Burroughs

Reference Manual

B 20 Systems X.25 Network Gateway

(Relative to Release Level 4.0)

Distribution Code SA

Priced Item
Printed in U.S.A.
October 1984

1176104

Burroughs

Reference Manual

B 20 Systems X.25 Network Gateway

(Relative to Release Level 4.0)

Distribution Code SA

Priced Item
Printed in U.S.A.
October 1984

1176104

Burroughs cannot accept any financial or other responsibilities that may be the result of your use of this information or software material, including direct, indirect, special or consequential damages. There are no warranties extended or granted by this document or software material.

You should be very careful to ensure that the use of this software material and/or information complies with the laws, rules, and regulations of the jurisdictions with respect to which it is used.

The information contained herein is subject to change without notice. Revisions may be issued from time to time to advise of such changes and/or additions.

Correspondence regarding this publication should be forwarded, using the Documentation Evaluation Form at the back of the manual, or remarks may be addressed directly to Burroughs Corporation, Corporate Product Information East, 209 W. Lancaster Ave., Paoli, PA 19301, U.S.A.

LIST OF EFFECTIVE PAGES

Page	Issue
iii	Original
iv	Blank
v thru viii	Original
1-1 thru 1-9	Original
1-10	Blank
2-1 thru 2-10	Original
3-1 thru 3-6	Original
4-1 thru 4-42	Original
5-1 thru 5-15	Original
5-16	Blank
6-1 thru 6-9	Original
6-10	Blank
7-1 thru 7-10	Original
A-1 thru A-8	Original
B-1 thru B-7	Original
B-8	Blank
C-1	Original
C-2	Blank
D-1, D-2	Original
Glossary-1 thru Glossary-3	Original
Glossary-4	Blank
1, 2	Original

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

TABLE OF CONTENTS (Cont.)

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1
GENERAL INFORMATION

PRODUCT OVERVIEW

The X.25 Network Gateway is a software product that allows a B20 master, cluster, or standalone workstation to operate on packet-switching public data networks that support CCITT Recommendation X.25. The workstation is connected to a network by an RS-232 port with a synchronous modem, at speeds of up to 9600 baud.

The X.25 Network Gateway provides three levels of access to a public data network (PDN): packet access method (supported by the X.25 Network Gateway system service), Multimode Terminal Program and Sequential Access Method. The X.25 Network Gateway also provides the X.25 status program for monitoring the current status of the X.25 Network Gateway system service. See Figure 1-1 below.

The X.25 Network Gateway is comprised of five programs:

1. The X.25 Network Gateway System Service implements the X.25 protocol for Burroughs workstations. The system service must be installed at the standalone workstation or master workstation in cluster configurations before using any of the other components of the X.25 Network Gateway.

   This system service supports the X.25 Packet Access Method. It consists of a set of procedural interfaces that enables user programs to send and receive individual X. 25 data and control packets and to directly monitor the establishment of X.25 connections. The Packet Access Method is used by the other components of the X.25 Network Gateway for X.25 operations and is available to the communications programmer for use in custom designed X.25 based software.

   The X.25 Network Gateway System Service is accessible through the Packet Access Method requests from standalone and master workstations and from cluster workstations with the system service installed at their master workstation.

2. The X.25 Status Monitor allows the workstation user to display the status of X.25 connections on the video display of standalone and master workstation and of cluster workstations with the system service installed at their master workstation.

3. The X.25 byte stream provides device-independent access to X.25 by way of the Sequential Access Method. The X.25 byte stream provides the application programmer with tools for transmitting and receiving data over PDNs without requiring extensive knowledge of the X.25 protocol.

4. The X.25 byte stream Configuration File Editor is employed by the user to create configuration files for X.25 byte streams. X. 25 byte stream configuration files are used to define X.25 communications options for a particular X.25 byte stream.

5. The Multimode Terminal Program (MTP) allows a workstation end user to communicate with host computers over PDNs.

Packet Access Method

The packet access method enables the user's program to send and receive individual X.25 control and data packets, and to directly monitor the establishment of X.25 connections. This level allows the communications system programmer to design sophisticated CCITT Recommendation X.25-based communications applications such as interfaces to other computer networks, server programs, and packet assembly/disassembly (PAD) facilities. Its use requires extensive knowledge of the CCITT Recommendation X.25.

Packet access operations perform the following functions:

- Call establishment and clearing operations begin and end calls over virtual circuits.

- Data transfer operations transfer data over an established virtual circuit.

- A status operation monitors the status of the X.25 Network Gateway system service.

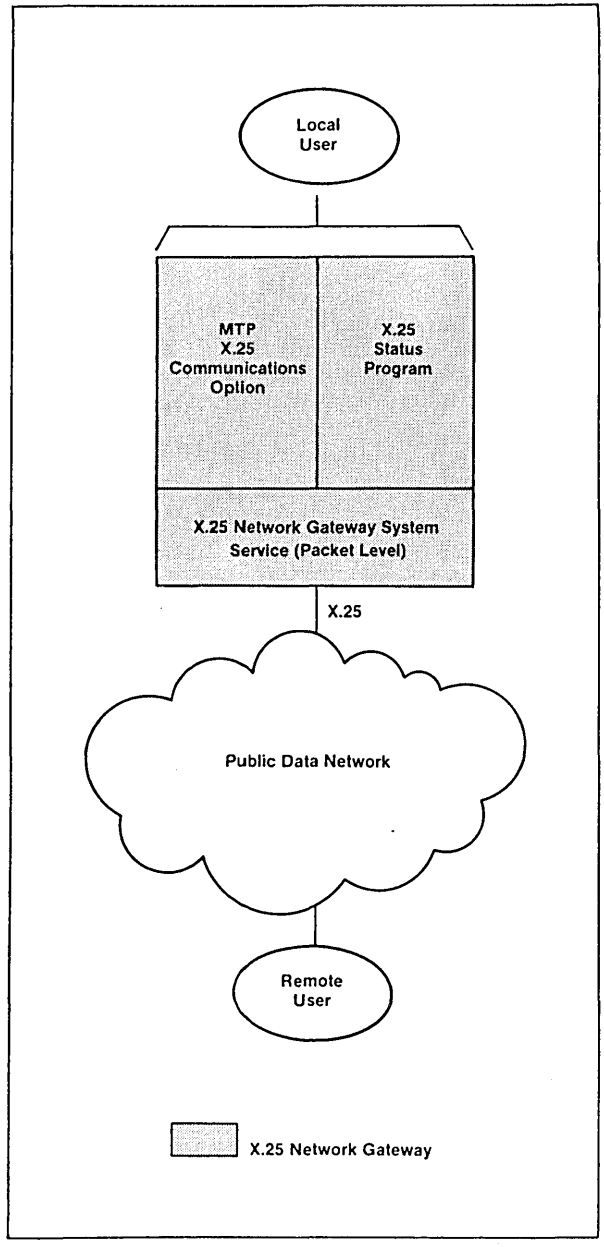


Figure 1-1. X.25 Network Gateway.

These operations are used by the other levels of the X.25 Network Gateway for CCITT Recommendation X.25 communications. Packet access method operations are served by the X.25 Network Gateway system service, which is the B20 implementation of CCITT Recommendation X.25 packet and link level protocols.

Sequential Access Method

The Sequential Access Method (SAM) enables the user to send data to other computer systems over a PDN without the sophisticated techniques required by the pocket access method. This level of access is appropriate for the applications programmer because its use does not require a detailed knowledge of the CCITT Recommendation X.25.

Sequential access operations perform packet assembly and disassembly in a manner compatible with CCITT Recommendations X.3 and X.29.

Multimode Terminal Program (MTP)

The Multimode Terminal Program (MTP) is designed for the unsophisticated communications user. This allows a user's workstation to appear as an intelligent terminal to a host computer on a PDN through CCITT Recommendation X.25.

The Multimode Terminal Program User's Guide provides an introduction to MTP as well as basic operating information. Those using it should read the Multimode Terminal Program User's Guide first. The Multimode Terminal Program Reference Manual provides detailed information about MTP.

RELEASE LEVEL

This manual supports release level 4.0 of X.25 Network Gateway software. This release contains the following files:

```
 [f0]<sys>crashDump.sys
 [f0]<sys>Fdsys.version
 [f0]<sys>fileHeader.sys
 [f0]<sys>mfd.sys
 [f0]<sys>sysImage.sys
 [f0]<sys>install.sub
 [f0]<sys>log.sys
 [f0]<sys>badBlk.sys
*[f0]<sys>diagTest.sys
*[f0]<sys>bootExt.sys
 [f0]<sys>x25config.sys
 [f0]<B20X25-1>MTP.Run
 [f0]<B20X25-1>Mte-Ini
 [f0]<B20X25-1>X25.Run
 [f0]<B20X25-1>X25Mon.Run
 [f0]<B20X25-1>Mte-Hlp
 [f0]<B20X25-1>X25config.Run
 [f0]<B20X25-1>X25SAM.LIB
 [f0]<B20X25-1>X25ss.cmd
```

* Only present on 8" diskette

New Features Included in Release 4.0

1. Supports the use on the B25 series.

2. Supports operation on Telenet and Transpac. The "Install X.25 command" has been modified to allow you to specify the "[PDN Type]". The allowed entries are as follows:

   default - Telenet, Tymnet
   "2" - TransPac

3. Maximum number of Virtual Circuits is limited to 32.

4. The X.25 Network Gateway operates at data rates between 2400 and 9600 baud.

Applications of Release 4.0

This version of the X.25 Network Gateway can be used on the following Public Data Networks.

1. Tymnet
2. Telenet
3. Transpac

Model	:	B21, B22, B25
Host O.S	:	B20 Operating System Level 4.0 or higher
Link Control	:	HDLC
Access Procedures	:	LAPB

Limitations/Deviations

1. Permanent Virtual Circuits are not supported for the 4.0 release.

2. Packet sizes of 16, 32, 64, 256, 512, and 1024 are also not supported.

3. WriteX25Packet procedural call is a blocking operation which is not complete until the Gateway receives Packet Level acknowledgement for data packets sent out. As a result, it is not easy to issue multiple "WriteX25Packet" without creating multiple processes all using the same logical channel, which requires some coding to take advantage of the "Window size" feature.

4. Data packets longer than 128 bytes are not accepted by the ReadX25Packet and block further data packets from being read.

MEMORY UTILIZATION

The X.25 Network Gateway System Service requires 35,800 bytes and a variable amount of memory for buffers and control structures at the master or standalone workstation where it is installed.

The amount of memory required for buffers and control structures is based on installation parameters and is calculated as follows:

Number of Bytes = (Number of Circuits * 176) + [(2 * Number of Circuits) + 8) * Default window size * (Max packet size + 12)]

(* indicates multiplication.)

Using the default installation parameters for default window size (2) and max packet size (128), approximately 750 bytes of memory are required per circuit.

The maximum size of the memory used for buffers and control structures is 42 K-bytes.

The X.25 Status Monitor requires 31,700 bytes at the workstation where it is run.

X.25 bytestreams requires 3,900 bytes and a 1,024 byte buffer per open byte stream (supplied by the user program) at the cluster, master or standalone workstation where a program using X.25 bytestreams is run.

The X.25 Bytestream Configuration File Editor requires 4,600 bytes at the cluster, master or standalone workstation where it is run.

MTE requires 71,700 bytes and as much memory as is available for its display memory at the cluster, master or standalone workstation where it is run. The maximum size of MTP display memory is 64 K-bytes.

REQUIRED FILES

Individual X.25 Network Gateway programs require only some of the files comprising the product. If only some X.25 Network Gateway programs are to be used, the files required for the unused programs need not be present on your hard disk. Required files for each program are as follows:

* X.25 Network Gateway System Service
X25.Run

X.25 Status Monitor
X25Mon.Run

X.25 Bytestreams
X25Sam.Lib

X.25 Bytestream Configuration File Editor
X25Config.Run

Multimode Terminal Emulator
MTE.Run
Mte-Ini
Mte-Hlp

* The X.25 Network Gateway System Service must be installed before any other X.25 Network Gateway programs may be used.

An X.25 bytestream accesses a default configuration file, [Sys]<sys>X25Config.sys, if no configuration file is specified when the bytestream is opened. Users can create their own Configuration Files with the X.25 Bytestream Configuration File Editor. "Configure X.25" command has to be invoked for creating configuration files.

SOFTWARE DEPENDENCIES

This release has been qualified to run on Operating System release level 4.0 or higher.

HARDWARE DEPENDENCIES

The X.25 Network Gateway uses the workstation onboard serial channels for X.25 communications. No special workstation hardware is required to run the X.25 Network Gateway.

Typically, the connection between the workstation and a Public Data Network (PDN) for X.25 communications consists of a leased telephone line with synchronous modems. (This equipment is normally supplied by the PDN).

SUPPORT OF CCITT RECOMMENDATIONS

The B20 X.25 Network Gateway supports CCITT Recommendations X.3, X.21bis, X.25, X.28, and X.29. The "Network Interaction" subsections within the sections below describe how the protocols defined in these recommendations are supported.

SECTION 2
CONCEPTS

SWITCHING

Circuit and Message Switching

Switching, which is defining the communications path between parties, is a major element in all communications systems. Two schemes are used in switching: (1) preallocating the path for the duration of the communication, and (2) dynamically allocating the path while the communication is in progress.

Telephone, Telex, and TWX systems, examples of preallocating schemes, use a fixed path for the duration of a call. These systems use a switching method known as circuit switching, in which switching entities (for example, switchboards) set up a connection between two parties. This connection is dedicated to the parties until one party terminates it. Information transferred along this connection is conveyed without delay as all switching decisions were performed when the communication was established.

Mail, telegraph, and message systems, examples of dynamic allocation schemes, operate by dynamically allocating paths between parties as information is moved from the sender to the receiver. These systems use a switching method known as message switching, in which switching entities (for example, the post office) route information dynamically without first establishing the entire source-to-destination path. In message switching, the communications path between parties is shared by other users. However, delays in information transfer occur while switching decisions are made and implemented.

Historically, circuit switching was the preferred method as it has very low delays in data transfer, thus allowing virtually interactive communication. However, circuit switching is not efficient since a connection is reserved between two parties even if information is not actually being transferred. As most communications have gaps where information is not being exchanged, a circuit-switched call often causes underutilization of the communications medium.

Message switching allows consistently high utilization of a communications path by sharing (multiplexing) it among a number of users and filling the gaps in one user's communication with information from another's. However, message switching has traditionally required human, time-consuming switching decisions at the switching entities. Further, a user transferring very long messages can dominate a message-switched system, thereby causing excessive delays for other users.

Packet Switching

Packet switching is a type of message switching in which messages are divided into small segments, or packets. These packets are transferred by a message-switched method and reassembled at their destination. The advent of high-speed digital computers allowed the development of very fast message-switching entities that can move packets along a dynamically allocated route at high speeds. Packet switching offers the traditionally high media utilization characteristic of message switching without the long switching delays and susceptibility to single-user monopolization that have plagued such systems.

CCITT RECOMMENDATION X.25

As packet switching became more popular in computer communications during the 1970s, a number of corporations began offering public data communications services over packet-switched PDNs. Because there was no common standard for internal operation, PDNs differed in design and operation. The resulting variety of user interface requirements made it apparent that a standard interface allowing communication with all PDNs was required for user equipment.

In 1976, the CCITT Sixth Plenary Assembly adopted Recommendation X.25, the standard device-independent interface between packet-switched PDNs and user devices operating in the packet-switched mode. In February, 1980, a revised Recommendation X.25 was issued. The B20 uses this revised standard for the implementation of X.25 as defined in this Manual (see Appendix C).

CCITT Recommendation X.25 defines a protocol to connect a user's terminal (data terminal equipment, or DTE) to a network node (data communications equipment, or DCE). The protocol requires that CCITT Recommendation X.25 support be resident in both the DTE and the DCE. X.25 Network Gateway is an example of required DTE-resident X.25 software. (Although some PDNs use CCITT Recommendation X.25 for their internal protocol, only a PDN's external behavior, from the perspective of the user, must match the CCITT Recommendation X.25 protocol.)

Protocol Levels

CCITT Recommendation X.25 specifies three protocol levels:

- The physical level defines the electrical or modem interface between the DTE and DCE, based on Recommendation X.21 or X.21bis (RS-232).

- The link level defines the link access procedure for transferring packets accurately between the DTE and DCE. The link level consists of procedures for link setup, data transfer, and link disconnect. The protocol used is LAPB, a symmetric version of the HDLC protocol.

- The packet level defines procedures for call establishment, data transfer, flow control, error recovery, and call clearing between two communicating DTEs. Up to 4,096 logical channels can be multiplexed over a single link, using the packet-level protocol.

X.25 Multiplexing

The multiplexing capability of the CCITT Recommendation X.25 packet level allows the single physical link between the DTE and the DCE to be shared by multiple users. Users transfer data over logical channels, known as virtual circuits. Despite sharing the same physical line to the DCE, these logical channels appear to users as private communications links to the parties with which they are communicating.

Intermixing of data from various logical channels onto the physical link is handled transparently to users by the CCITT Recommendation X.25 packet level. Each transferred packet is marked with a logical channel number. This number enables the CCITT Recommendation X.25 packet level to match packets and logical channels.

Up to 4,096 logical channels are supported by the CCITT Recommendation X.25 packet level; any number can be active simultaneously. Link use is allocated so no logical channel can monopolize the physical line. This random multiplexing technique allows maximum use of the connection between the DTE and the DCE, independent of the number of users and the amount of data transferred.

CCITT RECOMMENDATIONS X.3, X.28, AND X.29

CCITT Recommendations X.3, X.28, and X.29 define the protocols for interfacing nonpacket-mode asynchronous (stop/start) terminals to a PDN. (See Figure 2-1 below.) Special software is required for these protocols to process the data from asynchronous terminals into a format compatible with CCITT Recommendation X.25. This software is termed a packet assembly/disassembly (PAD) facility. Within the PDN, a PAD is basically another packet-mode DTE.

Recommendation X.3 defines the parameters that specify an asynchronous terminal's characteristics to a PAD.

Recommendation X.28 defines the procedure for communicating these parameters between an asynchronous terminal and a PAD.

Recommendation X.29 defines the procedure for communicating between a PAD and a true packet-mode DTE over the PDN. This procedure allows a true packet-mode DTE to access Recommendation X.3 parameters in a manner similar to that of an asynchronous terminal.

LOGICAL CHANNELS AND VIRTUAL CIRCUITS

A logical channel is one multiplexed data stream over a physical line. A virtual circuit is a two-party connection that uses a logical channel.

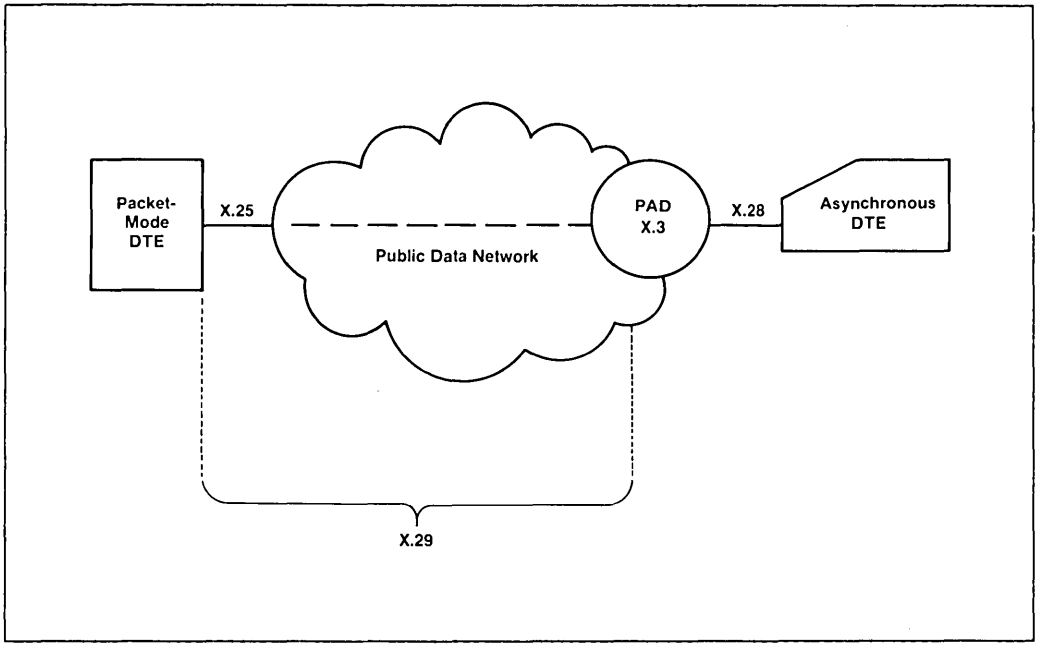


Figure 2-1. Public Data Network Protocols.

X.25 Network Gateway defines 4,096 logical channels for each DTE-DCE physical link. These logical channels are divided into 16 logical channel groups. The logical channels within each group are numbered from 0 to 255. (See Figure 2-2 below.)

Logical channel 0 of logical channel group 0 is reserved for protocol operations. The remaining logical channels are divided into virtual circuit ranges at installation: two-way (both incoming and outgoing), incoming, outgoing, and permanent. The ordering of these virtual circuits on the DTE-DCE link is (1) permanent, which must begin with logical channel 1 of logical channel group 0, (2) incoming, (3) two-way, and (4) outgoing. Only virtual circuit 0 and one other virtual circuit need be present. Although virtual circuit ranges do not have to be contiguous, the logical channel numbers within each virtual circuit range must be contiguous.

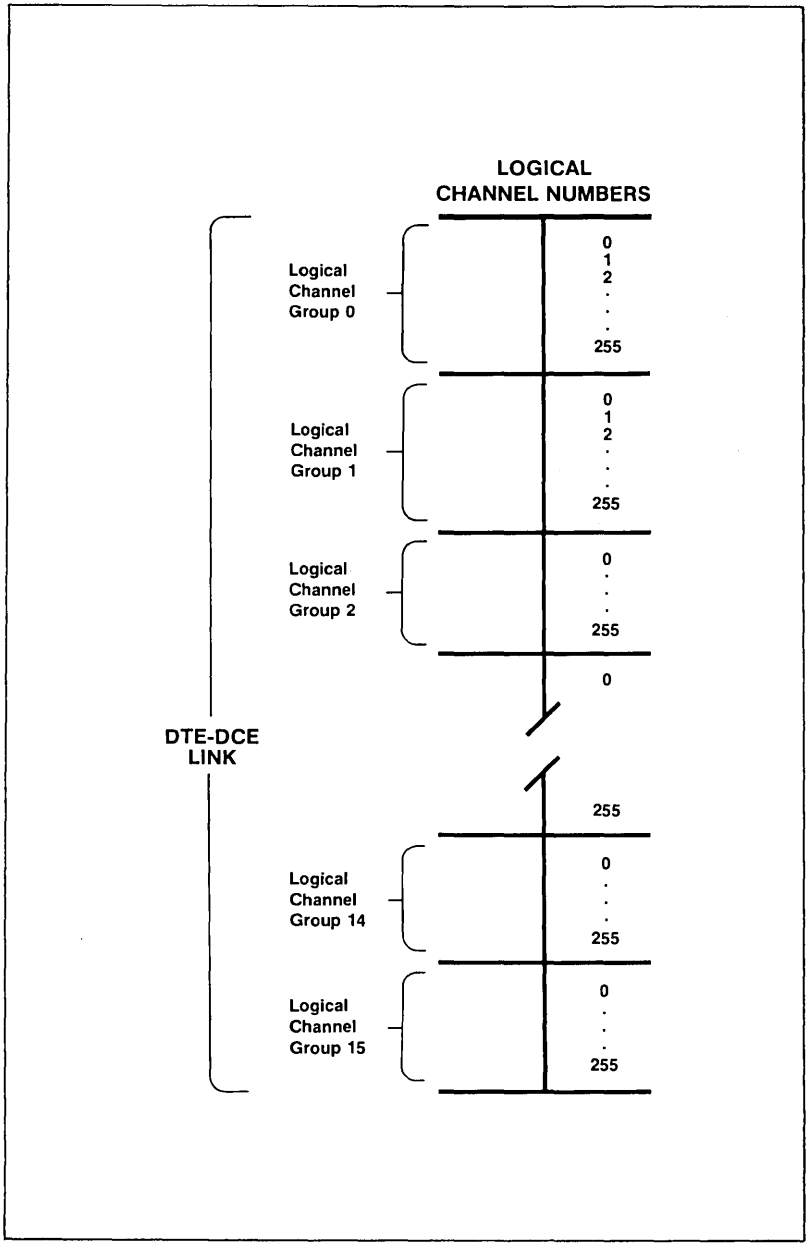


Figure 2-2. Logical Channels on a DTE-DCE Link.

Figures 2-3 through 2-5 below show several examples of virtual circuit ordering. In Figure 2-3, 19 virtual circuits are divided among four virtual circuit ranges; all are within logical channel group 0. These virtual circuits are ordered as follows:

- three permanent virtual circuits, beginning with logical channel 1,
- seven incoming virtual circuits, beginning with logical channel 4,
- six two-way virtual circuits, beginning with logical channel 11, and
- three outgoing virtual circuits, beginning with logical channel 29.

In Figure 2-4 below, all 32 virtual circuits are two-way virtual circuits, beginning with logical channel 240 in logical channel group 12 and continuing to logical channel 15 in logical channel group 13.

In Figure 2-5 below, ten virtual circuits are divided between two ranges, each range in a different logical channel group. These virtual circuits are ordered as follows:

- four permanent virtual circuits, beginning with logical channel 1 in logical channel group 0, and
- six two-way virtual circuits, beginning with logical channel 11 in logical channel group 2.

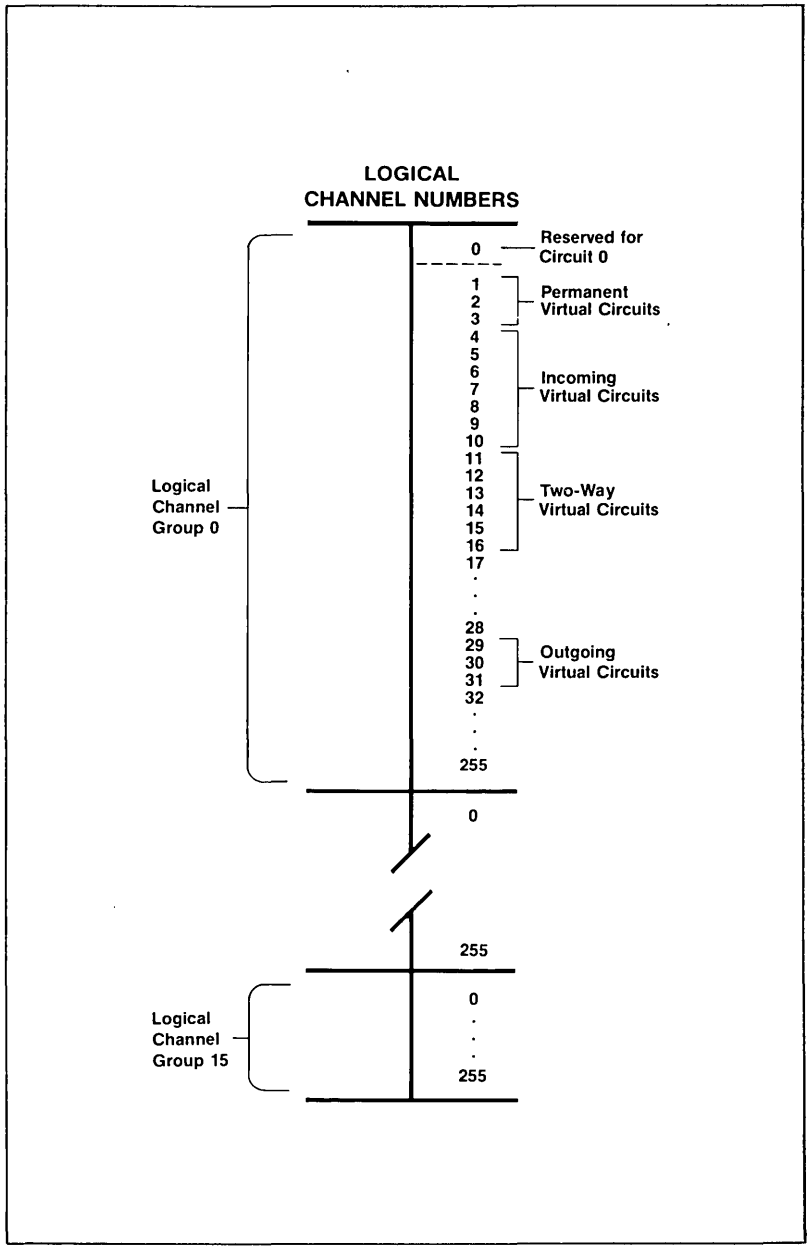


Figure 2-3. Example of Virtual Circuit Definition: Four Ranges within One Logical Channel Group.

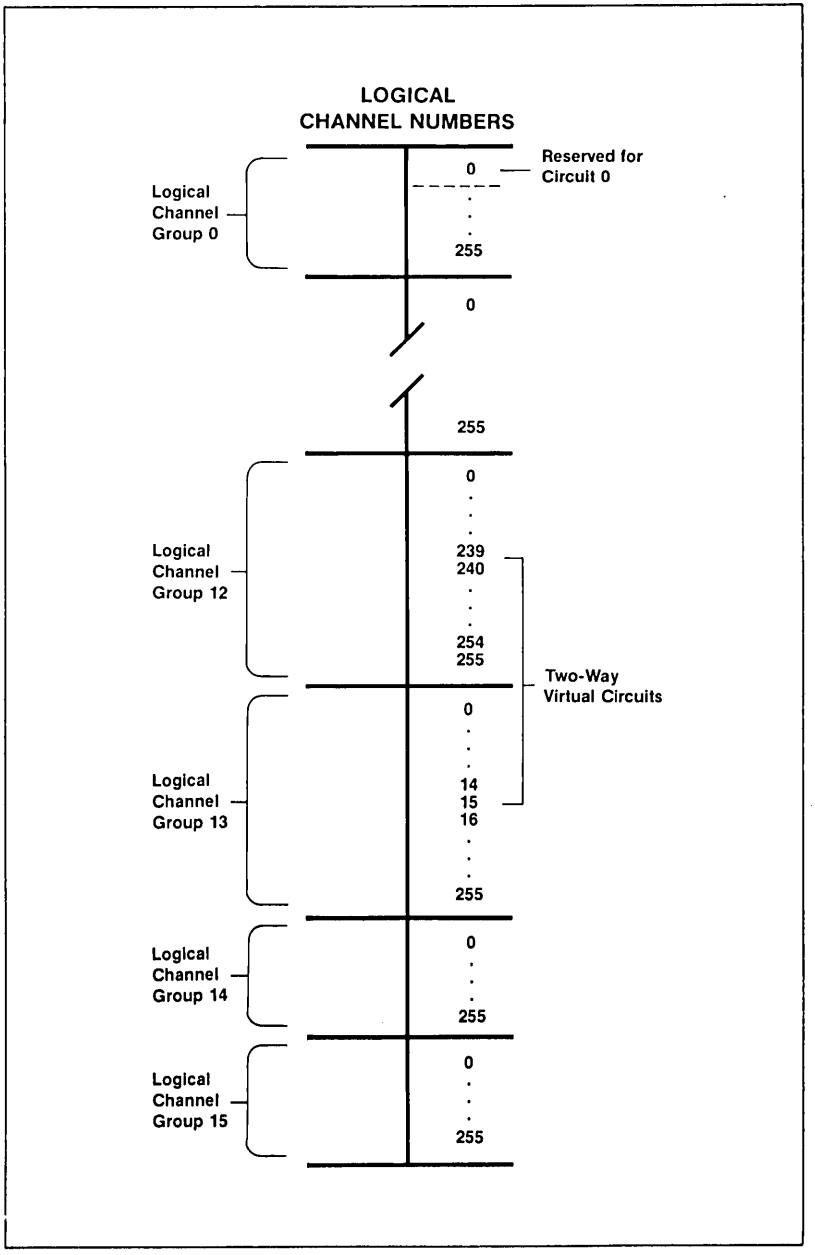


Figure 2-4. **Example of Virtual Circuit Definition: One Range Spanning Two Logical Channel Groups.**

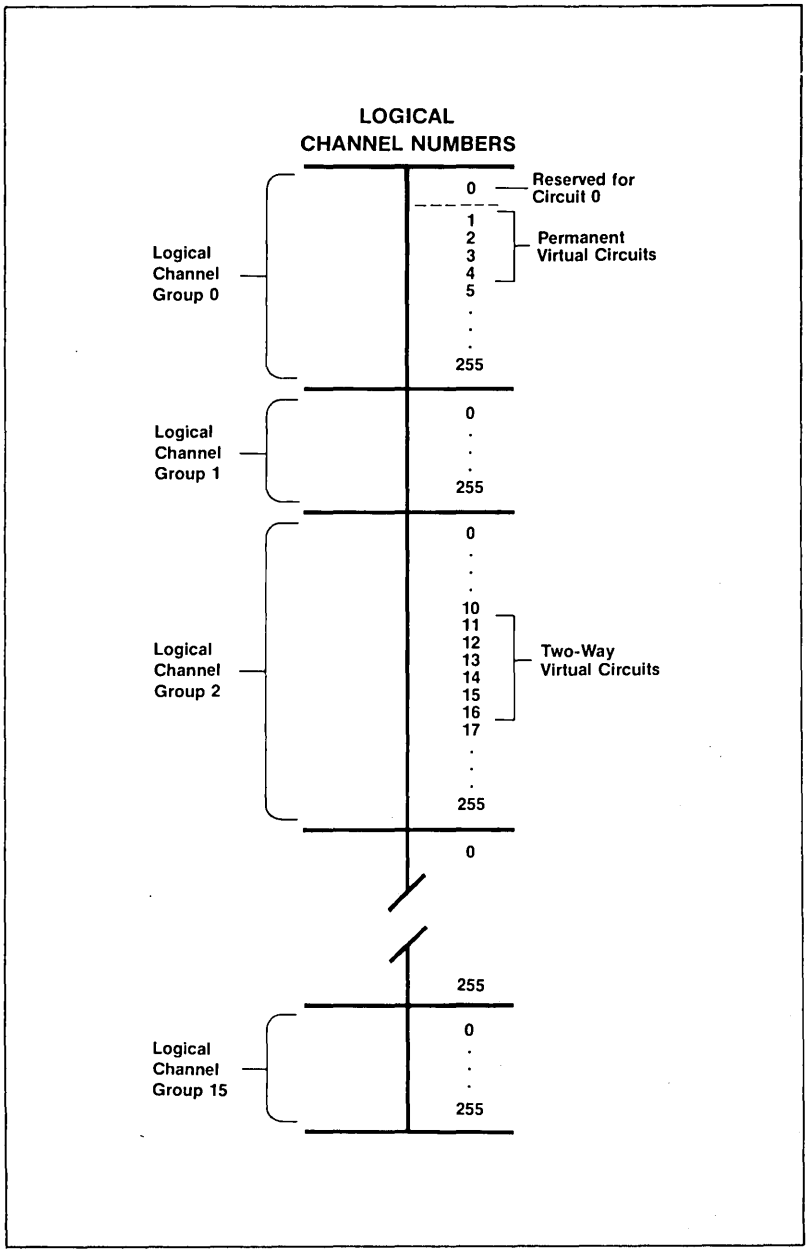


Figure 2-5. **Example of Virtual Circuit Definition: Two Ranges, Each in a Different Logical Channel Group.**

SECTION 3
INSTALLATION PROCEDURES: INSTALL X.25 COMMAND

INSTALLATION INSTRUCTIONS

1. Login as follows.

   ```
   Command  Path                        press RETURN
   Path
     Volume                  sys        press RETURN
     Directory               sys        press RETURN
     [Default file prefix]
     [Password]
   ```

   If your hard disk has a volume password on [d0], fill this password into the [Password] field. Press GO.

2. Power down all cluster workstations.

3. Insert the product distribution diskette, labelled X.25 Network Gateway in drive [f0]. (Do NOT press the RESET button).

4. Install the product as follows:

   ```
   Command Software Installation    Press GO
   ```

5. You will be prompted to power down your cluster stations if you have not already done so. Press GO.

6. The message "INSTALLATION OF BURROUGHS X25 NETWORK GATEWAY COMPLETE" will be displayed. When installation is complete, remove the product distribution diskette and save it as an archive.

7. Power up your cluster workstations.

8. If this is the first time the B22 has been used for synchronous communication, power it off and have a qualified service technician remove the memory-I/O board. Make sure that the communications channel used for connection to the host computer is configured for external clock. This is discussed further in Appendix D of this manual. When this is complete, return the board to its slot, and power the B22 back up.

If you have a B21, no adjustments are necessary.

INSTALLATION HINTS

The following hints should be considered if problems are encountered during installation of the X.25 Network Gateway.

1. If you are using a B22, make sure that the clocking switches on the I/O memory board are set for external clock. This is discussed further in Appendix D of this manual.

2. You should have a modem and a leased line, supplied by either the Public Data Network to which you are subscribing or by your local telephone network.

3. The X.25 Network Gateway expects and requires a full-duplex connection. Both the modem to which the B20 is connected and the modem which the PDN is using at the other end of your leased line must be strapped for continuous carrier. If either modem is not connected properly, the X.25 Gateway will not operate correctly.

OPERATIONAL HINTS

1. The X.25 Network Gateway can theoritically support 4096 virtual circuits. However, the present release can support only 32. When the X.25 Network Gateway is installed by the "Install X.25" command, the parameter to be entered for "Number of VCs" should not exceed 32.

2. The parameter to be entered for "Number of VCs" in the "Install X.25" command should be always one more than the number of virtual circuits to be used.

3. Default values are built in for all the parameters for the "Install X.25" command.

4. The system has to be rebooted each time the Gateway is installed.

5. The Gateway should only be used on hard-disk systems.

(VC is the abbreviation for virtaul circuit, and LC is the abbreviation for logical channel. All numbers are decimal.)

```
Install X.25
  [Net address]
  [Number of VCs]
  [LC group number for two-way VCs]
  [Starting LC niumber for two-way VCs]
  [Number of incoming-only VCs]
  [LC group number for incoming-only VCs]
  [Starting LC number for incoming-only VCs]
  [Number of outgoing-only VCs]
  [LC group number for outgoing-only VCs]
  [Starting LC number for outgoing-only VCs]
  [Number of permanent VCs]
  [Channel (A or B)]
  [Modulus (8 or 128)]
  [Max packet size]
  [Default packet size]
  [Default window size]
  [PDN type]
```

where

[Net address]
: is the calling address to be inserted into all CALL REQUEST packets (for further information, see the "Packet Access Method" section below). Up to 14 digits may be entered.

[number of VCs]
: is the number of all virtual circuits to be allocated. This number can be between 1 and 32. The default if either 1, for a standalone workstation, or the total number of workstations, including the master in a cluster configuration, where the X.25 system service is required.

NOTE

While up to 32 virtual circuits can be allocated during installation, the actual number allocated must equal the number of PDN virtual circuits to which the user has subscribed plus 1 (for virtual circuit 0).

The number of two-way, incoming, outgoing, and permanent virtual circuits cannot exceed the number of allocated virtual circuits. X.25 Network Gateway calculates the number of two-way circuits by subtracting the number of incoming, outgoing, and permanent virtual circuits from the number of allocated virtual circuits.

[LC group number for two-way VCs]
is the logical channel group number for two-way virtual circuits. The number can be between 0 (the default) and 15.

[Starting LC number for two-way VCs]
is the starting logical channel number for two-way virtual circuits. The number can be between 0 and 255. The default is the channel number of the last outgoing-only virtual circuit plus 1.

[Number of incoming-only VCs]
is the number of virtual circuits to be allocated for incoming calls only. The number can be between 0 (the default) and the total number of allocated virtual circuits.

[LC group number for incoming-only VCs]
is the logical channel group number for incoming virtual circuits. The number can be between 0 (the default) and 15.

[Starting LC number for incoming-only VCs]
is the starting logical channel number for incoming virtual circuits. The number can be between 0 and 255. The default is the number of permanent virtual circuits plus 1.

[Number of outgoing-only VCs]
is the number of virtual circuits to be allocated to outgoing calls only. The number can be between 0 (the default) and the total number of allocated virtual circuits.

[LC group number for outgoing-only VCs]
is the logical group number for outgoing virtual circuits. The number can be between 0 (the default) and 15.

[Starting LC number for outgoing-only VCs]
is the starting logical channel number for outgoing virtual circuits. The number can be between 0 and 255. The default is the channel number of the last two-way virtual circuit plus 1.

[Number of permanent VCs]
is the number of permanent virtual circuits and can be between 0 (the default) and the total number of allocated virtual circuits.

[Channel (A or B)]
is the communications channel and is either Channel A or Channel B (the default).

[Modulus (8 or 128)]
is the type of packet sequence numbering. For this release, 8 is the only value supported.

[Max packet size]
is the maximum packet size, in bytes, and can be one of the following: 16, 32, 64, 128 (the default), 256, 512, or 1024. Release level 4.0 supports only a value of 128.

[Default packet size]
is the default packet size, in bytes, and can be one of the following: 16, 32, 64, 128 (the default), 256, 512, or 1024.

[Default window size]
is the default window size. The number can be between 1 and 6. The default is 2.

SECTION 4

PACKET ACCESS METHOD

OVERVIEW

The packet access method provides access to the capabilities of the X.25 Network Gateway system service through standard BTOS mechanisms. In this manner, it enables the user's program to send and receive individual X.25 data and control packets, and to directly monitor the establishment of X.25 connections to a PDN.

The packet access method allows the communications system programmer to design sophisticated, CCITT Recommendation X.25-based communication products such as interfaces to other computer networks, server programs, and PAD facilities. Thus, the packet access method is intended for users who are very knowledgeable about CCITT Recommendation X.25. In particular, the application program must understand the format and contents of X.25 packets, as specified in Section 6 of CCITT Recommendation X.25 (referenced in Appendix C of this Manual).

USAGE

A standard X.25 user session consists of (1) a call establishment phase, (2) a data transfer phase, and (3) a call clearing phase. Status can be monitored during a session.

The call establishment phase consists of either (1) placing an outgoing call or receiving an incoming call or (2) reserving a permanent virtual circuit.

The data transfer phase consists of reading and/or writing data.

The call clearing phase is used to terminate a session on a nonpermanent virtual circuit.

Five operations enable a calling process to begin (establish) and end (clear) calls over virtual circuits: AcceptX25Call, ClearX25Call, ConnectX25Permanent, InitiateX25Call, and NotifyNextIncomingCall. These operations should be used as follows:

- Reserving a permanent virtual circuit:

  ConnectX25Permanent

- Placing an outgoing call:

  InitiateX25Call

- Receiving an incoming call:

  NotifyNextIncomingCall

  then either

  AcceptX25Call (to complete the establishment of a call)

  or

  ClearX25Call (to reject the establishment of a call)

- Terminating a call:

  ClearX25Call

Once a virtual circuit has been established, the following four operations can be used to transfer

data over it: ReadX25Packet, ResetX25Call, WriteX25Interrupt, and WriteX25Packet. QueryX25Status can be used to monitor status. These operations can be used in any sequence.

The packet access method operations are served by the X.25 Network Gateway system service, which is the B20 implementation of CCITT Recommendation X.25 packet and link level protocols.

Specifying and Setting M, D, and Q Bits

A user can specify (for transmitted packets) or examine (for received packets) relevant M, D, and Q bit values on a per-packet basis with packet access method operations:

- AcceptX25Call specifies values for a transmitted CALL ACCEPT packet.
- InitiateX25Call specifies values for a transmitted CALL REQUEST packet and returns values from a received CALL ACCEPT packet.
- NotifyNextIncomingCall returns values from a received CALL REQUEST packet.
- ReadX25Packet returns values from a received DATA packet.
- WriteX25Packet specifies values for a transmitted DATA packet.

Not all bits are meaningful for all packets:

Packet	Valid Bits		
	M	D	Q
ACCEPT CALL		x	
INCOMING CALL		x	
DATA	x	x	x

Invalid bits are ignored, if specified, or set to 0, if returned. For data packets, all bit settings are considered valid by the X25 Network Gateway.

For received CALL REQUEST packets, the entire packet is returned to the buffer specified in the NotifyNextIncomingCall operation. The user can then examine the bit values.

For received CALL ACCEPT packets, the entire packet is returned to the buffer specified in the

InitiateX25Call operation. The user can then examine the bit values.

For all other packets, bit values are specified or examined with the general format indicator parameter. The format of the general format indicator is shown below in Table 4-1. Refer to Section 6.3 of CCITT Recommendation X.25 for additional information concerning these bits.

Table 4-1. General Format Indicator

Bit	Meaning
0	More data indicator (M bit). Indicates start and end of multi-packet messages (DATA packets only).
1-5	Reserved.
6	Delivery confirmation bit (D bit). Indicates local (immediate) or end-to-end (delayed) delivery confirmation (DATA, INITIATE CALL, and ACCEPT CALL packets).
7	Qualified data bit (Q bit). Indicates special packet content (DATA packets only).

For AcceptX25Call, InitiateX25Call, and WriteX25-Packet operations, the general format indicator is a request parameter. Bit values are specified by the user in the bGfi field of the request.

For ReadX25Packet operations, the general format indicator is a response parameter. Bit values are returned to the memory location pointed to by the user-specified parameter pGfiRet.

OPERATIONS: SERVICES

Packet access method operations are categorized by function in Table 4-2 below.

Table 4-2. Packet Access Method Operations by Function

Call Establishment and Clearing	
	AcceptX25Call
	ClearX25Call
	ConnectX25Permanent
	InitiateX25Call
	NotifyNextIncomingCall
Data Transfer	
	ReadX25Packet
	ResetX25Call
	WriteX25Interrupt
	WriteX25Packet
Status Monitoring	
	QueryX25Status

Call Establishment and Clearing

AcceptX25Call
: requests transmission of an ACCEPT CALL packet to complete the establishment of an incoming call.

ClearX25Call
: requests transmission of a CLEAR REQUEST packet to either refuse an incoming call or terminate an existing call.

ConnectX25Permanent
: reserves a permanent virtual circuit.

InitiateX25Call
: requests transmission of a CALL REQUEST packet to initiate a call to the called number.

NotifyNextIncomingCall
: notifies the X.25 Network Gateway system service that the calling process wishes to receive an incoming call.

Data Transfer

ReadX25Packet
: returns DATA and INTERRUPT packets received by the X.25 Network Gateway system service from the PDN.

ResetX25Call
: requests that a RESET REQUEST packet be transmitted in order to clear all outstanding read and write requests and reinitializes the flow control procedures for a virtual circuit without clearing the call.

WriteX25Interrupt
: transmits an INTERRUPT packet over the PDN.

WriteX25Packet
: causes transmission of a DATA packet over the PDN.

Status Monitoring

QueryX25Status
: returns a set of statistics about activity over the PDN.

AcceptX25Call

Description

The AcceptX25Call service requests transmission of an ACCEPT CALL packet to complete the establishment of an incoming call. AcceptX25Call can be issued after the successful completion of a NotifyNextIncomingCall operation.

Only one AcceptX25Call can be outstanding for a virtual circuit at any given time. If more are queued, the excess requests are returned with a status code of 8502 ("Maximum number of this request has been queued").

To take advantage of the Fast Select facility or to engage in facility negotiation, the optional facilities field and/or the user data field in the CALL ACCEPTED packet can be sent to the caller. Details of the required facilities and user data fields formats are given in Section 7 of CCITT Recommendation X.25. For further information on facilities, see the "Network Interaction" subsection below.

Procedural Interface

```
AcceptX25Call (vch, pFacilities, sFacilities,
               pUserData, sUserData, bGfi):
               ErcType
```

where

vch — is the virtual circuit handle returned by a NotifyNextIncomingCall operation.

pFacilities
sFacilities — describe a buffer containing facilities data to be included in the CALL ACCEPTED packet.

pUserData
sUserData — describe a buffer containing user data to be included in the CALL ACCEPTED packet. This field is used for Fast Select calls only.

bGfi — is the one-byte general format indicator. Bit 6 of this byte can be

set if the D bit is to be used during the call (bit 7 is the most significant bit). (See the "Specifying and Setting M, D, and Q Bits" subsection above.)

Request Block

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	6
2	nReqPbCb	1	2
3	nRespPbCb	1	0
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	166
12	vch	2	
14	reserved	2	
16	bGfi	1	
17	reserved	1	
18	pFacilities	4	
22	sFacilities	2	
24	pUserData	4	
28	sUserData	2	

ClearX25Call

Description

The ClearX25Call service requests transmission of a CLEAR REQUEST packet to either refuse an incoming call or terminate an existing call. If a call is cleared by the PDN, the status code 8513 ("DCE clear") is returned to any outstanding data transfer requests.

Only one ClearX25Call can be outstanding for a virtual circuit at any given time. If more are queued, the excess requests are returned with a status code of 8502 ("Maximum number of this request has been queued").

Procedural Interface

```
ClearX25Call (vch, bReason, pUserData,
              sUserData): ErcType
```

where

vch	is the virtual circuit handle.
bReason	is a one-byte diagnostic code to be included in the CLEAR REQUEST packet.
pUserData sUserData	describe a buffer containing user data to be included in the CLEAR REQUEST packet. This field should be used only if ClearX25Call is used to reject a Fast Select call (this is not checked by the X.25 Network Gateway system service).

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	6
2	nReqPbCb	1	1
3	nRespPbCb	1	0
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	168
12	vch	2	
14	bReason	1	
15	reserved	3	
18	pUserData	4	
22	sUserData	2	

ConnectX25Permanent

Description

The ConnectX25Permanent service reserves a permanent virtual circuit for a user. ConnectX25-Permanent returns a virtual circuit handle that can then be used to reference the permanent virtual circuit in subsequent data transfer requests.

Only one ConnectX25Permanent can be outstanding for a virtual circuit at any given time. If more are queued, the excess requests are returned with a status code of 8502 ("Maximum number of this request has been queued").

No packets are transmitted by ConnectX25-Permanent.

Procedural Interface

```
ConnectX25Permanent (lcn, pVchRet): ErcType
```

where

lcn — is a logical channel number in the range 1 through the maximum number of allocated virtual circuits. The logical channel number must have been specified as a permanent virtual circuit at installation.

pVchRet — is the memory address of a word to which the virtual circuit handle is returned.

sVchRet is always 2.

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	2
2	nReqPbCb	1	0
3	nRespPbCb	1	1
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	175
12	lcn	2	
14	pVchRet	4	
18	sVchRet	2	2

InitiateX25Call

Description

The InitiateX25Call service requests transmission of a CALL REQUEST packet to initiate a call to the called number, which can be any user in the PDN. The local network address number, as specified at installation, is automatically filled in as the calling number by the X.25 Network Gateway system service.

Only one InitiateX25Call can be outstanding for a virtual circuit at any given time. If more are queued, the excess requests are returned with a status code of 8502 ("Maximum number of this request has been queued").

Optional parameters for facilities and user data can be specified. The specified parameters must be in accordance with applicable standards, either those of CCITT Recommendations X.25 and X.29, those of other international or national standards bodies, or those unique to the specific PDN. For further information on facilities, see below to the "Network Interaction" subsection of this section.

Procedural Interface

```
InitiateX25Call (iNet, pVchRet, pPacketRet,
                 sPacketMax, psPacketRet,
                 pCalled, sCalled, pFacilities,
                 sFacilities, pUserData,
                 sUserData, bGfi): ErcType
```

where

iNet — is the communications network identifier, for which the only value currently allowed is 0.

pVchRet — is the memory address of a word to which the virtual circuit handle is to be returned.

pPacketRet
sPacketMax — describe a buffer into which either the ACCEPT CALL packet is copied if the call is established or the CLEAR

REQUEST packet is copied if the call is rejected.

psPacketRet is the memory address of a word to which the size of the received ACCEPT CALL or CLEAR REQUEST packet is to be returned.

pCalled
sCalled describe a buffer containing the called number, in ASCII. (The number is converted to BCD and placed in the CALL REQUEST packet by the X.25 Network Gateway system service.)

pFacilities
sFacilities describe a buffer containing facilities data to be included in the CALL REQUEST packet.

pUserData
sUserData describe a buffer containing user data to be included in the CALL REQUEST packet. The buffer can contain up to 16 bytes for normal calls or 128 bytes for Fast Select calls. The first four bytes identify high-level protocols (for further information, see CCITT Recommendation X.25).

bGfi is the one-byte general format indicator. Bit 6 of this byte should be set if the D bit is to be used during the call (bit 7 is the most significant bit). (See the "Specifying and Setting M, D, and Q Bits" subsection above.)

sVchRet is always 2; ssPacketRet is always 2.

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	6
2	nReqPbCb	1	3
3	nRespPbCb	1	3
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	167
12	iNet	2	
14	bGfi	1	
15	reserved	3	
18	pFacilities	4	
22	sFacilities	2	
24	pUserData	4	
28	sUserData	2	
30	pCalled	4	
34	sCalled	2	
36	pVchRet	4	
40	sVchRet	2	2
42	pPacketRet	4	
46	sPacketMax	2	
48	psPacketRet	4	
52	ssPacketRet	2	2

NotifyNextIncomingCall

Description

The NotifyNextIncomingCall service notifies the X.25 Network Gateway system service that the calling process wishes to receive an incoming call of a certain protocol and within a certain port range. For NotifyNextIncomingCall to successfully complete, both the protocol and the port number of a received CALL REQUEST packet must match those specified in NotifyNextIncomingCall.

The X.25 Network Gateway system service queues NotifyNextIncomingCall requests. This queue can contain as many requests as there are virtual circuits per line. If more are queued, the excess requests are returned with a status code of 8502 ("Maximum number of this request has been queued").

The first byte of the user data field of the CALL REQUEST packet contains a protocol identifier. This protocol identifier is matched against the set of protocol identifiers specified in NotifyNextIncomingCall. This set is specified by mask/value pairs, where the mask is a byte that is logically ANDed with the received protocol identifier, and the value is compared with the result. If mask/value pairs are not specified (sRgClass of 0), any incoming protocol is accepted.

If no protocol is specified in the CALL REQUEST packet (that is, user data are not present), 0 is used as the value of the received protocol identifier.

The port number is the last two digits of the called address in CALL REQUEST packets. A port number is valid only if the number of digits of the incoming address is two digits longer than the network address specified at installation. This incoming port number must be within the specified range of high and low port numbers for NotifyNextIncomingCall to successfully complete and return the call to the user. If a port number is not present in the called address of the INCOMING CALL packet, Port 0 is assumed.

No packets are transmitted by NotifyNextIncoming-Call.

Procedural Interface

```
NotifyNextIncomingCall (iNet, pVchRet,
                        pPacketRet, sPacketMax,
                        psPacketRet, pRgClass,
                        sRgClass, lowPort,
                        highPort, timeOut):
                        ErcType
```

where

iNet — is the communications network identifier, for which the only value currently allowed is 0.

pVchRet — is the memory address of a word to which the virtual circuit handle is to be returned.

pPacketRet
sPacketMax — describe a buffer to which the received INCOMING CALL packet is copied.

psPacketRet — is the memory address of a word to which the size of the received INCOMING CALL packet is to be returned.

pRgClass
sRgClass — describe an array of protocol class identifiers; each identifier consists of a one-byte mask that is followed by a one-byte value. If sRgClass is 0, checking for protocols is disabled for this NotifyNextIncomingCall.

lowPort — is a word containing the lower bound on the range of port numbers (the last two digits of the called address) for which calls should be notified. The required format for the port number is a pair of ASCII digits, with the high-order digit in the high-order byte of the word.

highPort — is a word containing the upper bound on the range of port numbers for

which calls should be notified. The required format for the port number is a pair of ASCII digits, with the high-order digit in the high-order byte of the word.

timeOut — is a word containing the maximum amount of time, in units of 100 ms, to wait; however, the X.25 Network Gateway system service rounds the value up to the next second. A value of 0 indicates no waiting; NotifyNextIncomingCall returns with the status code 8507 ("User-specified time out") if no packets have been received from the PDN. A value of 0FFFFh indicates to wait indefinitely.

ssPacketRet is always 2; sVchRet is always 2.

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	8
2	nReqPbCb	1	1
3	nRespPbCb	1	3
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	165
12	iNet	2	
14	lowPort	2	
16	highPort	2	
18	timeOut	2	
20	pRgClass	4	
24	sRgClass	2	
26	pPacketRet	4	
30	sPacketMax	2	
32	psPacketRet	4	
36	ssPacketRet	2	2
38	pVchRet	4	
42	sVchRet	2	2

QueryX25Status

Description

The QueryX25Status service returns a set of statistics concerning activity over the PDN. Information can be provided for either a single logical channel or several logical channels.

Procedural Interface

```
QueryX25Status (iNet, pStatusBuffer,
                sStatusBuffer, lcnFirst,
                lcGnFirst, nLc, vch, pnLcRet):
                ErcType
```

where

iNet — is the communications network identifier, for which the only value currently allowed is 0.

pStatusBuffer
sStatusBuffer — describe a buffer to which the status information is to be returned. For further information, see the "Network Statistics Buffer" subsection below.

lcnFirst — is the logical channel number of the first of several virtual circuits for which circuit-specific statistics are to be returned. lcnFirst is ignored if vch is not 0.

lcGnFirst — is the logical channel group number of the first of several virtual circuits for which circuit-specific statistics are to be returned. lcGnFirst is ignored if vch is not 0.

nLc — is the number of logical channels for which circuit-specific statistics are to be returned. nLc is ignored if vch is not 0. If vch is 0 and nLc is 0FFFFh, information is given for all virtual circuits on the network specified by iNet. If sStatusBuffer is too small, the

information is truncated. If vch and nLc are both 0, logical channel statistics are not returned.

vch — is the virtual circuit handle of a single virtual circuit for which statistics are to be returned. If vch is not 0, lcnFirst, lcGnFirst, and nLc are ignored. If vch is 0, statistics for several logical channels are returned, based on the values in lcnFirst, lcGnFirst, and nLc.

pnLcRet — is the memory address of a byte to which the number of logical channels for which statistics are actually returned is to be returned.

Request Block

snLcRet is always 1.

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	8
2	nReqPbCb	1	0
3	nRespPbCb	1	2
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	174
12	iNet	2	
14	lcnFirst	1	
15	lcGnFirst	1	
16	nLc	2	
18	vch	2	
20	pStatusBuffer	4	
24	sStatusBuffer	2	
26	pnLcRet	4	
30	snLcRet	2	1

Network Statistics Buffer

The network statistics buffer is set up by the X.25 Network Gateway system service and is pointed to by pStatusBuffer and sStatusBuffer. This buffer is described in Table 4-3 below. The first 64 bits contain line information and are always returned. If the user requests logical channel status, the X.25 Network Gateway system service appends logical channel status blocks. One 32-bit logical channel status block is appended for each logical channel.

Table 4-3. Network Statistics Buffer

Offset	Field	Size (bytes)
0	iLine	2
2	nVcConf	2
4	nVcActive	2
6	nVcOutofOrder	2
8	fLinkLevelUp	1
9	fPacketLevelUp	1
10	nLinkResets	2
12	nDteRestarts	2
14	nDceRestarts	2
16	dataPacketsReceived	4
20	dataPacketsTransmitted	4
24	nIncomingCalls	2
26	nOutgoingCalls	2
28	timeLastChange	4
32	localNetAddress	7
39	cbLocalNetAddress	1
40	cTicks	4
44	cRxActiveTicks	4
48	cTxActiveTicks	4
52	reserved	12
64	rgLcStatus	nLc*32

where

iLine is the communications line (0 for Channel A or 1 for Channel B) in use.

nVcConf is the maximum number of configured virtual circuits.

nVcActive is the number of virtual circuits currently active.

nVcOutofOrder
: is the number of virtual circuits currently out of order.

fLinkLevelUp
: is TRUE if the X.25 Network Gateway link level protocol is operational.

fPacketLevelUp
: is TRUE if the X.25 Network Gateway packet level protocol is operational.

nLinkRestarts
: is the number of link level protocol restarts that have occurred since the connection of the physical line.

nDteRestarts
: is the number of packet level protocol restarts initiated by the X.25 Network Gateway system service since the last link level protocol restart.

nDceRestarts
: is the number of packet level protocol restarts initiated by the PDN since the last link level protocol restart.

dataPacketsReceived
: is the number of data packets received since the last packet level protocol restart.

dataPacketsTransmitted
: is the number of data packets transmitted since the last packet level protocol restart.

nIncomingCalls
: is the number of calls answered since the last packet level protocol restart.

nOutgoingCalls
: is the number of calls established since the last packet level protocol restart.

timeLastChange
: is the system date and time, in B20 format (32 bits), that was recorded the last time either the

link level changed state or the packet level became operational, whichever is most recent.

localNetAddress
: is the local network address that is inserted in all outgoing CALL REQUEST packets, in BCD.

cbLocalNetAddress
: is the number of digits in the local network address.

cTicks
: is the number of elapsed 100 ms timer ticks since the last link level protocol restart.

cRxActiveTicks
: is the number of elapsed timer ticks since the last link level protocol restart in which the receive side of the line was active.

cTxActiveTicks
: is the number of elapsed timer ticks since the last link level protocol restart in which the transmit side of the line was active.

rgLcStatus
: is an array of nLc logical channel status blocks. One block is written for each virtual circuit that is monitored. For further information, see Table 4-4 below.

Table 4-4. Logical Channel Status Blocks

Offset	Field	Size (bytes)
0	lcn	1
1	lcGn	1
2	fActive	1
3	fOutofOrder	1
4	dataPacketsReceived	2
6	dataPacketsTransmitted	2
8	nDteReset	2
10	nDceReset	2
12	nUser	2
14	remoteNetAddress	7
21	cbRemoteNetAddress	1
22	receiveWindowSize	1
23	sendWindowSize	1
24	receivePacketSize	2
26	sendPacketSize	2
28	lastCause	1
29	lastDiagnostic	1
30	callState	1
31	circuitType	1

where

lcn is the logical channel number.

lcGn is the logical channel group number.

fActive is TRUE if the virtual circuit is active.

fOutofOrder is TRUE if the virtual circuit is out of order.

dataPacketsReceived
: is the number of data packets received since this virtual circuit was established.

dataPacketsTransmitted
: is the number of data packets transmitted since this virtual circuit was established.

nDteReset
: is the number of flow control resets initiated by the X.25 Network Gateway system service over this virtual circuit since it was established.

nDceReset
: is the number of flow control resets initiated by the PDN over this virtual circuit since it was established.

nUser
: is the number of the user who last established a connection over this virtual circuit.

remoteNetAddress
: is the called (remote) network address.

cbRemoteNetAddress
: is the number of digits in the called (remote) network address.

receiveWindowSize
: is the maximum number of data packets the PDN will send over this virtual circuit without acknowledgment by the X.25 Network Gateway system service. This is a sliding window; that is, more than one unacknowledged packet can be outstanding. If the maximum number is reached, no more packets can be sent until some (or all) of the outstanding packets are acknowledged.

sendWindowSize
: is the maximum number of data packets the X.25 Network Gateway system service will send to the PDN over this virtual circuit before waiting for acknowledgment. This is a sliding window; that is, more than one unacknowledged packet can be

outstanding. If the maximum number is reached, no more packets can be sent until some (or all) of the outstanding packets are acknowledged.

receivePacketSize
: is the size of the largest data packet the PDN will send to this virtual circuit.

sendPacketSize
: is the size of the largest data packet the PDN will accept from this virtual circuit.

lastCause
: is the reason why this virtual circuit was previously cleared or reset. See Appendix B.

lastDiagnostic
: is the diagnostic from the previous time this virtual circuit was cleared or reset. See Appendix B.

callState
: is the internal state of the virtual circuit, as defined in Annex 2 of CCITT Recommendation X.25. It can take one of the values shown in Table 4-5 below.

Table 4-5. CallState Values

Value	Acronym	Meaning
0	P1	Ready
1	P2	DTE waiting
2	P3	DCE waiting
3	P6	DTE clear
4	P7	DCE clear
5	D1	Flow control ready
6	D2	DTE reset
7	D3	DCE reset
8	R2	DTE restart
9	R3	DCE restart

circuitType is the type of virtual circuit; it can take one of the following values:

0 = two-way virtual circuit

1 = permanent virtual circuit

2 = incoming-only virtual circuit

3 = outgoing-only virtual circuit

ReadX25Packet

Description

The ReadX25Packet service returns a DATA or INTERRUPT packet received by the X.25 Network Gateway system service from the PDN. The returned status code is 0 ("OK") for a DATA packet, or 8519 ("Interrupt data") for an INTERRUPT packet. In either case, only the data portion of the full X.25 packet is returned.

Only two ReadX25Packet requests can be outstanding for a virtual circuit at any given time. If more are queued, the excess requests are returned with a status code of 8502 ("Maximum number of this request has been queued").

No packets are transmitted by ReadX25Packet.

Procedural Interface

```
ReadX25Packet (vch, pPacketRet, sPacketMax,
               psPacketRet, pGfiRet, timeOut):
               ErcType
```

where

vch — is the virtual circuit handle.

pPacketRet
sPacketMax — describe a buffer to which the user data portion of the next DATA or INTERRUPT packet received over the specified virtual circuit is to be returned.

psPacketRet — is the memory address of a word to which the size of the received DATA or INTERRUPT packet is to be returned.

pGfiRet — is the memory address of a byte into which the general format indicator (see Table 4-1 above) is to be returned.

timeOut — is a word containing the maximum amount of time, in units of 100 ms, to wait; however, the system rounds the value up to the next second. A

value of 0 indicates no waiting; ReadX25Packet returns with the status code 8507 ("User-specified time out") if no packets have been received from the PDN. A value of 0FFFFh indicates to wait indefinitely.

Request Block

ssPacketRet is always 2; sGfiRet is always 1.

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	6
2	nReqPbCb	1	0
3	nRespPbCb	1	3
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	170
12	vch	2	
14	timeOut	2	
16	reserved	2	
18	pPacketRet	4	
22	sPacketMax	2	
24	psPacketRet	4	
28	ssPacketRet	2	2
30	pGfiRet	4	
34	sGfiRet	2	1

ResetX25Call

Description

The ResetX25Call service requests that a RESET REQUEST packet be transmitted in order to clear all outstanding read and write requests and to reinitialize the flow control procedures for the specified virtual circuit without clearing the call.

Only one ResetX25Call can be outstanding for a virtual circuit at any given time. If more are queued, the excess requests are returned with a status code of 8502 ("Maximum number of this request has been queued").

Procedural Interface

```
ResetX25Call (vch, bReason): ErcType
```

where

vch — is the virtual circuit handle.

bReason — is a one-byte diagnostic code to be included in the RESET REQUEST packet.

Request Block

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	4
2	nReqPbCb	1	0
3	nRespPbCb	1	0
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	173
12	vch	2	
14	bReason	1	
15	reserved	1	

WriteX25Interrupt

Description

The WriteX25Interrupt service transmits an INTERRUPT packet over the PDN, bypassing the normal flow control procedures associated with DATA packet transmission. Unlike the WriteX25-Packet operation, WriteX25Interrupt transfers only a single byte of interrupt data; DATA packet formatting is performed by the X.25 Network Gateway system service.

Only one WriteX25Interrupt can be outstanding for a virtual circuit at any time. If more are queued, the excess requests are returned with a status code of 8502 ("Maximum number of this request has been queued").

Procedural Interface

```
WriteX25Interrupt (vch, bInterruptData): ErcType
```

where

vch is the virtual circuit handle.

bInterruptData
is the byte of interrupt data to be transmitted over the specified virtual circuit.

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	4
2	nReqPbCb	1	0
3	nRespPbCb	1	0
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	172
12	vch	2	
14	bInterruptData	1	
15	reserved	1	

WriteX25Packet

Description

The WriteX25Packet service causes transmission of a DATA packet over the PDN. As with ReadX25-Packet, the packet buffer should contain only user data. The three- or four-octet packet header is built by the X.25 Network Gateway system service, and contains the qualifier, delivery confirmation, and the data bits specified in the general format indicator (see Table 4-1 above).

Only five WriteX25Packet requests can be outstanding for a virtual circuit at any given time. If more are queued, the excess requests are returned with a status code of 8502 ("Maximum number of this request has been queued").

Procedural Interface

```
WriteX25Packet (vch, bGfi, pPacket, sPacket):
                ErcType
```

where

vch — is the virtual circuit handle.

bGfi — is the one-byte general format indicator in which M, D, and Q bits are specified. For further information on these bits, see the "Specifying and Setting M, D, and Q Bits" subsection above.

pPacket
sPacket — describe a buffer containing the data to be transmitted over the specified virtual circuit. sPacket should be less than or equal to the Maximum Packet Size parameter used when installing X.25 Network Gateway.

Offset	Field	Size (bytes)	Contents
0	sCntInfo	2	6
2	nReqPbCb	1	1
3	nRespPbCb	1	0
4	userNum	2	
6	exchResp	2	
8	ercRet	2	
10	rqCode	2	171
12	vch	2	
14	bGfi	1	
15	reserved	3	
18	pPacket	4	
22	sPacket	2	

NETWORK INTERACTION

Facilities

The X.25 Network Gateway supports, with a few exceptions, both the CCITT facilities (options) described in CCITT Recommendation X.25 and the non-CCITT facilities of a particular PDN as long as the facilities of that PDN conform to the facilities formats defined in CCITT Recommendation X.25.

In general, it is the user's responsibility (1) to coordinate the facilities used with the appropriate PDN administration, (2) to ensure that the PDN to which the X.25 Network Gateway is connected supports the desired facilities, and (3) to ensure that the specification of a facility agrees with the parameters expected by the PDN. The X.25 Network Gateway checks outgoing per-call facilities to ensure that their length does not exceed 63 bytes (CCITT specification); the facilities data are truncated if this length is exceeded.

CCITT facilities fall into three general categories:

- o those transparent to the X.25 Network Gateway,
- o those nontransparent to the X.25 Network Gateway (that is, requiring special processing), and
- o those not supported by the X.25 Network Gateway.

Transparent Facilities

Most CCITT facilities can be used without affecting (are transparent to) the X.25 Network Gateway. Except for those facilities listed below in the "Nontransparent Facilities" and "Nonsupported Facilities" subsections, the X.25 Network Gateway operates without regard for or knowledge of the facilities environment in which it is used or of the facilities data contained in the packets.

If the X.25 Network Gateway system service is communicating with a PDN that supports non-CCITT facilities, these facilities are handled transparently if they conform to the CCITT facilities standards.

Nontransparent Facilities

Nontransparent facilities consist of installation facilities and per-call facilities.

Installation Facilities The following facilities must be specified at the X.25 Network Gateway installation:

- nonstandard default window size (other than 2),
- nonstandard default packet size (other than 128), and
- logical channel ranges for two-way, incoming and/or outgoing virtual circuits.

For further information, see the Install X.25 command parameters in the "Installation Procedures: Install X.25 Command" section above.

Per-call Facilities Some per-call facilities affect the X.25 Network Gateway operation; therefore, it must recognize when these facilities are used. To do this, the X.25 Network Gateway examines the facility fields of received call-establishment packets (ACCEPT CALL and CALL REQUEST) for these per-call facilities. The X.25 Network Gateway modifies its operation appropriately if these per-call facilities are found.

The facility fields are examined using an algorithm based on CCITT Recommendation X.25. This algorithm is independent of all facilities other than the ones being searched for; however, the coding of the facility fields must match CCITT Recommendation X.25. If these fields are coded in other formats, errors can occur.

Facilities data are transferred between the calling process and the X.25 Network Gateway system

service using three operations: AcceptX25Call, InitiateX25Call, and NotifyNextIncomingCall.

The AcceptX25Call and InitiateX25Call operations enable data to be included in the facilities field of transmitted ACCEPT CALL and CALL REQUEST packets, respectively. The InitiateX25Call and NotifyNextIncomingCall operations return the ACCEPT CALL and CALL REQUEST packets, respectively, which contain facilities fields transmitted by the remote DTE.

Facilities data are scanned for three per-call facilities:

- Fast Select,
- window size negotiation, and
- packet size negotiation.

Fast Select Facility The X.25 Network Gateway examines the facility fields of both incoming and outgoing CALL REQUEST packets for the Fast Select facility. A Fast Select facility is a Type A facility with a facility code of 1 and a parameter field with its high-order bits set. No special actions are taken for a restricted Fast Select facility beyond those taken for a nonrestricted Fast Select facility. If the Fast Select facility is used, the X.25 Network Gateway generates and accepts call establishment and call clearing packets according to the Fast Select packet formats during call establishment. Once call establishment is complete, the X.25 Network Gateway reverts to normal packet format.

Window and Packet Size Negotiation Facilities
The X.25 Network Gateway examines the facilities fields of both received and transmitted CALL ACCEPT packets for the window size and packet size negotiation facilities. If either of these facilities is used, the X.25 Network Gateway modifies its parameters for window and/or packet size for the duration of the call.

The window size negotiation facility is a Type B facility with a facility code of 43h. The elements of the parameter field are assumed to contain appropriate values for window sizes; therefore, these values are not validated.

The packet size negotiation facility is a Type B facility with a facility code of 42h. The elements of the parameter field are assumed to contain appropriate values for packet sizes log base 2; therefore, these values are not validated.

For further information on these facilities, see Section 7 of CCITT Recommendation X.25.

Nonsupported Facilities

The following facilities are not supported by the X.25 Network Gateway:

- datagram service and
- packet retransmission.

Multiplexing

When a packet is transmitted to the link level, the number of the virtual circuit on which it is transmitted is saved. The next time the link level is ready to accept a packet for transmission, virtual circuit 0 is examined for a restart-related packet requiring transmission. If no restart-related packet is pending, all other active virtual circuits are examined in circuit number order, starting at the virtual circuit following the last virtual circuit to transmit. The first virtual circuit found with a pending requirement to transmit a packet is allowed to transmit. Thus, no particular virtual circuit can dominate transmission.

Flow Control

When a data packet is received on a virtual circuit, a flow-control packet (RR or RNR; see Section 7 of CCITT Recommendation X.25) acknowledging that data packet is made pending for transmission. If additional data packets are received before the flow-control packet can be issued, a single flow-control packet acknowledging multiple data packets is made pending for transmission. If the type of flow-control packet to be sent changes (from RR to RNR, or vice versa) once it has been made pending, the type is altered accordingly. Thus, a flow-control packet is sent to acknowledge a data packet as soon as

transmission is possible. With this scheme, a minimum number of flow-control packets is transmitted.

Error Handling

Call-progress signals as indicated in received RESTART REQUEST, CLEAR REQUEST, RESET REQUEST, and DIAGNOSTIC packets are stored for each virtual circuit as they are received. The last call-progress signals received are accessible for each virtual circuit with the QueryX25Status operation. A process can specify the value of the diagnostic code field when it requests that CLEAR and RESET REQUEST packets be generated. For CLEAR and RESET REQUEST packets generated by the X.25 Network Gateway in response to errors, see Appendix B for the values inserted in the diagnostic code field.

Error conditions for requests to the X.25 packet occur in three circumstances:

- the request contains invalid parameters or is invalid in the context of prior requests,
- the request is superseded by a later request from the same application system, or
- the state of a virtual circuit or of the X.25 Network Gateway system service as a whole makes it impossible to fulfill the request.

Requests are checked for the validity of their parameters and the propriety of their occurrence. If an error condition is detected, the request is immediately returned with an appropriate status code. If a valid and proper request is received, the X.25 Network Gateway system service attempts to fulfill the request. In most cases, a request cannot be fulfilled immediately and must be held by the X.25 Network Gateway system service for some period. If a request awaiting fulfillment is canceled by a later action from the application system or the X.25 Network Gateway system service, the request is canceled and an appropriate status code is returned. Generally, it cannot be assumed that requests are returned in the sequence of their issuance in either error or normal situations. However, requests for reads and writes are returned in the sequence of their issuance if

errors occur while they are being held by the X.25 Network Gateway system service.

SECTION 5
SEQUENTIAL ACCESS METHOD

OVERVIEW

The X.25 Sequential Access Method (X.25 SAM) provides device-independent input/output through the Burroughs Sequential Access Method. Because this level of access can be used without detailed knowledge of CCITT Recommendation X.25, it is appropriate for the application programmer.

X.25 SAM uses byte streams (for further information on byte streams, see the "Sequential Access Method" section of the B20 Operating System Manual). Each X.25 byte stream corresponds to a virtual circuit that is established when the byte stream is opened and cleared when the byte stream is closed. Thus, X.25 SAM provides the user with the tools needed to send data to other computer systems over an X.25 PDN without requiring the sophisticated techniques of the packet access method.

Device-dependent information for an X.25 byte stream is provided by a configuration file that controls the call establishment parameters for the virtual circuit corresponding to the byte stream. Configuration files can be created or modified with the Configure X.25 command.

X.25 SAM performs packet assembly and disassembly in a manner compatible with CCITT Recommendations X.3 and X.29.

X.25 SAM consists of object module procedures contained in the library X25SAM.LIB. Programs using X.25 SAM must include this library at link time and must include X.25 bytes streams in their SamGen module.

DEVICE FILE SPECIFICATION

The device file specification supplied to the OpenByteStream operation to open an X.25 byte stream is as follows:

```
[X25]n&{node}[volname]<dirname>filename
```

where

n is a communication channel/network specifier. It is recommended that this field should be either blank or 0.

&{node}[volname]<dirname>filename describes an optional configuration file containing the operational characteristics. A default configuration file is used if none is specified. The default file is [Sys]<Sys>X25Config.Sys.

CONFIGURATION FILES

A configuration file is generated with the Configure X.25 command. Configure X.25 takes the file specification for a configuration file as a parameter, then displays a form that prompts for configuration file parameters.

Command: Configure X.25

```
Configure X.25
  Configuration file name _____________
```

X.25 Configuration File Parameters

The following form is displayed after the user enters the configuration file name in Configure X.25 command.

```
X.25 Parameters
  [Disable transmit buffering?]                                    _____
  [Initiate or accept call (default = accept)]                     _____
  [Local or end-to-end acknowledgement (default = local)]          _____
  [Read timeout (default = forever)]                               _____
  [Called address]                                                 _____
  [Reverse charging?]                                              _____
  [Call data]                                                      _____
  [Low port (default = 0)]                                         _____
  [High port (default = 99)]                                       _____
  [Notify timeout (default = forever)]                             _____
```

where

[Disable transmit buffering?]
is either Yes or No (the default). Entering Yes disables internal data buffering from WriteByte and WriteBsRecord operations.

[Initiate or accept call (default = accept)]
specifies if the call is to be initiated by the X.25 Network Gateway or by the remote user.

For initiate, an OpenByteStream operation causes an InitiateX25Call operation to be issued with the following attributes specified in the configuration file:

1. called address,
2. call data, and
3. reverse charging.

Call data and reverse charging attributes are optional.

For accept, an OpenByteStream operation causes a NotifyNextIncomingCall operation with the following values specified in the configuration file:

1. low port,
2. high port, and
3. time out.

If an incoming call received within the timeout interval matches the specified port values, an AcceptX25Call operation is issued to complete the establishment of the call.

[Local or end-to-end acknowledgement (default = local)]
is either local or end-to-end. Entering end-to-end specifies the delivery confirmation (D-bit) feature. This feature allows the user to specify the value of the D bit in X.25 DATA packets transmitted by an X.25 byte stream. The D bit is set if end-to-end acknowledgement is specified and reset if local acknowledgement is specified.

[Read timeout (default = forever)]
is the timeout interval, in seconds, for read operations; the maximum value is 6500. A value of 0 indicates data are to be returned only if buffered data are available at the byte stream or at the X.25 Network Gateway. Any value exceeding 6500 indicates to wait forever.

[Called address]
is the network address to be called and is used only if initiate is specified in the Accept Call field.

[Reverse charging?]
is either Yes or No (the default). Entering Yes specifies a "collect call" is made when a call is initiated (initiate is specified in the Accept Call field).

[Call data] can contain up to 12 characters of ASCII data to be transmitted to the remote user during call establishment, and is used only if initiate is specified in the Accept Call field.

[Low port (default = 0)]
indicates the lower bound of the port range for calls to be accepted and is used only if accept is specified in the Accept Call field.

[High port (default = 99)
indicates the upper bound of the port range for calls to be accepted, and is used only if accept is specified in the Accept Call field.

[Notify timeout (default = forever)]
contains the timeout interval, in seconds, for the OpenByteStream operation; the maximum value is 6500. A value of 0 indicates data are to be returned only if buffered data are available at the byte stream or at the X.25 Network Gateway. Any value exceeding 6500 indicates to wait forever. The time out is used only if accept is specified in the Accept Call field.

DEVICE-INDEPENDENT OPERATIONS

The following device-independent sequential access method operations are supported:

CheckPointBs
CloseByteStream
OpenByteStream
PutBackByte
ReadBsRecord
ReadByte
ReadBytes
ReleaseByteStream
WriteBsRecord
WriteByte

For further information on these operations, see the "Sequential Access Method" section in the B20 Operating System Manual.

Device-dependent operations are not required.

X.25 SAM can be opened for read, write or modify operations. For compatibility, the text and append modes are allowed in the OpenByteStream operation. The text mode is treated as read mode, and the append mode is treated as write mode.

Multiple X.25 byte streams can be concurrently opened by an individual user. Each open byte stream transfers data over a single X.25 virtual circuit.

The X.25 protocol allows data to be transferred during the establishment of a virtual circuit. If initiate is specified in the configuration file, data can be transferred during call establishment. If accept is specified in the configuration file, any data received during call establishment are placed in the read buffer. These data are accessible with standard read operations once the byte stream is opened.

NETWORD INTERACTION

Packet Assembly/Disassembly Facility (PAD)

The X.25 byte stream transfers and receives data from the X.25 Network Gateway in packets of variable size. Packets are buffered within the byte stream and are assembled/disassembled as user output/input is performed by the X.25 SAM write/read operations. Separate write and read buffers are provided.

Outgoing data are stored in the write buffer until the transmit packet size for the X.25 Network Gateway is reached, at which time the actual transfer is performed. The configuration file can be used to disable this write buffering feature; data are then transferred for each X.25 SAM write operation.

Each packet of incoming data is stored in the read buffer until all data in the packet are read, at which time a new incoming packet is solicited from the X.25 Network Gateway. The write buffer is flushed before every read request to the X.25 Network Gateway to avoid a deadlock condition that causes the user to wait for a response to outgoing data being held in the buffer.

X.29 Procedures

CCITT Recommendation X.29 defines procedures for exchanging information between a PAD and a packet-mode data terminal equipment (DTE) over a public data network. X.25 SAM incorporates elements of these procedures into its operation to support interaction (1) among users of Burroughs Corporation equipment and (2) between users of Burroughs Corporation equipment and users of other equipment.

Byte streams opened by X.25 SAM support CCITT Recommendation X.29 procedures. The provided X.29 support depends on the direction of call establishment. This asymmetric operation is required by the nature of CCITT Recommendation X.29, which only allows calls to be established in one direction. That is, a PAD can establish calls to a packet-mode DTE, but a packet-mode DTE cannot establish calls to a PAD. However, once the call is established, data can be transferred in both directions.

For calls accepted by X.25 SAM, the workstation functions as a packet-mode DTE communicating with a PAD according to CCITT Recommendation X.29. This allows data to be exchanged with an asynchronous DTE over a public data network.

For calls initiated by X.25 SAM, the workstation functions as a PAD interfacing an asynchronous DTE to a public data network. This allows the use of commercial time-sharing facilities that communicate with asynchronous DTEs over public data networks using PADs. In this case, the user workstation is still a packet-mode DTE using the X.25 Network Gateway to exchange data according to CCITT Recommendation X.25 packet switching protocol. Actual public data network PADs are not required. Rather, the packet-mode capability is enhanced by allowing the user's workstation to function in place of both an asynchronous DTE and a PAD while maintaining the benefits of a packet-mode DTE.

Supported Procedures

X.25 SAM supports most CCITT Recommendation X.29 procedures for communicating control and user information between a PAD and a X.25 packet-mode device; this is transparent to the user. X.29 support specifications are published in CCITT Recommendation X.29.

X.25 SAM's support of CCITT Recommendation X.29 procedures allows a workstation-to-workstation communication over a PDN through X.25 SAM. Support of X.25 SAM initiating a call is compatible with that of X.25 SAM accepting a call.

X.25 SAM can be used to communicate with pure packet-mode DTEs if the DTE initiates and/or accepts calls with the CCITT Recommendation X.29 protocol identifier in the Call User Data field of the CALL REQUEST packet. Only CCITT Recommendation X.29 calls are initiated/accepted by X.25 SAM. Because X.29 support of X.25 SAM is essentially passive, the only action required during the call is that the remote DTE must use the X.29 protocol identifier during call establishment.

Operation

CCITT Recommendation X.29 information is transmitted within X.25 data packets. This information provides special error recovery as well as call termination and parameter setting procedures for PAD operation. CCITT Recommendation X.29 messages are automatically intercepted and generated by X.25 SAM independently of the operations that package user data.

In support of CCITT Recommendation X.29, X.25 SAM transmits and receives data packets, regardless of the mode (read only or write only) for which the byte stream is opened. Thus, the open mode of an X.25 byte stream affects only user access to directions of data transfer. Bidirectional data transfer occurs regardless of the mode.

Packet-Mode DTE

When X.25 SAM accepts a call (the workstation functions as a packet-mode DTE), X.25 SAM supports standard CCITT Recommendation X.29 control procedures for PAD functions. This support can be termed passive because X.25 SAM does not set PAD parameters to force any particular mode of operation of the PAD or of the asynchronous terminal. X.25 SAM recognizes and issues PAD messages when necessary to ensure correct data transfer with the PAD, but in no other case.

Actions taken on receipt of PAD messages (Q bit set) are shown in Table 5-1.

Table 5-1. Actions Taken by X.25 SAM (as a Packet-Mode DTE) on Receipt of X.29 Messages

Messages	Action
Illegal message	Send X.29 error message
Error message	Clear the call
Indication of break	Send a set-parameters message to reset parameter 8 for normal delivery of data to asynchronous DTE
Any other message	Ignored.

An invitation-to-clear message is transmitted if the ReleaseByteStream or CloseByteStream operation is called. This message asks the remote user to clear the call once the remote user has received all transmitted data. The byte stream itself does not clear the call. However, termination of user's program results in the B20 Operating System clearing the call. Thus, loss of data can result if the user's program terminates before the PAD has responded to the invitation-to-clear message by clearing the call.

Asychronous DTE/PAD

When X.25 SAM initiates a call (the workstation functions as an asynchronous DTE interfaced to a PAD), X.25 SAM supports standard CCITT Recommendation X.29 control procedures for PAD functions, but it does not act on the PAD parameters set by the packet-mode DTE. That is, while the packet-mode DTE can set and read back values of Recommendation X.3 PAD parameters, the X.25 SAM operation is not modified by the values of these parameters.

Actions taken on receipt of PAD messages (Q bit set) are shown in Table 5-2.

Parameter values are maintained for the 12 X.3 parameters (see CCITT Recommendation X.3). Any other parameters included in a parameter field are considered an error, and an X.29 error message is issued. The call is cleared if the ReleaseByteStream or CloseByteStream operation is called.

Limitations

The following are not supported by X.25 SAM:

* transmission of indication-of-break message,
* local specification or interpretation of X.29 parameters,
* asynchronous requests to X.25,
* M bit,
* interrupt packets,
* facilities other than reverse charging,
* permanent virtual circuits, and
* nonstandard default packet sizes.

Table 5-2. Actions Taken by X.25 SAM (as an Asynchronous DTE/PAD) on Receipt of X.29 Messages

Message	Action
Illegal message	Send X.29 error message
Error message	Clear the call
Indication of break	Ignored
Invitation to clear	Clear the call
Read parameters	If a parameter field is present, send parameters indication with values of specific parameters If no parameter field is present, send the values of all parameters
Set and read parameters	If a parameter field is present, store new values of specific parameters and send parameters indication with their values If no parameter field is present, reset all parameters to 0 and send parameters indication with values of all parameters

Calls to the Packet Level

The following operations are handled automatically by the X.25 SAM:

AcceptX25Call
ClearX25Call
InitiateX25Call
NotifyNextIncomingCall
ReadX25Packet
WriteX25Packet

* AcceptX25Call

  Issued: When the OpenByteStream operation is requested, the configuration file indicates accept call, and the NotifyNextIncomingCall operation completes without error.

  Parameters:

  Vch = internal variable

  pFacilities, sFacilities = 0

  bGfi = user-specified value of D bit from the configuration file; Q and M bits = 0

  pUserData, sUserData = 0

* ClearX25Call

  Issued:
  1. When the ReleaseByteStream or CloseByteStream operation is called and the call is locally initiated.
  2. When an unrecoverable X.25 error occurs.
  3. When an invitation-to-clear message is received and the call is remotely initiated.

  Parameters:

  Vch = internal variable

  bReason = 17

  pUserData, sUserData = 0

* InitiateX25Call

  Issued: When the OpenByteStream operation is called and the configuration file indicates initiate call.

  Parameters:

  iNet = 0

  pVchRet = internal variable

  pPacketRet, sPacketRet = internal buffer

  psPacketRet = internal variable

  pCalled, sCalled = the user-specified value from the configuration file

  pFacilities, sFacilities = either 0 or the memory address of a string containing the code for the reverse charging facility

  pUserData, sUserData = X.29 protocol plus the user-specified values from the configuration file

  bGfi = the user-specified value of the D bit from the configuration file; Q and M bits = 0

* NotifyNextIncomingCall

  Issued: When the OpenByteStream operation is called and the configuration file indicates accept call.

Parameters:

iNet = 0

pVchRet = internal value

pPacketRet, sPacketRet = internal buffer

psPacketRet = internal variable

pRgClass, sRgClass = X.29 protocol mask-value pair

lowport, highport = the user-specified values from the configuration file

timeOut = the user-specified value from the configuration file

* ReadX25Packet

Issued: When a user read or write request read is made and the read buffer is empty.

Parameters:

vch = internal variable

pPacketRet, sPacketRet = internal buffer

psPacketRet = internal variable

pGfiRet = internal variable

timeOut - is either the user-specified value from the configuration file for user read operations or is a or 0 following user write operations.

* WriteX25Packet

Issued: 1. When a user write request is made

2. When an X.29 message is sent

Parameters:

vch = internal variable

bGfi = the user-specified value of the D bit from the configuration file; M bit = 0; Q bit = 1 for X.29 messages, 0 for user data

pPacket, sPacket = internal variables

The ConnectX25Permanent, ResetX25Call, and WriteX25Interrupt operations are never issued by the X.25 SAM.

Error Handling

X.29 error handling is transparent to the user.

The X.29 Network Gateway performs the following error handling procedures:

* If X.25 SAM receives a message violating CCITT Recommendation X.29 procedures, it sends an X.29 error message, but does not clear the call. If X.25 SAM sends a message violating CCITT Recommendation X.29 procedures, X.25 SAM will receive an X.29 error message; X.25 SAM will then clear the call.

* If an X.25 error occurs during a SAM operation, the call is cleared, but the byte stream is not closed. A ReleaseByteStream or CloseByteStream operation must be done to close the byte stream.

* If a user-specified time out occurs, the operation in question is terminated, but the call is not cleared.

SECTION 6

MULTIMODE TERMINAL PROGRAM X.25 COMMUNICATIONS

OVERVIEW

The Multimode Terminal Program (MTP) enables a user who is unsophisticated in communications to operate over PDNs.

All MTP functions and capabilities are documented in the Multimode Terminal Program Reference Manual and the Multimode Terminal Program User's Guide. This section describes only the details of the interaction between MTP X.25 communications and the PDNs and is written for the programmer who is knowledgeable about communications.

NETWORK INTERACTION

Packet Assembly/Disassembly Facility

MTP transfers and receives data from the X.25 Network Gateway in packets of variable size. Packets are buffered within MTP and are assembled/disassembled as user output/input is performed. Separate write and read buffers are provided.

Data transfer is handled by an individual process that transmits DATA packets to the X.25 Network Gateway. As data are received for transmission, the process constructs a packet to be sent to the X.25 Network Gateway. When the request is completed, another packet is constructed for transmission to the X.25 Network Gateway. If the X.25 Network Gateway falls behind, the transmission process sends a second DATA packet. Two DATA packets are maintained until either transmission is complete or the X.25 Network Gateway catches up.

Each incoming DATA packet is stored in the read buffer until all data in the packet are read by MTP at which time a new incoming DATA packet is solicited from the X.25 Network Gateway.

X.29 Procedures

CCITT Recommendation X.29 defines procedures for exchanging information between a PAD and a packet-mode DTE over a public data network. The MTP incorporates elements of these procedures into its operation to support interaction (1) between users of B20 equipment and (2) among users of B20 equipment and users of other equipment.

MTP supports CCITT Recommendation X.29 procedures. The provided X.29 support depends on the direction of call establishment. This asymmetric operation is required by the nature of CCITT Recommendation X.29, which only allows calls to be established in one direction. That is, a PAD can establish calls to a packet-mode DTE, but a packet-mode DTE cannot establish calls to a PAD.

For calls accepted by MTP, the workstation functions as a packet-mode DTE communicating with a PAD according to CCITT Recommendation X.29. This allows the X.25 communications option to exchange data with an asynchronous DTE over a public data network.

For calls initiated by MTP, the workstation functions as a PAD interfacing an asynchronous DTE to a public data network. This allows the use of commercial time-sharing facilities that communicate with asynchronous DTEs over public data networks with PADs. In this case, the workstation still functions as a packet-mode DTE using the X.25 Network Gateway to exchange data according to CCITT Recommendation X.25 packet switching protocol. Actual public data network PADs are not required. Rather, the packet-mode capability is enhanced by allowing the user's workstation to function in place of both an asynchronous DTE and a PAD while maintaining the benefits of a packet-mode DTE.

Supported Procedures

The Multimode Terminal Program supports most CCITT Recommendation X.29 procedures for communicating control and user information between a PAD and a packet-mode DTE; this is transparent to the user. X.29 support specifications are published in CCITT Recommendation X.29.

MTP's support of CCITT Recommendation X.29 procedures allows B 20-to-B 20 communication over a PDN through MTP. Support of the MTP initiating a call is compatible with that of the MTP accepting a call.

MTP can be used to communicate with pure packet-mode DTEs if the DTE initiates and/or accepts calls with the X.29 protocol identifier in the Call User Data field of the CALL REQUEST packet. Only CCITT Recommendation X.29 calls are initiated/accepted by MTP. Because support of X.25 communications is essentially passive, the only X.29 action required during the call is the use of the X.29 protocol identifier during call establishment.

MTP is affected by two X.3 parameters: echo and break. These parameters affect both the echo (full-duplex vs. half-duplex) in conversational transmission, and the action taken when the BREAK key is pressed.

Operation

X.29 information is transmitted within X.25 DATA packets. Special error recovery as well as call termination and parameter setting procedures for PAD operation are supported. X.29 messages are automatically intercepted and generated by MTP independently of the operations that package user data.

Packet-Mode DTE

When MTP accepts a call (the B 20 functions as a packet-mode DTE), it supports standard CCITT Recommendation X.29 control procedures for PAD functions. This support can be termed passive because MTP does not set PAD parameters to force any particular mode of the operation of the PAD

or of the asynchronous DTE. MTP recognized and issues PAD messages when necessary to ensure correct data transfer with the PAD, but in no other case.

Actions taken on receipt of PAD messages (Q bit set) are shown in Table 6-1 below.

Table 6-1. Actions Taken by the MTP (as a Packet-Mode DTE) on Receipt of X.29 Messages

Message	Action
Illegal message	Send X.29 error message, then clear the call.
Error message	Clear the call.
Indication of break	Set-parameters message; reset parameter 8 for normal delivery of data to asynchronous DTE.
Any other message	Ignored.

An invitation-to-clear message is transmitted when the connection is terminated. This message asks the remote user to clear the call once the remote user has received all transmitted data. MTP itself does not clear the call. However, termination of MTP results in the BTOS Operating System clearing the call. Thus, loss of data can result if MTP terminates before the PAD has responded to the invitation-to-clear message by clearing the call.

Asynchronous DTE/PAD

When MTP initiates a call (the B 20 functions as an asynchronous DTE interfaced to a PAD), the MTP supports standard CCITT Recommendation X.29 control procedures for PAD functions, but it acts only on the echo and break PAD parameters set by the packet-mode DTE. That is, while the

packet-mode DTE can both set and read back values of CCITT Recommendation X.3 PAD parameters, MTP X.25 communications operation is modified only by the values of the echo and break parameters.

Actions taken on receipt of PAD messages (Q bit set) are shown in Table 6-2 below.

Parameter values are maintained for the 12 X.3 parameters (see CCITT Recommendation X.3). Any other parameters included in a parameter field are flagged as an error and the call is cleared.

Limitations

The following are not supported by MTP X.25 communications:

- user specification or interpretation of X.29 parameters, except echo and break options,
- D bit,
- M bit,
- facilities,
- permanent virtual circuits, and
- nonstandard default packet sizes.

Calls to the Packet Level

The following operations are issued automatically by MTP X.25 communications: AcceptX25Call, ClearX25Call, InitiateX25Call, NotifyNextIncomingCall, ReadX25Packet, WriteX25Interrupt, and WriteX25Packet.

- AcceptX25Call

  Issued: When the CODE-f2, then GO, keys are pressed and the NotifyNextIncoming-Call operation completes without an error.

Parameters:

Vch = internal variable

pFacilities, sFacilities = 0

bGfi = 0 (all bits are off)

pUserData, sUserData = 0

Table 6-2. Actions Taken by MTP (as an Asynchronous DTE/PAD) on Receipt of X.29 Messages

Message	Action
Illegal message	Send X.29 error message, then clear the call.
Error message	Clear the call.
Indication of break	Ignored.
Invitation to clear	Clear the call.
Read parameters	If a parameter field is present, send parameters indication with values of specific parameters. If no parameter field is present, send the values of all parameters.
Set and read parameters	If a parameter field is present, store new values of specific parameters and send parameters indication with their values. If no parameter field is present, reset all parameters to 0 and send parameters indication with values of all parameters.

- ClearX25Call

  Issued:
  a. When the user initiates (CODE-f1) or accepts (CODE-f3) another call.
  b. When an X.29 error occurs.

  Parameters:

  vch = internal variable

  bReason = 0

- InitiateX25Call

  Issued: When the Initiate Call command (CODE-f1, then GO) is executed.

  Parameters:

  iNet = 0

  pVchRet = internal variable

  pPacketRet, sPacketMax = internal buffer

  psPacketRet = internal variable

  pCalled, sCalled = user-specified values from the MTP sign-on form

  pFacilities, sFacilities = not supported

  pUserData, sUserData = X.29 protocol plus user-specified value from the MTP sign-on form

  bGfi = 0 (all bits are off)

- NotifyNextIncomingCall

  Issued: When the Accept Call command (CODE-f2, then GO) is executed.

Parameters:

iNet = 0

pVchRet = internal value

pPacketRet, sPacketMax = internal buffer

psPacketRet = internal variable

lowPort, highPort = user-specified value from the MTP sign-on form

timeOut = 0FFFFh (wait indefinitely)

- ReadX25Packet

  Issued: Reads are done synchronously. There is always one read outstanding unless MTP's input buffer is full.

  Parameters:

  Vch = internal variable

  pPacketRet, sPacketMax = internal buffer

  psPacketRet = internal variable

  pGfiRet = internal variable

  timeOut = 0FFFFh (wait indefinitely)

- WriteX25Interrupt

  Issued: When BREAK is pressed and the break options specify an interrupt packet is to be sent.

  Parameters:

  Vch = internal variable

  bInterruptData = 0

- WriteX25Packet

  Issued:
  a. When characters are transmitted to the host computer.
  b. When an X.29 message is sent.

  Parameters:

  Vch = internal variable

  bGfi = D bit = 0; M bit = 0; Q bit = 1 for X.29 messages, 0 for user data

  pPacketRet, sPacketMax = internal variables

The ConnectX25Permanent and ResetX25Call operations are never issued by MTP X.25 communications option.

Error Handling

X.29 errors cause the call to be cleared.

SECTION 7
X.25 STATUS PROGRAM

The X.25 status program displays information regarding the state of the X.25 Network Gateway system service. The command is

X.25 Status

VIDEO DISPLAY

When the X.25 status program is operational, it displays status information on the video display in one of three formats (see Figures 7-1 through 7-3 below). The information is updated at one-second intervals.

The following items describe the parts of the video display. They are keyed to the circled numbers in Figure 7-1.

General Information

1. Current date and time.

2. X.25 LINE MONITOR

   signifies the X.25 status program is active.

Line Status Data

Items 3 through 21 indicate the status of the line as a whole. The significance of these data depends on the X.25 Network Gateway system service state (see Item 7). If the state is operational, the other line status data reflect the current state of the line. If the state is connected or disconnected, the other line status data reflect the state of the line when it was last operational. These data, with the exception of the restarts (Items 10 and 13), are reset each time the X.25 Network Gateway system service state goes from connected to operational.

3. CHANNEL

   is the communications Channel (A or B) of the SIO communications controller for the X.25 Network Gateway system service being monitored. This value is entered as a

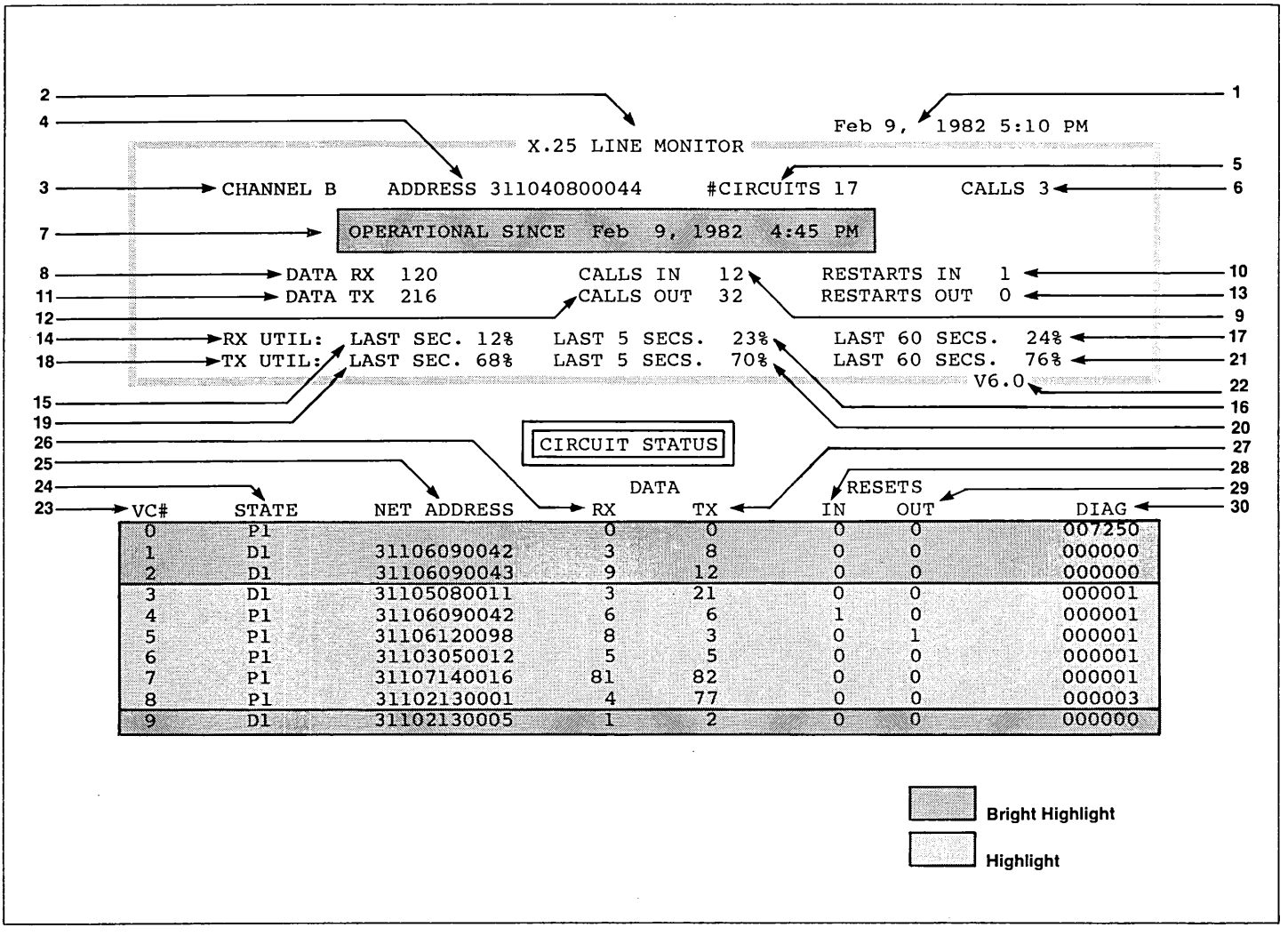


Figure 7-1. Operational Status of the X.25 Network Gateway.

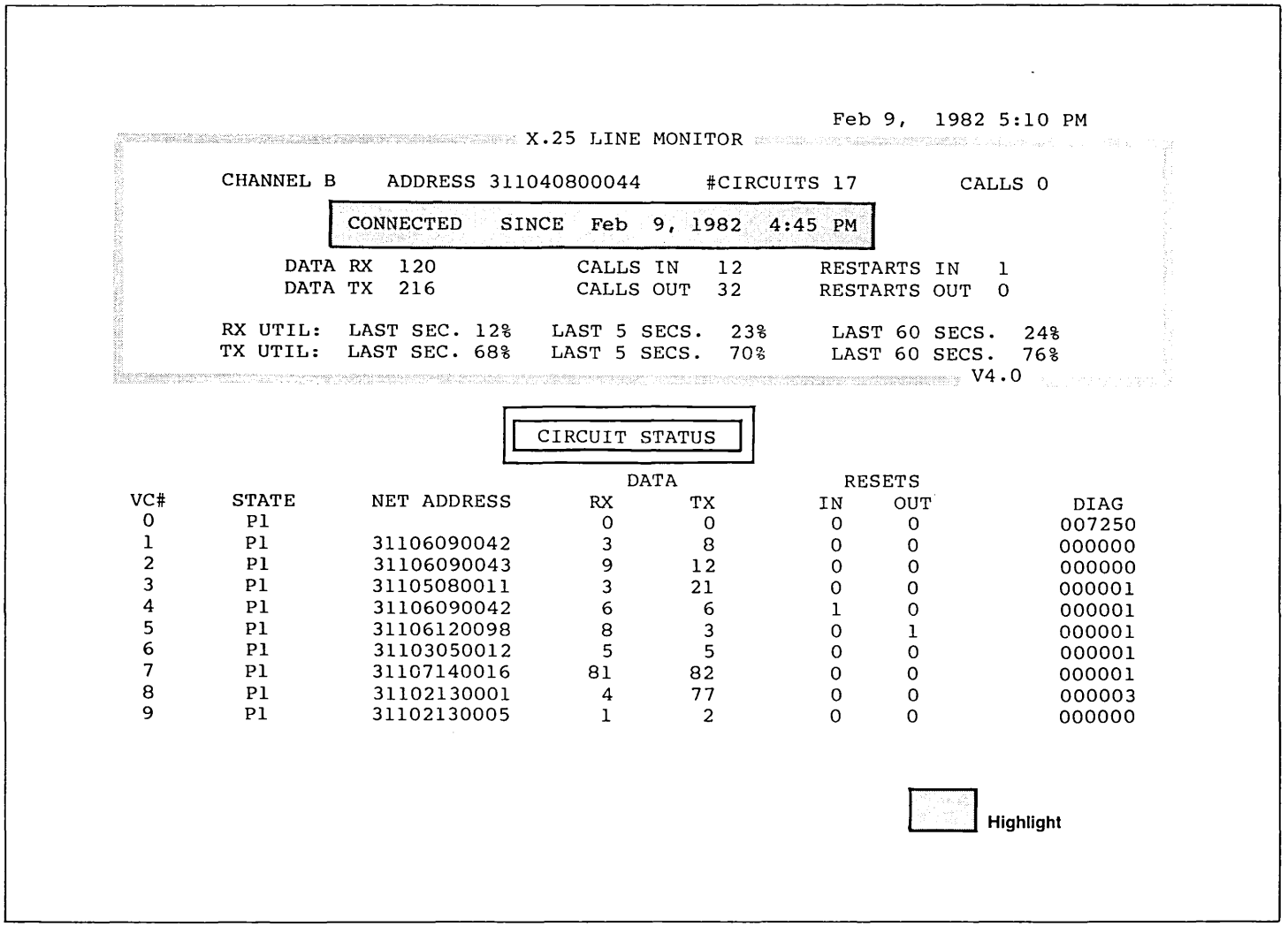
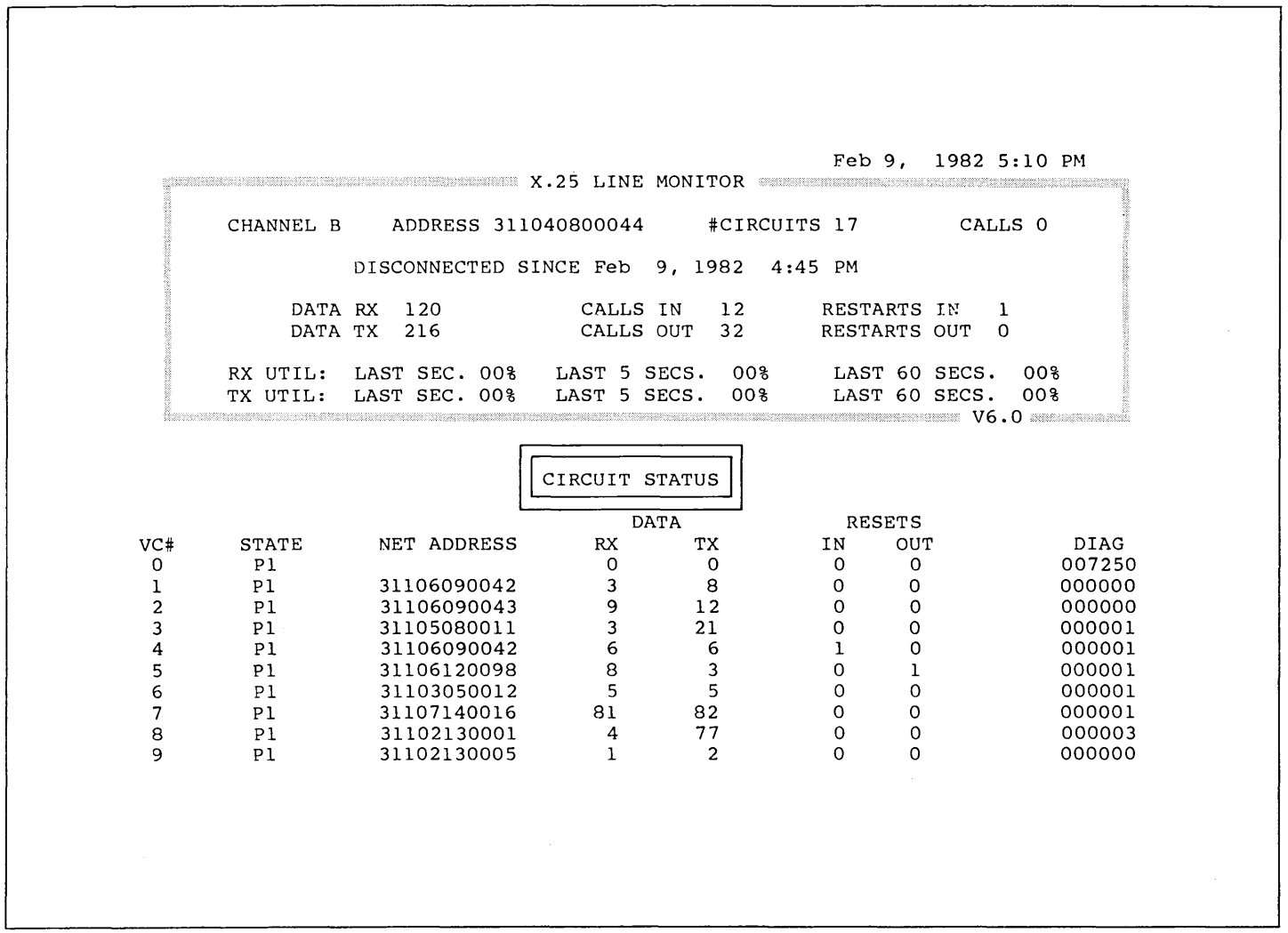


Figure 7-2. Connected Status of the X.25 Network Gateway.



VC#	STATE	NET ADDRESS	DATA RX	DATA TX	RESETS IN	RESETS OUT	DIAG
0	P1		0	0	0	0	007250
1	P1	31106090042	3	8	0	0	000000
2	P1	31106090043	9	12	0	0	000000
3	P1	31105080011	3	21	0	0	000001
4	P1	31106090042	6	6	1	0	000001
5	P1	31106120098	8	3	0	1	000001
6	P1	31103050012	5	5	0	0	000001
7	P1	31107140016	81	82	0	0	000001
8	P1	31102130001	4	77	0	0	000003
9	P1	31102130005	1	2	0	0	000000

Figure 7-3. Disconnected Status of the X.25 Network Gateway.

parameter when the X.25 Network Gateway is installed.

4. ADDRESS

   is the network identification code for the X.25 Network Gateway system service being monitored. This code is assigned by the PDN, and is entered as a parameter when the X.25 Network Gateway is installed.

5. #CIRCUITS

   is the number of virtual circuits allocated on the PDN that are being monitored. This value is entered as a parameter when the X.25 Network Gateway is installed.

6. CALLS

   is the number of virtual circuits currently connected.

7. x...x SINCE y...y

   is the state of the X.25 Network Gateway system service where

   x...x is one of the following (the field has a different video attribute, depending on the state):

   - OPERATIONAL--the X.25 Network Gateway is communicating with the PDN. This is shown in bright highlight.
   - CONNECTED--the connection to the PDN is established. This is shown in highlight.
   - DISCONNECTED--no connection to the PDN. This is shown in normal video.

   y...y is the date and time when the current state was entered.

8. DATA RX (received)

   is the number of data packets received since the X.25 Network Gateway system service became operational.

9. CALLS IN

   is the number of incoming calls established on nonpermanent virtual circuits since the X.25 Network Gateway system service became operational.

10. RESTARTS IN

    is the number of times the X.25 Network Gateway system service was restarted by the PDN since the X.25 Network Gateway system service was installed.

11. DATA TX (transmitted)

    is the number of data packets transmitted since the X.25 Network Gateway system service became operational.

12. CALLS OUT

    is the number of calls established on nonpermanent virtual circuits by the user since the X.25 Network Gateway system service became operational.

13. RESTARTS OUT

    is the number of times the X.25 Network Gateway system service has been restarted by the user since the X.25 Network Gateway system service was installed.

Line Utilization

Items 14 through 21 are the approximate percentage usage of the line for the transmit and receive directions for the previous 1, 5, and 60 seconds.

14 through 17. RX UTIL

   is the usage on the receive direction (RX UTIL--Item 14) during the previous second (LAST SEC.--Item 15), the previous 5 seconds (LAST 5 SECS.--Item 16), and the previous 60 seconds (LAST 60 SECS.--Item 17).

18 through 21. TX UTIL

is the usage on the transmit direction (TX UTIL--Item 18) during the previous second (LAST SEC.--Item 19), the previous 5 seconds (LAST 5 SECS.--Item 20), and the previous 60 seconds (LAST 60 SECS.--Item 21).

Version

22. Vn.n

is the version of the X.25 status program.

Circuit Status Data

The status of ten circuits at a time can be displayed in the circuit status area. For more than ten circuits, the scroll feature is used (see the "Keyboard Input" subsection below).

The status line for a circuit consists of eight entries (Items 23 through 30). The significance of these entries depends on the state of the circuit. Circuits can be in one of three states, which are determined by the X.25 Network Gateway system service state and the current use of the virtual circuit. Each state has a particular video attribute:

- Out of order (normal video). Normal video indicates a virtual circuit is out of order. If the X.25 Network Gateway system service is operational, a permanent virtual circuit is out of order if either it has not been initiated by the PDN or a fatal protocol error occurred; a nonpermanent circuit is out of order if a fatal protocol error occurred. If the X.25 Network Gateway system service state is not operational (disconnected or connected), then all virtual circuits are out of order.

- Idle (highlight). Highlight indicates a virtual circuit is idle. A virtual circuit is idle if it is not out of order but is not in use.

- In use (bright highlight). Bright highlight indicates a virtual circuit is in use. A

virtual circuit is in use when a user is connected to it.

Circuit status for permanent virtual circuits reflect activity over the permanent virtual circuit since it was initialized by the PDN. Circuit status for permanent virtual circuits is reinitialized when the permanent virtual circuit's state changes from out of order to idle.

Circuit status for virtual circuits in use reflect activity on that virtual circuit since the current call was established. Circuit status for virtual circuits that are idle or out of order reflect activity on the virtual circuit during the last call established on that virtual circuit. Circuit status for nonpermanent virtual circuits are reinitialized when a call is established on the virtual circuit.

23. VC#

    is the virtual circuit number; it is preceded by a p if the circuit is permanent. Virtual circuit 0 is reserved for protocol functions and is always in use when the X.25 Network Gateway system service is operational.

24. STATE

    is the circuit state. The B20 implementation of CCITT Recommendation X.25 defines ten internal circuit states (independent of the idle/in-use/out-of-order state discussed above) that correspond approximately to the circuit states defined in CCITT Recommendation X.25. A two-character acronym is displayed for the current state of the circuit (see Table 7-1).

25. NET ADDRESS

    is the remote user address, that is, the PDN identification code of the user at the remote end of the call. (This field is blank for permanent virtual circuits.)

26. DATA RX

    is the number of packets (circuit data) received since call establishment (or, since

Table 7-1. Circuit State Acronyms

Acronym	Meaning
R1	Ready
R2	DTE restart
R3	DCE restart
P2	DTE waiting
P3	DCE waiting
P6	DTE clear
P7	DCE clear
D1	Flow control ready
D2	DTE reset
D3	DCE reset

circuit initialization for permanent virtual circuits).

27. DATA TX

    is the number of packets (circuit data) transmitted since call establishment (or, since circuit initialization for permanent virtual circuits).

28. RESETS IN

    is the number of times the PDN has reset the protocol since call establishment (or, since circuit initialization for permanent virtual circuits).

29. RESETS OUT

    is the number of times the X.25 Network Gateway system service has reset the protocol since call establishment (or, since circuit initialization for permanent virtual circuits).

30. DIAG (diagnostic)

    are the cause and diagnostic codes transmitted in the last clear/reset/restart packet over this virtual circuit. For a permanent virtual circuit, the cause and

diagnostic codes are always from the last reset. For virtual circuit 0, the cause and diagnostic codes are from the last restart. For all other virtual circuits,

- if the virtual circuit is in use, the cause and diagnostic codes are from the last reset;
- if the virtual circuit is idle, the cause and diagnostic codes are from the last clear;
- if the virtual circuit is out of order, the cause and diagnostic codes can be from either the last reset or the last clear.

The cause code is displayed in the first three digits, and the diagnostic code is displayed in the last three digits; see Appendix B.

KEYBOARD INPUT

The following keys can be used during status monitoring; all other keys (except ACTION-FINISH) are ignored.

- FINISH--returns to the Executive.
- SCROLL UP/SCROLL DOWN--if more than ten circuits are allocated for the X.25 Network Gateway system service being monitored, the scroll keys are active and allow the user to select the ten circuits for which data are displayed in the circuit status area. Circuit data are displayed in circuit number order. However, circuit numbers may not be contiguous depending on how circuit ranges were specified at installation of the X.25 Network Gateway.

APPENDIX A
STATUS CODES

X.25 PACKET ACCESS METHOD

Error conditions for requests to the X.25 Network Gateway packet access method occur in three circumstances:

- The request either contains invalid parameters or is invalid in the context of prior requests.
- The request is superseded by a later request.
- The state of a virtual circuit or of the packet level as a whole makes it impossible to fulfill the request.

Incoming requests are checked for the validity of their parameters and the propriety of their occurrence when received. If an error condition is detected, the request is immediately returned with an appropriate status code.

If a valid and proper request is received, the packet level attempts to fulfill the request. A fulfilled request is returned with a status code of 0 ("OK"). In most cases, the request cannot be immediately fulfilled and must be held by the packet level for some period. If a request awaiting fulfillment is canceled by a later action from the user or the packet level, the request is returned with an appropriate status code.

Generally, it cannot be assumed that requests are returned in the sequence of their issuance for either error or normal completion. However, requests for reads and writes are returned in the sequence of their issuance when errors occur while they are being held by the packet level.

Decimal Value	Hexadecimal Value	Meaning
8500	2134	Link level down. The link level of the X.25 Network Gateway system service is not operational. This situation occurs either at power up before communication with the

PDN is established or during operation if an irrecoverable link level error occurs. The X.25 Network Gateway system service link level software should reestablish communication as soon as possible. If the link level remains down for an extended period, an irrecoverable error at the physical level or the link level exists, and a PDN representative should be contacted.

8501	2135	Packet level down. The packet level of the X.25 Network Gateway system service is not operational. This situation occurs (1) at power up, (2) during operation following a link level failure and subsequent reestablishment of link level communications, or (3) following an irrecoverable packet level error condition. The X.25 Network Gateway system service should reestablish the packet level as soon as possible. If the packet level remains down for an extended period, a PDN representative should be contacted.
8502	2136	Maximum number of this request has been queued. Previously submitted requests of this type must be completed before more can be issued. The maximum number of each request type is o NotifyNextIncomingCall requests: the number of virtual circuits per line. o ReadX25Packet requests: two per virtual circuit. o WriteX25Packet requests: five per virtual circuit.

- o all other packet access method operation requests: one per virtual circuit.

Generally, since the packet level should complete requests in a short period, the request should be resubmitted. If this condition persists, QueryX25Status should be used to examine the state of the X.25 Network Gateway system service to determine the cause of the delay.

8503	2137	X.25 Network Gateway system service is busy. Insufficient memory is available for the X.25 Network Gateway system service to process any more requests at this time. In normal operation, the X.25 Network Gateway system service should complete enough requests to free the memory required to accept new requests. If this error persists, reinstallation of the X.25 Network Gateway with additional memory should be considered.
8504	2138	Process termination. All requests were (or shortly will be) returned and all virtual calls were (or shortly will be) cleared because the user's process has terminated.
8505	2139	Bad port parameter. A NotifyNextIncomingCall operation contains a port range with one of two error conditions: - o The high port number is less than the low port number. - o The low and/or high port number is not in ASCII digits.

8506	213A	No virtual circuit available. An InitiateX25Call operation was received, but all virtual circuits were either in use or out of order.
8507	213B	User-specified time out. A ReadX25Packet or a NotifyNextIncomingCall operation could not be fulfilled by the packet level during the specified maximum time.
8508	213C	Virtual circuit in use. A request was received for a virtual circuit (or permanent virtual circuit) in use by some other user.
8509	213D	Call collision. An incoming call was received on a virtual circuit before an InitiateX25Call operation that had been allocated to that virtual circuit could be completed. The process should resubmit the InitiateX25Call operation.
8510	213E	Call cleared. An AcceptX25Call operation was made on a circuit for which no call was pending.
8511	213F	Virtual circuit not in use. A request was received for a virtual circuit (or permanent virtual cicuit) that was not allocated to any user.
8512	2140	DTE clear. Either an erroneous packet was received from the PDN, or the process requested that the call be terminated. The X.25 Network Gateway system service cleared the call that was on this virtual circuit. Data in the process of being transferred may have been lost.

8513	2141	DCE clear. The PDN cleared the call that was on this virtual circuit. Data in the process of being transferred may have been lost.
8514	2142	DTE reset. Either an erroneous packet was received from the PDN, or the process requested the call be reset. The X.25 Network Gateway system service reset the call on this virtual circuit. Data in the process of being transferred may have been lost.
8515	2143	DCE reset. The PDN reset the call on this virtual circuit. Data in the process of being transferred may have been lost.
8516	2144	DTE restart. An erroneous packet was received from the PDN, and the X.25 Network Gateway system service was restarted. All active calls were cleared. Data in the process of being transferred may have been lost.
8517	2145	DCE restart. The PDN restarted the packet level. All active calls were cleared. Data in the process of being transferred may have been lost.
8518	2146	Virtual circuit not in data transfer mode. A read, write, reset, or interrupt request was received for a virtual circuit that was not in the correct state. Either no call was present, or the circuit was in the process of being cleared or reset.
8519	2147	Interrupt data. This indicates normal completion of a read request, but with an interrupt data packet rather than a normal data

		packet. Interrupt data are returned to the process before any normal packets being held for the process by the packet level.
8520	2148	Virtual circuit out of order. An irrecoverable error occurred on this virtual circuit, and the X.25 Network Gateway system service declared it out of order. All calls on this circuit were cleared. The circuit can be restored only by the PDN.
8521	2149	Internal time out. The PDN did not respond to the packet generated by the request in the required time period. The process should resubmit the request.
8522	214A	Invalid virtual circuit number. Either (1) a request was received for a virtual circuit with a vch parameter that is either out of bounds or is 0 (circuit 0 is reserved), or (2) a ConnectX25Permanent operation was received for a nonpermanent virtual circuit.
8523	214B	Data truncated. Data to be returned to the process exceeded the size of sPacketRet as specified by the process. The data were truncated to the size of the buffer.
8524	214C	No buffer. A read or write operation was attempted, with sBuffer equal to 0.
8525	214D	Permanent circuit. ClearX25Call or AcceptX25Call was issued with the vch parameter of a permanent virtual circuit.

8526	214E	The number of VCs requested in the "Install X.25" parameter field is not enough to support the current system configuration. There must be at least one VC for a standalone workstation or one VC per workstation in a cluster environment.
8527	214F	The number of VCs requested in the "Install X.25" parameter field exceeds the maximum number of 32 VCs supported by the X.25 Network Gateway.
8528	2150	Invalid maximum packet size value. The allowed values are 16, 32, 64, 128 (default), 256, 512, or 1024.
8529	2151	Maximum number of "Permanent VCs" exceeds the total number of VCs assigned.
8530	2152	Number of "Incoming Only VCs" exceeds the total number of VCs assigned.
8531	2153	Number of "Outgoing Only VCs" exceeds the total number of VCs assigned.
8532	2154	Total of all "Outgoing Only VCs", "Incoming Only VCs" and "Permanent VCs" exceeds the assigned number of VCs.

Decimal Value	Hexa-decimal Value	Meaning
2350	092E	X.25 error occurred during operation. If an X.25 error occurs during a byte stream operation, the call is cleared, but the byte stream is not closed. A ReleaseByteStream of CloseByteStream operation must be done to close the byte stream.
2351	092F	Time out. The specified time out elapsed before the X.25 Network Gateway System Service Operation finished the operation in question is terminated, but the call is not cleared.

APPENDIX B

X.25 CAUSE AND DIAGNOSTIC CODES

CCITT Recommendation X.25 specifies two diagnostic fields within CLEAR REQUEST, RESTART REQUEST, and RESET REQUEST packets. The Cause field indicates the reason for the operation, and the Diagnostic field provides additional information.

CCITT Recommendation X.25 defines cause and diagnostic codes for clear, restart, or reset operations initiated by a PDN. These codes are listed below in the subsections "CCITT Standard Cause Codes" and "CCITT Standard Diagnostic Codes."

For clear, restart, or reset operations initiated by a DTE, CCITT Recommendation X.25 specifies that the value of the Cause field is 000 and that the value of the Diagnostic field is defined by the DTE.

CLEAR REQUEST, RESTART REQUEST, and RESET REQUEST packets can be generated locally either by the X.25 Network Gateway system service itself (in response to protocol errors) or by the user of the X.25 Network Gateway system service (for application-specific reasons). The diagnostic codes included in CLEAR REQUEST, RESTART REQUEST, and RESET REQUEST packets generated by the X.25 Network Gateway system service are listed below in the "B20 Diagnostic Codes" subsection.

The same set of values is used in CLEAR REQUEST, RESTART REQUEST, and RESET REQUEST packets; however, some codes are appropriate only to one packet type. For packets generated by the user, the user must supply the value of the diagnostic code.

The most recent cause code for a virtual circuit is displayed as the first three digits of the virtual circuit's Diagnostic (DIAG) field in the circuit status area of the X.25 status program. The cause code is also returned by the QueryX25-Status operation.

The most recent diagnostic code for a virtual circuit is displayed as the last three digits of the virtual circuit's Diagnostic (DIAG) field in the circuit status area of the X.25 status program. The code is also returned by the QueryX25Status operation.

CCITT STANDARD CAUSE CODES

CLEAR Packets

Code	Definition
000	DTE originated.
001	Number is busy.
003	Invalid facility request.
005	Network congestion.
009	Out of order.
011	Access barred.
013	Not obtainable.
017	Remote procedure error.
019	Local procedure error.
021	RPOA out of order.
025	Reverse charging acceptance not subscribed (can be received only if the reverse charging user facility is used).
033	Incompatible destination.
041	Fast Select facility acceptance not subscribed (can be received only if the Fast Select user facility is used).

RESET Packets

Code	Definition
000	DTE originated.
001	Out of order (applicable only to permanent virtual circuits).
003	Remote procedure error.
005	Local procedure error.
007	Network congestion.

009	Remote DTE operational (applicable only to permanent virtual circuits).
015	Network operational.
017	Incompatible destination (applicable only to permanent virtual circuits).

RESTART Packets

Code	Definition
000	DTE originated.
001	Local procedure error.
003	Network congestion.
007	Network operational.

B 20 DIAGNOSTIC CODES

Code	Definition
000	No additional information.
001	Process termination. The virtual circuit is being cleared (permanent virtual circuit is being reset) because the process using it terminated.
002	Reset request time out. The maximum number of transmissions of a RESET REQUEST packet were performed without receipt of a RESET CONFIRM packet for the remote DTE.
003	CALL REQUEST time out. No CALL CONFIRM or CLEAR INDICATION packet was received in response to a CALL REQUEST packet.
004	Window time out. Unacknowledged packets were outstanding for an excessive period of time.
005	No answer. A CLEAR REQUEST packet was transmitted in response to an INCOMING CALL packet because no NotifyNextIncomingCall

operations that match the incoming call were present.

006 No session established.

For virtual circuits: a packet other than an INCOMING CALL packet was received, but no virtual circuit was established.

For permanent virtual circuits: a packet other than a RESET REQUEST packet with a cause code of 007 ("Network operational") was received while the permanent virtual circuit was waiting to be initialized with such a packet.

007 Invalid interrupt confirm.
An INTERRUPT CONFIRM packet was received, but no INTERRUPT packet was outstanding.

008 Bad p(r).
The p(r) sequencing number of a received RR, RNR, or DATA packet was in error.

009 Bad p(s).
The p(s) sequencing number of a received DATA packet was in error.

010 No packet header.
A packet was received with a length of less than three bytes.

011 Bad packet ID.
A packet has been received with a packet ID that was not recognized by the X.25 Network Gateway system service.

012 Packet too small.
A packet was received with a length less than the minimum number of bytes specified for its type.

013 Packet too big.
A packet was received with a length greater than the maximum number of bytes specified for its type.

014 Remote error in P2.
A packet was received that is inappropriate in the current state.

015	Remote error in P3. The received packet is inappropriate in the current state.
016	Remote error in P7. The received packet is inappropriate in the current state.
017	Remote error in D1. The received packet is inappropriate in the current state.
018	Remote error in D2. The received packet is inappropriate in the current state.
019	Remote error in D3. The received packet is inappropriate in the current state.
020	Remote error in P1. The received packet is inappropriate in the current state.
021	Remote error in R3. The received packet is inappropriate in the current state.

CCITT STANDARD DIAGNOSTIC CODES

Code	Definition
000	No additional information.
001	Invalid p(s).
002	Invalid p(r).
016	Packet type invalid.
017	Packet type invalid for state r1.
018	Packet type invalid for state r2.
019	Packet type invalid for state r3.
020	Packet type invalid for state p1.
021	Packet type invalid for state p2.
022	Packet type invalid for state p3.

023 Packet type invalid for state p4.

024 Packet type invalid for state p5.

025 Packet type invalid for state p6.

026 Packet type invalid for state p7.

027 Packet type invalid for state d1.

028 Packet type invalid for state d2.

029 Packet type invalid for state d3.

032 Packet not allowed.

033 Packet not allowed; unidentifiable packet.

034 Packet not allowed; call on one-way logical channel.

035 Packet not allowed; invalid packet type on a permanent virtual circuit.

036 Packet not allowed; packet on unassigned logical channel.

037 Packet not allowed; REJECT packet not subscribed to.

038 Packet not allowed; packet too short.

039 Packet not allowed; packet too long.

040 Packet not allowed; invalid general format identifier.

041 Packet not allowed; RESTART packet with other than 0 in bits 1 through 4 and 9 through 16.

042 Packet not allowed; packet type is not compatible with the facility.

043 Packet not allowed; unauthorized interrupt confirmation.

044 Packet not allowed; unauthorized interrupt.

048 Timer expired.

049 Timer expired for INCOMING CALL packet.

050 Timer expired for CLEAR REQUEST packet.

051 Timer expired for RESET REQUEST packet.

052 Timer expired for RESTART REQUEST packet.

064 Call establishment problem.

065 Call establishment problem; facility code not allowed.

066 Call establishment problem; facility parameter not allowed.

067 Call establishment problem; invalid called address.

068 Call establishment problem; invalid calling address.

APPENDIX C

REFERENCES

BURROUGHS CORPORATION REFERENCES

Multimode Terminal Program Reference Manual.

Multimode Terminal Program User's Guide.

OTHER REFERENCES

Public Data Networks, CCITT--Comite Consultatif Internationale de Telegraphique et Telephonique International Telegraph and Telephone Consultive Committee Seventh Plenary Assembly, Document No. 7, Study Group 7, Contribution No. 489, (June 1980).

- Recommendation X.3, "Packet Assembly/Disassembly Facility (PAD) in a Public Data Network."
- Recommendation X.21bis, "Use on Public Data Networks of Data Terminal Equipments (DTEs) Which Are Designed for Interfacing to Synchronous V-Series Modems."
- Recommendation X.25, "Interface Between Data Terminal Equipment (DTE) and Data Circuit-Terminating Equipment (DCE) for Terminals Operating in the Packet Mode on Public Data Networks."
- Recommendation X.28, "DTE/DCE Interface for a Start-Stop Mode Data Terminal Equipment Accessing the Packet Assembly/Disassembly Facility (PAD) in a Public Data Network Situated in the Same Country."
- Recommendation X.29, "Procedures for the Exchange of Control Information and User Data Between a Packet Mode DTE and a Packet Assembly/Disassembly Facility (PAD)."

The CCITT documents are also contained in the following book:

McGraw-Hill's Compilation of Data Communications Standards, edited by Harold C. Folts and Harry R. Karp (1978).

APPENDIX D

HARDWARE CONSIDERATIONS

The X.25 Network Gateway is intended for use with synchronous modems such as the Bell 201, 208, or 209 Data sets (Data-Phone 2400, 4800,or 9600 Service). Modems must be either leased from the telephone company, purchased from independent vendors offering compatible products, or supplied by the PDN vendor.

In the typical system, the B20 system is permanently connected to the PDN over a leased line.

When other considerations permit, Burroughs recommends the following optional modem features:

- Internally Timed Transmitter,
- Switched Carrier,
- Without New Sync, and
- Four-wire operation.

For synchronous operation with an internally timed modem, the B22 system switches must be set for external clock, meaning that clocking is external to the workstation but internal to the modem. For the B21 family of workstations, this is done automatically under program control.

If the modem is connected to communications Channel B of the B22 workstation, the required switch settings on the I/O-Memory Board are shown in Table D-1 below.

Table D-1. Switch Settings for I/O-Memory Board

Switch	Setting
5	ON
6	ON
7	OFF
8	OFF

A double-male RS-232-C extension cable must be used to connect the workstation to the modem. It should be a straight-through terminal-to-modem cable rather than the crossover (null modem) type that is used to connect the workstation to another terminal.

RS-232-C signals used in synchronous operation are shown in Table D-2 below.

Table D-2. RS-232-C Signals in Synchronous Operation

Pin Number	Signal Name
1, 7	Ground
2	Transmit Data
3	Receive Data
4	Request to Send
5	Clear to Send
6	Data Set Ready
8	Data Carrier Detect
15	Transmit Clock
17	Receive Clock
20	Data Terminal Ready
22	Ring Indicator

GLOSSARY

Balanced Link-Access Protocol. A balanced link-access protocol is a particular implementation of a high-level data link control. Also see High-Level Data Link Control.

CCITT. Comite Consultatif Internationale de Telegraphique et Telephonique International Telegraph and Telephone Consultive Committee .

Data Communications Equipment. Data communications equipment provides communications services.

Data Terminal Equipment. Data terminal equipment, such as the B20 workstation, uses communications services.

DCE. See Data Communications Equipment.

DTE. See Data Terminal Equipment.

Extended Packet-Sequence Numbering. Extended packet-sequence numbering is an option that allows the maximum size of the window of outstanding packets to be increased.

Facility. Facility is a generic term for an option provided within CCITT Recommendation X.25.

Fast Select Facility. The Fast Select facility, an option within CCITT Recommendation X.25, allows both the called and calling data terminal equipment to transfer 128 bytes of data during call establishment and/or clearing.

HDLC. See High-Level Data Link Control.

High-Level Data Link Control. High-level data link control is a bit-synchronous, link access protocol. Variants of the HDLC protocol are used as the link level protocol in the X.25 Network Gateway and in the B20 cluster. Also see Balanced Link-Access Protocol.

LAPB. See Balanced Link-Access Protocol.

LC. See Logical Channel.

Link Level Protocol. Link level protocol, also known as link access protocol, defines the link access procedure for transferring data between

data terminal equipment and data communications equipment.

Logical Channel. A logical channel is one multiplexed data stream over a physical line.

MTP. See Multimode Terminal Program.

Multimode Terminal Program. The Multimode Terminal Program (MTP) is software that allows a master, cluster, or standalone workstation to operate on public packet-switching networks that support CCITT Recommendation X.25.

Multiplexing. Multiplexing is a technique that enables several data streams to be sent over a single physical link between the data terminal equipment and the network. The link is shared by multiple users although it appears as a private link to each user.

One-Way Virtual Circuits. A one-way virtual circuit accepts either incoming or outgoing calls, but not both. Also see Permanent Virtual Circuit, Virtual Circuit, and Two-Way Virtual Circuit.

Packet. A packet is the basic unit of data transfer over an X.25 communications network. A packet contains control information as well as data.

Packet Assembly/Disassembly Facility. The packet assembly/disassembly (PAD) facility is the special software in a public data network that processes the data from non-packet mode terminals into a format compatible with CCITT Recommendation X.25.

Packet Level Protocol. The packet level protocol is the X.25 layer that defines procedures for communication between two pieces of data terminal equipment through a public data network.

Packet Switching. Packet switching is a type of data transfer that occupies a network communications link only during the time of actual data transmission. The data are transmitted in small segments, called packets.

PAD. See Packet Assembly/Disassembly Facility.

PDN. See Public Data Network.

Permanent Virtual Circuit. A permanent virtual circuit does not require call establishment and cannot be cleared. Also see One-Way Virtual Circuit, Two-Way Virtual Circuit, and Virtual Circuit.

Physical Level Protocol. The physical level protocol defines the modem interface between data terminal equipment and the data communications equipment.

Protocol. A protocol is a set of procedures that govern the exchange of data over a communications link.

Protocol Identifier. A protocol identifier is information, transferred during call establishment, that identifies the higher level protocol to be used for a virtual call.

Public Data Network. A public data network is a regulated provider of communications services.

Two-Way Virtual Circuit. A two-way virtual circuit accepts both incoming and outgoing calls. Also see One-Way Virtual Circuits, Permanent Virtual Circuits, and Virtual Circuits.

VC. See Virtual Circuit.

Virtual Circuit. A virtual circuit is a communications link between data terminal equipment through a store-and-forward packet network. Although data packets are not necessarily kept in sequence during transmission, they are kept in sequence at the receiving end of the link. Also see One-Way Virtual Circuit, Permanent Virtual Circuit, and Two-Way Virtual Circuit.

Virtual Circuit Handle. A virtual circuit handle is a 16-bit number used to reference a virtual circuit.

Window. (1) A window is the visible portion of display memory contained in the data frame of the video display. (2) In a public data network, a window is the number of X.25 packets that can be transmitted before receiving an acknowledgment. The default window size is two; the maximum window size is seven unless extended packet-sequence numbering is used.

INDEX

Documentation Evaluation Form

Title: B 20 Systems X.25 Network Gateway
Reference Manual

Form No: 1176104
Date: October, 1984

Burroughs Corporation is interested in receiving your comments and suggestions regarding this manual. Comments will be utilized in ensuing revisions to improve this manual.

Please check type of Comment/Suggestion:

☐ Addition ☐ Deletion ☐ Revision ☐ Error ☐ Other

Comments:

From:

Name _____

Title _____

Company _____

Address _____

_____

Phone Number _____ Date _____

Remove form and mail to:

Burroughs Corporation
Corporate Product
Information East
209 W. Lancaster Ave.
Paoli, PA 19301 U.S.A.