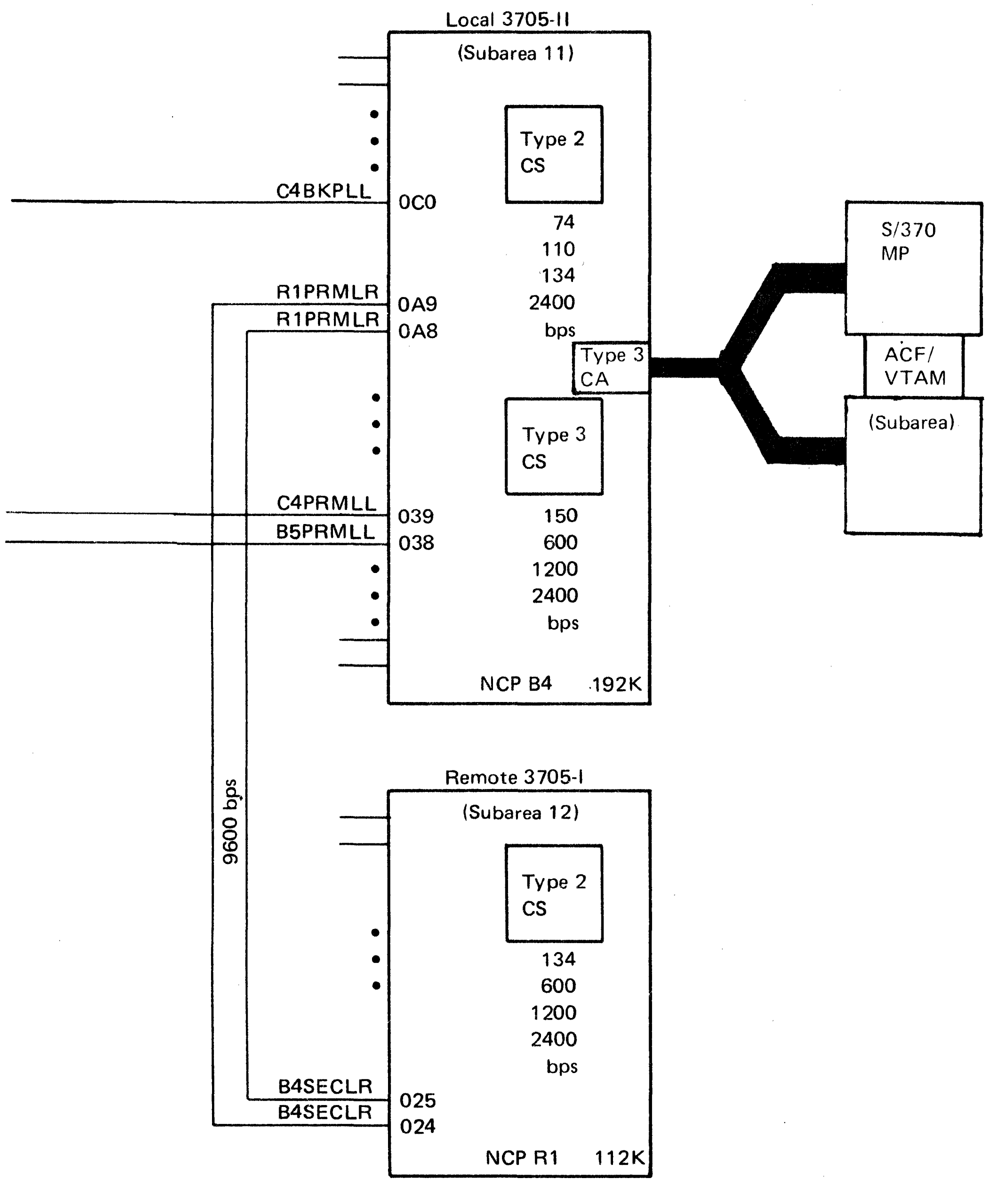
```
SDLCGP      GROUP    LNCTL=SDLC,                                                          X
                     DIAL=NO
LOCLN       LINE     ADDRESS=(020,021),   PRINCIPAL SDLC LINK TO LOCAL CTLR               X
                     SPEED=7200,                                                          X
                     CLOCKNG=(EXT,EXT),                                                   X
                     TADDR=C1,            PHYSICAL ADDR OF REMOTE CONTROLLER              X
                     NEWSYNC=NO,          IN WHICH THIS PROGRAM WILL BE                   X
                     DUPLEX=FULL          EXECUTED --- NOTE THAT ADDR IS SAME AS          X
                                          SPECIFIED IN PU MACRO OF LOCAL NCP
LOC3705A    PU       PUTYPE=4             TYPE 4 PHYS UNIT; MAXOUT DEFAULT=7
ALTLOCLN    LINE     ADDRESS=022,         ALT (BACKUP) SDLC LINK TO LOCAL CTLR            X
                     SPEED=2400,                                                          X
                     CLOCKNG=EXT,                                                         X
                     TADDR=C1,                                                            X
                     NEWSYNC=NO,                                                          X
                     DUPLEX=FULL
LOC3705B    PU       PUTYPE=4
            GENEND
            END
/*
//
```

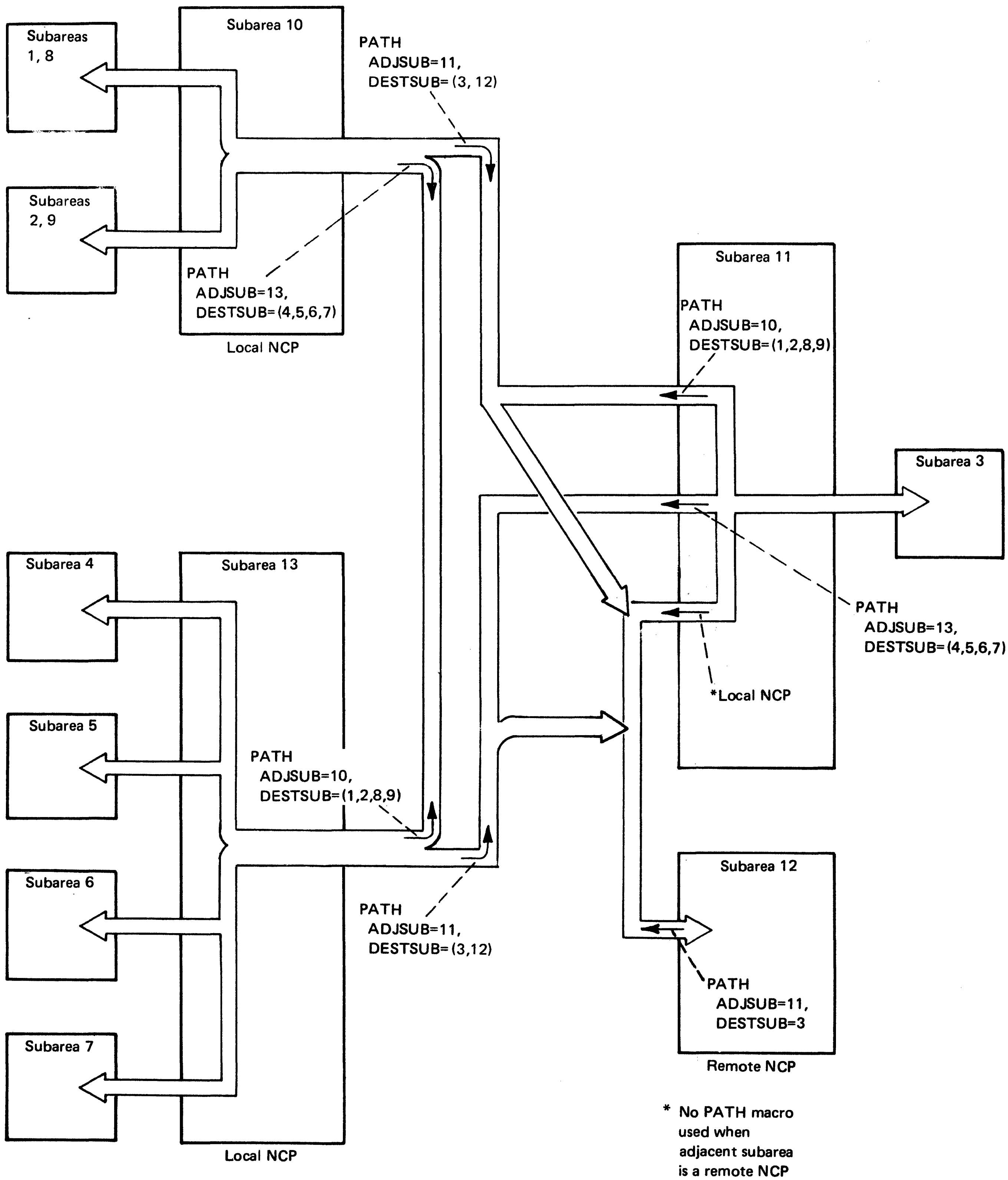


Local 3705-I
(Subarea 10)
S/370
(Subareas 1, 8)
ACF/TCAM
Type 2
CS
Type 2
CA
74
110
137
2400
bps
S/370
(Subareas 2, 9)
ACF/VTAM
Type 2
CS
037
B4BKPLL
2400 bps
Type 2
CA
45
134
1200
2400
bps
029
B5SECLL
028
B4SECLL
NCP C4
176K
2400 bps
S/370
(Subarea 4)
ACF/VTAM
Local 3705-II
(Subarea 13)
Type 4
CA
Type 2
CS
74
110
134
2400
bps
9600 bps
S/370
(Subarea 5)
ACF/VTAM
Type 4
CA
Type 3
CS
150
600
1200
2400
bps
S/370
(Subarea 6)
ACF/TCAM
Type 4
CA
039
C4PRMLL
038
B4SECLL
S/370
(Subarea 7)
ACF/VTAM
Type 4
CA
NCP B5
192K

Sample 8

The configuration illustrated on these pages includes three 3705s (attached to a total of seven host processors) in which local network control programs are executed and one 3705 in which a remote NCP is executed, along with the local-local and local-remote links joining the 3705s. Since this example is intended to show the network relationships between the 3705s and host processors, as established by the link and path specifications in the NCP source statements that follow, other SDLC, BSC, and start-stop stations and their links are omitted. On the next page appears the network paths established by the PATH macros in the sample programs.




Subareas 1, 8
Subarea 10
PATH ADJSUB=11, DESTSUB=(3, 12)
Subareas 2, 9
PATH ADJSUB=13, DESTSUB=(4,5,6,7)
Local NCP
Subarea 11
PATH ADJSUB=10, DESTSUB=(1,2,8,9)
Subarea 3
PATH ADJSUB=13, DESTSUB=(4,5,6,7)
*Local NCP
Subarea 4
Subarea 13
Subarea 5
PATH ADJSUB=10, DESTSUB=(1,2,8,9)
Subarea 6
PATH ADJSUB=11, DESTSUB=(3,12)
Subarea 12
PATH ADJSUB=11, DESTSUB=3
Remote NCP
Subarea 7
Local NCP
* No PATH macro used when adjacent subarea is a remote NCP

```
*   SAMPLE PROGRAM 8-C4
*
*   NETWORK CONTROL PROGRAM (LOCAL), GENERATED UNDER OS/VS, ACCESS METHODS:
*      ACF/VTAM, ACF/TCAM
*
C4NCP      BUILD     TYPGEN=NCP,            SAMPLE PROGRAM 8-C4              X
                     SUBAREA=10,                                             X
                     MAXSUBA=15,                                             X
                     MAXSSCP=8,                                              X
                     MODEL=3705,                                             X
                     MEMSIZE=176,                                            X
                     CA=(TYPE2,TYPE2),                                       X
                     NCPCA=(ACTIVE,ACTIVE),                                  X
                     CATRACE=(YES,255),                                      X
                     BFRS=60,                                                X
                     ANS=YES,                                                X
                     ABEND=YES,                                              X
                     PNLTEST=YES,                                            X
                     DIALTO=60,                                              X
                     ENABLTO=0.1,                                            X
                     SLODOWN=12,                                             X
                     TRACE=(YES,50),                                         X
                     NEWNAME=C4NCP,                                          X
                     LOADLIB=LC4NCP,                                         X
                     OBJLIB=C4EY252,                                         X
                     JOBCARD=MULTI,                                          X
                     LESIZE=450,                                             X
                     ASMXREF=YES,                                            X
                     PARTIAL=NO,                                             X
                     QUALIFY=SN50,                                           X
                     UNIT=SORTWK
           SYSCNTRL  OPTIONS=(BHSASSC,RIMM,RCOND,BACKUP,LNSTAT,SESINIT,DVSINIT, X
                     SESSION,SSPAUSE,XMTLMT,MODE,RECMD,RCNTRL,ENDCALL)
           HOST      INBFRS=3,              REPRESENTS HOST PROCESSOR 1      X
                     SUBAREA=(1,8),                                          X
                     MAXBFRU=8,                                              X
                     UNITSZ=256,                                             X
                     BFRPAD=17,             ACF/TCAM MIN. BUFFER PAD         X
                     DELAY=4,                                                X
                     STATMOD=YES
           HOST      INBFRS=3,              REPRESENTS HOST PROCESSOR 2      X
                     SUBAREA=(2,9),                                          X
                     MAXBFRU=8,                                              X
                     UNITSZ=256,                                             X
                     BFRPAD=0,              ACF/VTAM BUFFER PAD VALUE        X
                     DELAY=4,                                                X
                     STATMOD=YES
           PATH      ADJSUB=11,                                              X
                     DESTSUB=(3,12)
           PATH      ADJSUB=13,                                              X
                     DESTSUB=(4,5,6,7)
           CSB       SPEED=(74,110,134,2400)                                 X
                     MOD=0,                                                  X
                     TYPE=TYPE2
           CSB       SPEED=(45,134,1200,2400),                               X
                     MOD=1,                                                  X
                     TYPE=TYPE2
LUP001     LUPOOL    NUMBER=8
*
* * * * *  MACRO INSTRUCTIONS FOR NON-SDLC STATIONS OMITTED          * * * * *
*
* * * * *  MACRO INSTRUCTIONS FOR SDLC STATIONS FOLLOW:               * * * * *
*
SDLCGRP    GROUP     LNCTL=SDLC,                                             X
                     ACTIVTO=420.0,                                          X
                     TYPE=NCP,                                               X
```

```
                        DIAL=NO,                                              X
                        REPLYTO=3
B4SECLL      LINE       ADDRESS=028,                                          X
                        CLOCKNG=EXT,                                          X
                        DUPLEX=FULL,                                          X
             TADDR-     TADDR=C1,                                             X
                        SPEED=2400,                                           X
                        NRZI=YES,                                             X
                        POLLED=NO
B4SECPU      PU         ANS=CONT,                                             X
                        PUTYPE=(4,LOCAL),                                     X
                        SUBAREA=11
B5SECLL      LINE       ADDRESS=029,                                          X
                        CLOCKNG=EXT,                                          X
                        DUPLEX=FULL,                                          X
                        TADDR=C1,                                             X
                        SPEED=2400,                                           X
                        NRZI=YES,                                             X
                        POLLED=NO
B5SECPU      PU         ANS=CONT,                                             X
                        PUTYPE=(4,LOCAL),                                     X
                        SUBAREA=13
B4BKPLL      LINE       ADDRESS=037,                                          X
                        CLOCKNG=EXT,                                          X
                        DUPLEX=FULL,                                          X
                        SPEED=2400,                                           X
                        NRZI=YES,                                             X
                        POLLED=NO,                                            X
                        TADDR=C1
B4BKPPU      PU         ANS=CONT,                                             X
                        PUTYPE=(4,LOCAL)
*
* * * * *  (OTHER GROUP, LINE, SERVICE, AND PU MACROS FOR SDLC STATIONS)  * * * * *
*
             GENEND
             END
/*
//
```

```
*  SAMPLE PROGRAM 8-B4
*
*  NETWORK CONTROL PROGRAM (LOCAL), GENERATED UNDER OS/VS, ACCESS METHOD:
*     ACF/VTAM
*
B4NCP      BUILD     TYPGEN=NCP,              SAMPLE PROGRAM 8-B4            X
                     SUBAREA=11,                                             X
                     MAXSUBA=15,                                             X
                     MAXSSCP=8,                                              X
                     MODEL=3705-2,                                           X
                     MEMSIZE=192,                                            X
                     CA=TYPE3,                                               X
                     CHANTYP=TYPE3,                                          X
                     CATRACE=(YES,255),                                      X
                     BFRS=60,                                                X
                     ANS=YES,                                                X
                     ABEND=YES,                                              X
                     PNLTEST=YES,                                            X
                     DIALTO=60,                                              X
                     ENABLTO=0.1,                                            X
                     SLODOWN=12,                                             X
                     TRACE=(YES,50),                                         X
                     NEWNAME=B4NCP,                                          X
                     LOADLIB=LB4NCP,                                         X
                     OBJLIB=B4SNA52,                                         X
                     JOBCARD=NO,                                             X
                     LESIZE=450,                                             X
                     ASMXREF=YES,                                            X
                     PARTIAL=NO,                                             X
                     QUALIFY=SN50,                                           X
                     UNIT=SORTWK
           SYSCNTRL  OPTIONS=(BHSASSC,RIMM,RCOND,BACKUP,LNSTAT,SESINIT,DVSINIT,   X
                     NAKLIM,SESSION,SSPAUSE,XMTLMT,MODE,RECMD,RCNTRL,ENDCALL)
           HOST      INBFRS=3,                REPRESENTS HOST PROCESSOR 3    X
                     TIMEOUT=420.0,                                          X
                     SUBAREA=3,                                              X
                     MAXBFRU=8,                                              X
                     UNITSZ=256,                                             X
                     BFRPAD=0,                ACF/VTAM BUFFER PAD VALUE      X
                     DELAY=4
           CSB       SPEED=(74,110,134,2400),                                X
                     MOD=0,                                                  X
                     TYPE=TYPE2
           CSB       SPEED=(150,600,1200,2400)                               X
                     MOD=1,                                                  X
                     TYPE=TYPE3
           PATH      ADJSUB=10,                                              X
                     DESTSUB=(1,2,8,9)
           PATH      ADJSUB=13,                                              X
                     DESTSUB=(4,5,6,7)
LUP001     LUPOOL    NUMBER=8
*
*  *  *  *  *  MACRO INSTRUCTIONS FOR NON-SDLC STATIONS OMITTED      *  *  *  *  *
*
*  *  *  *  *  MACRO INSTRUCTIONS FOR SDLC STATIONS FOLLOW:          *  *  *  *  *
*
SDLCGRP    GROUP     LNCTL=SDLC,                                             X
                     TYPE=NCP,                                               X
                     DIAL=NO,                                                X
                     ACTIVTO=420.0,                                          X
                     REPLYTO=3
C4PRMLL    LINE      ADDRESS=039,             PRINCIPAL LOCAL-LOCAL LINK TO  X
                     CLOCKNG=EXT,               NCP C4                       X
                     DUPLEX=FULL,                                            X
                     SPEED=2400,                                             X
                     NRZI=YES,                                               X
                     POLLED=YES,                                             X
```

```
                    RETRIES=(,3,5)
          SERVICE   ORDER=C4PRMPU
C4PRMPU   PU        ANS=CONT,                                               X
                    ADDR=C1,                                                X
                    RETRIES=(,3,5),                                         X
                    IRETRY=YES,                                             X
                    PUTYPE=(4,LOCAL)                                        X
                    SUBAREA=10
C4BKPLL   LINE      ADDRESS=0C0,          BACKUP LOCAL-LOCAL LINK TO        X
                    CLOCKNG=EXT,            NCP C4                          X
                    DUPLEX=FULL,                                            X
                    SPEED=2400,                                             X
                    NRZI=YES,                                               X
                    POLLED=YES,                                             X
                    RETRIES=(,3,5)
          SERVICE   ORDER=C4BKPPU
C4BKPPU   PU        ANS=CONT,                                               X
                    PUTYPE=(4,LOCAL).
B5PRMLL   LINE      ADDRESS=038,          PRINCIPAL LOCAL-LOCAL LINK TO     X
                    CLOCKNG=EXT,            NCP B5                          X
                    DUPLEX=FULL,                                            X
                    SPEED=2400,                                             X
                    NRZI=YES,                                               X
                    POLLED=YES
          SERVICE   ORDER=B5PRMPU
B5PRMPU   PU        ANS=CONT,                                               X
                    ADDR=C1,                                                X
                    RETRIES=(,3,5),                                         X
                    IRETRY=YES,                                             X
                    PUTYPE=(4,LOCAL),                                       X
                    SUBAREA=13
R1PRMLR   LINE      ADDRESS=(0A8,0A9),    LINK TO REMOTE NCP R1             X
                    SPEED=9600,                                             X
                    CLOCKNG=EXT,                                            X
                    DUPLEX=FULL,                                            X
                    POLLED=YES,                                             X
                    RETRIES=15,                                             X
                    NRZI=NO
          SERVICE   ORDER=R1PRMPU
R1PRMPU   PU        PUTYPE=(4,REMOTE),                                      X
                    ANS=CONT,                                               X
                    IRETRY=YES
*
* * * * * (OTHER GROUP, LINE, SERVICE, AND PU MACROS FOR SDLC STATIONS) * * * * *
*
          GENEND
          END
/*
//
```

```
*  SAMPLE PROGRAM 8-B5
*
*  NETWORK CONTROL PROGRAM (LOCAL), GENERATED UNDER OS/VS, ACCESS METHODS:
*      ACF/TCAM, ACF/VTAM
*
B5NCP     BUILD     LOADLIB=LB5NCP,             SAMPLE PROGRAM 8-B5          X
                    OBJLIB=B5NCP,                                            X
                    JOBCARD=NO,                                              X
                    CATRACE=(YES,255),                                       X
                    LESIZE=450,                                              X
                    MAXSSCP=8,                                               X
                    QUALIFY=SN50,                                            X
                    MEMSIZE=192,                                             X
                    NCPCA=(ACTIVE,ACTIVE,ACTIVE,ACTIVE),                     X
                    CA=(TYPE4,TYPE4,TYPE4,TYPE4),                            X
                    MODEL=3705-2,                                            X
                    TYPGEN=NCP,                                              X
                    PWROFF=YES,                                              X
                    MAXSUBA=15,                                              X
                    SUBAREA=13,                                              X
                    ANS=YES,                                                 X
                    ENABLTO=0.1,                                             X
                    ABEND=YES,                                               X
                    BFRS=60,                                                 X
                    PNLTEST=YES,                                             X
                    NEWNAME=B5NCP,                                           X
                    DIALTO=60,                                               X
                    ASMXREF=YES,                                             X
                    PARTIAL=NO,                                              X
                    SLODOWN=12,                                              X
                    TRACE=(YES,50),                                          X
                    UNIT=SORTWK
          SYSCNTRL  OPTIONS=(BHSASSC,RIMM,RCOND,BACKUP,LNSTAT,SESINIT,       X
                    DVSINIT,NAKLIM,SESSION,SSPAUSE,XMTLMT,MODE,RECMD,        X
                    RCNTRL,ENDCALL)
          HOST      INBFRS=3,                   REPRESENTS HOST PROCESSOR 4  X
                    SUBAREA=4,                                               X
                    MAXBFRU=8,                                               X
                    UNITSZ=256,                                              X
                    BFRPAD=0,                   ACF/VTAM BUFFER PAD VALUE    X
                    DELAY=4
          HOST      INBFRS=3,                   REPRESENTS HOST PROCESSOR 5  X
                    SUBAREA=5,                                               X
                    MAXBFRU=8,                                               X
                    UNITSZ=256,                                              X
                    BFRPAD=0,                   ACF/VTAM BUFFER PAD VALUE    X
                    DELAY=4
          HOST      INBFRS=3,                   REPRESENTS HOST PROCESSOR 6  X
                    SUBAREA=6,                                               X
                    MAXBFRU=8,                                               X
                    UNITSZ=256,                                              X
                    BFRPAD=17,                  ACF/TCAM MIN BUFFER PAD VALUE X
                    DELAY=4
          HOST      INBFRS=3,                   REPRESENTS HOST PROCESSOR 7  X
                    SUBAREA=7,                                               X
                    MAXBFRU=8,                                               X
                    UNITSZ=256,                                              X
                    BFRPAD=0,                   ACF/VTAM BUFFER PAD VALUE    X
                    DELAY=4
          PATH      ADJSUB=10,                                               X
                    DESTSUB=(1,2,8,9)
          PATH      ADJSUB=11,                                               X
                    DESTSUB=(3,12)
          CSB       SPEED=(74,110,134,2400),                                 X
                    MOD=0,                                                   X
                    TYPE=TYPE3
LUP001    LUPOOL    NUMBER=8
```

```
*
* * * * *  MACRO INSTRUCTIONS FOR NON-SDLC STATIONS OMITTED             * * * * *
*
* * * * *  MACRO INSTRUCTIONS FOR SDLC STATIONS FOLLOW:                 * * * * *
*
GPB4SEC    GROUP     LNCTL=SDLC,                                                X
                     ACTIVTO=420.0,                                             X
                     TYPE=NCP,                                                  X
                     DIAL=NO,                                                   X
                     REPLYTO=3
B4SECLL    LINE      ADDRESS=038,'                                              X
                     CLOCKNG=EXT,                                               X
                     DUPLEX=FULL,                                               X
                     TADDR=C1,                                                  X
                     SPEED=2400,                                                X
                     NRZI=YES,                                                  X
                     POLLED=NO
B4SECPU    PU        ANS=CONT,                                                  X
                     PUTYPE=(4,,LOCAL)                                          X
                     SUBAREA=11
GPC4PRI    GROUP     LNCTL=SDLC,                                                X
                     TYPE=NCP,                                                  X
                     ACTIVTO=420.0,                                             X
                     DIAL=NO,                                                   X
                     REPLYTO=3
C4PRMLL    LINE      ADDRESS=039,                                               X
                     CLOCKNG=EXT,                                               X
                     DUPLEX=FULL,                                               X
                     SPEED=2400,                                                X
                     NRZI=YES,                                                  X
                     POLLED=YES,                                                X
                     RETRIES=(,3,5)
           SERVICE   ORDER=C4PRMPU
C4PRMPU    PU        ANS=CONT,                                                  X
                     ADDR=C1,                                                   X
                     RETRIES=(,3,5),                                            X
                     IRETRY=YES,                                                X
                     PUTYPE=(4,LOCAL),                                          X
                     SUBAREA=10
*
* * * * *  (OTHER GROUP, LINE, SERVICE, AND PU MACROS FOR SDLC STATIONS)  * * * * *
*
           GENEND
           END
/*
//
```

```
*   SAMPLE PROGRAM 8-R1
*
*   NETWORK CONTROL PROGRAM (REMOTE), GENERATED UNDER OS/VS
*
R1NCP        BUILD    TYPGEN=NCP-R,               SAMPLE PROGRAM 8-R1            X
                      SUBAREA=12,                                                X
                      MAXSUBA=15,                                                X
                      MODEL=3705,                                                X
                      MEMSIZE=112,                                               X
                      BFRS=60,                                                   X
                      ANS=YES,                                                   X
                      ABEND=YES,                                                 X
                      PNLTEST=YES,                                               X
                      SLODOWN=12,                                                X
                      TRACE=(YES,50),                                            X
                      NEWNAME=R1NCP,                                             X
                      LOADLIB=LR1NCP,                                            X
                      OBJLIB=R1EY152,                                            X
                      JOBCARD=MULTI,                                             X
                      LESIZE=350,                                                X
                      ASMXREF=YES,                                               X
                      PARTIAL=NO,                                                X
                      OLT=YES,                                                   X
                      QUALIFY=SN50,                                              X
                      UNIT=SORTWK
             SYSCNTRL OPTIONS=(BHSASSC,RIMM,RCOND,BACKUP,LNSTAT,SESINIT,         X
                      DVSINIT,NAKLIM,SESSION,SSPAUSE,XMTLMT,MODE,RECMD,          X
                      RCNTRL,ENDCALL)
             PATH     ADJSUB=11,                                                 X
                      DESTSUB=3
LUP001       LUPOOL   NUMBER=3
*
*  *  *  *  *  MACRO INSTRUCTIONS FOR NON-SDLC STATIONS OMITTED
*
*  *  *  *  *  MACRO INSTRUCTIONS FOR SDLC STATIONS FOLLOW:             *  *  *  *  *
*
GRP3705      GROUP    DIAL=NO,                                                   X
                      LNCTL=SDLC,                                                X
                      ACTIVTO=60.0
B4SECLR      LINE     ADDRESS=(024,025),                                         X
                      SPEED=9600,                                                X
                      CLOCKNG=EXT,                                               X
                      NRZI=NO,                                                   X
                      DUPLEX=FULL,                                               X
                      TADDR=C1,                                                  X
                      RETRIES=10
B4SECPU      PU       PUTYPE=(4,LOCAL)
*
*  *  *  *  *  (OTHER GROUP, LINE, SERVICE, AND PU MACROS FOR SDLC STATIONS)   *  *  *  *  *
*
             GENEND
             END
/*
//
```

Appendix J: Coding Examples for Switched Lines and Multiple Terminal Access Operation

This appendix provides examples illustrating how operations over switched lines, with and without multiple terminal access operation, can be accommodated within the network control program.

How to Establish Operation over the Switched Telephone Network (Network Control Mode)

Assume that your data communications network includes the following types of terminals that can call the communications controller.

- IBM 1050
- IBM 2740 (basic)
- IBM 2740 with Record Checking

Assume also that two of these types—1050 and 2740 with Record Checking—can be called by the controller.

For each type of terminal that will call the controller a separate line is required. There are accordingly three lines that must be specified as capable of receiving incoming calls. The two lines associated with the 1050 and the 2740 with Record Checking must be also be specified as capable of calling these terminals, and must therefore be placed in dial sets. In the LINE macro for the basic 2740 you would specify CALL=IN; in the LINE macros for the other two terminal types you would specify CALL=INOUT, and also specify the name of the DIALSET macro that represents the respective dial sets.

Because the lines are to be used for call-in operation, each line must have a logical connection definition. The call-in definition is a TERMINAL macro in which CTERM=YES is specified; the call-out definition is a TERMINAL macro in which CTERM=NO is specified. The basic 2740, because the controller does not call it, can be represented by a single TERMINAL macro in which CTERM=NO is specified.

The macro instructions for this combined call-in and call-out arrangement would appear as:

```
SW1050      DIALSET     LINES=(G1L1)
SW2740F     DIALSET     LINES=(G2L1)
G2740A      GROUP       LNCTL=SS,
                        DIAL=YES
BASICLN     LINE        ADDRESS=022,
                        SPEED=134,
                        POLLED=NO,
                        CALL=IN
BASIC       TERMINAL    CTERM=YES,
                        TERM=2740-1,
                        FEATURE=NOCHECK
GRSW1050    GROUP       LNCTL=SS,
                        DIAL=YES
G1L1        LINE        ADDRESS=020,
                        SPEED=134,
                        CALL=INOUT,
                        AUTO=021,
                        DIALSET=SW1050,
                        POLLED=YES
T1050LCD    TERMINAL    TERM=1050,
                        CTERM=YES,
                        ADDR=81F9,
                        POLL=81F0,
                        FEATURE=(ATTN,BREAK)
T1050       TERMINAL    TERM=1050,
                        CTERM=NO,
                        ADDR=81F9,
                        POLL=81F0,
                        FEATURE=(ATTN,BREAK),
                        DIALSET=SW1050,
                        DIALNO=1234
G2740F      GROUP       LNCTL=SS,
                        DIAL=YES
G2L1        LINE        ADDRESS=024,
                        SPEED=134,
                        CALL=INOUT,
                        AUTO=025,
                        DIALSET=SW2740F,
                        POLLED=NO
LCD2740F    TERMINAL    TERM=2740-1,
                        CTERM=YES,
                        FEATURE=CHECK
T2740F      TERMINAL    TERM=2740-1,
                        CTERM=NO,
                        FEATURE=CHECK,
                        DIALSET=SW2740F,
                        DIALNO=(5678)
```

Example of Alternate-Port Switched Network Backup Operation

The following source statements illustrate how switched network backup using the alternate-port technique can be established for BSC communication lines (not applicable for 3270s).

In this example, two switched lines—one with and one without an automatic calling unit (ACU) attached—are provided as backup to three principal nonswitched BSC lines, one of which is multipoint. One of the backup lines is specified for call-in and call-out operation; the other is specified for call-out operation only. Only call-out operation is involved in switched backup use.

Operands illustrating how switched network backup operation is specified are shown. Other macros and operands that may be required, including configuration macros for other lines that may be in the network, are not shown.

```
          BUILD      TYPGEN=PEP,ENABLTO=90                1
          SYSCNTRL   OPTIONS=(BACKUP,...)
BKUPDSET  DIALSET    LINES=(BKUPLN1,BKUPLN2)              2
BSCLNGP   GROUP      LNCTL=BSC,DIAL=NO, TYPE=PEP...       2
BSCLN1    LINE       TYPE=NCP, POLLED=YES,DIALALT=BKUPDSET
T1        TERMINAL   DIALNO=(5789),...                    3
T2        TERMINAL   DIALNO=(5841),...                    3
T3        TERMINAL   DIALNO=(6007),...                    3
BSCLN2    LINE       TYPE=NCP,DIALALT=BKUPDSET,...
T4        TERMINAL   DIALNO=(6217),...                    3
BSCLN3    LINE       TYPE=PEP,DIALALT=BKUPDSET,...        4
T5        TERMINAL   DIALNO=(5472),...                    3
BKUPGP    GROUP      LNCTL=BSC,DIAL=YES,TYPE=NCP,
                     DIALSET=BKUPDSET
BKUPLN1   LINE       AUTO=122,MPTALT=YES,CALL=OUT,...     5,6
BKUPLN2   LINE       MPTALT=YES,CALL=INOUT,...            6
CT1       TERMINAL   CTERM=YES,...                        7
          GENEND
```

---

[1] Timeout duration must be sufficient to permit operator to complete the manual dialing procedure for switched backup lines.

[2] All lines in the backup dialset and all lines for which the backup dial set provides backup lines must have the same characteristics (as explained in the description of the GROUP macro). All lines in the backup dial set must be specified as MPTALT=YES because one of the principal lines for which backup is provided is a multipoint line (POLLED=YES).

[3] DIALNO operand is required for use of automatic calling unit on line BKUPLN1. (Dial digits cannot be dynamically changed via TCAM.)

[4] Backup operation applies only when line is operating in network control mode.

[5] AUTO operand (and the ACU whose address it specifies) is required for automatic calling operation.

[6] MPTALT=YES is required because one of principal lines for which backup is provided is a multipoint line.

[7] CTERM=YES is required because line is specified for call-in as well as call-out operation.

How to Specify Multiple Terminal Access Operation (Network Control Mode)

The manner in which the multiple terminal access facility of the network control program can be put to use is illustrated by the following example. Assume the following:

- The data communications network includes three types of start-stop terminals:

  IBM 1050
  IBM 2740 Model 1 (basic)
  IBM 2740 Model 1 with Record Checking feature

- All three types of terminals will communicate with the controller over a single switched line, referred to as a multiple terminal access line (MTA line), as follows:

  —All three types are to be able to call the controller. This is referred to as *call-in MTA operation.*

  —In addition, the controller must be able to call 1050s and 2740s with record checking, but not basic 2740s, over the MTA line. This is known as *call-out MTA operation.*

- The network will include, in addition to these MTA terminals, other 1050 terminals, identical to the MTA 1050s.

To arrange for MTA operation based on the foregoing, you would proceed as follows:

Step 1: Define Line Groups to Represent Individual Terminal Types

In MTA operation, a *type* of terminal refers not to a particular numeric designation, such as IBM 1050 or 2740, but to a unique combination of the line control scheme and transmission code used. For example, a 1050 using BCD code and a 1050 using EBCD code represent two different types of MTA terminals, though they are both 1050s. Similarly, a 2740 without the Record Checking feature and a 2740 equipped with this feature represent two MTA terminal types.

In this example, there are three MTA terminal types. Each type must be represented in a line group definition. This may be a regular non-MTA group definition (GROUP, LINE and TERMINAL macros), or a "stand-alone" group definition consisting only of a GROUP macro. A stand-alone group need be defined only when there is no non-MTA group for the same type of terminal.

As stated above, one of the types of MTA terminals—the 1050—is also used in non-MTA operation. Because this type of terminal will be represented in a non-MTA line group, a stand-alone MTA line group for this type is not needed.

For the two types of 2740s, however, stand-alone groups are required:

```
G2740A  GROUP   LNCTL=SS,
                POLLED=NO, (for basic 2740)
                DIAL=YES,
                TERM=2740-1
G2740F  GROUP   LNCTL=SS,
                POLLED=NO,  (for 2740 with record
                DIAL=YES,   checking)
                TERM=2740-1,
                FEATURE=(CHECK)
```

Each stand-alone GROUP macro must be named; the name is referred to by the MTALCST macro.

(In this example, all operands relating to MTA operation are shown. For purposes of this example, it is assumed that the default values for omitted operands are appropriate. Some operands that could be omitted, such as LNCTL=SS, are nonetheless shown.)

Step 2: Define MTA Line Group

The next step is to define an *MTA line group*. This is a regular line group, defined in the same manner as any regular line group (not a stand-alone group), except that you specify MTA, rather than a specific type of terminal, in the TERM operand. (The TERM operand may appear in the TERMINAL or LINE macros, or in the GROUP macro.)

GROUP macro:

The GROUP macro need specify only DIAL=YES; for purposes of this example, assume that for all other operands the default values are appropriate.

```
G1        GROUP    LNCTL=SS,
                   DIAL=YES
```

LINE macro:

Assume that the MTA line (1) is a duplex line that will operate at 134.5 bits per second (the data rate of the 1050s and 2740s), (2) is attached to address 020 in the controller, and (3) has its automatic calling unit attached to address 021. You would therefore specify, in the LINE macro, ADDRESS=020, SPEED=134, AUTO=021, DUPLEX=FULL.

Because the line will communicate with 1050s, which must be polled, you would also code POLLED=YES.

You would code CALL=INOUT (or omit the CALL operand), because calls over the line will be initiated both by terminals (1050 and both 2740 types) and by the controller (for the 1050s and 2740s with record checking only), as stated at the beginning of this example.

Call-in operation requires that you associate an *MTA list* with this line. *Call-out* operation requires that you associate a *dial set* with the line. (These are defined as described under Steps 4 and 5, below.) You would therefore code the MTALIST and DIALSET operands in the LINE macro.

Assuming that no other operands are required because their default values are appropriate, the complete LINE macro would be:

```
G1L1      LINE     ADDRESS=020,
                   SPEED=134,
                   AUTO=021,
                   DUPLEX=FULL,
                   POLLED=YES,
                   CALL=INOUT,
                   MTALIST=MTALST,
                   DIALSET=SWITMTA
```

TERMINAL macro for call-in operation:

Following each LINE macro for an MTA line over which MTA terminals will call the controller there must be a single TERMINAL macro in which CTERM=YES is specified. CTERM=YES designates this TERMINAL macro as the "call-in logical connection" terminal. This TERMINAL macro must appear between the LINE macro and the next LINE macro. (If this MTA line group contained several lines, instead of just one, a separate logical connection TERMINAL macro would be required for each line.)

```
G1L1TU   TERMINAL     TERM=MTA,
                      CTERM=YES,
                      FEATURE=(ATTN,BREAK),
                      ATTN=ENABLED
```

TERMINAL macros for call-out operation:

Each MTA terminal type that the controller will call over the MTA line must be represented by its own TERMINAL macro. As mentioned earlier, in this example the controller will call 1050s and 2740s with record checking, but not basic 2740s.

Assume (1) that only one 1050 will be called, and that its telephone number is to be defined in the network control program, and (2) that several 2740s with record checking will be called, and that the host processor will provide the appropriate telephone number with each call-out request. One TERMINAL macro is accordingly required for the 1050 and one for all 2740s with record checking. The DIALNO operand for the 1050 specifies the digits in the telephone number, followed by the length of the number; DIALNO for the 2740s specifies only the length, as the number will be provided within each call-out request from the host processor.

In the TERMINAL macros you would specify in the TERM operand *not* the actual terminal type, but *MTA*. You would, however, specify the features with which the terminal is equipped, and in the case of the 1050, specify enabling of the attention feature, if required by the application. Also, for the 1050, you would specify the polling and addressing characters needed.

These TERMINAL macros must specify the line control selection table entry and the dial set to be associated with the terminal, as explained in Steps 3 and 5.

The TERMINAL macros for call-out operation are therefore:

```
G1L1TA   TERMINAL     TERM=MTA,
                      FEATURE=(ATTN,BREAK),
                      ATTN=ENABLED,
                      ADDR=81F9,          (for 1050)
                      POLL=81F0,
                      LCST=MTA1050,
                      DIALSET=SWITMTA,
                      DIALNO=(3251,4)
G1L1TB   TERMINAL     TERM=MTA,
                      LCST=MTA2740F,      (for 2740 with
                      DIALSET=SWITMTA,    record checking)
                      DIALNO=(,4)
```

Step 3: Define Line Control Selection Table

One entry in the line control selection table is required for each combination of MTA terminal type and set of operating parameters to be identified to the network control program. *(Operating parameters* refers to the characteristics of a terminal apart from its line control scheme and transmission code; for example, the carriage return rate and the length (in characters) of the print line.)

Assume that all of the 1050s and all of the 2740s with record checking have the same operating parameters. Also assume that some of the basic 2740s have a printer line length of 130 characters and others have a line length of 95 characters, all other parameters being identical. The basic 2740s therefore have two sets of operating parameters, thus requiring two MTALCST macros; the 1050s and 2740s with record checking require only one each:

```
MTA1050         MTALCST          GROUP=GRSW1050,
                                 SPEED=134,
                                 CODE=EBCD,
                                 LCTYPE=1050,
                                 LINESIZ=130
MTA274A1        MTALCST          GROUP=G2740A,
                                 SPEED=134,
                                 CODE=COR,
                                 LCTYPE=2740A,
                                 LINESIZ=130
MTA274A2        MTALCST          GROUP=G2740A,
                                 SPEED=134,
                                 CODE=COR,
                                 LCTYPE=2740A,
                                 LINESIZ=95
MTA2740F        MTALCST          GROUP=G2740F,
                                 SPEED=134,
                                 CODE=EBCD,
                                 LCTYPE=2740F,
                                 LINESIZ=95
```

If the network control program is to call the type of terminal represented by one of the MTALCST macros, the LCST operand of the TERMINAL macro representing that type must name the MTALCST macro. In this example, the program is to call the 1050 represented by the MTALCST macro labeled MTA1050 and 2740s represented by the MTALCST macro labeled MTA2740F. You would therefore code LCST=MTA1050 and LCST=MTA2740F in the TERMINAL macros labeled G1L1TA and G1L1TB, respectively, as shown in Step 2.

Step 4: Specify MTA Lists, MTA Tables, and 1050 Polling Character List

MTA lists:
Associated with each MTA line must be a list of MTA terminal types for which the NCP must check when answering a call over the line. The MTALIST macro (which must directly follow the MTALCST macros) defines this list. In this example only one MTALIST macro is needed because there is only one combination of terminals for which the program must check. The name of the MTALIST macro must be specified in the MTALIST operand of the LINE macro.

The order in which you specify the MTA terminal types in the MTALIST macro is not important; the program always attempts to identify the terminal types in the same predetermined sequence.

```
MTALST       MTALIST LCTYPE=(1050,2740A,2740F)
```

MTA tables:

For call-in operation, each MTA terminal type must be identified in an MTA table defined by an MTATABL macro.

Because there are three MTA terminal types, three MTATABL macros are required. Each must specify the names of the MTALCST macros representing the combinations of operating parameters that the network control program must recognize. For the 1050s and the 2740s with record checking, there is only one set of parameters. The basic 2740s, on the other hand, have two sets. These differ in the length of the print line. Notice that the LCST operand of the MTATABL macro for the basic 2740s therefore names two MTALCST macros, whereas the others name only one each.

```
MTATABL      LCST=(MTA1050),
             CODE=EBCD,
             LCTYPE=1050
MTATABL      LCST=(MTA274A1,MTA274A2),
             CODE=COR,
             LCTYPE=2740A
MTATABL      LCST=(MTA2740F),
             CODE=EBCD,
             LCTYPE=2740F
```

1050 polling list: For the 1050 terminals that will call the controller, an MTAPOLL macro is required, listing all of the 1050 polling characters. Assuming for this example that all such terminals have the same polling characters, 81F0 (hexadecimal), you would code:

```
MTAPOLL   POLL=(81F0)
```

Step 5: Specify Dial Sets for Call-Out Operation

Because the line over which the controller communicates with MTA terminals is to be used for call-out operations, it must be included in a dial set. (This is the normal requirement for call-out lines, regardless of whether they are used for MTA operation.)

The dial set in this example consists of one line. The DIALSET macro therefore names a single LINE macro. Assuming that no alternate dial set is required and that the default values for the QLIMIT, QLOAD and RESERVE operands are appropriate, the DIALSET macro appears as:

```
SWITMTA        DIALSET          LINES=(G1L1)
```

The name of this dial set is referred to by the DIALSET operand of the LINE macro appearing in Step 2.

The DIALSET operand of the TERMINAL macro for each terminal that the controller will call must name the dial set to be used.

* * * * *

All the macros required for the example have now been coded. Figure J-1 shows all of the macros in a single sequence, with all macro sequencing requirements observed. Figure J-2 shows the logical relationships between the macros, as established by the pointers shown.

VTAM Note: Under certain conditions a VTERM macro may be needed directly following a TERMINAL macro in which CTERM=YES is specified. See the *ACF/VTAM Installation* guide for information on the purpose of the VTERM macro and for the conditions for its use.

```
SWITMTA      DIALSET      LINES=(G1L1)
MTA1050      MTALCST      GROUP=GRSW1050,
                          SPEED=134,
                          CODE=EBCD,
                          LCTYPE=1050,
                          LINESIZ=130
MTA274A1     MTALCST      GROUP=G2740A,
                          SPEED=134,
                          CODE=COR,
                          LCTYPE=2740A,
                          LINESIZ=130
MTA274A2     MTALCST      GROUP=G2740A,
                          SPEED=134,
                          CODE=COR,
                          LCTYPE=2740A,
                          LINESIZ=95
MTA2740F     MTALCST      GROUP=G2740F,
                          SPEED=134,
                          CODE=EBCD,
                          LCTYPE=2740F,
                          LINESIZ=95
MTALST       MTALIST      LCTYPE=(1050,2740A,2740F)
             MTAPOLL      POLL=(81F0)
             MTATABL      LCST=(MTA1050),
                          LCTYPE=1050,
                          CODE=EBCD
             MTATABL      LCST=(MTA274A1,MTA274A2),
                          LCTYPE=2740A,
                          CODE=COR
             MTATABL      LCST=(MTA2740F),
                          LCTYPE=2740F,
                          CODE=EBCD
```

Figure J-1. Macro Instructions Required for MTA (Part 1 of 2)

```
G2740A      GROUP       LNCTL=SS,
                        POLLED=NO,
                        DIAL=YES,
                        TERM=2740-1
G2740F      GROUP       LNCTL=SS,
                        POLLED=NO,
                        DIAL=YES,
                        TERM=2740-1,
                        FEATURE=(CHECK)
G1          GROUP       LNCTL=SS,
                        DIAL=YES
G1L1        LINE        ADDRESS=020,
                        SPEED=134,
                        AUTO=021,
                        DIALSET=SWITMTA,
                        DUPLEX=FULL,
                        MTALIST=MTALST,
                        POLLED=YES,
                        CALL=INOUT
G1L1TU      TERMINAL    TERM=MTA,
                        CTERM=YES,
                        FEATURE=(ATTN,BREAK),
                        ATTN=ENABLED
G1L1TA      TERMINAL    TERM=MTA,
                        FEATURE=(ATTN,BREAK),
                        ATTN=ENABLED,
                        ADDR=81F9,
                        POLL=81F0,
                        LCST=MTA1050,
                        DIALSET=SWITMTA,
                        DIALNO=(3251,4)
G1L1TB      TERMINAL    TERM=MTA,
                        LCST=MTA2740F,
                        DIALSET=SWITMTA,
                        DIALNO=(,4)
GRSW1050    GROUP       LNCTL=SS,
                        POLLED=YES,
                        DIAL=YES,
                        TERM=1050,
                        FEATURE=(ATTN,BREAK)
              (LINE and TERMINAL macros for
                        GRSW1050 Group)
                        .
                        .
                        .
```

Figure J-1. Macro Instructions Required for MTA (Part 2 of 2)

MTA Line Group Definition (Step 2)

```
G1         GROUP      LNCTL=SS,
                      DIAL=YES

G1L1       LINE       ADDRESS=020,
                      SPEED=134,
                      AUTO=021,
                      DIALSET=SWITMTA,
                      DUPLEX=FULL,
                      MTALIST=MTALST,
                      POLLED=YES,
                      CALL=INOUT

G1L1TU     TERMINAL   TERM=MTA,
                      CTERM=YES,
                      FEATURE=(ATTN,BREAK),
                      ATTN=ENABLED

G1L1TA     TERMINAL   TERM=MTA,
                      FEATURE=(ATTN,BREAK),
                      ATTN=ENABLED,
                      ADDR=81F9,
                      POLL=81F0,
                      LCST=MTA1050,
                      DIALSET=SWITMTA,
                      DIALNO=(3251,4)

G1L1TB     TERMINAL   TERM=MTA,
                      LCST=MTA2740F,
                      DIALSET=SWITMTA,
                      DIALNO=(,4)
```

Dial Set Definition (Step 5)

```
SWITMTA      DIALSET      LINES=(G1L1)
```

MTA List and Polling List Definitions (Step 4)

```
MTALST       MTALIST      LCTYPE=(1050,2740A,2740F)

             MTAPOLL      POLL=(81F0)
```

MTA Table Definition (Step 4)

```
           MTATABL    LCST=(MTA1050),
                      LCTYPE=1050,
                      CODE=EBCD

           MTATABL    LCST=(MTA274A1,MTA274A2)
                      LCTYPE=2740A,
                      CODE=COR

           MTATABL    LCST=(MTA2740F),
                      LCTYPE=2740F,
                      CODE=EBCD
```

Line Control Selection Table Definition (Step 3)

```
MTA1050      MTALCST      GROUP=GRSW1050,
                          SPEED=134,
                          CODE=EBCD,
                          LCTYPE=1050,
                          LINESIZ=130

MTA274A1     MTALCST      GROUP=G2740A,
                          SPEED=134,
                          CODE=COR,
                          LCTYPE=2740A,
                          LINESIZ=130

MTA274A2     MTALCST      GROUP=G2740A,
                          SPEED=134,
                          CODE=COR,
                          LCTYPE=2740A,
                          LINESIZ=95

MTA2740F     MTALCST      GROUP=G2740F,
                          SPEED=134,
                          CODE=EBCD,
                          LCTYPE=2740F,
                          LINESIZ=95
```

Line Group Definitions (Step 1)

```
Non-MTA Line Group Definitions:

GRSW1050     GROUP      LNCTL=SS,
                        POLLED=YES,
                        DIAL=YES,
                        TERM=1050
                        FEATURE=ATTN,
                                BREAK)
     (LINE and TERMINAL macros
        for GRSW1050 group)
                 .
                 .
                 .

Stand-alone Line Group Definition:

G2740A       GROUP      LNCTL=SS,
                        POLLED=NO,
                        DIAL=YES,
                        TERM=2740-1

G2740F       GROUP      LNCTL=SS,
                        POLLED=NO,
                        DIAL=YES,
                        TERM=2740-1,
                        FEATURE=(CHECK))
```

Figure J-2. Relationships of MTA Macro Instructions

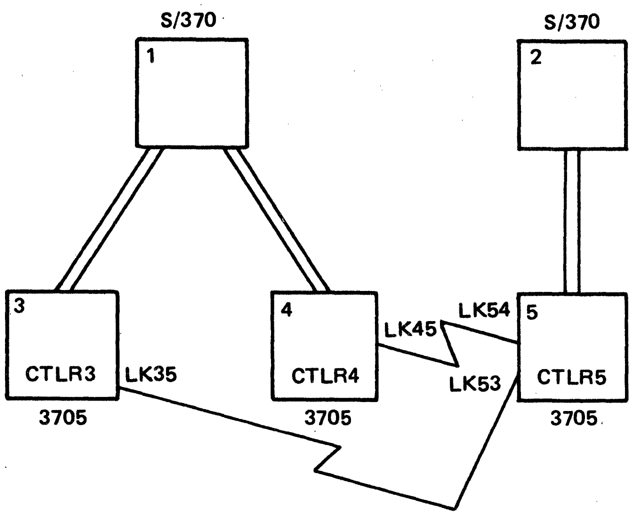
Appendix K. Network Path Specification Examples

This appendix shows for several network configurations how to use PATH macros to establish the desired network paths. Each example includes a diagram of the configuration, a similar diagram showing the desired paths, and the relevant operands of the BUILD, HOST, PATH, and PU macros defining the subareas and paths between them. After the NCP source statements appear the corresponding ACF/TCAM and ACF/VTAM statements for the paths shown.

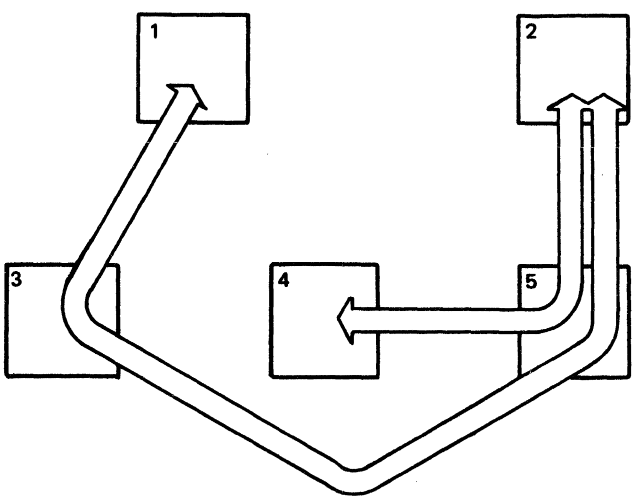
Note: The labels appearing on the TCAM IEDRTDEF statements must correspond to the labels of the TCAM TERMINAL macros (not shown) that represent the controllers. In these examples, the same names are used as labels on the PU macros representing the controllers.

Example 1

Network Configuration:



Desired Paths:



NCP Statements:

For subarea 3:

```
        BUILD   SUBAREA=3,...
          ⋮
        HOST    SUBAREA=1,...
          ⋮
        PATH    ADJSUB=5,DESTSUB=2
LGP1    GROUP
LK35    LINE    LNCTL=SDLC,...
CTLR5   PU      PUTYPE=(4,LOCAL),SUBAREA=5,...
          ⋮
```

For subarea 4:

```
        BUILD   SUBAREA=4,...
          ⋮
        HOST    SUBAREA=1,...
          ⋮
        PATH    ADJSUB=5,DESTSUB=2
LGP1    GROUP
LK45    LINE    LNCTL=SDLC,...
CTLR5   PU      PUTYPE=(4,LOCAL),SUBAREA=5,...
          ⋮
```

For subarea 5:

```
        BUILD   SUBAREA=5,...
          ⋮
        HOST    SUBAREA=2,...
          ⋮
        PATH    ADJSUB=3,DESTSUB=1
LGP1    GROUP
LK54    LINE    LNCTL=SDLC,...
CTLR4   PU      PUTYPE=(4,LOCAL),SUBAREA=4,...
LK53    LINE    LNCTL=SDLC,...
CTLR3   PU      PUTYPE=(4,LOCAL),SUBAREA=3,...
          ⋮
```

ACF/TCAM Statements:

For subarea 1:

```
IEDRTDEF        TERMNAM=CTLR3,SUBAREA=(5,2)
```

For subarea 2:

```
IEDRTDEF        TERMNAM=CTLR5,SUBAREA=(1,3,4)
```

ACF/VTAM Statements:

For subarea 1:

```
PATH            ADJSUB=3,DESTSUB=(5,2)
```

For subarea 2:

```
PATH            ADJSUB=5,DESTSUB=(1,3,4)
```

Example 2

NCP Statements:

For subarea 3:

```
          BUILD   SUBAREA=3,...
            •
            •
          HOST    SUBAREA=1,...
            •
            •
          PATH    ADJSUB=5,DESTSUB=2
LGP1      GROUP
LK34      LINE    LNCTL=SDLC,...
CTLR4     PU      PUTYPE=(4,LOCAL),SUBAREA=4,...
LK35      LINE    LNCTL=SDLC,...
CTLR5     PU      PUTYPE=(4,LOCAL),SUBAREA=5,...
            •
            •
```

For subarea 4:

```
          BUILD   SUBAREA=4,...
            •
            •
          HOST    SUBAREA=1,...
            •
            •
          PATH    ADJSUB=3,DESTSUB=(5,2)
LGP1      GROUP
LK43      LINE    LNCTL=SDLC,...
CTLR3     PU      PUTYPE=(4,LOCAL),SUBAREA=3,...
            •
            •
```

For subarea 5:

```
          BUILD   SUBAREA=5,...
            •
            •
          HOST    SUBAREA=2,...
            •
            •
          PATH    ADJSUB=3,DESTSUB=(4,1)
LGP1      GROUP
LK53      LINE    LNCTL=SDLC,...
CTLR3     PU      PUTYPE=(4,LOCAL),SUBAREA=3,...
            •
            •
```

ACF/TCAM Statements:

For subarea 1:

```
IEDRTDEF        TERMNAM=CTLR3,SUBAREA=(5,2)
```

For subarea 2:

```
IEDTRDEF        TERMNAM=CTLR5,SUBAREA=(1,3,4)
```

ACF/VTAM Statements:

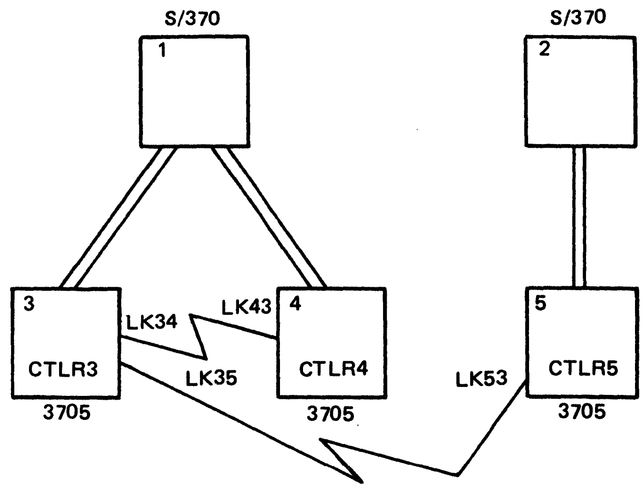
For subarea 1:

```
PATH            ADJSUB=3,DESTSUB=(5,2)
```

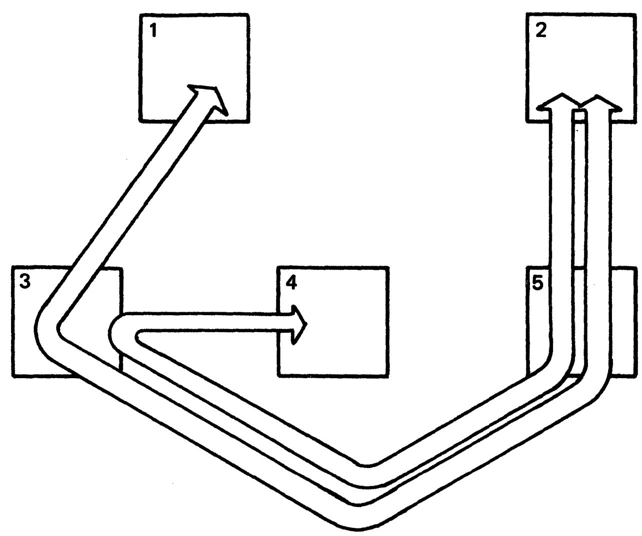
For subarea 2:

```
PATH            ADJSUB=5,DESTSUB=(1,3,4)
```

Network Configuration:

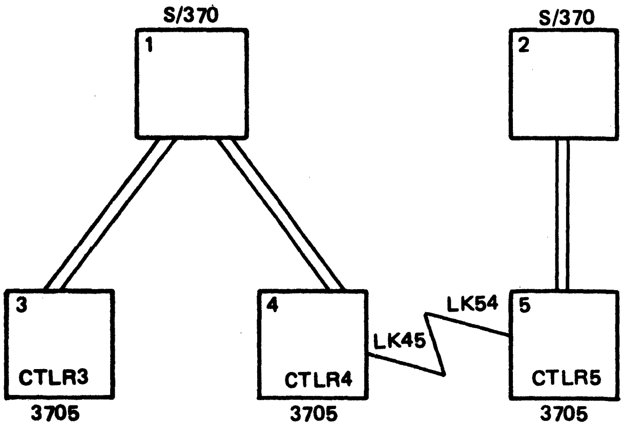


Desired Paths:

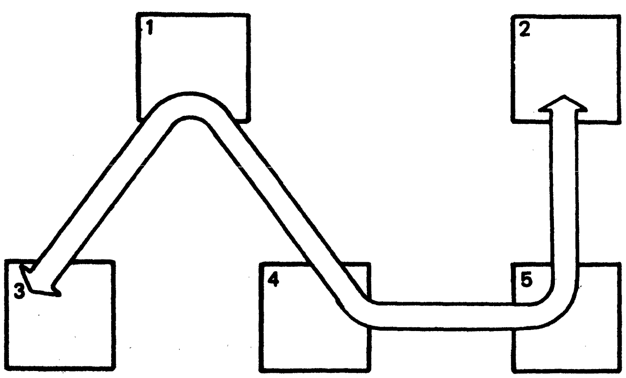


Example 3

Network Configuration:



Desired Path:



NCP Statements:

For subarea 3:

```
         BUILD    SUBAREA=3,...
           :
         HOST     SUBAREA=1,...
           :
         PATH     ADJSUB=1,DESTSUB=(4,5,2)
           :
```

For subarea 4:

```
         BUILD    SUBAREA=4,...
           :
         HOST     SUBAREA=1,...
           :
         PATH     ADJSUB=5,DESTSUB=2
         PATH     ADJSUB=1,DESTSUB=3
LGP1     GROUP
LK45     LINE     LNCTL=SDLC,...
CTLR5    PU       PUTYPE=(4,LOCAL),SUBAREA=5,...
           :
```

For subarea 5:

```
         BUILD    SUBAREA=5,...
           :
         HOST     SUBAREA=2,...
           :
         PATH     ADJSUB=4,DESTSUB=(1,3)
LGP1     GROUP
LK54     LINE     LNCTL=SDLC,...
CTLR4    PU       PUTYPE=(4,LOCAL),SUBAREA=4,...
           :
```

ACF/TCAM Statements:

For subarea 1:

```
IEDRTDEF         TERMNAM=CTLR4,SUBAREA=(5,2)
```

For subarea 2:

```
IEDRTDEF         TERMNAM=CTLR5,SUBAREA=(1,3,4)
```

ACF/VTAM Statements:

For subarea 1:

```
PATH             ADJSUB=4,DESTSUB=(5,2)
```

For subarea 2:

```
PATH             ADJSUB=5,DESTSUB=(1,3,4)
```

Example 4

NCP Statements:

For subarea 3:

```
         BUILD   SUBAREA=3,...
           ⋮
         HOST    SUBAREA=1,...
           ⋮
         PATH    ADJSUB=4,DESTSUB=(5,2)
LGP1     GROUP
LK34     LINE    LNCTL=SDLC,...
CTLR4    PU      PUTYPE=(4,LOCAL),SUBAREA=4,...
           ⋮
```

For subarea 4:

```
         BUILD   SUBAREA=4,...
           ⋮
         HOST    SUBAREA=1,...
           ⋮
         PATH    ADJSUB=5,DESTSUB=2
LGP1     GROUP
LK43     LINE    LNCTL=SDLC,...
CTLR3    PU      PUTYPE=(4,LOCAL),SUBAREA=3,...
LK45     LINE    LNCTL=SDLC,...
CTLR5    PU      PUTYPE=(4,LOCAL),SUBAREA=5,...
           ⋮
```

For subarea 5:

```
         BUILD   SUBAREA=5,...
           ⋮
         HOST    SUBAREA=2,...
           ⋮
         PATH    ADJSUB=4,DESTSUB=(1,3)
LGP1     GROUP
LK54     LINE    LNCTL=SDLC,...
CTLR4    PU      PUTYPE=(4,LOCAL),SUBAREA=4,...
           ⋮
```

ACF/TCAM Statements:

For subarea 1:

```
IEDRTDEF         TERMNAM=CTLR4,SUBAREA=(5,2)
```

For subarea 2:

```
IEDRTDEF         TERMNAM=CTLR5,SUBAREA=(1,3,4)
```

ACF/VTAM Statements:

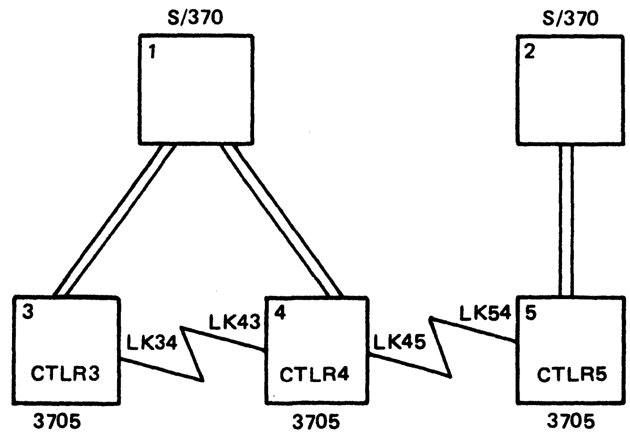
For subarea 1:

```
PATH             ADJSUB=4,DESTSUB=(5,2)
```

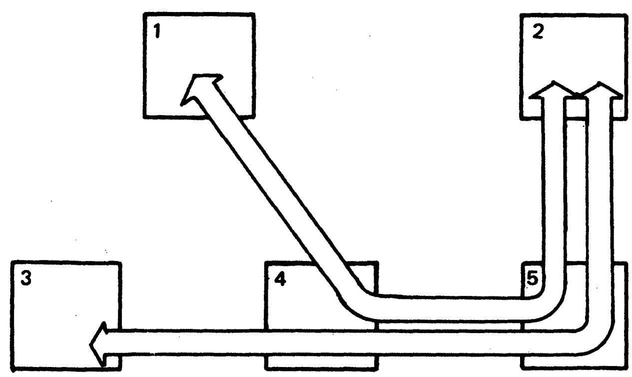
For subarea 2:

```
PATH             ADJSUB=5,DESTSUB=(1,3,4)
```

Network Configuration:

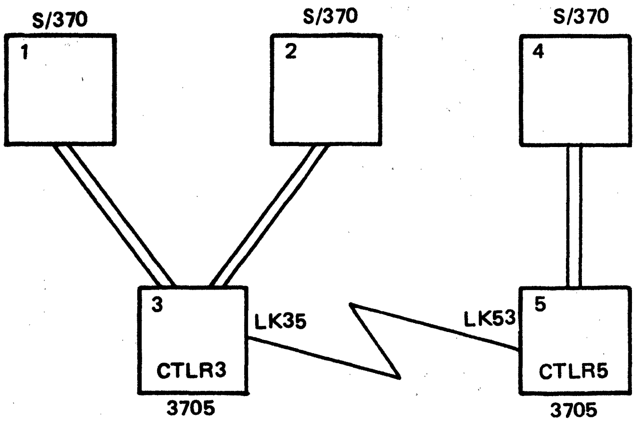


Desired Paths:

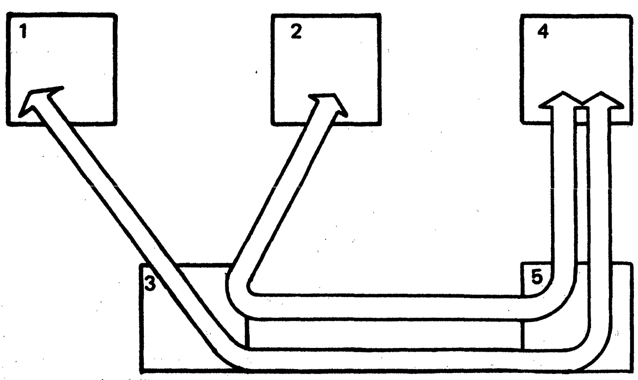


Example 5

Network Configuration:



Desired Paths:



NCP Statements:

For subarea 3:

```
        BUILD   SUBAREA=3,...
          •
        HOST    SUBAREA=1,...
        HOST    SUBAREA=2,...
          •
        PATH    ADJSUB=5,DESTSUB=4
LGP1    GROUP
LK35    LINE    LNCTL=SDLC,...
CTLR5   PU      PUTYPE=(4,LOCAL),SUBAREA=5,...
          •
```

For subarea 5:

```
        BUILD   SUBAREA=5,...
          •
        HOST    SUBAREA=4,...
          •
        PATH    ADJSUB=3,DESTSUB=(1,2)
LGP1    GROUP
LK53    LINE    LNCTL=SDLC,...
CTLR3   PU      PUTYPE=(4,LOCAL),SUBAREA=3,...
          •
```

ACF/TCAM Statements:

For subarea 1:

```
IEDRTDEF        TERMNAM=CTLR3,SUBAREA=(2,4,5)
```

For subarea 2:

```
IEDRTDEF        TERMNAM=CTLR3,SUBAREA=(1,4,5)
```

For subarea 4:

```
IEDRTDEF        TERMNAM=CTLR5,SUBAREA=(1,2,3)
```

ACF/VTAM Statements:

For subarea 1:

```
PATH            ADJSUB=3,DESTSUB=(2,4,5)
```

For subarea 2:

```
PATH            ADJSUB=3,DESTSUB=(1,4,5)
```

For subarea 4:

```
PATH            ADJSUB=5,DESTSUB=(1,2,3)
```

Example 6

NCP Statements:

For subarea 4:

```
        BUILD   SUBAREA=4,...
          :
        HOST    SUBAREA=1,...
          :
        PATH    ADJSUB=5,DESTSUB=(2,3,6)
LGP1    GROUP
LK45    LINE    LNCTL=SDLC,...
CTLR5   PU      PUTYPE=(4,LOCAL),SUBAREA=5,...
          :
```

For subarea 5:

```
        BUILD   SUBAREA=5,...
          :
        HOST    SUBAREA=2,...
          :
        PATH    ADJSUB=4,DESTSUB=1
        PATH    ADJSUB=6,DESTSUB=3
LGP1    GROUP
LK54    LINE    LNCTL=SDLC,...
CTLR4   PU      PUTYPE=(4,LOCAL),SUBAREA=4,...
LK56    LINE    LNCTL=SDLC,...
CTLR6   PU      PUTYPE=(4,LOCAL),SUBAREA=6,...
          :
```

For subarea 6:

```
        BUILD   SUBAREA=6,...
          :
        HOST    SUBAREA=3,...
          :
        PATH    ADJSUB=5,DESTSUB=(1,2,4)
LGP1    GROUP
LK65    LINE    LNCTL=SDLC,...
CTLR5   PU      PUTYPE=(4,LOCAL),SUBAREA=5,...
          :
```

ACF/TCAM Statements:

For subarea 1:

```
IEDRTDEF        TERMNAM=CTLR4,SUBAREA=(2,3,5,6)
```

For subarea 2:

```
IEDRTDEF        TERMNAM=CTLR5,SUBAREA=(1,3,4,6)
```

For subarea 3:

```
IEDRTDEF        TERMNAM=CTLR6,SUBAREA=(1,2,4,5)
```

ACF/VTAM Statements:

For subarea 1:

```
PATH            ADJSUB=4,DESTSUB=(2,3,5,6)
```

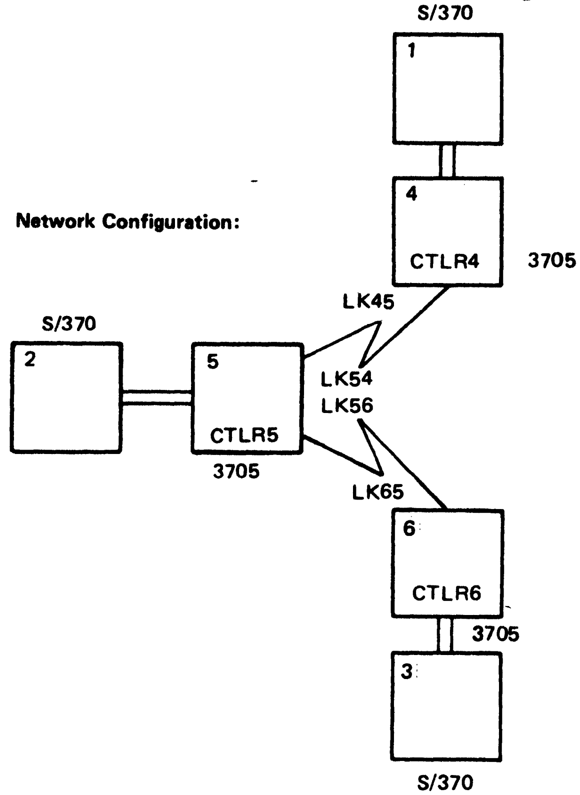
For subarea 2:

```
PATH            ADJSUB=5,DESTSUB=(1,3,4,6)
```

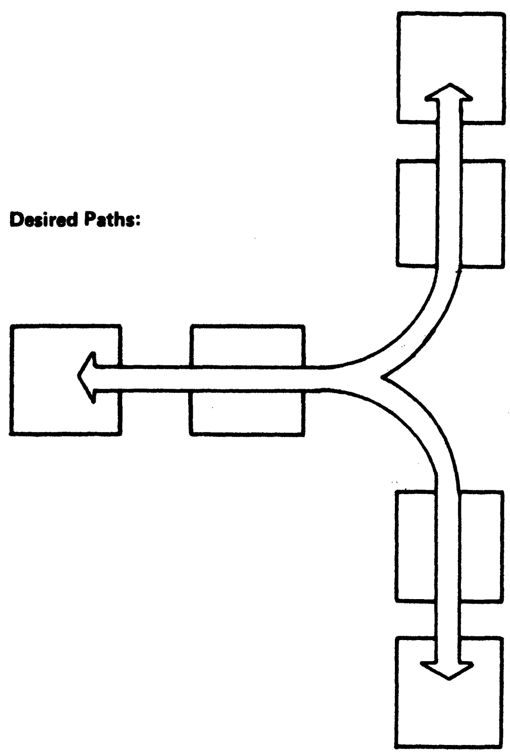
For subarea 3:

```
PATH            ADJSUB=6,DESTSUB=(1,2,4,5)
```

Network Configuration:

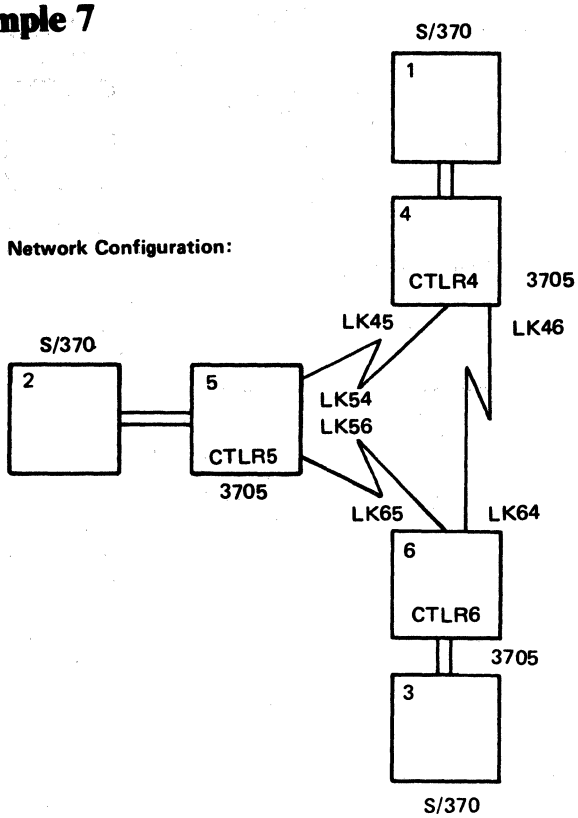


Desired Paths:

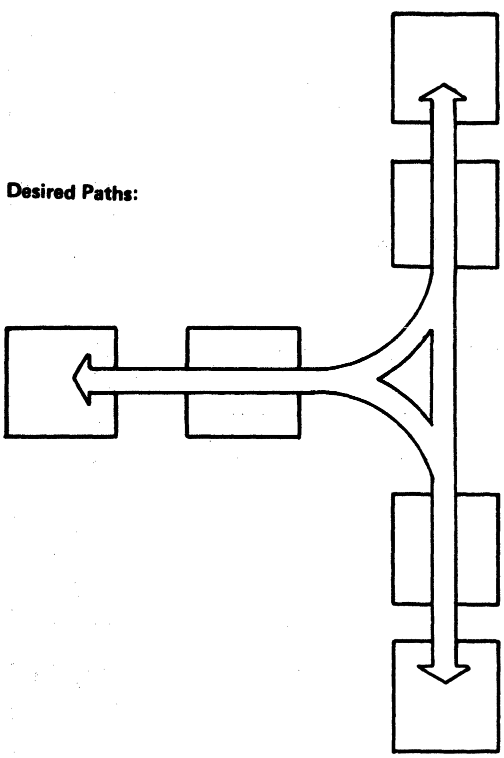


Example 7

Network Configuration:



Desired Paths:



NCP Statements:

For subarea 4:

```
          BUILD    SUBAREA=4,...
            :
          HOST     SUBAREA=1,...
            :
          PATH     ADJSUB=5,DESTSUB=2
          PATH     ADJSUB=6,DESTSUB=3
LGP1      GROUP
LK45      LINE     LNCTL=SDLC,...
CTLR5     PU       PUTYPE=(4,LOCAL),SUBAREA=5,...
LK46      LINE     LNCTL=SDLC,...
CTLR6     PU       PUTYPE=(4,LOCAL),SUBAREA=6,...
            :
```

For subarea 5:

```
          BUILD    SUBAREA=5,...
            :
          HOST     SUBAREA=2,...
            :
          PATH     ADJSUB=4,DESTSUB=1
          PATH     ADJSUB=6,DESTSUB=3
LGP1      GROUP
LK54      LINE     LNCTL=SDLC,...
CTLR4     PU       PUTYPE=(4,LOCAL),SUBAREA=4,...
LK56      LINE     LNCTL=SDLC,...
CTLR6     PU       PUTYPE=(4,LOCAL),SUBAREA=6,...
            :
```

For subarea 6:

```
          BUILD    SUBAREA=6,...
            :
          HOST     SUBAREA=3,...
            :
          PATH     ADJSUB=4,DESTSUB=1
          PATH     ADJSUB=5,DESTSUB=2
LGP1      GROUP
LK65      LINE     LNCTL=SDLC,...
CTLR5     PU       PUTYPE=(4,LOCAL),SUBAREA=5,...
LGP2      GROUP
LK64      LINE     LNCTL=SDLC,...
CTLR4     PU       PUTYPE=(4,LOCAL),SUBAREA=4,...
            :
```

ACF/TCAM Statements:

For subarea 1:

```
IEDRTDEF       TERMNAM=CTLR4,SUBAREA=(2,3,5,6)
```

For subarea 2:

```
IEDRTDEF       TERMNAM=CTLR5,SUBAREA=(1,3,4,6)
```

For subarea 3:

```
IEDRTDEF       TERMNAM=CTLR6,SUBAREA=(1,2,4,5)
```

ACF/VTAM Statements:

For subarea 1:

```
PATH           ADJSUB=4,DESTSUB=(2,3,5,6)
```

For subarea2:

```
PATH           ADJSUB=5,DESTSUB=(1,3,4,6)
```

For subarea 3:

```
PATH           ADJSUB=6,DESTSUB=(1,2,4,5)
```

Appendix L. Supplemental Information for Airlines Line Control Users

This appendix contains information for specifying communication lines that use the airlines line control (ALC) procedure. It supplements the information in the remainder of this publication, and is applicable only when you are defining a control program that performs emulation functions (PEP) for an IBM 3705-II Communications Controller in which RPQ numbers 858911 and 858912 are installed.

To define a line as using ALC, rather than BSC or start-stop line control, specify LNCTL=ALC in the GROUP macro and observe the requirements indicated under each of the following macro instructions and operands mentioned.

Note: LNCTL=ALC is valid only for a line serviced by a type 3 communication scanner in a 3705-II controller.

BUILD Macro Instruction

Use a BUILD macro that defines a program to perform emulation functions (PEP). ALC inposes no special requirements on coding the BUILD macro and its operands.

CSB Macro Instruction

No line for which LNCTL=ALC is specified should be specified in the WRAPLN operand of the CSB macro unless all lines serviced by the scanner are ALC lines. Otherwise, ALC imposes no special requirements on coding of the CSB macro and its operands.

GROUP Macro Instruction

Each ALC line must be included in a line group, represented by a GROUP macro. No ALC line may be included in more than one line group, and all lines in an ALC line group must be ALC lines.

All ALC lines are nonswitched (DIAL=NO).

In general, the operands of the GROUP macro that apply to nonswitched BSC lines apply also to ALC lines. Specific requirements for coding these operands area as follows.

Invalid Operands

CHAREC	QUIETCT
DELAY	REPLYTO
EOB	TEXTTO
EOT	

Valid Operands

The following operands are valid for an ALC line group. Specific requirements are indicated.

DIAL: Specify DIAL=NO or omit this operand.

LNCTL: Specify LNCTL=ALC

Line Macro Instruction

In general the operands of the LINE macro that apply to nonswitched BSC lines apply also to ALC lines. Certain LINE macro operands may be specified instead in the GROUP macro. They are the same for an ALC line as for a non-ALC line. Specific requirements for coding the operands of the LINE macro are as follows.

Invalid Operands

The following operands are invalid for ALC lines and if coded are ignored.

AUTO QUIET CHECKRING CODETADDR DUALCOM
TERM FEATUREUNITXC MULTI

Valid Operands

The following operands are valid for an ALC line. Specific requirements are indicated.

ADDRESS: ALC lines are arranged in pairs and attached to consecutive line interface addresses, the lower of which must be an even address. The even address is used for transmit operations. The next higher (odd) address is used for receive operations. In the ADDRESS operand specify only the even (transmit) address and the associated subchannel address.

RCVADDR: For example, ADDRESS=(0A0,43).

In the RCVADDR operand specify the odd (receive) address and the associated subchannel address. For example, RCVADDR=(0A1,42).

The ADDRESS and RCVADDR operands are required.

Note: Any line interface address you specify for an ALC line must be an address associated with a type 3 communication scanner.

Specify the subchannel address in the ADDRESS and RCVADDR operands in the same way as for a non-ALC line. Because an ALC line must be associated with a type 4 channel adapter, you may specify multiple emulation subchannel addresses for the line interface address. (See Figure L-2 for an example).

BUFSIZE: Specify BUFSIZE=(n1,[n2]), in which n1 represents the emulation mode buffer size for the transmit (even) line address specified in the ADDRESS operand and n2 represents the emulation mode buffer size for the receive (odd) line address specified in the RCVADDR operand.

n1 and n2 may be any of the following values:

8	64	160
16	96	192
32	128	224

If you omit this operand, a buffer size of 32 [BUFSIZE=(32,32)] is assumed.

CHNPRI: No special requirements apply to the use of this operand.

CLOCKNG: No special requirements apply to the use of this operand.

CU: Specify CU=2703 or omit this operand.

CUTYPE: Specify either CUTYPE=1006, CUTYPE=2946, or CUTYPE=2948, as appropriate, for the type of station attached to the ALC line. If a mixture of station types is attached to the line, specify CUTYPE=MIXD.

DATRATE: No special requirements apply to the use of this operand.

DISABLE: Specify DISABLE=NO or omit this operand.

DUPLEX: Specify DUPLEX=FULL.

INTPRI: No special requirements apply to the use of this operand.

MODEM: Specify MODEM=OPTION2 or omit this operand.

NEWSYNC: No special requirements apply to the use of this operand.

PAD: Specify PAD=YES or omit this operand.

RCVADDR: (See ADDRESS operand in this appendix.)

SPEED: Line speeds between 2400 and 9600 bps, inclusive, are valid.

GENEND Macro Instruction

ALC imposes no special requirements on coding of the GENEND macro and its operands.

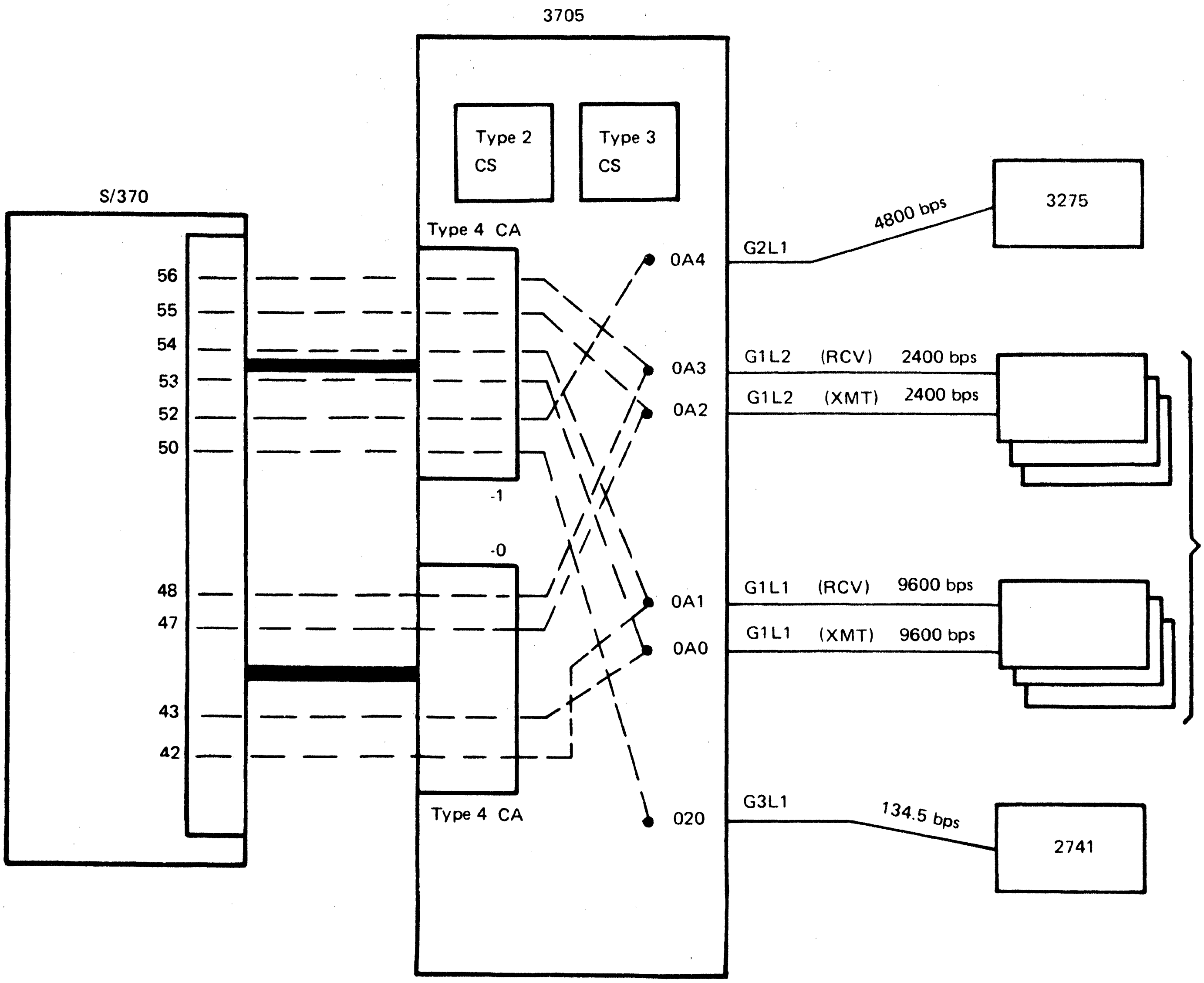


Figure L-1. Network Configuration for Sample Program for ALC Users

```
          BUILD    MODEL=3705-2,                           X
                   CA=(TYPE4, TYPE4),                      X
                   LOCHAN=(40, 50),                        X
                   HICHAN=(4F, 5F),                        X
                   TEST=YES,                               X
                   LINETRC=YES,                            X
                   DYNADMP=(YES,, 5F),                     X
                   LOADLIB=LOD3705,                        X
                   OBJLIB=OBJ3705,                         X
                   NEWNAME=ALC01

          CSB      TYPE=TYPE2,                             X
                   MOD=0,                                  X
                   SPEED=(134, 600),                       X
                   WRAPLN=020

          CSB      TYPE=TYPE3,                             X
                   MOD=1,                                  X
                   SPEED=(150, 600, 1200),                 X
                   WRAPLN=0A4

G1        GROUP    LNCTL=ALC,                              X
                   CLOCKNG=EXT,                            X
                   DUPLEX=FULL,                            X
                   CUTYPE=MIXD

G1L1      LINE     ADDRESS=(0A0, 43, 53-1),                X
                   RCVADDR=(0A1, 42, 54-1),                X
                   SPEED=9600,                             X
                   BUFSIZE=(64, 32),                       X
                   INTPRI=3,                               X
                   CHNPRI=HIGH

G1L2      LINE     ADDRESS=(0A2, 47, 55-1),                X
                   RCVADDR=(0A3, 48, 56-1),                X
                   SPEED=2400

G2        GROUP    LNCTL=BSC,                              X
                   CODE=EBCDIC

G2L1      LINE     ADDRESS=(0A4, 52-1),                    X
                   SPEED=4800,                             X
                   TERM=3275,                              X
                   CU=2701

G3        GROUP    LNCTL=SS,                               X
                   CLOCKNG=INT

G3L1      LINE     ADDRESS=(020, 50-1),                    X
                   SPEED=134,                              X
                   DUPLEX=FULL,                            X
                   UNITXC=NO,                              X
                   TERM=2741

          GENEND
          END
```

Figure L-2. Sample Emulation Program for ALC Users

Glossary

This glossary contains definitions reproduced from the *American National Dictionary for Information Processing*, copyright 1977 by the Computer and Business Equipment Manufacturers Association, copies of which may be purchased from the American National Standards Institute at 1430 Broadway, New York, New York 10018.

ANSI definitions are identified by an asterisk. The symbol (ISO) at the beginning of a definition indicates that the definition has been approved for inclusion in the *Data Processing Vocabulary* of the International Organization for Standardization. The symbol (SC1) indicates that the definition is from an early working paper of ISO Technical Committee 97/Subcommittee 1 and that agreement has not yet been reached among its members.

ACF: Advanced Communications Function.

ACF/NCP/VS: Advanced Communications Function/Network Control Program/Virtual Storage *See also network control program.*

ACF/TCAM: *Advanced Communications Function for the Telecommunications Access Method.*

ACF/VTAM: *Advanced Communications Function for the Virtual Telecommunications Access Method.*

adjacent domains: Domains sharing a common network control program (ACF/NCP/VS) or two domains connected by a cross-domain link. See also *domain.*

adjacent subareas: Two subareas connected by a subchannel, a local-local SDLC link, or a local-remote SDLC link. See also *subarea.*

Advanced Communications Function (ACF): A group of IBM program products (ACF/TCAM, ACF/VTAM, ACF/NCP/VS) that use the concepts of Systems Network Architecture (SNA), including distribution of function and resource sharing. The Multisystem Networking Facility of ACF/TCAM, ACF/VTAM, and ACF/NCP/VS allows the interconnection of two of more domains into one consolodated and coordinated multiple-domain network.

application program: (1) A program written for or by a user that applies to a particular application. (2) A program used to connect and communicate with terminals in a network, enabling users to perform application-oriented activities.

buffer: (1) *A routine or storage used to compensate for a difference in rate of flow, or time of occurrence of events, when transmitting data from one device to another. (2) An area of storage that is temporarily reserved for use in performing an input/output operation, into which data is read or from which data is written.

communication line: A physical connection, such as a wire or a telephone circuit, between communications controllers or between a communications controller and a station. Synonym for line.

communications controller: A type of communication control unit whose operations are controlled by a program stored and executed in the unit. It manages the details of line control and routing of data through a network. It can route data to the host processor, or it can route data to or from a cluster controller or terminal. Examples are the IBM 3704 and 3705 Communications Controllers.

DOS/VS: Disk Operating System/Virtual Storage.

emulation mode The function of a network control program that enables it to emulate a transmission control unit. Contrast with *network control mode*.

host access method: The access method, either ACF/TCAM or ACF/VTAM, that controls data communication with a domain.

host processor: The System/370 with its operating system, access methods, and application programs; the host processor oversees the entire domain. The system services control point (SSCP) is located in the host processor.

load module A program in a format suitable for loading into storage for execution.

logical unit (LU): One of three type of network addressable units; an end user's means of accessing the facilities and services of a network in order to communicate with another end user.

LU: Logical unit.

LU-LU session: In SNA, a session between two logical units in the network. It provides communication between two end users, each associated with one of the logical units.

message: A combination of characters and symbols transmitted from one point to another in a network.

multipoint line: A line or circuit interconnecting several stations. Contrast with *point-to-point line.*

network: (1) The assembly of equipment through which connections are made between installations. (2) A configuration in which two or more station installations are connected.

network addressable unit: In SNA, a logical unit, a physical unit, or a system services control point. It is the origin or the destination of information transmitted in the transmission subsystem layer. Each network addressable unit has a network address that represents it to the transmission subsystem layer.

network control mode: The functions of a network control program that enable it to direct a communications controller to perform activities such as polling, device addressing, dialing, and answering. Contrast with *emulation mode.*

network control program: A control program for the

IBM 3704 and 3705 Communications Controllers generated by the user from a library of IBM-supplied modules. ACF/NCP/VS operates only in the IBM 3705 Communications Controller.

nonswitched line: A connection between a remote station and a communications controller that does not have to be established by dialing. See also *point-to-point line* and *multipoint line.*

OS/VS: Operating System/Virtual Storage.

partitioned emulation programming (PEP) extension: A function of network control program that enables a communications controller to operate some communication lines in network control mode while simultaneously operating others in emulation mode.

path information unit (PIU): In SNA, the unit of transmission consisting of a transmission header (TH) and either a basic information unit (BIU) or a BIU segment.

PEP: Partitioned emulation programming.

physical unit (PU): In SNA, one of three types of network addressable units; a PU is associated with each mode whose existence has been defined to the system services control point (SSCP). A physical unit controls the resources local to its associated node. The SSCP establishes a session with the physical unit as part of the bring-up process.

PIU: Path information unit.

point-to-point line: A line that connects a remote station to a host processor; it may be either switched or nonswitched. Contrast with *multipoint line.*

polling: A technique by which a device is periodically interrogated to determine whether it needs servicing.

primary LU: A logical unit (LU) that issues a Bind Session request to establish an LU-LU session; the logical unit that controls an LU-LU session. Contrast with *secondary LU.*

secondary LU: In an LU-LU session, the partner that receives the Bind Session request. Contrast with *primary logical unit.*

SSCP-LU session: An SNA session between the system services control point (SSCP) and a logical unit (LU). It is used to support logical-unit-related control and use of the communication system. Each logical unit in the network must participate in a session with the SSCP that provides services for that logical unit.

SSCP-PU session: An SNA session between the system services control point (SSCP) and a physical unit (PU) that is used to control the physical configuration of a network and to control an individual node.

station: A point in a data communication network at which data can enter or leave.

subarea: In SNA: (1) A subfield in the network address. (2) The group of network addressable units sharing a common subarea address. The network address space is partitioned into subareas and each subarea is further divided into elements. Subareas are the basic units of routing in SNA.

switched line: A communication line in which the connection between the communications controller and a remote station is established by dialing.

system services control point (SSCP): In SNA, a network addressable unit that provides configuration, maintenance, and session services via a set of command processors—network services—supporting physical units and logical unit. The SSCP must be in session with each logical unit and each physical unit for which it provides these services.

Index

SC30-3142-0

ACF/NCP/VS Network Control Program System Support Programs Installation Printed in U.S.A. SC30-3142-0

IBM®

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601

Cut or Fold Along This Line

ACF/NCP/VS

Network Control Program

System Support Programs

Installation

Order No. SC30-3142-0

READER'S COMMENT FORM

This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you would like a reply, complete the following (Please Print):

Your Name ____________________ Date __________

Company Name ____________________

Department ____________________

Street Address ____________________

City __________ State __________ Zip Code __________

Thank you for your cooperation. No postage stamp is necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)

SC30-3142-0

ADDITIONAL COMMENTS:

Cut or Fold Along Line

ACF/NCP/VS Network Control Program System Support Programs Installation

Please Do Not Staple

Fold and Tape .... Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Dept. E02
P.O. Box 12195
Research Triangle Park
North Carolina 27709

Fold and Tape .... Fold and Tape

IBM®

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, N.Y. 10604

IBM World Trade Americas/Far East Corporation
Town of Mount Pleasant, Route 9, North Tarrytown, N.Y., U.S.A. 10591

IBM World Trade Europe/Middle East/Africa Corporation
360 Hamilton Avenue, White Plains, N.Y., U.S.A. 10601