SOFTWARE
SPECIFICATION

Document Type: External Reference Specification
Product Name: LOCAL COMMUNICATIONS CONTROLLER
Product Number: I039*1, 2 & 3

This document is a working paper only and does not represent any official intent on the part of CONTROL DATA CORPORATION. Consequently, it is to be used for reference purposes by CONTROL DATA personnel only. The document or information contained therein may not be distributed to clients. Sales commitments are not to be made on the basis of this document.

<table>
<thead>
<tr>
<th>Change No.</th>
<th>SCN No.</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Original</td>
<td></td>
<td>4/01/71</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td>6/04/71</td>
</tr>
<tr>
<td>2</td>
<td>New Issue</td>
<td>11/01/71</td>
</tr>
<tr>
<td>3</td>
<td>New Issue</td>
<td>5/12/72</td>
</tr>
</tbody>
</table>
LOCAL COMMUNICATIONS CONTROLLER

PL10
**DESIGN DOCUMENT AUTHORIZATION**

**PRODUCT NUMBER/NOMENCLATURE**
1039*1.0, 1039*2.0 and 1039*3.0

**Local Communications Controller (LCC)**

<table>
<thead>
<tr>
<th>APPROVAL SIGNATURES</th>
<th>DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PREPARED BY: S. M. Lammers, J. C. Wiblishauser</td>
<td>S/M. Signed</td>
</tr>
<tr>
<td>PROJECT MANAGER: S. M. Lammers, J. C. Wiblishauser</td>
<td>S/M. Signed</td>
</tr>
<tr>
<td>SUPERVISOR: K. H. Kimmerle</td>
<td></td>
</tr>
<tr>
<td>DEPARTMENT MANAGER: J. C. Kellogg</td>
<td>J. C. Kellogg</td>
</tr>
<tr>
<td>DIRECTOR - SOFTWARE</td>
<td></td>
</tr>
<tr>
<td>- HARDWARE: M. J. McHale</td>
<td>M. J. McHale</td>
</tr>
<tr>
<td>GENERAL MANAGER: H. N. Frazier</td>
<td>H. N. Frazier</td>
</tr>
</tbody>
</table>

**REVISIONS**

<table>
<thead>
<tr>
<th>LTR</th>
<th>DESCRIPTION</th>
<th>INITIAL</th>
<th>DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>New issue: Section 3</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Section 4 and Section 5</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>are updated. New Section 6</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mode 4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**DISCLAIMER:** This document is a working paper only and does not necessarily represent any official intent on the part of Control Data Corporation.
1.0 INTRODUCTION

2.0 APPLICABLE DOCUMENTS

2.1 General Documents
2.2 Documents Pertaining to Link Control Procedures
2.3 External Documents

3.0 GENERAL DESCRIPTION

3.1 General Characteristics

3.1.1 Function Description
3.1.2 Hardware Configuration
3.1.3 Subsystem Software
3.1.4 Interface with Higher Level Processor

3.2 General Operating Characteristics

3.2.1 Multi Lines
3.2.2 Error Reporting
3.2.3 Error Recovery
3.2.4 LCC Buffer Threshold
3.2.5 Basic HLP Operation

4.0 MODE 2 PROCEDURE

4.1 General Description of Mode 2

4.1.1 Configuration
4.1.2 Mode 2 Features
4.1.3 Station Level Functions

4.2 LCC/Remote Station Interface Procedure

4.2.1 Information Flow: Two-Way Simultaneous
4.2.2 Format of Blocks
4.2.3 Error Recovery

4.3 Operation with HLP

4.3.1 Request Entry Format: HLP to LCC
4.3.2 Request Entry Format: Stream
4.3.3 Command Format: Stream
4.3.4 Command Format: HLP to LCC
4.3.5 Response Formats
4.4 Overview of Mode 2

5.0 MODE 3 - INTERACTIVE TTY PROCEDURES

5.1 General Description of Mode 3

5.1.1 Configuration
5.1.2 Mode 3 (TTY) Operation

5.2 Interactive Mode 3 (TTY) Procedures
5.3 Interface with HLP

5.3.1 Request Entry Format: HLP to LCC
5.3.2 Command Format: HLP to LCC
5.3.3 Request Entry Format: Stream
5.3.4 Command Format: Stream
5.3.5 Response List

5.4 Overview of Mode 3

6.0 MODE 4 PROCEDURE

6.1 General Description of Mode 4

6.1.1 Configuration
6.1.2 Mode 4 Features

6.1.2.1 Terminal Identification
6.1.2.2 Determination of Initial Activity on Terminals and Stations of Multi-Station Terminals
6.1.2.3 Read Command Option for Mode 4 Terminals
6.1.2.4 Code Conversion
6.1.2.5 Dynamic Selection of a TM on a Line
6.1.2.6 Multi-drop Lines
6.1.2.7 Automatic Dialing, Mode 4 Terminals
6.1.2.8 Automatic Dialing, Teletype
b.1.2.9 Line Disconnect
b.1.2.10 Error Handling Option for Mode 4 Terminals
b.1.2.11 Control Characters Added/Deleted
b.1.2.12 Generation and Validation of Messages between the Terminal and the LCC Needed to Perform the Operation Requested by the HLP.
b.1.2.13 Error Recovery
b.1.2.14 Site Polling Option

b.1.3 Interface With the HLP

b.1.3.1 Line List
b.1.3.2 Request List

b.1.3.2.1 Request List for Dialed and Single Terminal Dedicated Lines
b.1.3.2.2 Request List for Multi-Drop Lines
b.1.3.2.3 Establishment of Request Lists by the HLP
b.1.3.2.4 Format of Requests

b.1.3.3 Commands

b.1.3.3.1 Command Format, HLP to LCC
b.1.3.3.2 Command Format, Stream

b.1.3.4 Buffers
b.1.3.5 Response List

b.1.3.5.1 Normal Unsolicited Response Format - Type 1
b.1.3.5.2 Normal Solicited Response - Type 2
b.1.3.5.3 Normal Solicited Response - Type 3
6.1.3.5.4 Abnormal Unsolicited Response - Type 4
6.1.3.5.5 Solicited Response Codes
6.1.3.5.6 Unsolicited Response Codes

6.1.3.6 Use of the Interface by the HLP
   6.1.3.6.1 HLP's Role in Error Recovery
   6.1.3.6.2 Intervene Key Processing
   6.1.3.6.3 Termination of Stream Commands by the HLP
   6.1.3.6.4 Autodisconnect Requiring Action by the HLP

6.1.4 Interface With the Terminal
   6.1.4.1 Error Correction Procedure With Read Commands

6.1.5 Overview

Appendix A - Glossary of Terms
1.0 INTRODUCTION

This ERS reflects how the LCC subsystem interacts with a higher level processor {HLP} and how the LCC interacts with communication lines.

The LCC subsystem interacts with the HLP via a defined protocol which is specified in the Software Interface for PL Compatible Trunk Specification. This specification is the guide to all subsystems which interact within a network of CDC CYBER central processors, stations and subsystems. Thus, the LCC subsystem can be a part of this network.

Since there is an array of types of remote terminals, line/terminal protocols have been established for interaction with the array of terminal types. These protocols are referred to as "Modes". Each mode is numbered, thus, for each line/terminal protocol there is a MODE X which is associated with that defined protocol.

This ERS is organized so that information associated with a particular mode is in a section dealing with that mode.
2.0 APPLICABLE DOCUMENTS

2.1 General Documents

Buffer Controller Engineering Specification No. 52410600
Buffer Controller Normal Channels No. 52410700
Buffer Controller Block Transfer Coupler No. 52410800
CDC Compatible Truck Interface Specification No. 52418100
LCC11 Engineering Specification (Hardware) No. 52420800

Software Interface for PL Compatible Trunk No. 52426200
ERS for MSBT Software {1042*1}
ERS for HSBT Software {1041*1}

Software Interface Procedures PL10/ICL Compatible Trunk, No. 52426200, dated 7/70.


Graphics Display Subsystem interfacing with Data-Phone

Data Set 201A or 201B, Engineering Spec. No. 14046000, Feb. 1970, Revision D

Remote Multi-Station Display Subsystem with USASCII

216-4 Controller Engineering Spec. No. 14048700, October, 1969; Revision A.
2.2 Documents Pertaining to Link Control Procedures

Memorandum to: Members of USASCII X3.4: Proposed Interactive Procedures. {Submitted by R. F. Meyer, February 20, 1970.}


6000 INTERCOM 4 ERS {Section Pertaining to TTY} No. E043*4.0 - E013*3.4.

2.3 External Documents

USA Standard Code for Information Interchange USA X3.4 - 1968.
Bell System Data Communications Technical Reference Manual
Data Set 201A and 201B Interface Specification - Sept. 1962
Data Set 203-Type, May 1969
Wideband 303-Type, 1966
Data Set 103A Interface Specification, February 1967.
3.0 GENERAL DESCRIPTION

3.1 General Characteristics

There are generally two levels of software associated with a communications subsystem - the application and the driver.

It is the responsibility of the application to interpret and process data associated with the terminals while it is the responsibility of the driver to maintain an orderly flow of data between the application and the terminal network.

The LCC is a driver level product which is concerned with the control and maintenance of an orderly flow of data between central site facilities and the terminal network. There is, however, one fundamental difference between this and previous communications drivers. This difference centers around the concept of distributive computing made possible by the presence of the programmable LCC Subsystem.

In general a communications driver consists of the functions listed below. The general division of functions between the HLP and the LCC is as indicated:

Performed by LCC

- line multiplexing {scanning for assembled data}
- line control {HDX/FDX considerations, speed, etc}
- Link control {link error recovery, interaction with remote station}
- network control {polling}
Performed by HLP device control (code compression, device not ready/down recovery, escape sequence processing). The LCC is subordinate to the Higher Level Processor and dominant over the link. In order that the two elements be properly coordinated in the communications process, a series of pre-defined interactions are exchanged via HLP Memory over the CDC compatible trunk. This trunk, which is 32 bits wide, is also the data path between the LCC and the HLP. In a logical sense, the HLP Memory module is the common element between the HLP and the LCC subsystem.

Figure 3.1-2 shows a basic LCC subsystem. The subsystem is modular and can be configured to accommodate a variety of applications within the design constraints.
Figure 3.1-2 Basic LCC Configuration

The LCC services up to 48 mixed, synchronous and asynchronous transmission channels. Data can be transmitted in both directions on each channel with transmission rates ranging from 75 to 50,000 bits per second (bps), depending on the capability of the individual channels. Maximum instantaneous throughput rate (total for all channels) is 400,000 bps; for example, 200,000 bps input and 200,000 bps output or 300,000 bps input and 100,000 bps output, etc.

3.1.1 Functional Description

The major functions of the LCC are:

- Assembly/Disassembly - Data is transmitted as a serial bit stream on the transmission channels. The data set adapters
perform serial to 8-bit assembly on input and 8-bit to serial disassembly on output. In addition, subsystem software performs 8-bit/32-bit assembly/disassembly for data exchanged with the higher level computer.

- Multiplexing - The subsystem software scans all channels periodically to transmit or receive data at the rate dictated by the channels. High speed lines are serviced more often than low-speed lines.
- System Interface - The LCC processes the HLP's Requests and commands and builds Responses indicating the status of the Request and Command.

In previous communications systems some of these functions were handled by the central computer. The LCC reduces the input/output overhead of the main computer by performing many of the control and housekeeping functions associated with data communications.

3.1.2 Hardware Configuration

Figure 3.1-3 shows a fully expanded LCC subsystem. The principle hardware elements are:

- Buffer Controller - A small scale stored-program computer used as a programmable control element in the communications controller.
- Memory - A 4K, or 8K 200 nanosecond, 16-bit memory used for storage of program and data. The 8K memory is achieved by expanding the basic 4K in a 4K increment.
- Normal Channels - Act as interfaces for the exchange of data between the data sets and the buffer controller. The number of normal channels required depends on the number of data sets to be serviced.
- Coupler - An interface between the buffer controller block transfer channel and a standard 32-bit CDC compatible trunk.
- Expansion Cabinet - Required in configurations that include more than 16 communication lines. It allows the buffer controller to address up to 48 communication channels.
- Data Set Adaptor - Acts as an interface between a data set (modem) and normal channels. One DSA is required for each transmission channel. During input operations, the DSA assembles the serial bit stream furnished by the data set, into 8-bit characters. During output operations the reverse disassembly operation occurs. Three types of DSA's are available to accommodate a variety of data sets. Each DSA is a separate pluggable module. There are positions available for 48 DSA's in a fully expanded LCC.
- Data Set - Data sets or modulator - demodulators (modem) are not part of the LCC but each transmission channel serviced by the LCC must be terminated by a data set. The function of a data set is to convert a serial bit stream to a form suitable for transmission over a voice grade line or other transmission channel. It also performs the reverse operation, demodulation. A variety of data sets, furnishing
a wide range of transmission rates, are available. Care should be taken in selecting modems that are not data sensitive.

The LCC with two-way simultaneous modes does not drop the write line on full duplex synchronous models between transmissions; thus, an $0_{16}$ is used as time fill.

- Cyclic Encoder - This element is an error detection device that generates the cyclic redundancy code based on the polynomial $x^{16} + x^{15} + x^2 + 1$.

- Automatic Calling Adapter - This unit together with an Automatic Calling Unit (ACU) allows the LCC to call any telephone number in a switched network. ACU's, like data sets, are not part of the LCC. Each switched transmission channel requiring automatic calling capabilities must have an automatic calling adapter (ACA) as well as a data set adapter. ACA's plug into DSA positions within the LCC and thus reduce the maximum number of DSA's that can be installed. This unit is used if the software is available. Check the sections in this ERS which deal with modes to see if this unit is supported.

- Real-Time Clock - The Real-Time Clock or oscillator is used to detect link time-out conditions. It has a fixed setting of 10 milliseconds. It may require an adjustment on site.
EXAMPLE:

\[ \begin{array}{c}
\text{Real Time Clock Period} \\
\text{5 msec.} \quad \text{10 msec.} \quad \text{5 msec.} \\
\end{array} \]
Figure 3.1-3. Fully Expanded LCC Subsystem
3.1.3 Subsystem Software

The LCC subsystem software, resident in the buffer controller memory, controls all subsystem operations. Subsystem characteristics are thus highly software dependent.

3.1.3.1 Subsystem Software Organization

The software is basically divided into two general categories. One category is the interface with the higher level processor (HLP), see Section 3.1.4 for details. The other category is line protocol.

Line protocol infers a defined mode of interaction with a remote terminal. Since there are several types of remote terminals, there are several different line protocols (modes). The LCC subsystem software can have several modes (i.e., line protocols) of software interacting with the one set of software which interfaces with the HLP. Thus, an LCC can support one or more modes of communication lines. There need not be a mode per line.

An LCC could support 48 lines all interacting with one mode, or it could support 48 lines with groups of lines interacting with individual modes.

3.1.3.2 LCC Performance

There are two important factors affecting LCC performance. They are core restraints and data throughput.
Core Restraints

The amount of core required for the LCC software of a required configuration varies according to the number of different line protocols (modes) to be supported. The software which supports a line protocol (mode) interacts with other line protocol (mode) software through a base set of software.

The amount of core for a given configuration may be calculated by using the following formula:

\[ X \text{ words of core} = 2000 + A + \text{MODE2} + \text{MODE3} + \ldots + \text{MODEX} \]

2000 = fixed words (16 bit word) of core used as base software

\[ A = \text{ additional program size for the DSA driver} \]

4 lines = 0 words of core
8 lines = 60 words of core
24 lines = 180 words of core
32 lines = 240 words of core
40 lines = 300 words of core
48 lines = 360 words of core

\text{MODE2 through MODEX are the core equations which are unique to each mode. Reference the proper section of this ERS which deals with the modes to be configured; place their equations into the above equation.}
Data Throughput

The data throughput {the number of lines the LCC can service} depends on the largest cumulative line rate which can be driven by the LCC without data loss. Throughput is based on the following factors:

1. The maximum LCC throughput is determined by the largest line service time. The service time is the time required by the LCC software to input or output a byte on a line. The service time may vary from protocol to protocol and increases as a function of increasing number of lines.

The computed services times require that the buffer controller memory rate is 200 nsec cycle time and the rate that the HLP accepts 64 bits of data does not exceed 4.4 usec/32 bits. If these rates are slower, the service time per line increases. The larger the service time, the less the maximum supported throughput.

2. When two or more protocols are mixed in a configuration, the protocol with the largest service time determines the maximum supported throughput of the mix.

3. When a value for throughput is determined for a given number of lines and protocol mix, any combination of supported line speeds may be configured such that:

\[ \sum \text{line speeds} \leq \text{throughput} \]

Each half duplex synchronous line and each full duplex
asynchronous line is considered as a single line operating at the rated one way bit per second rate while each full duplex synchronous line is considered as a single line operating at twice the rated one way bit per second rate.

4. The maximum supported throughput of the LCC is determined according to the following formula:

\[ T = \frac{8}{S_{\text{max}}} \]

where:
- \( T \) = maximum supported throughput in bps
- \( S_{\text{max}} \) = maximum line service time in seconds

5. Maximum line service times are as follows:

Mode 2 = \( \{20+n+p\} \times 10^{-b} \) sec.
Mode 3 = \( \{30+n+p\} \times 10^{-b} \) sec.
Mode 4 = \( \{20+n+p\} \times 10^{-b} \) sec.

where:
- \( n \) = Additional service increment based on number of lines:
  - 1-4 lines \( n=0 \)
  - 5-8 lines \( n=3 \)
  - 9-16 lines \( n=6 \)
  - 17-24 lines \( n=9 \)
  - 25-32 lines \( n=12 \)
  - 33-40 lines \( n=15 \)
  - 41-48 lines \( n=18 \)
- \( p \) = Additional service increment based on HLP memory access priority
  - highest priority \( p=0 \)
  - 2nd priority \( p=1.1 \)
  - 3rd priority

NOTE: Access priority means that at highest priority the HLP accepts data at a guaranteed rate of 4.4 usec per 32 bit transfer.
Second priority implies the HLP accepts data at a guaranteed rate of 5.5 usec per 32 bit transfer. Third priority implies that there is no guaranteed transfer rate. Thus if three LCC's were coupled to one HLP (such as the CDC 7077) the third LCC must be configured to support exceedingly slow lines (300 bps and less) or a "lost data" condition may arise.

Since the sum of the line speeds is less than or equal to the throughput, a configuration of lines can be calculated. Following are some computed configurations attained by the above calculations:

In a two-way simultaneous mode (such as Mode 2):

- up to 4 lines at 50K bps
- up to 16 lines at 9600 bps
- up to 30 lines at 4800 bps
- up to 48 lines at 2400 bps

In a two way alternate mode (such as Mode 3 and Mode 4):

- up to 30 lines at 9600 bps
- up to 48 lines at 4800 bps or less

For configurations where mixed line speeds are supported, no simple formula exists to calculate the maximum configuration for a given mix of line speeds. A reasonably close approximation may be obtained by replacing one or more lines of a given speed {bps} from a known configuration (a configuration generated by using the maximum throughput computed for the protocol with the largest service time).
with slower lines. The sum of the bit rate of the lines
being added must not exceed the {sum of the} bit rate
of the line{s} being replaced. A margin of error must be
allowed {usually by omitting one of the lower speed lines}. 
3.1.4 Interface with High Level Processor (HLP)

The following paragraphs describe the mechanism by which the HLP and the LCC communicate. The general structure is described here leaving the detailed codes to be described in later sections. Figure 3.1-4 and 3.1-5 graphically depict the interface lists.

3.1.4.1 Channel Flag

The Channel Flag is a hardware control line provided by the CDC compatible trunk from the HLP to the LCC. The primary function of the Channel Flag is to serve as a software interrupt in the LCC.

The Channel Flag is used by the HLP to notify the LCC that a line or lines are to be activated. Any further use of the channel flags are unique in that the various versions of the software and the respective sections should be consulted.

3.1.4.2 Line List

The Line List provides the basic linkage mechanism allowing the HLP to communicate with the LCC. The Line List as well as the Request lists, commands and responses reside in HLP memory. When a line is opened, an entry in the Line List directs the LCC to a Request List which allows the HLP and LCC coordinate the flow of data to/from the terminals.
The Line List is ordered according to line number; thus, the first entry in the list is associated with line 1 which is connected to DSA port 01 etc. There must be as many entries in the list as the highest numbered line attached to the LCC. That is, if the LCC is configured with lines 1 and 3 {lines 2, 3, and 4 are not present} then the Line List must be five entries long. When the Channel Flag is set, the LCC reads entries from the Line List for those lines that are currently inactive {i.e., closed}.

If the entry = -1 {i.e. is negative} there is no new activity for the line.

# -1 {i.e. is not negative} it contains a 32 bit base address of the Request List for this line.

3.1.4.3 Request List

Once established, the Request List is the basic vehicle by which the HLP and LCC coordinate the flow of data over the associated line. The Request Lists provide the basic routing functions as well as the controlling of the actual flow of data.

Once the base address of the Request List is read into the LCC, {i.e. the line is opened} it remains there until the line involved is closed. Periodic references to the Request Lists are made by the LCC to determine if a new request has been entered. As each request in the request list is completed {normal or abnormal} a response is sent to the HLP.
The Request List per line has two basic types of entry slots. The first entry slot of each list is dedicated for communications between the HLP and the LCC. The other entry slots are for control of data flow (streams).

The Request List must have as many stream entries as defined for the mode servicing the Request List. See the appropriate section of this ERS for further definition.

Streams are identified by number. An even number stream is dedicated to the LCC simplex output operation (i.e., the LCC transmits data to the remote terminal on even streams only). An odd number stream is dedicated to the LCC simplex Input operation (i.e., the LCC receives data from the remote terminal on odd streams only).

3.1.4.4 Command

Commands are 64 bits in length and supply detailed information to the LCC for proper execution of a request. There is one command per request.

Commands for streams have only two allowed functions to be executed. An input stream (odd numbered stream) can read data and an output stream (even numbered stream) can write data. The byte count and the buffer address is passed to the LCC.

Commands for HLP to LCC requests are of a utility nature and cause specific functions to be performed, but do not involve data transfers with terminals.
3.1.4.5 Response List

The Response List allows the LCC to communicate upward to the HLP. Entries in the Response List allows the LCC to inform the HLP of the current request, command, and system status. There are two types of responses: solicited and unsolicited. Solicited responses are those which are the result of a HLP request. Unsolicited responses are those which are the result of some unexpected stream or system condition. Response entries are 128 bits in length and are placed in a circular buffer (see Figure 3.1-5).

The Response List operates in a circular manner by the use of IN and OUT pointers. The pointers are 32 bit addresses. The IN pointer always points to the next point in the list to be filled. The OUT pointer always points at the next entry, to be emptied. The list is defined to be empty if IN = OUT. The list is defined as full when IN = OUT minus one entry. The LCC only changes the IN pointer since it is filling the list. The HLP only changes the OUT pointer since it is emptying the list. Both the LCC and HLP compare the IN and OUT pointers. The circular buffer allows the two processors to independently operate on the list.

A common Response List is used by all versions of LCC software.
Figure 3.1-1 Interface Lists Resident in the Higher Level Processor

0 ≤ N ≤ 47

N+1 = FFFF FFFF implies inactive slot
    = 00F3 FFFF implies the system is to be suspended. See Section 3.2.5.4
Circular Response List

Figure 3.1-5

N IS PREDEFINED BY LCC
3.2 General Operating Characteristics

3.2.1 Multi-Lines

The LCC software scans all communication lines periodically to transmit or receive data at the rate dictated by the line. This is the traditional multiplexing operation.

Communication lines connected to the LCC on lower number DSA ports (Data Set Adapter I/O slot within the normal channels) are serviced more often than the higher numbered DSA ports. Therefore, when a system is configured, the high speed lines must interface with the lower number DSA ports. Note LCC hardware specification, #52420800, Tables 2 through 8 for explanation on configurations.

3.2.2 Error Reporting

Error status is always reported to the HLP via the response field. Error status is reported even if successful recovery is made. Some of the error status consists of:

a. Error Recovery was initiated x times for the last block
b. The status of the Communications Adapter
c. The condition of last data block
d. The condition that the connected remote has its data set adapter carrier off
e. The condition that LCC buffers are saturated

3.2.3 Error Recovery

Error recovery for the CDC Compatible Trunk is initiated when the error status condition is detected. If the error occurs
during a data transfer from a line buffer, normal link error recovery is initiated and a response is written to the HLP reporting an intermittent coupler problem. If the error occurs during LCC work cycle (non-link operation), then the transfer is repeated until it is successful.

Error recovery on the link is described in the sections describing the different modes of LCC software.

3.2.4 LCC Buffer Threshold

The LCC software has a buffering scheme which allows it to dynamically rotate internal buffers. When use of these buffers reaches a saturation threshold, the LCC reports to the HLP that buffer saturation is present and aborts the request which was being processed when the condition was discovered. When the buffers are saturated, the LCC reports to the HLP that the line is being closed by doing the following. As each outstanding request is abnormally terminated, a response is written and when all stream requests have been terminated abnormally, a response is written that the line is closed.
3.2.5 Basic HLP Operations

3.2.5.1 Activate a Line

The HLP activates a line by first establishing a Request List for each line to be activated. The location of each Request List is entered in the appropriate slot in the Line List. The channel Flag is then set notifying the LCC that new entries have been made in the Line List. The LCC searches the Line List for all new entries. As each new entry is found, the line is activated. When the Line List has been completely searched, the Channel Flag is cleared and a response written indicating that the lines are activated.

Once the HLP sets the Channel Flag, it should not attempt to make new entries into the list until it receives the response indicating the LCC is through searching. At this time the HLP must set the newly opened line entries to -1.

3.2.5.2 Close a Line

The HLP has the option of declaring an active communication line inactive. The LCC discontinues servicing that line. No data is sent or received. When this action is taken the LCC terminates all requests for this stream. When they are terminated it notifies the HLP that the line is closed. The Request List is no longer defined. For data to be sent on this line again, the operation of activating the line is required again.
3.2.5.3 Autoload and System Initialization

The system autoload function consists of loading the LCC memory with the LCC software. The LCC system initialization consists of maintaining a dialogue with the HLP to obtain configuration information and to initialize the system. When initialization is complete, control is transferred to the LCC software which initiates standard HLP interface procedures, Section 3.1.4.

Initiation of the autoload sequence can be accomplished by program control; HLP initiates autoload across the CDC compatible trunk. {Program autoload is executable code.}

The HLP places the 64 16-bit autoload record {Fig. 3.2-1} in CSM starting at address 0 prior to initiating the LCC Subsystem autoload.

The 16-bit Autoload Control Word, Word 61 of the Autoload Record {Fig. 3.2-1}, specifies the operation type as follows:

- 0001 Autodump
- 0002 Autoload

The 32 bit Autoload Table Address, Words 62 and 63 of the Autoload Record, is a 32-bit HLP address that points to the Autoload Table {Fig. 3.2-2} used to support the above operations and communications via the software interface when the autoload phases are completed.
The first 16 words of the Autoload Table are always present. The last 16 words are present only when autoloading LCC software which supports mode 4 terminals.
Autoload Record Format

Autoload Control Word
Autoload Table First
Word Address

FIGURE 3.2-1

Autoload Table Format

Autoload
Response
Next Record
Address
Next Record Length
Dump Address
Request Table
Address
Response Table
Address
Response Table Length
Parameter 1
Parameter 2
Parameter 3
Parameter 4
Parameter 5
Line Select Mask 1
Line Select Mask 2
Line Select Mask 3
Line Select Mask 4
Line Select Mask 5
No. of Code Conversion Tables

FIGURE 3.2-2
The Autoload Table (Fig. 3.2-2) resides in HLP memory and is not relocated during the entire autoloading phase.

**Autoload Response:** The LCC writes 1 32-bit response into this area. The following codes, right justified, are used.

1 - Error Switch: N/A.

2 - Ready to read or to write another autoload record: Used in autodump and autoload.

4 - Operation completed: When mode 4 is not present this is used at completion of autoload. When mode 4 is present, this response is used to indicate the completion of the autoload sequence and to request the first and subsequent initialization blocks {defined below}.

6 - Ready to dialogue through I/O software interface: Used after internal initialization following completion of autoload.

16 - Error Response: LCC aborted operation and requires autoload to restart.

**Next Record Address:** This 32-bit HLP address indicates where the LCC must read or write the next record.

**Next Record Length:** This 16-bit word indicates the length {even number of 16-bit words} of the next record. A length equal to 0 terminates a autoload and forces the autodump operation to scan the Channel Flag.
Dump Address: This 16-bit address, used only for an autodump operation, indicates from which address the LCC must start the autodump of its own memory into HLP memory.

Request Table Address: This 32-bit HLP address points to the Line List {Section 3.1.4.2} in HLP memory used by the LCC.

Response Table Address: This 32-bit HLP address points to the Response Table {Section 3.1.4.5} in HLP memory used by the LCC.

Response Table Length: This is the length {number of 32-bit words} of the circular Response Table. This length includes the IN and OUT pointers.

Parameter Word 1: Not assigned

Parameter Word 2: Defines number of lines attached to the LCC operating in MODE 2.

Parameter Word 3: Defines the number of MODE 3 lines.

Parameter Words 4-5: Not assigned.

Parameters needed if MODE 4 lines are being serviced by the LCC subsystem.

Line Select Mask 1: Not assigned.

Line Select Mask 2: This is a 48-bit parameter used to select those lines, out of the 48 possible, that are MODE 2 lines. Bit 0 corresponds to line 0; bit 1 to line 1, etc.

Line Select Mask 3: This is a 48-bit parameter used to select those lines, out of the 48 possible, that are MODE 3 lines.

Line Select Mask 4: This is a 48-bit parameter used to select those lines, out of the 48 possible, that are MODE 4 lines.

Line Select Mask 5: Not assigned
Number of Code Conversion Tables: This parameter specifies the number, from 0 to 6, of code conversion tables to be allowed. This parameter need not be supplied if the code conversion tables and table pointers are assembled into the code by using the assembly option provided for this purpose.

As each autoloading record is processed, the LCC will return an autoloading response of 2 to the HLP. The HLP will reply by supplying the LCC with another autoloading record, updating the autoloading table and setting the channel flag. The autoloading sequence will continue until the HLP replies to an autoloading response from the LCC with a value of zero in the "next record length" field of the autoloading table. The system initializer (SI) will use these parameters to initialize the LCC subsystem.

When this has been done the LCC will respond to the HLP with an autoloading response code of 4, indicating that the SI is ready to accept MODE 4 Terminal Model Records and Line Parameter Records.

If MODE 4 is not supported, the LCC subsystem is considered to be totally initialized and the LCC initiates standard interface procedures, Section 3.1.4.

If MODE 4 is supported, the LCC maintains a dialogue with the HLP in which the LCC sends response codes of 4 to notify the HLP to send TM and LC records. When the HLP replies with a value of zero in the "next record length" field of the autoloading table, the LCC is considered to be totally initialized.
and the LCC initiates standard interface procedures, Section 3.1.4.

For MODE 4, the HLP must first send all of the Terminal Model Records and then send the Line Parameter Records to the LCC. These records may be sent all in one buffer or in several buffers. In addition, each individual Terminal Model Record or Line Parameter Record must be a multiple of 32 bits in length with a 16-bit padding word inserted in the record where needed.

The format of the Terminal Model Record is shown in Figure 3.2-3.
A terminal Model (TM) is a set of words describing a given terminal. One TM is required for each unique terminal configuration in the system. From 1 to 16 TM's may be used on a single line and up to 32 may be defined for the system. The TM consists of a 32-bit site descriptor and 1 to 16 32-bit device descriptors, depending on the number of devices on the terminal.

Each TM that can be assigned to a given line must have a site address different from that of any other TM that may be assigned to the same line. The reason for this is that the site address will be used to determine what type of site is present at any given time on a given line. Therefore, no ambiguities can be allowed.
Field A: this field is one bit in length and is equal to one
to indicate that this is a Terminal Model Record.

TM Number: this is the Terminal Model number. It has a value
from 0 to 31 {decimal}.

Protocol: this field is 3-bits in length and indicates the
communications protocol to be used. The following values
are defined:

0 - 200 UT protocol is used
4 - 216 protocol is used.

Code Set: this field indicates the code set used by the
terminal. The values are:

0 = ASCII
1 = internal BCD
2 = external BCD

Screen Size: indicates the size of the CRT screen. Values are:

0 = 20X50
1 = 13X80
2 = 8X80
3 = 16X80

Field B, The Number of Devices on the Terminal: this field
indicates the number of devices attached to the terminal and,
therefore, the number of device descriptors in the TM {the
padding word is not included in this count if one exists}. 
Terminal Type: indicates the type of terminal as follows:

0 = 200 UT
1 = 214
2 = 711
3 = {not used}
4 = 241 grid
5 = {not used}
6 = 216

Site Address: this is the site address to be used by all terminals on switched lines if the terminals are exactly described by this TM. A hex value of 0-F corresponds to the hexadecimal site address 70-7F {160-177 octal} respectively. If bit 11 equals one, then all possible site codes are to be polled and this is the only TM allowed on lines that select this TM.

Device Type: this is the first field in the device descriptor. This field describes the type of device being described by this device descriptor as follows:

0 = CRT
1 = card reader
2 = printer
3 = typewriter
4 = cassette
5 = paper tape
Station Address: the station address to be used. If the station code is not enabled for the particular terminal being described, this value is set to zero. Otherwise a value of 1 to F (hexadecimal) is to be used corresponding to an octal address of 141 to 157 respectively.

The format of the Line Parameter Records is shown in Figure 3.2-4. Words 0-3 of this record are present for all mode 4 lines, words 4-n are present only if the line is a dedicated, multi-point line.

<table>
<thead>
<tr>
<th>WORD</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>A</td>
<td>B</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>C</td>
<td>D</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>n</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   | Line Number |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   | No. of Sites |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   | ACU Port Number |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   | TM Selection Mask (32 bits) |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   | Site Code |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   | Additional TM/SITE Records (if any) |   |   |   |   |   |   |   |   |   |   |   |   |   |
|   |   |   | TM/SITE record or blank padding |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

FIGURE 3.2-4
Field A: always set to zero to indicate that this is a Line Parameter Record.

Field B: indicates 2 or 4 wire mode as follows:
   0 - 2 wire mode
   1 - 4 wire mode

Line Number: this field indicates the line number referred to by this record.

Field C: this field is reserved for future use when the ACU {Automatic Calling Unit} feature is implemented. This field when implemented will indicate whether autodial procedures are allowed on this line as follows:
   0 - autodial procedures are allowed
   1 - autodial procedures are not allowed on this line and ACU port number field has no meaning.

Field D: multipoint indicator with the following interpretation:
   0 - the line is point-to-point
   1 - the line is multi-point and words 4 to n are present.

Number of Sites: this field is used only if the line is a dedicated multi-point line. It indicates the number of terminals {sites} on the line, and also the number of TM/SITE words in this record {exclusive of the blank padding word if one exists}. 
ACU Port Number Reserved: this field is reserved for future use when the ACU feature is implemented. This field when implemented gives the number of the port attaching the automatic calling adapter to be used in automatic dialing procedures on this line. This field has meaning only if field C is zero and field D is zero.

TM Selection Mask: this field, 32 bits in length, contains the bit mask used to select the Terminal Models that are allowed on this line. Up to 16 bits may be set in this mask. See TM description above.

TM Number: the TM number describing the terminal represented by this word {TM/SITE record}.

Site Code: the value from 0-F {hex} corresponding to the site address 160-177 {octal} to be used by the terminal represented by the record in which this field is found.

3.2.8 Suspend

The HLP places the LCC into a suspended state which causes the LCC to temporarily stop all references to the HLP memory until the HLP sets the channel flag notifying the LCC to resume normal operations. During the suspended state, the LCC discontinues normal operations with the remotes. The HLP is notified when the LCC subsystem is in a suspended state.
In order to place the LCC into a suspended state, the HLP must place the code, 00F3H, into the N+1 slot of the line list, Figure 3.1-4.
4.0 MODE 2 PROCEDURE

Mode 2 link protocol in the LCC is intended to support the new line batch terminals (HSBT, MSBT, and LSBT). Mode 2 will only support synchronous lines which are point-to-point, full duplex, and non-switched. The transmission flow will be two-way simultaneous.

Mode 2 software resident in the LCC-11 memory provides a message framing and link control procedure.

This involves activities such as:

1. Insertion or stripping of link control characters that enclose the data blocks.
2. Addition or deletion of control characters that are required for transparent data block.
3. Adding or deleting station status information transmitted between the terminals and LCC.
4. Acknowledgment of transmissions.
5. Error Detection and Recovery - Each message block is protected by a cyclic redundancy check code. If the LCC senses an error, it asks the distant end for transmission of the block. If an error is detected at the distant end, the terminal device notifies the LCC which responds by retransmitting the block. The LCC furnishes error status information to the higher level computer for control and error logging purposes.
Station level control in the LCC permits transmission of several streams on a time-shared basis over a single communication line, which is connected point-to-point with the remote terminal. See Section 4.1.3.1 for a detailed explanation.

4.1 General Description of Mode 2

4.1.1 Configuration

Since Mode 2 interacts in a two-way simultaneous scheme, the number of lines supported are: (See Section 3.1.3 for further explanation)

- up to 4 lines at 50K bps
- up to 16 lines at 9600 bps
- up to 32 lines at 4800 bps
- up to 48 lines at 2400 bps

A mix of line speeds can be attained in the manner described in Section 3.1.3.

The amount of core required for Mode 2 software and configuration are as follows:

Mode 2 words of core = 1600 + 36L2 + 16AL2 + 16S2

1600 = fixed words of Mode 2 software

L2 = number of lines supported with Mode 2

AL2 = number of Mode 2 lines active (i.e., opened)

S2 = for all Mode 2 lines, the sum of active streams (i.e., sum of stream requests being processed by the LCC at an average operation time)
The number of line and streams serviced is dependent upon core size. In general, the maximum number of streams goes down as the number of lines increases. The following is the worst case under the given conditions:

A 4K {16 bit word} LCC will support with -
- 4 lines attached with 4 lines active up to 17 streams
- 5 lines attached with 5 lines active up to 14 streams

An 8K {16 bit word} LCC will support -
- 16 lines attached with 16 lines active up to 297 streams
- 32 lines attached with 32 lines active up to 183 streams
- 48 lines attached with 48 lines active up to 131 streams
4.1.2 Mode 2 Features

4.1.2.1 Control Characters Added/Deleted

For transmission, the LCC, which is dominant, frames all data blocks with a set of control characters. These control characters consist of an opening delimiter sequence, station control information, data {i.e., text}, framing control characters, and ending delimiters.

The LCC receives from the subordinate remote station data blocks which are bounded with similar control characters. The LCC interprets these control characters, ascertains if they are valid and strips them off of the data block before it records the data in the HLP data buffer.

4.1.2.2 Data Dependencies

Basically, the LCC is not data dependent. Data blocks must all be in the transparent format.

Transparency is invoked by the proper ASCII control characters {i.e., the LCC does not expect bit position eight of a data byte to reflect parity, and thus does not generate parity or check for valid parity} thus allowing any of the 256 possible bit combinations per data byte.
4.1.2.3 Error Recovery

If on receipt of a beta \(\bar{\beta}\) status block or an alpha \(\bar{\alpha}\) data block on a communication line, the LCC detects an error condition, the LCC initiates error recovery procedures for that block a maximum of N times \(N\) is defined at assembly time. When recovery procedures have been executed N times, the LCC exercises the Data Set Adapter to see if the condition is caused locally. The results of the exercise and the condition of a non-recoverable error are reported to the HLP, and then the line is closed and all HLP requests are terminated abnormally. The conditions which stimulate the LCC into recovery procedures are:

<table>
<thead>
<tr>
<th>CONDITION CAUSING RECOVERY</th>
<th>LCC RECOVERY ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Time out - a valid (\bar{\alpha}) response block for an (\bar{\alpha}) (\text{or} \bar{\beta}) transmission is not</td>
<td>The last (\bar{\alpha}) (\text{or} \bar{\beta}) block transmitted by the LCC is retransmitted.</td>
</tr>
</tbody>
</table>
received within the allowed time frame.

2. Incorrect Parity - an interpretable response block is received. However, the block has a control character parity error or cyclic redundancy check error.

3. Negative Response (NAK) - The remote station interpreted the last \( \times \) (or \( \beta \) ) transmission from the LCC as having a control character parity error or cyclic redundancy check error.

4. Improper Sequence - The expected \( \times \) (or \( \beta \) ) numbered response is not received. However, the parity and CRC are good.

An \( \alpha \) (or \( \beta \) ) message which requests a Repeat of the remote stations last \( \times \) (or \( \beta \) ) transmission is sent to the remote.

The last \( \alpha \) (or \( \beta \) ) block transmitted by the LCC is retransmitted.

The last \( \times \) (or \( \beta \) ) block transmitted by the LCC is retransmitted.

4.1.3 Station Level Functions

4.1.3.1 Multi-Streams

A stream is the simplex flow of data from one point to another
Streams are differentiated by numbers. For example, stream one has a different source/sink than stream two. Even numbered streams refer to output simplex flow of data from the LCC to a sink in the Remote Station and odd numbered streams refer to input simplex flow of data from a source in the remote station to a sink (i.e., buffer) in the HLP. The LCC coordinates the transmission of several streams (multi-streams) on a time-shared basis over a single communication line (note, that since this system is point-to-point, there is only one remote station per communication line).

An example of multi-stream activity within an LCC is the following. A HLP has a data buffer for stream 2 and a data buffer for stream 4. Both data streams are to be sent to the same remote station. The remote station has defined stream 2 as the printer buffer and stream 4 as the card punch buffer. The LCC sends a block \( \leq \text{HLP buffer size} \) of data for stream 2. While the remote station is printing the printer buffer, the LCC sends a block of data for stream 4. Thus, the remote station has sufficient data to drive devices simultaneously. Because the system is two-way simultaneous, similar multi-stream transmissions can be sent from the remote station to the LCC.
Graphical Description of Streams

Figure 4.1-1
4.1.3.2 Blocking of Data

The HLP defines a buffer area or areas within its core and the total data byte count for each buffer. This data is to be output to a remote station, stream x, line y. The total byte count may be greater than that which the remote station will receive. The LCC breaks up the transmission of this data into blocks of bytes (these blocks are multiples of four bytes) which are acceptable to the remote station. Thus, the data is transmitted in one or more blocks. The LCC inputs data from stream z, line y in blocks. If the HLP buffer is larger than the block size, the LCC packs successive blocks from stream z into the HLP buffer. However, if there is not sufficient buffer space for an input transmission block, the LCC does not input the data. See Section 3.2 for further detail.

Figure 4.1-2 is a graphic example for a blocked buffer.

The LCC logically divides the byte count into blocks. The size of a block is defined by an index passed to the LCC by the HLP. This index is used by the LCC to reference an internal assembled table which contains blocking values. The blocking values must be multiples of four bytes. The interface between the LCC and HLP has a hardware defined width of 32 bits (4 bytes). Transfers across the interface cannot be performed in smaller units than 4 bytes.
NOTE

In this example, the block size is 512 bytes.

Figure 4.1-2 Blocked Buffers
4.2 LCC/Remote Station Interface Procedure

4.2.1 Information Flow: Two-way Simultaneous

The LCC and the remote terminal interface by trading blocks of information. The LCC is dominant; thus, all trades are initiated by the LCC. There are three types of information trades: Control block, alpha \(\alpha\) and beta \(\beta\). The LCC (with type \(\alpha\) and type \(\beta\) information trades merged) is two-way simultaneous information flow. See Figure 4.2-1 for a graphic description. Once communications is established the LCC maintains this sequence of trades. Following every \(\alpha\) transmission, the LCC transmits a \(\beta\) block, then an \(\alpha\) block, then a \(\beta\) block, etc.

\[
\begin{array}{|c|c|c|c|c|c|c|}
\hline
\text{Status} & \text{Data} & \text{Status} & \text{Data} & \text{Status} & \text{Data} & \text{Status} \\
\hline
\text{Sends} & \alpha & \beta & \alpha & \beta & \alpha & \beta \\
\text{Receives} & \alpha & \beta & \alpha & \beta & \alpha & \beta \\
\hline
\end{array}
\]

Figure 4.2-1
Control Block Transmission

Control blocks are unique link trades which are done in a two-way alternate scheme. Control blocks fit outside the alpha/beta information trade. Thus the trade of control blocks must be done before alpha/beta communication trade is initiated or after the trade is terminated. Control blocks can carry a variety of different codes {i.e., meanings}.

This version of the LCC is implementing a restart function. The receipt by the remote or the LCC of a restart control block implies that the alpha/beta numbers are to be reset to an even state and that any outstanding trades of information are negated. If the LCC is reautoloade by the HLF when exchanges of data information were being traded by the LCC and the remote, the restart control block informs the remote that the last data block was not accepted.

Once the remote acknowledges receipt of the restart control block, the LCC considers communications to be established and initiates the trade of alpha/beta transmission blocks. The remote must always acknowledge receipt of a restart control block by transmitting a restart control block back.

If the remote sends a control block during normal trade of alpha/beta blocks, the LCC terminates all activity on that line until it can reply with a corresponding block {i.e., the line is closed and must be reopened}; See Section 4.3.5, unsolicited responses for further detail.
Graphic descriptions of restart control block trades are:

LCC

Sends: Restart \rightarrow Restart \rightarrow Restart \rightarrow \alpha \rightarrow \beta

Receives: ... \rightarrow Restart \leftarrow \alpha

Remote

Receives

IDLE POLL {LCC initiates restart in dormant system}

Figure 4.2-2

LCC

Sends: \alpha \rightarrow \beta \rightarrow \alpha \rightarrow \beta \rightarrow \alpha \rightarrow Restart \rightarrow \alpha

Receives: \beta \leftarrow \alpha \leftarrow \beta \leftarrow \beta \leftarrow \beta \leftarrow {\text{autoloading of LCC}} \leftarrow Restart \leftarrow \alpha

Remote

Receives

LCC initiates restart during trade of information

Figure 4.2-3

LCC

Sends: \alpha \rightarrow \beta \rightarrow \alpha \rightarrow \beta \rightarrow \alpha \rightarrow Restart \rightarrow \alpha

Receives: \beta \leftarrow \alpha \leftarrow \beta \leftarrow \beta \leftarrow \beta \leftarrow {\text{autoloading of remote}} \leftarrow Restart \leftarrow Restart

Remote

Receives

Remote initiates restart during trade of information. When the LCC sends the restart it is the same as in idle poll.

Figure 4.2-4
Alpha Transmission

Type α information trade consists of two information blocks. The LCC initiates the trade by transmitting a "status" block. After the complete block has been received by the remote station, the remote station replies with a "data" block. An α status block contains stream control information which is used by the remote station to direct its activities. An α data block contains specific data or text which is associated with an input {odd number} stream and which is destined for the HLP.

A graphic description of a sequence trade is the following. An arrow denotes a block of information and its direction. Read the graph left-to-right.

![Diagram](image)

Figure 4.2-5
Beta Transmission

Type $\beta$ information trade also consists of two information blocks. The LCC initiates the trade by transmitting a "data" block. After the complete data block is received by the remote, the remote responds with a "status" block. A $\beta$ data block contains data or text which is associated with an output {even numbered} stream and which is destined for the remote station. A $\beta$ status block contains status of streams in the remote, i.e., they are busy, they can accept data, etc.

A graphic description of a sequence of trades is the following. An arrow denotes a block of information and the direction of flow. Read the graph from left-to-right.

![Diagram]

Figure 4.2-6

If there is no information to trade, but communications are to be maintained, the LCC and remote can send an IDLE block {a block with no data or status} in place of the data or status block.
The LCC with type α and type β information trades merged is two-way simultaneous, non-continuous information flow. See figure 4.2-8 for a graphic description. The LCC always maintains this sequence of trades. Following every α transmission, the LCC transmits a β block and then an α block, etc.
Alpha and beta blocks are differentiated from each other by a sequential number scheme. Each block transmitted contains an even or odd sequence number. The even/odd numbers transmitted by the LCC for \( \alpha \) type information blocks are the ASCII graphic characters, p and q, and for \( \beta \) type information blocks are the ASCII graphic characters, 0 and l. The even/odd numbers expected by the LCC in the responding blocks sent by the remote for \( \alpha \) type information blocks are the ASCII characters, 0 and l, and for \( \beta \) type information blocks are the ASCII characters, p and q. See figure 4.2-9 for a graphic description of this numbering scheme. If the responding block number is not as it should be, the LCC initiates error recovery.

There are two types of transmission blocks: information blocks and error information blocks. There are two versions of information blocks: data information block and status information block. Data information blocks contain data which is associated with streams and which is destined for the HLP/or Remote Station. Status Information blocks contain only stream information. These blocks are the conversation media used by the LCC and the remote batch to coordinate their activities.
Error information blocks also have two versions: repeat block and negative response block. The repeat blocks are sent by the LCC to the remote to inform the remote that it must retransmit a block. The negative response block is sent by the remote to the LCC to inform the LCC that the remote could not accept the last information block sent by the LCC. See figure 4.2-10 and 4.2-11 for a graphic representation of the N/REP blocks.
Figure 4.2-8

Figure 4.2-9

Figure 4.2-10 {See Section 4.2.2.1}

Figure 4.2-11

x = repeat block
y = repeat block

n = negative reply
4.2.2 Format of Blocks

Both types of blocks, information blocks and error information blocks, are made up of a combination of the following fields:

OD, CI, DF, ED, CRC

where:

OD = Opening Delimiter. All blocks must have an opening delimiter. This sequence of ASCII characters define the beginning of a transmission block.

CI = Control Information. All information blocks contain some station control information. Station control information is passed with stream identifiers and the current status of those streams.

DF = Data Framing control characters. Data information blocks have the data {i.e., text or routing information} framed with ASCII control characters.

ED = Ending Delimiter. All blocks must have an ending delimiter which signifies the end of a block.

CRC = Cyclic Redundancy Check. All blocks must have this field. It consists of two eight bit bytes.

NOTE: All ASCII control characters must follow standard ANSI rules concerning character parity {i.e. odd parity}. 
4.2.2.1 Opening Delimiters

Three forms of opening delimiters exist for blocks. There is a form for control blocks, a form for information blocks and a form for error information blocks.

Control block opening delimiter is:

\[ \text{DLE } Z \]

DLE: is the ASCII communication control character referred to as Data Link Escape.

Z: is the ASCII upper case character Z. This opening delimiter flags that this block is a control block.

Information block opening delimiters are:

\[ \text{DLE } XSN \]

DLE: is the ASCII communication control character referred to as Data Link Escape.

XSN: Transmission Sequence Number

\( x \) XSN: is the odd/even sequencing number defined in Section 4.2.1 for type information trades. The LCC transmits the graphic characters p and q. The Remote Station responds with the graphic characters 0 and 1. See Figure 4.2-9.

\( \beta \) XSN: is the odd/even sequencing number defined in Section 4.2.1 for type information trades. The LCC transmits the graphic characters 0 and 1. The remote station responds with the graphic characters p and q. See Figure 4.2-9.
Error information block opening delimiters are:

1. DLE REP
2. DLE N

DLE: is the ASCII communication control character referred to as Data Link Escape.

REP: is the ASCII graphic character X in type information trades and is y in type information trades. This opening delimiter flags that this block is a REPeat block. See Section 4.2.3, Error Recovery.

N: is the ASCII character N. This opening delimiter is used by the remote to flag the last message was not received correctly. See Section 4.2.3, Error Recovery.

4.2.2 Control Information

Station Control Information

Station control information is made up of 8-bit bytes referred to as SID (Stream IDentifier). Each SID (8-bit byte) contains the current status of a stream along with its identifier. A SID has the following 8-bit structure.

P1XXXXXY

where:

P: is the parity of the SID

1: is the flag that says this is a non-communication control character.

XXXX: is the stream number of a source/link path. Even numbered streams refer to output simplex paths from a source (i.e. buffer) in the HLP to a sink in the
remote station. Odd numbered streams refer to input simplex paths from a source in the remote to a sink \(\text{i.e., buffer}\) in the HLP. The maximum stream number is 15 \(\text{decimal}\), and the minimum stream number is zero \(\text{i.e., } 0 \leq \text{XXXX} \leq 15\).

\(\text{YY:}\) defines the current status of this stream. See Figure 4.2-12 for the meanings of these bits.
Control Block Information Code (CBC)

The control block information code is made up of 8-bit bytes. The code has the following 8-bit structure:

\[ \text{P} \text{XXXXXXX} \]

where:

- \( P \) - is the parity of the code.
- \( 1 \) - is the flag that says this is a non-communication control character.
- XXXXXX - is the actual code. The only defined code is 000001 = restart.

**WHEN LCC IS TRANSMITTING**

<table>
<thead>
<tr>
<th>INPUT STREAM (ODD NUMBER)</th>
<th>YY</th>
<th>OUTPUT STREAM (EVEN NUMBER)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTS</td>
<td>00</td>
<td>RTS</td>
</tr>
<tr>
<td>—</td>
<td>01</td>
<td>DATA</td>
</tr>
<tr>
<td>—</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>AS</td>
<td>11</td>
<td>AS'</td>
</tr>
</tbody>
</table>

**WHEN LCC IS RECEIVING**

<table>
<thead>
<tr>
<th>INPUT STREAM (ODD NUMBER)</th>
<th>YY</th>
<th>OUTPUT STREAM (EVEN NUMBER)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTS</td>
<td>00</td>
<td>CTS</td>
</tr>
<tr>
<td>DATA</td>
<td>01</td>
<td>BUSY</td>
</tr>
<tr>
<td>SOURCE IDLE</td>
<td>10</td>
<td>SINK IDLE</td>
</tr>
<tr>
<td>NA</td>
<td>11</td>
<td>NA</td>
</tr>
</tbody>
</table>

Figure 4.2-12
CTS = Clear to Send, the stream sink is conditioned and Ready to receive data from the source.

RTS = Request to Send, the stream source wants to send data to the sink.

DATA = The stream source is delivering the following data to the sink.

BUSY = The stream sink is unable to receive data, it is currently occupied with previous data.

IDLE = The source/Sink is idle, no activity is going on - however, it is not yet prepared to send/receive data.

N/A = Not available, the source/sink is not available for any activity. {Example of reason for N/A is the remote sink, such as a printer, is down.}.

A/S = Abort Stream, the LCC is terminating current operations on this stream. {Example of reason for this status is the HLP has gone down, or the HLP has terminated the stream, etc.}.

### 4.2.2.3 Data Framing

Data Information blocks are the only blocks which contain data framing characters. These characters precede the data. They define the characteristics of the following data characters. See figure 4.2-23 for a train graph of block formats. The data framing characters are the following ASCII Control Characters:
NOTES:

1. PATH 1 IS THE IDLE BLOCK
2. PATH 2 IS THE α STATUS BLOCK SENT BY THE
   LCC TO THE REMOTE, OR IT IS THE γ status BLOCK
   RESPONSE SENT BY THE REMOTE TO THE LCC.
3. PATH 3 IS THE β DATA BLOCK SENT BY THE LCC TO
   THE REMOTE, OR IT IS THE α DATA BLOCK RESPONSE
   SENT BY THE REMOTE TO THE LCC.
4. PATH 4 IS THE Repeat BLOCK SENT BY THE LCC.
5. PATH 5 IS THE NEGATIVE RESPONSE BLOCK SENT BY THE
   REMOTE TO THE LCC.
   XSN = SEQUENTIAL α or γ NUMBER (I.E. P, Q, R, 1)
   STAT = STREAM STATUS YY

Figure 4-13
Block Format Train
TSTX \{DLE, STX\} = Start of Transparent Text. This character sequence delimits the end of station control information and the start of transparent data \{text\} in a block. The following data characters are 8-bit bytes with no parity.

4.2.2.4 Ending Delimiters

All blocks must have an ending delimiter. The ending delimiters are the following ASCII control characters.

ETB = End of this transmission block. This character is used only on idle blocks or control blocks. It is never used with a data block.

TETB = End of transparent text for this transmission \{DLE,ETB\} block. However, more transparent text for this stream is pending \{i.e. an implied Request to Send\}.

TETX = End of transparent text for this transmission \{DLE,ETX\} block and for this message or file.
4.2.2.5 Cyclic Redundancy Check [CRC]

A maximum of two CRC bytes must be accumulated at both the sending and receiving stations during the transmission of a message to facilities error detection. They are transmitted following the ending delimiters of blocks.

**Function:**

1. The accumulation is started by the appearance of the first opening delimiter character of a block. The accumulation begins immediately with the opening DLE.
2. All characters, with the exception of those listed in 4 below, transmitted after the start of the accumulation are included in the accumulation, through and including the end-of-block or end-of-text delimiter. {ETB, TETB, or TETX}.
3. The CRC bytes are transmitted following the end-of block or end-of-text delimiter. The accumulation is reset and restarted with the appearance of the opening delimiter of the following transmission block.
4. Characters which are excluded from the accumulation are as follows:

a. SYN characters \{synchronization characters appearing after the opening delimiter and before the TSTX.\}

b. TSYN characters \{DLE, SYN\} appearing after the opening delimiter.

c. One DLE of DLE sequences that appear in transparent text. All ASCII control characters have character parity in bit 8. Transparent text uses bit 8 for data, thus allowing all 256 bit combinations in each byte. The Data Link Escape \{DLE\} character is used to introduce transparent text transmission and subsequently to distinguish control character bit configurations which may be present in the data stream. Once within the text \{i.e., subsequent to a DLE STX\}, control codes are recognized as such only when immediately preceded by a DLE code. To preclude data simulation of this sequence, the following procedure is followed: The transmitting station monitors the data stream and when the bits within a byte position duplicate the DLE bit configuration, it inserts an additional DLE character into the stream and follows it with the data DLE character. The receiving station interprets DLE, DLE character sequences to be representative of a single data byte with a bit configuration.
identical to that of the DLE character to summarize, transmitting terminals add DLE's and receiving terminals delete DLE's to preserve the integrity of legitimate control code sequences.

The generating polynomial \( G \{x\} \) used in the LCC cyclic encoder is:

\[ G \{x\} = x^{16} + x^{15} + x^2 + 1 \]

The remote station must use the same polynomial.

All \( n-k \) data and control bits included in the accumulation, along with the \( k \) redundancy bits transmitted after the data and control bits, form a code polynomial of the form - \( x^{n-1} + x^{n-2} + \ldots + x + 1 \) - that is evenly divisible by \( G \{x\} \).

With the LCC cyclic encoder unit, it is necessary to pass \( 16 \) binary zeros through the encoder after all data and control bits have been passed through in order to come up with the proper redundancy bits. It is important that a compatible implementation at a remote station take these properties of code polynomial into account.

4.2.2.6 **Synchronization**

The ASCII character used to maintain bit and character synchronization (meaning the line is sampled at regular intervals) is \(^{c}SYN\). In Mode 2 a minimum of two SYN characters must precede each transmission block. If the transmitting end must \(^{c}time-fill\) once it has sent the data framing characters, see Section 4.2.2.3; it must time-fill with TSYN \{DLE SYN\}. 
4.2.2.2 Specific Block Format

Each variation of block type has a unique combination of the above specified fields. See Figures 4.2-14, -15, -16 for examples of these combinations in an information flow.

A. Information Blocks

1. Data Blocks: OD, CI, DF, {DATA}, ED, CRC

   OD = DLE, XSN, XSN is sequential transmission number

   CI = One SID (only) with YY status equal to DATA

   DF = The data framing characters DLE, STX

   ED = The ending delimiter characters DLE, ETX, or DLE, ETB

   CRC

2. Status Blocks: OD, CI, ED, CRC

   OD = DLE, XSN, XSN is sequential transmission number

   CI = A variable number of SIDs

   Example: S S S S S S
            I I I I I I
            D D D D D D
            0 1 3 4 5 8
            RTS CTS CTS RTS IDLE RTS

   ED = ETB

   CRC

3. Idle Block: OD, ED, CRC

   OD = DLE, XSN, XSN is sequential transmission number

   ED = ETB

   CRC
B. Error Information Block

1. Repeat Blocks: OD, ED, CRC
   OD = DLE, x, y
   ED = ETB
   CRC

2. Negative Response Blocks: OD, ED, CRC
   OC = DLE, N
   ED = ETB
   CRC

C. Control Block

1. Restart Block: OD, CBC, ED, CRC
   OD = DLE, Z
   CBC = 11000001
   ED = ETB
   CRC
### Control Data Corporation Division

**Document Class:** ERS  
**Page No.:** 4-32

**Product Name:** LCC  
**Product Model No.:** 1039L-10  
**Machine Series:** PL10

<table>
<thead>
<tr>
<th>Status</th>
<th>Data</th>
<th>Status</th>
<th>Data</th>
<th>Status</th>
<th>Data</th>
<th>Status</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
</tr>
<tr>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
</tr>
<tr>
<td>E 1BC</td>
<td>E BC</td>
<td>E 1BC</td>
<td>E BC</td>
<td>E 1BC</td>
<td>E BC</td>
<td>E 1BC</td>
<td>E BC</td>
</tr>
<tr>
<td>CTS</td>
<td>RTS</td>
<td>CTS</td>
<td>RTS</td>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
</tr>
<tr>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
</tr>
<tr>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
</tr>
<tr>
<td>E 4BC</td>
<td>E 4BC</td>
<td>E 4BC</td>
<td>E 4BC</td>
<td>E 4BC</td>
<td>E 4BC</td>
<td>E 4BC</td>
<td>E 4BC</td>
</tr>
<tr>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
</tr>
</tbody>
</table>

Above Figure Shows Data, Status, Idle Blocks  
**Figure 4.2-14**

<table>
<thead>
<tr>
<th>Status</th>
<th>Data</th>
<th>Status</th>
<th>Data</th>
<th>Status</th>
<th>Data</th>
<th>Status</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
</tr>
<tr>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
</tr>
<tr>
<td>E 03BC</td>
<td>E BC</td>
<td>E 03BC</td>
<td>E BC</td>
<td>E 03BC</td>
<td>E BC</td>
<td>E 03BC</td>
<td>E BC</td>
</tr>
<tr>
<td>RTS</td>
<td>CTS</td>
<td>RTS</td>
<td>CTS</td>
<td>RTS</td>
<td>DATA</td>
<td>RTS</td>
<td>DATA</td>
</tr>
<tr>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
</tr>
<tr>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
</tr>
<tr>
<td>E 36X</td>
<td>E 36X</td>
<td>E 36X</td>
<td>E 36X</td>
<td>E 36X</td>
<td>E 36X</td>
<td>E 36X</td>
<td>E 36X</td>
</tr>
<tr>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
</tr>
</tbody>
</table>

Above Figure Shows Data, Status, Idle, Rep Blocks  
**Figure 4.2-15**

<table>
<thead>
<tr>
<th>Status</th>
<th>Data</th>
<th>Status</th>
<th>Data</th>
<th>Status</th>
<th>Data</th>
<th>Status</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
</tr>
<tr>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
</tr>
<tr>
<td>RTS</td>
<td>CTS</td>
<td>RTS</td>
<td>CTS</td>
<td>RTS</td>
<td>DATA</td>
<td>RTS</td>
<td>DATA</td>
</tr>
<tr>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
<td>S</td>
<td>D IEC</td>
</tr>
<tr>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
<td>LpDTR</td>
<td>LOTR</td>
</tr>
<tr>
<td>E 1BC</td>
<td>E 1BC</td>
<td>E 1BC</td>
<td>E 1BC</td>
<td>E 1BC</td>
<td>E 1BC</td>
<td>E 1BC</td>
<td>E 1BC</td>
</tr>
<tr>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
<td>CTS</td>
<td>DATA</td>
</tr>
</tbody>
</table>

Above Figure Shows Data, Status, Idle, N Blocks  
**Figure 4.2-16**

---

AA 3777  
PRINTED IN U.S.A.
4.2.3 Error Recovery

The LCC initiates a time out feature. After the completion of a block transmission, the LCC sets a time of N milliseconds (the value N is determined at assembly time). If the value N is reached before a valid response is received from the remote, the LCC initiates error recovery.

The remote station will receive LCC transmission blocks in one of the following three conditions:

a. Invalid - An invalid transmission block is one whose opening {DLE-XSN} and closing {ETB/ETX} delimiters were not recognized to be in proper form.

b. Valid and Incorrect - A valid and incorrect transmission block is one that is valid, but has character parity or CRC error.

c. Valid and Correct - A valid and correct transmission is one that is valid and has no character parity or CRC error.

Responses to transmission blocks depends on the condition and content of the block as shown in Figures 4.2-17, -18. Figures 4.2-19 through 4.2-27 graphically show information flow and error recovery.
<table>
<thead>
<tr>
<th>REMOTE TERMINAL RECEIVES</th>
<th>REMOTE TERMINAL RESPONDS WITH</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Valid and Correct Block with:</td>
<td>a. A new data or status block with an updated sequence number. The responding block number is the expected block number.</td>
</tr>
<tr>
<td>a. Data or Status Block in Proper Sequence</td>
<td>b. A repetition of the last correct data or status block, if the corresponding block number is not correct.</td>
</tr>
<tr>
<td>b. Data or Status Block in Improper Sequence</td>
<td>c. A repetition of the last correct data or status block.</td>
</tr>
<tr>
<td>c. Repeat Request Block (REP)</td>
<td></td>
</tr>
<tr>
<td>2. Valid and Incorrect Block</td>
<td>2. DLE, N, ETB, CRC</td>
</tr>
<tr>
<td>3. Invalid Block</td>
<td>3. No response</td>
</tr>
</tbody>
</table>

Figure 4.2-17
<table>
<thead>
<tr>
<th>LCC RECEIVES</th>
<th>LCC RESPONDS WITH</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Valid and Correct Block with:</td>
<td>a. A new data or status block</td>
</tr>
<tr>
<td></td>
<td>with an updated sequence number</td>
</tr>
<tr>
<td>a. Data or Status Block in Proper Sequence</td>
<td>b. A retransmission of the last correct data or</td>
</tr>
<tr>
<td></td>
<td>status block</td>
</tr>
<tr>
<td>b. Data or Status Block in Improper Sequence</td>
<td>c. A retransmission of the last correct data or</td>
</tr>
<tr>
<td></td>
<td>status block</td>
</tr>
<tr>
<td>c. Negative Response {N}</td>
<td>2. A Repeat block (DLE, REP, ETB, CRC)</td>
</tr>
<tr>
<td></td>
<td>3. A time out and the last correct data or status</td>
</tr>
<tr>
<td>2. Valid and Incorrect Block</td>
<td>block</td>
</tr>
<tr>
<td>3. Invalid block or No Response</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.2-18
Error Recovery Information Flow

Examples: Invalid or No Response Block With Time Out

Notes:
1. "Timed Out" means that the time allotted to transmit an α/β block has elapsed and a valid transmission has not occurred. The value of ΔT is established at assembly time.
2. "Lost in Time Out" is retransmitted.
FIGURE 4.2-21  Error Recovery Information Flow
4.2-22  Examples: $\alpha / \beta$ Valid and Incorrect Blocks With Repeat Response
Error Recovery Information Flow

Examples: $\alpha/\beta$

Valid and Incorrect Blocks With Negative Response

Figure 4.2-23

Figure 4.2-24
FIGURE 4.2-25  Error Recovery Information Flow

Example: Valid and Incorrect Block With Repeat Response and Invalid or No Response Block With Time Out
**FIGURE 4.2-26**  
Error Recovery Information Flow

Example: Valid and Incorrect Block With Negative Response and Out of Sequence
Error Recovery Information Flow

Example: Valid and Incorrect Blocks with Negative and Repeat Responses
4.3 Operation With HLP

The HLP activates a line {i.e., tells the LCC to service a line} by establishing a Request List and entering its address in the Line List. The HLP then sets the channel flag causing the LCC to search the Line List. The LCC activates all new entries in the Line List, clears the channel flag, and reports to the HLP that the line or lines are in an idle poll state and that the channel flag is cleared.

When a line is in an idle poll state the LCC periodically polls the remote station to see if it is ready to establish communications {i.e. trade blocks of information}. If the LCC does not get a response to the poll {i.e. a time out occurs}, the LCC will poll the remote station again. When the LCC receives the required response to a poll, it initiates information trades with the remote and scans the HLP lists for requests.

If the HLP wants to take that line out of an idle poll state it must issue an HLP to LCC request to close the line. This request must be accompanied by a channel flag, because the LCC does not reference the Request List for a line when the line is in an idle poll state.

Once communications with the terminal has been established, the LCC continually scans the Request List for that line.
Associated with each request is a command. Commands direct the LCC to perform some function. Commands for streams have only two allowed functions to be executed. An input stream (odd numbered stream) can read data and an output stream (even numbered stream) can write data. Input stream requests to read will be rejected by the LCC if an unsolicited RTS has not been received from the remote terminal. Stream commands also define data buffers.

For input streams, the buffers should be a multiple of the transmission block size defined for this stream request. As each full transmission block (transmission blocks are multiples of four bytes) for the requested stream is received from the remote, the LCC does the following checks:

1. If the block received is not a full block (the maximum number of bytes for the requested transmission block size and not necessarily a multiple of four bytes), and is terminated successfully with the byte count reflecting the number of bytes successfully received and the response status reflects that the ending delimiter was an ETB.

2. If the block received is a full block and is terminated with an ETB ending delimiter, the LCC checks the HLP buffer byte count to see if the remaining portions of the buffer is smaller than the block of bytes to be received from the remote. If there is sufficient room for another full block, the LCC continues to receive data. If there is not
sufficient room for another full block, the stream request is terminated successfully with the byte count reflecting the number of bytes successfully received, and the response status reflects that the ending delimiter was an ETB.

3. If the stream request defines a transmission block size, the value obtained via the block type index received in the stream request, that is larger than the HLP data buffer byte count, the stream request is rejected as abnormal. It is up to the HLP to insure the integrity of the transmission block size defined in the stream request. If the HLP specifies a block type of zero, the LCC inputs data into HLP buffer until it is full, i.e., only one block of data is expected, or until an ending delimiter is received. See Section 4.3.3 for an explanation of transmission block size and Section 4.1.3.2 for an explanation of blocking of data.

4. If a transmission block is received from the remote which is terminated with an ETX {denotes the last block of the message} the stream request is terminated successfully. The byte received successfully {byte count can be zero, if a transmission block is received with no data bytes} and the response status reflects that an ETX was received.

5. If an HLP data buffer is terminated with an ETB, the request is terminated successfully with the byte count reflecting the number of bytes successfully received. The response, also, implies that the remote has more data to send {i.e., implied unsolicited response of a RTS}. The HLP must issue
another read request for this stream. This implied RTS is also true for the conditions stated above in 1 and 2.

6. If the LCC subsystem receives a data information block from the remote, it accepts the block and increments the HLP byte count by zero. Thus, if the block were terminated with a TETX, the LCC could terminate a read request successfully with a byte count of zero.

For output stream requests the LCC subsystem does the following:

1. The LCC transmits the HLP data in as many even multiples of the transmission block size as possible. Each transmission block is transmitted with the ending delimiter TETB. The last block of data in the HLP buffer is transmitted with the stream requested ending delimiter TETB or TETX.

2. The LCC subsystem software transmits the last block from an HLP data buffer "even" if it is not a full transmission block or an even multiple of four bytes. The intermediate blocks transmitted from a buffer must be multiples of four bytes.

3. A stream request defining an HLP data buffer byte count of zero is treated as a proper request, a data information block is transmitted with no data and the request is successfully terminated with a byte count of zero. See Section 3.2.3.

4. If the HLP specifies a block type of zero, the LCC outputs data from the HLP data buffer in one transmission block.
It is up to the HLP to verify that the remote terminal will accept a transmission block of an indeterminate length.

5. If the HLP issues a write with ETB request, the LCC subsystem transmits all blocks with the ending delimiter TETB. The receipt of this by the remote terminal ending delimiter infers that more data is coming, i.e., the file is not complete. Thus it is important that the HLP issue consecutive write with ETB requests until the end of the file or write with TETX.

6. The LCC subsystem does not terminate a stream write with ETB request until the remote terminal replies {in a status transmission block} with a CTS status. Thus, the remote is ready to receive more data. {See Figure 5.2-12.}

7. The LCC subsystem does not terminate a stream write with ETX request until the remote terminal replies with an IDLE status. This implies that the remote terminal has successfully output the last block of data.
4.3.1 Request Entry Format: HLP to LCC

The following is the expansion of the first entry slot in the request list as shown in the second box of Figure 3.1-4.

```
Word 1
00 15 16 17 26 27 28 31
| DESTINATION FIELD | S | I |
```

```
Word 2
00 31
| BASE ADDRESS OF COMMAND |
```

Bits 17-26 and 28-31 should be filled with zeroes.

<table>
<thead>
<tr>
<th>WORD</th>
<th>BITS</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>00-15</td>
<td>Destination Field: Supplied by the HLP and is returned in the solicited response.</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>S Field: Status of this request slot:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>°0° - This request is active or has been completed. The LCC clears this bit when it initiates this request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>°1° - This request is ready for initiation. The HLP sets this bit when it enters a new request.</td>
</tr>
<tr>
<td>27</td>
<td>I Field: Interrupt and Response Options</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0 - Do not send interrupt with request response.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1 - Send interrupt with request response.</td>
<td></td>
</tr>
</tbody>
</table>
### 4.3.2 Request Entry Format: Stream

The following is an expansion of the other entry slots used for streams as shown in the third box of Figure 3.1-4.

<table>
<thead>
<tr>
<th>Word</th>
<th>Bits</th>
<th>Description</th>
</tr>
</thead>
</table>
| 1    | UD-15| **Destination Field**: Required in all requests.  
It is returned in the solicited response.  
This field is free for HLP use. |
| 16   | S | **S Field**: Status of this request slot.  
°0° - This request is active or has been  
completed. The LCC clears this bit  
when it initiates this request.  
°1° - This request is ready for initiation.  
The HLP sets this bit when it enters  
a new request. |
| 1    | P | **P Field**: Priority Request  
°0° - Normal priority  
°1° - Preemptive priority |
<table>
<thead>
<tr>
<th>WORD</th>
<th>BITS</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>24-27</td>
<td>I Field: Interrupt and Response Options</td>
<td></td>
</tr>
<tr>
<td></td>
<td>00XX - Do not send response or interrupt when request is initiated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>01XX - Not valid bit combination.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>10XX - Send response when request is initiated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>11XX - Send response and interrupt when request is initiated.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XX00 - Do not send interrupt when request is completed.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XX10 - Send interrupt if request terminates abnormally.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XX01 - Not valid bit combination.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XX11 - Send interrupt when request is completed or terminated.</td>
<td></td>
</tr>
</tbody>
</table>
4.3.3 Command Format: Stream

<table>
<thead>
<tr>
<th>Word</th>
<th>Bits</th>
<th>Function Code (Hex)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>00-07</td>
<td>00</td>
<td>Read - The LCC stores the data it receives from the line and stream (defined in the request) into the data buffer.</td>
</tr>
<tr>
<td></td>
<td>0A-15</td>
<td></td>
<td>B Field - An index for a table containing various block byte counts. The index may range from 1-7. The index must point to the proper entry in the table describing the block size (block sizes must be multiples of 4 bytes) for this particular stream. The entries in the table are defined at assembly time. If the B field is zero, no blocking is associated with the request and the data buffer byte count is used as the blocking factor.</td>
</tr>
<tr>
<td>80</td>
<td></td>
<td></td>
<td>Write transparent text with ETB.</td>
</tr>
<tr>
<td>81</td>
<td></td>
<td></td>
<td>Write transparent text with EOM (i.e. ETX).</td>
</tr>
<tr>
<td>WORD</td>
<td>BITS</td>
<td>FUNCTION CODE (HEX)</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>16-31</td>
<td></td>
<td>Byte Count</td>
<td>This field specifies the number of 8-bit data bytes to be processed (read or write) from the data buffer. The byte count must be less than 7FFF hex.</td>
</tr>
</tbody>
</table>
4.3.4 Command Format: HLP to LCC

<table>
<thead>
<tr>
<th>WORD</th>
<th>BITS</th>
<th>FUNCTION CODE (HEX)</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>00-07</td>
<td>FL</td>
<td>Close a Communication Line</td>
</tr>
</tbody>
</table>

This function causes the LCC to terminate all outstanding stream requests to this line. Since this is a requested action when the stream requests are terminated, their responses will reflect an abnormal termination. Before the LCC writes the completion response to this request it scans each entry in the Line Request Queue. If there is a new request it is a terminated abnormally. When all the entries have been scanned all references to this line are eliminated, the line is considered to be inactive.
<table>
<thead>
<tr>
<th>WORD</th>
<th>BITS</th>
<th>CODE [HEX]</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>F2</td>
<td></td>
<td></td>
<td><strong>Terminate Stream Request</strong></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>A normal response is given for</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>successful completion of this request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The LCC can be in one of three</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>conditions with the defined stream when</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>a Terminate Stream request is issued.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1). For the defined stream the LCC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>has processed and is executing a</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>stream request. The LCC will ter-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>minate that request, it's response</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>will reflect an abnormal</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>termination. The byte count in the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>response will reflect the number of</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>bytes successfully received/trans-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>mitted.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2). For the defined stream the LCC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>does not have a stream request in the</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>core. However, internal flags infer</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>that implied RTS or CTS have been</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>traded with the remote terminal. The</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>LCC clears all internal flags. No</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>special stream response is written.</td>
</tr>
<tr>
<td>WORD</td>
<td>BITS</td>
<td>FUNCTION CODE {HEX}</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>1</td>
<td>00-07</td>
<td>F4</td>
<td>3). For the defined stream the LCC does not have a stream request in the LCC core and has no internal flags. In all the three cases, the LCC transmits an abort stream SID to the remote terminal, See Section 4.2.2.2. Field A = input or output stream number. Transfer the LCC storage to the HLP. Field A defines the beginning LCC address to be transferred. Field A1 defines the number of LCC words to be transferred. This number must be in increments of four words. Field B defines the 32-bit base address in the HLP. Configuration definition for line X. This request should be issued at Open Line time only. If it is issued at other times, undetermined results may occur. This request defines the time-out type {bits 13,14,15 of Field B} and the maximum number of</td>
</tr>
<tr>
<td>WORD</td>
<td>BITS</td>
<td>FUNCTION CODE (HEX)</td>
<td>DESCRIPTION</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>streams for this line {bits 27, 28, 29, 30, 31}. The time-out type is an index similar to that defined for the B {block-type} Field in a stream request. The index must point to the proper entry in the table describing the terminal supported on this line. The index values are the following: Index 0 = 733-10, CDC CYBER High Speed Batch Terminal. Indexes 1-7, not defined. Data Set Adapter Loop Request. This request instructs the LCC to loop the Data Set Adapter, i.e., the data character in the request is transmitted 256 times out of the LCC and is immediately returned on the input side of the communication line. No data is transmitted to the remote terminal. The number of unsuccessful receipts is placed into the</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fb</td>
<td></td>
</tr>
</tbody>
</table>
termination response which is written in the HLP.

This request can be issued at open line time only. Upon completion of the test, the line is considered closed. The operation defined above is the same operation initiated by the LCC when the link reaches the maximum number of error recovery attempts and exercises (loops) the line. Bits 24 through 31 of field B contain the character to be involved in the loop-back operation.

Data Set Loop request. This request is similar to the one defined above, however it has to be issued to the AT&T 303 Modem Series, or to any other modem which conforms to the same loop function possible in the AT&T 303 modems. It is suggested that this request not be issued on a line which does not provide this function.
**NOTE:** All commands which are starred (*) have the solicited response written in the HLP when the command is initiated. When the command is totally satisfied, an unsolicited response is written in the HLP, notifying it that the operation is complete (i.e., the line is closed, or the system is suspended, etc.)

4.3.5 Response Formats

Type 1 Normal Unsolicited Response

<table>
<thead>
<tr>
<th>Stream #*, if Applies</th>
<th>Line #</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Response Code</td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>
Normal Solicited Response to HLP-LCC Requests and to illegally formatted requests (HLP/LCC and Stream)

Type 2

<table>
<thead>
<tr>
<th>Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-15</td>
</tr>
<tr>
<td>Destination Field</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>Response Code</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>Base Address of Command</td>
</tr>
</tbody>
</table>

Type 3

Normal Solicited Response to a Stream Request

<table>
<thead>
<tr>
<th>Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-15</td>
</tr>
<tr>
<td>Destination Field</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>L G</td>
</tr>
<tr>
<td>Response Code</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>Base Address of Command</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>REXMIT</td>
</tr>
<tr>
<td>COUNT</td>
</tr>
<tr>
<td>LINK</td>
</tr>
<tr>
<td>STATUS</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>BYTE Count</td>
</tr>
</tbody>
</table>
Type 4 Abnormal Unsolicited Response

<table>
<thead>
<tr>
<th>Line No.</th>
<th>Response Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>L</td>
</tr>
<tr>
<td>1</td>
<td>G</td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Field A</td>
</tr>
<tr>
<td>7</td>
<td>Field B</td>
</tr>
<tr>
<td></td>
<td>Field C</td>
</tr>
<tr>
<td></td>
<td>Field D</td>
</tr>
</tbody>
</table>

Type 4 Field Meanings

There are two Type 4 responses. They are response 44 {link has errored out} and 45 {intermittent SAC problems}. Response 44 uses fields B, C, D.

**Field B = Link Status = 8 bits;** B, S, C, NE, N, NA, SAC, TO
- B = illegal data SID
- S = out of sequence block
- C = CRC or character parity error
- NE = no ending delimiter
- N = NAK was received
- NA = Not Assigned
- SAC = had intermittent SAC problem
- TO = link time out

**Field C = DSA ERROR RESULTS = Data Set loop exercise results**
- = 0, All attempts were successful
- ≠ 0, Number of unsuccessful attempts
Field D = Carrier On/Off Status

- 00  Carrier is on and data set is ready
- 01  Carrier was on but data set not ready
- 02  Carrier was not on, data set ready
- 03  Neither carrier on or data set ready

Response 45 uses Field A.

Field A = Number of intermittent SAC rejects in this pass.

Type 3 Field Meanings

REXMIT Count = Number of error recovery attempts on the last block.

Byte Count = Number of 8-bit bytes received/stored in data buffer.

RESPONSE CODE DEFINITIONS

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>LG</td>
<td>INT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Response Code Word

LG = Log

INT = Interrupt

UNSOl = Unsolicited Response

Bits 8 to 15 = the response code specifying the status of the operation.
<table>
<thead>
<tr>
<th>Code</th>
<th>Type</th>
<th>Associated Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>00, 81</td>
<td>HLP solicited responses on initiation of a stream request.</td>
</tr>
<tr>
<td>1-20</td>
<td></td>
<td></td>
<td>Not assigned.</td>
</tr>
<tr>
<td>21</td>
<td>3</td>
<td>80</td>
<td>Write stream request {function code=80} terminated normally with an End of Block, another buffer of data is required.</td>
</tr>
<tr>
<td>22</td>
<td>3</td>
<td>00, 81</td>
<td>Stream request {read or write} terminated normally with an End of Message {ETX}, this file is completed.</td>
</tr>
<tr>
<td>23</td>
<td>2</td>
<td>F5</td>
<td>The HLP request to configure the line is completed.</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>F2</td>
<td>The HLP request to terminate an active stream request has been initiated. Response code 28 will eventually be given for this stream.</td>
</tr>
<tr>
<td>25</td>
<td>2</td>
<td>F4</td>
<td>Request to transfer LCC memory into the HLP is complete.</td>
</tr>
<tr>
<td>26</td>
<td>2</td>
<td>F1</td>
<td>Close line request is initiated.</td>
</tr>
<tr>
<td>27</td>
<td>2</td>
<td>F2</td>
<td>The request to terminate an inactive {LCC is not currently processing a stream request defined by the HLP request to terminate} stream request is complete and successful.</td>
</tr>
<tr>
<td>28</td>
<td>3</td>
<td>80, 81, 0</td>
<td>Stream request terminated abnormally.</td>
</tr>
<tr>
<td>29</td>
<td>2</td>
<td>All</td>
<td>Request rejected, illegal parameter.</td>
</tr>
<tr>
<td>Code</td>
<td>Type</td>
<td>Associated Command</td>
<td>Meaning</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td>2A</td>
<td>3</td>
<td>80, 81, 0</td>
<td>System is at buffer threshold, this request is rejected.</td>
</tr>
<tr>
<td>2B</td>
<td>4</td>
<td>F6</td>
<td>Request to initiate DSA loop is completed. The format of this response is the same as the abnormal unsolicited response number 44, except the base address of the command is in words 4 and 5.</td>
</tr>
<tr>
<td>2C</td>
<td>4</td>
<td>F7</td>
<td>Request to initiate Data Set loop is completed. The format of this response is the same as the abnormal unsolicited response number 44, except the base address of the command is in words 4 and 5.</td>
</tr>
<tr>
<td>2D</td>
<td></td>
<td></td>
<td>Not assigned.</td>
</tr>
<tr>
<td>2E</td>
<td>2</td>
<td>F3</td>
<td>The HLP request to suspend the system has been initiated.</td>
</tr>
<tr>
<td>2F</td>
<td></td>
<td></td>
<td>Not assigned.</td>
</tr>
</tbody>
</table>

**UNSOULICITED RESPONSE**

<table>
<thead>
<tr>
<th>Code</th>
<th>Type</th>
<th>Associated Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>1</td>
<td></td>
<td>Stream has a RTS.</td>
</tr>
<tr>
<td>41</td>
<td>1</td>
<td></td>
<td>Stream has an Idle status.</td>
</tr>
<tr>
<td>42</td>
<td>1</td>
<td></td>
<td>Stream has Not Available status.</td>
</tr>
<tr>
<td>43</td>
<td>1</td>
<td></td>
<td>Stream has a Busy status.</td>
</tr>
<tr>
<td>44</td>
<td>1</td>
<td></td>
<td>Link has errored out, this line as logically closing, all outstanding requests are being terminated abnormally. HLP must issue a close line request.</td>
</tr>
<tr>
<td>Code</td>
<td>Type</td>
<td>Associated Command</td>
<td>Meaning</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---------------------</td>
<td>---------</td>
</tr>
<tr>
<td>45</td>
<td>1</td>
<td></td>
<td>System had intermittent SAC rejects.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Field A of response contains the current count.</td>
</tr>
<tr>
<td>46</td>
<td>1</td>
<td></td>
<td>System buffers are saturated; this line is logically closed, HLP must issue close line.</td>
</tr>
<tr>
<td>47</td>
<td>1</td>
<td></td>
<td>Illegal SID in status block.</td>
</tr>
<tr>
<td>48</td>
<td>1</td>
<td></td>
<td>Had a disastrous DSA malfunction; this line is logically closed, HLP must issue close line.</td>
</tr>
<tr>
<td>49</td>
<td>1</td>
<td></td>
<td>Channel flag is cleared.</td>
</tr>
<tr>
<td>4A</td>
<td></td>
<td></td>
<td>Not assigned.</td>
</tr>
<tr>
<td>4B</td>
<td></td>
<td></td>
<td>Not assigned.</td>
</tr>
<tr>
<td>4C</td>
<td>1</td>
<td></td>
<td>Remote terminal has initiated restart {i.e., it is rewinding its files}. This line is logically closed, HLP must issue close line.</td>
</tr>
<tr>
<td>4D</td>
<td>1</td>
<td></td>
<td>Unsolicited response; the system is in a suspended state.</td>
</tr>
<tr>
<td>4E</td>
<td>1</td>
<td></td>
<td>Unsolicited response; the line is closed.</td>
</tr>
<tr>
<td>4F</td>
<td>1</td>
<td></td>
<td>Unsolicited response; the line is in an idle poll state.</td>
</tr>
<tr>
<td>50-60</td>
<td></td>
<td></td>
<td>Not assigned</td>
</tr>
</tbody>
</table>

**BOTH SOLICITED AND UNSOLICITED RESPONSES**

| 61   | 3    |                     | Read stream request has terminated normally, it has an unsolicited RTS with it. |

**Means logged**
4.4 Overview of Mode 2

Although it is not within the scope of this document to describe the precise nature of the HLP and terminal software, this section is included to give an understanding of the functions performed and the general flow of the operations.

The major functions of the HLP can be summarized as follows:
1. LCC Subsystem autoload - cause the LCC software to be established in the LCC memory.
2. Set up the Line Table in the HLP memory.
3. Format Request and Command Table entries in the HLP memory to direct LCC activities.
4. Scan the Response Table for completed operations.
5. Manage the allocation of data buffers in HLP memory.
6. Control the SAC interface.
7. Coordinate activities with remote batch terminal.
8. Data manipulation functions
   a. Code compressions/decompression
   b. Escape code processing
9. Interfacing with application programs.

On the other side of the LCC, the remote batch terminals (RBT) provides user access to the system. Its major functions can be summarized as follows:
1. Maintaining communication with the LCC.
2. Inputting data from cards
   a. Code conversion
b. Code compression

3. Outputting data to printers/punches
   a. Code conversion
   b. Code decompression
   c. Forms control

4. Controlling data to/from the CRT/keyboard.
5. Coordinating activities with the HLP.

b. Buffering the transmission blocks exchanged with the LCC until acknowledged by the LCC.

Establishment of the communication system begins by autoloading the LCC. A 64 word autoload/auto dump bootstrap is sent across the CDC compatible trunk from the HLP memory to the LCC memory. This bootstrap program then loads an initializer program from the HLP memory. The initializer program in turn loads the standard software program and constructs the necessary tables in the LCC. At this point, the LCC has begun normal operation and is waiting for a channel flag from the HLP (no communication has started; the LCC is in a ready state).

Some point in time after the autoload, the decision will be reached in the HLP that communication should be established with a terminal or terminals. The HLP must first build a command configuring each terminal, then a Request List is constructed for each terminal to be activated. The first request points to the configuring command. The location of the Request
List is entered into the appropriate slot in the Line List. The Channel Flag is then set and the LCC goes into action. The Line List is searched for all new entries and as each is found, the line is considered opened {although communication has not yet been established with the terminal}. The LCC enters each new line into an Idle Poll state such that Restart control blocks are repeated continuously until answered by the remote terminal.

At the time the HLP opens a line to the LCC, the condition of the remote batch terminal is unknown. The remote operator may have autoloaded the RBT prior to the HLP opening the line. In this case, the RBT will be in a ready state when the first Idle Poll is sent by the LCC and communication will be established immediately.

A second condition exists when the RBT is not ready when the line is opened {i.e. off or in an off line mode}. The LCC will continue in the Idle Poll state until the remote replies or is told to close the line by the HLP. The remote operator may autoload the RBT into the ready state when he desires to communicate with the HLP.

The Idle Poll state in the LCC terminates when a Restart control block is received from the RBT. The LCC then changes from Idle Poll to a state of normal communications {α/β sequences}. 
During normal communications, the LCC continually scans the Request List looking for work.

Once communication has been established with the terminal, the HLP may write data to the terminal and the RBT may send data to the HLP.

At the terminal when the card reader becomes active, a RTS is sent to the LCC. The LCC in turn notifies the HLP that the RBT wishes to send data on a particular stream via an unsolicited response. The HLP should then establish a data buffer [potentially several hundred bytes long] describe it with a read command and then points to the command with a request in the Request List. The LCC then decodes the request and command and returns a CTS to the terminal for that data stream. The terminal is then free to send one block of data to the LCC. If after putting the data into the HLP buffer, the buffer is not full, another CTS is automatically returned to the terminal.

When the buffer is full, the request is completed and a response written. If the LCC has not received an ETX indicating end of message, the response will indicate that another buffer and read command are required.

During the process of inputting data, the HLP may find the need to send data to the terminal to be printed. The HLP would allocate a convenient sized buffer, fill it with some of the data to be printed, describe it with a write command and point
to the command with a request. The LCC decodes the request/command and sends a RTS to the terminal. The terminal in turn determines that the device is operable and that buffer space for one block of data is available. Given that these conditions are met, a CTS is returned to the LCC. The LCC then responds with the first data block. This sequence of the terminal guaranteeing buffer space and the LCC responding with data segments continues until the HLP buffer is empty. The LCC then writes a response to the HLP indicating that the request is complete. The HLP can then allocate the next data buffer describing it with a command and request. The previously described sequence would again be repeated. When the HLP allocates the last buffer of the message, it should instruct the LCC to identify the last block as the end of message (EOM).

The HLP may wish to improve the response time between requests. To accomplish this, the HLP may build a new request/command sequence as soon as it determines that the "$" bit in the current request has been set to 0. The new request will be picked up by the LCC automatically when the current request is complete.

During the life of a request, the HLP may wish to terminate the data flow on a particular stream. To accomplish this, the HLP would issue a terminate command to the LCC. The LCC in turn would send an Abort Stream code in the SID to the terminal informing it of the HLP action. Upon completion of this
action, the appropriate responses are written to the HLP.

Having completed the exchange of data with the HLP the remote operator may wish to enter the terminal into an off-line mode. He would enter a command at the keyboard to the HLP instructing the HLP to relinquish control of the terminal. The HLP would then respond by instructing the RBT to go off line, followed by a close line command to the LCC. The LCC would cease communicating with the terminal and the remote operator would then be free to perform off line functions.
5.0 MODE 3 INTERACTIVE TTY PROCEDURES

Mode 3 of the LCC is intended to support Model 33/35 teletypes [ASR and KSR] and CDC's 713 terminal. These terminals are collectively referred to as TTY in the remainder of this section. In this implementation, asynchronous lines of speeds 110-300 bps are driven point-to-point, two-way alternate and may be either switched or non-switched. Teletype transmissions consist of a stream of data and format characters [e.g. carriage return]. There are no communications control characters framing a TTY message. This stream is passed between the TTY and HLP without being examined by the LCC software except as noted in Section 5.1.2.

5.1 General Description of Mode 3 [TTY]

5.1.1 Configuration

The Interactive Teletype software will support a strictly TTY configuration or operate in conjunction with Mode 2 software in supporting mixed TTY and remote batch configurations.

Mode 3 [TTY] terminals attach to lines of speed 110-300 bps. The lines are operated point-to-point, two-way alternate and may be either switched or non-switched. Note that Mode 3 [TTY] terminals operate in the half duplex mode and require a direct printing keyboard.
The Mode 3 LCC and subsystem will support 48 TTY's on lines up to 300 bps. For mixed configurations involving batch terminals, refer to Section 3.1.1.

The amount of core required for a given Mode 3 configuration may be calculated by the following formula:

$$\text{Mode 3} = 1750 + 36L_3 + 24AL_3$$

1750 = fixed Mode 3 program size

$L_3$ = number of Mode 3 lines

$AL_3$ = number of active Mode 3 lines

Applications of the above formula shows that:

- a 4K LCC will support 4 TTY lines
- a 8K LCC will support 48 TTY lines

5.1.2 Mode 3 Features

5.1.2.1 Communications Channels

Mode 3 supports 110-300 bps asynchronous communications channels in a half duplex manner. The channels may be either switched or non-switched.

Each line must be terminated at the LCC by a modem which conforms to the RS232 interface standard. For switched network operation, the modem may or may not be strapped for automatic answer. If the auto answer option is present, the LCC will automatically answer the call, but if auto answer is not present, manual intervention is necessary to place the modem in data mode.
At the conclusion of a call (switched network) the LCC will automatically disconnect the line by dropping the Data Terminal Ready lead to the associated modem. If the users local common carrier exchange operates on the called-party-hold principle, special operator action is necessary to ensure that the line is disconnected.

5.1.2.2 Data Dependencies

In general, the LCC V.2.0 is not sensitive to the contents of the TTY data stream except as specified in Table 5.1.

During an input operation, the LCC packs successive characters into the HLP buffer which was previously defined by a Request/Command sequence. The format of the buffer is as follows:

<table>
<thead>
<tr>
<th>16-bit word</th>
</tr>
</thead>
<tbody>
<tr>
<td>C 1</td>
</tr>
<tr>
<td>C 2</td>
</tr>
</tbody>
</table>

{C_i represent 8-bit characters}

The first character received is placed into the upper byte of the buffer word and the second character is placed into the lower byte. Each successive pair of characters is similarly packed into successive words of the buffer. Should the buffer be smaller than the incoming message {i.e. the buffer fills before the end of message character is detected} an error response is returned to the HLP and all characters of the incoming message segment are lost.
Each input message is subject to a lost data check. Should this condition occur, the contents of the input buffer are erased and the following message is sent to the TTY:

\[
\text{\{CR\} \{LF\} \{LF\} \{LF\} REPEAT LINE \{LF\} \{CR\} \{LF\} \{LF\}}
\]

\[
\text{\{CR\} = Carriage return}
\]

\[
\text{\{LF\} = Line Feed}
\]

The TTY operator is expected to re-enter the message.

The output operation is similar to the input operation, but in reverse. The LCC software does not form a parity bit for each character. The HLP provides the bit if needed.

In either case (input or output) the data is passed between the buffer in the HLP and the TTY without conversion or other manipulations upon it on the part of the LCC software except as noted below. The data stream is assumed to consist of a series of 8-bit characters from the ASCII code set \{USASI X3.4-1968\}.

The following table indicates the characters within the TTY input data stream that the LCC software recognizes and responds to:
<table>
<thead>
<tr>
<th>Code Detected in Input Stream</th>
<th>Operation Performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>LF {Line Feed}</td>
<td>When the LCC detects LF in the input stream, it sends a response to the HLP which indicates termination with LF. This response may be interpreted by the HLP as it sees fit {e.g. the LF may be used by the processing application to denote that a segment of a larger message has been entered, but that the message is not yet complete}. The LF is not placed into the HLP buffer. The LCC sends a carriage return to the TTY following a message that terminated with LF. This serves to position the carriage and to inform the operator of the TTY that more data can be entered. When the LCC detects CR in the input stream, it sends a response to the HLP which indicates termination with CR. This response may be interpreted by the HLP as it sees fit {e.g. the CR may be used to indicate that the final segment of a message has been entered}. The CR is not placed into the HLP buffer. The LCC sends a LF to the TTY following a message that terminated with CR.</td>
</tr>
</tbody>
</table>

**TABLE 5.1**
<table>
<thead>
<tr>
<th>Code Detected in Input Stream</th>
<th>Operation Performed</th>
</tr>
</thead>
<tbody>
<tr>
<td>NULL or DEL</td>
<td>NULL or DEL characters are ignored by the LCC. They are not placed into the HLP buffer.</td>
</tr>
<tr>
<td>X-OFF or EOT</td>
<td>Keyboard input: treated the same as for CR except that no LF is sent to the TTY. Paper tape input: A. Used as message delimiters - X-OFF and EOT are assumed to terminate a message if either is encountered after a string of data characters with no intermediate LF or CR. The LCC sends a solicited response to the HLP which indicates termination with CR. This response may be interpreted by the HLP as it sees fit (e.g. the X-OFF or EOT may be used to indicate that the final segment of a message has been received).</td>
</tr>
</tbody>
</table>
The X-OFF or EOT is not placed into the HLP buffer.

In addition to the above processing, the LCC sets up a .2 second timeout at the time the X-OFF or EOT is encountered.

If no input is received within the .2 second, it is assumed that tape motion has stopped and an unsolicited response which indicates X-OFF or EOT is sent to the response list. DEL/NULL characters received during the timeout interval cause the timeout to be reset. When it is determined that tape motion has stopped, the LCC places the line in the idle state and is prepared to accept further input from the keyboard. READTI must be issued to restart the tape reader.

B. Used as single control characters - if X-OFF and EOT are not encountered immediately after a string of data characters, they are assumed to be control characters. No solicited response indicating termination with CR is sent to the HLP. The rest of the processing (i.e. they are not put into the HLP buffer, and the timeout processing) is the same as if they were message delimiters.

TABLE 5.1 Continued
Any Input During An Output Operation

Whenever input is detected during an output operation, the LCC sends a response to the HLP which indicates that the remote operator has interrupted the output cycle.

The interrupted output is terminated by the LCC. This feature is useful when a long file is being transmitted in relatively short segments. The "interrupt" may be interpreted by the HLP as a request to discontinue or delay present output.

The LCC software does not inspect the TTY output data stream; therefore, the HLP software may insert as many carriage return {CR} and line feed {LF} characters as necessary for format purposes.

5.1.2.3 Automatic Answering

In order for the HLP to know when an open TTY line becomes active, the LCC issues a response to the HLP when Data Set Ready is detected.
5.2 Interactive TTY Procedure

The interactive TTY procedure implemented for Mode 3 consists of an implicit I/O scheme built around the concepts of unsolicited input and preemptive output. That is, the LCC is automatically prepared to accept data from a TTY line upon occurrence of Data Set Ready status from the associated modem. This occurs whether or not the HLP has issued a command which defines the receiving buffer in the HLP.

Whenever the LCC is not actively transmitting/receiving data over a TTY line, that line is in an unsolicited input mode. This can be considered as the normal idle state of an opened TTY line. Any input which occurs on a TTY line while it is idle is assumed to originate from the keyboard. When data begins appearing on a line, the line is no longer idle. If it has not already done so, the HLP is requested to issue a READ command to define the buffer area. The request from the LCC is via a response inserted in the response list. Lost data will occur if the LCC does not detect the READ when 9 bytes are input.

Commands for output are honored whenever the TTY line is idle. In this instance, the output preempts the idle state of the line. The line returns to the idle state when the output is complete.

A command for output received while the line is not idle (i.e. an input is in progress) is held until the line once again becomes idle.
5.3 Interface with the HLP

The HLP activates a line (i.e., tells the LCC to service a line) by setting the Channel Flag, and by setting the line tables in a predefined manner (see Section 3.1.4.2 for further details). The LCC activates all new entries in the line table, clears the Channel Flag, and reports to the HLP that the line or lines are activated.

A Teletype has two streams defined. There is a single input stream (from either paper tape or the keyboard) and a single output stream (to either paper tape or the printer). TTY streams are two-way alternate in nature; thus, input and output cannot occur simultaneously.

There are three request entries associated with TTY. Stream assignments for TTY are as follows:

<table>
<thead>
<tr>
<th>HLP/LCC Request</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stream 0:</td>
</tr>
<tr>
<td>TTY Printer or Paper Tape Punch</td>
</tr>
<tr>
<td>Stream 1:</td>
</tr>
<tr>
<td>TTY Keyboard or Paper Tape Reader</td>
</tr>
</tbody>
</table>

5.3.1 Request Entry Format: HLP to LCC

<table>
<thead>
<tr>
<th>Word 1</th>
<th>00</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>27</th>
<th>28</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>S</td>
<td>I</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Word 2</th>
<th>00</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

32-Bit Address of Command
Bits 17-26 and 28-31 should be loaded with zeroes.

<table>
<thead>
<tr>
<th>WORD</th>
<th>BIT(S)</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>00-15</td>
<td>Destination Field: Supplied by the HLP and is returned in solicited responses.</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>S Field: Status of this request slot.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;0&quot; - This request is active or has been completed. The LCC clears this bit when it initiates this request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;1&quot; - This request is ready for initiation. The HLP sets this bit when it enters a new request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>I Field: Interrupt and Response Options</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0 - Do not send interrupt with request response.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 - Send interrupt with request response.</td>
</tr>
</tbody>
</table>

5.3.2 Command Format: HLP to LCC

<table>
<thead>
<tr>
<th>Word 1</th>
<th>Function Code</th>
<th>A1</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>00</td>
<td>07</td>
<td>08</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Word 2</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>31</td>
</tr>
<tr>
<td>WORD</td>
<td>BIT(S)</td>
</tr>
<tr>
<td>------</td>
<td>--------</td>
</tr>
<tr>
<td>1</td>
<td>00-07</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Dump - Transfer the LCC storage to the HLP. Field A defines the beginning LCC address to be transferred. Field A defines the number of LCC words to be transferred. This number must be in increments of four words. Field B defines the 32-bit base address in the HLP.

* NOTE: All commands which are starred {*} have the solicited response written in the HLP when the request is initiated. When the command is totally satisfied, an unsolicited response is written in the HLP, notifying it that the operation is complete {e.g., the line is closed}.

5.3.3 Request Entry Format: Stream

<table>
<thead>
<tr>
<th>Word 1</th>
<th>00</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>23</th>
<th>24</th>
<th>27</th>
<th>28</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>DESTINATION</td>
<td>S</td>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Word 2</th>
<th>00</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>32-Bit Address of Command</td>
<td></td>
</tr>
</tbody>
</table>

Bits 17-23 and 28-31 should be zero filled.
<table>
<thead>
<tr>
<th>WORD</th>
<th>BIT{S}</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>00-15</td>
<td>Destination Field: Required in all requests. It is returned in the solicited responses.</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>$ Field: Status of this request slot.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;0&quot; - This request is active or has been completed. The LCC clears this bit when it initiates this request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;1&quot; - This request is ready for initiation. The HLP sets this bit when it enters a new request.</td>
</tr>
<tr>
<td>1</td>
<td>24-27</td>
<td>I Field: Interrupt and Response Option</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0XXX - Do not send response or interrupt when request is initiated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10XX - Send response when request is initiated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11XX - Send response and interrupt when request is initiated.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>XX00 - Do not send interrupt when request is completed.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>XX10 - Send interrupt if request terminates abnormally.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>XX11 - Send interrupt when request is completed or terminated.</td>
</tr>
</tbody>
</table>

5.3.4 Command Format: Stream

<table>
<thead>
<tr>
<th>00</th>
<th>07</th>
<th>08</th>
<th>15</th>
<th>16</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td>FUNCTION CODE</td>
<td></td>
<td></td>
<td>BYTE COUNT</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
32-Bit Address of Data Buffer

Bits 08-15 should be loaded with zeroes.

<table>
<thead>
<tr>
<th>WORD</th>
<th>BIT(S)</th>
<th>FUNCTION CODE(HEX)</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>00-07</td>
<td>00</td>
<td>Read - The LCC stores the data it receives from the line and stream (defined in the request) into the data buffer.</td>
</tr>
<tr>
<td></td>
<td>01</td>
<td></td>
<td>READTI - The initial read command when input from paper tape is expected.</td>
</tr>
<tr>
<td>80</td>
<td></td>
<td></td>
<td>Write (an ETB Response is returned).</td>
</tr>
<tr>
<td>81</td>
<td></td>
<td></td>
<td>Write (an EOM Response is returned).</td>
</tr>
<tr>
<td>16-31</td>
<td></td>
<td></td>
<td>Byte Count - This field specifies the number of 8-bit data bytes to be processed (read or write) from the data buffer.</td>
</tr>
</tbody>
</table>

### 5.3.4.1 Input Stream Commands

There are two forms of input commands - those associated with keyboard input and those associated with paper tape input.

It is tacitly assumed by the LCC that the HLP is aware of which device of the TTY is sending data and is directing the LCC accordingly.

All DEL and NULL characters which appear in the TTY input stream are discarded by the LCC.
Note that there are two descriptions of the READ command {function code 00}. The first describes the operation when input is from paper tape and the second describes the operation when input is from the keyboard.

5.3.4.2 READTI {Function Code 01}

This command is to be issued prior to the initial input from paper tape. This command differs from {paper tape} READ {function code 00} only in that X-ON is sent to the TTY prior to data input.

Note that READTI is to be issued prior to data input from paper tape. It is assumed that the HLP is aware of the forthcoming PT input through either prior entry at the keyboard or some other stimulus.

It may happen that tape input occurs before the LCC recognizes the READTI. In this instance, the input is assumed to originate from keyboard, the X-ON is not sent, and any data received up to the time READTI is detected is treated as though it originated from the keyboard.

Should an entire tape message be received before READTI is detected, that message is treated as though it originated from the keyboard.
5.3.4.3 READ {Function Code 00} Paper Tape Operation

This command is issued for all subsequent inputs from paper tape after READTI. READ, unlike READTI, does not send X-ON prior to data input.

The primary function of READ is to identify the HLP buffer and buffer length in which input data from the associated TTY line is to be placed. READ may be issued either before or after the input flow has begun. Should the TTY input begin before the HLP buffer is defined, a response is sent to the HLP which demands READ be promptly issued.

The buffer defined by READ is terminated normally upon the LCC encountering any of the following conditions:
1. Line Feed {LF}
2. Carriage Return {CR}
3. X-OFF {if immediately preceeded by data}
4. EOT {if immediately preceeded by data}

The terminating character {LF, CR} is defined as the first encountered in a message segment. This determines the response which will be sent.

The terminating character is not placed into the HLP buffer and succeeding DEL characters are ignored as is the CR or LF which occurs after the initial LF or CR. See examples below.
The character X-OFF and EOT may be used as message delimiters as well as their usual control functions. X-OFF and EOT are assumed to terminate a message if either is encountered after a string of data characters with no intermediate LF or CR. The message response indicates termination with CR when X-OFF and EOT terminated the message.

Message segments from paper tape are assumed to be in one of the following forms and the various forms may appear in any order on the tape insofar as the LCC is concerned.
This portion is passed to the HLP memory buffer.

This portion is not passed to the HLP memory buffer.

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>{1}</td>
<td>d</td>
<td>d</td>
<td>...</td>
<td>d</td>
<td>LF</td>
</tr>
<tr>
<td>{2}</td>
<td>d</td>
<td>d</td>
<td>...</td>
<td>d</td>
<td>CR</td>
</tr>
<tr>
<td>{3}</td>
<td>d</td>
<td>d</td>
<td>...</td>
<td>d</td>
<td>X-OFF</td>
</tr>
<tr>
<td>{4}</td>
<td>d</td>
<td>d</td>
<td>...</td>
<td>d</td>
<td>EOT</td>
</tr>
<tr>
<td>{5}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X-OFF</td>
</tr>
<tr>
<td>{6}</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>EOT</td>
</tr>
</tbody>
</table>

\[ d = \text{data} \]

In the above forms, the number of \texttt{DEL} or NULL characters is assumed to be a minimum of three. It is recommended that not fewer than 3 \texttt{DEL} or NULL characters be sent to provide adequate time delays in HLP-LCC communications.

The characters of \texttt{X-OFF} and \texttt{EOT} are handled in a special manner by the LCC. When either is encountered in the input stream, a response is not immediately sent. Rather, a timeout of .2 seconds is set up.
If no input is received within the .2 seconds, it is assumed that tape motion has stopped and a response which indicates X-OFF or EOT is sent to the response list. DEL/NULL characters received during the timeout interval cause the timeout to be reset. When it is determined that tape motion has stopped, the LCC places the line in the idle state and is prepared to accept further input from the keyboard. READTI must be issued to restart the tape reader.

If, however, input does occur before expiration of the timeout, the response is cancelled and the input is accepted. It is assumed that this data originates from the paper tape reader.

5.3.4.4 READ [Function Code 00] Keyboard Operation

This command is issued for all keyboard input operations. READ may be issued either before or after the input flow has begun. The input flow is defined as the first character encountered in a message. Should the input flow begin prior to issuance of READ, a response is sent to the HLP demanding that READ be given to define the HLP input buffer.

The READ command is terminated whenever one of the following characters are encountered.

1. LF
2. CR
3. EOT Treated as CR, but timeouts not set as with tape; and no LF is returned as for CR.
4. X-OFF
The ending character determines which response is to be sent.

If the message terminates with LF, the LCC automatically sends a CR prior to expecting a subsequent message, and conversely, if the message terminates with CR, the LCC sends a LF.

5.3.4.5 Output Stream Commands

Commands for output are honored whenever the line is open and idle. In no instance of output does the LCC inspect the output stream for special characters as it does for input. It is assumed that proper format codes are contained in the message and that the carriage is properly positioned for the subsequent input/output.

5.3.4.6 WRITE {Function Codes 80 and 81}

These commands are used to output data to the TTY. They operate with either the printer or paper tape.

When the LCC outputs data to the TTY, it does not inspect the output stream for special characters as in the case of input; therefore, the HLP may insert as many carriage control characters into the output stream as necessary for format purposes and must insure proper positioning of the carriage for the following I/O operation.

The output operation of WRITE may be issued at any time by the HLP. The LCC initiates this operation whenever a message is
not presently being received. If an input is in progress when
the WRITE is issued, the input is allowed to continue to
completion. The termination of WRITE occurs when the given
number of bytes has been transmitted.

The LCC does not distinguish between WRITE {function code 80}
and WRITE {function code 81}; the difference centers entirely
around the response sent to the HLP. The HLP may find the
distinct responses useful for its own control.

5.3.5 Response List

The LCC informs the HLP of current request, command and system
status by updating the response list.

The response list is a circular list contained in the HLP
memory module that is preceded by two 32 bit pointers: the IN
and OUT pointers.

```
   0  31
IN
OUT
Entry-Response
Entry Response n
```

The IN pointer is updated by the LCC and specifies the next
response entry to be filled by the LCC. The OUT pointer is
updated by the HLP and specifies the next entry to be processed
by the HLP.
There are two types of Responses: solicited and unsolicited. Solicited responses are those which are the result of a HLP request. Unsolicited responses are those which are the result of some unexpected stream or system condition.

Responses are 128 bits in length and either provide the central system with information about a previous request or tell the central system that something requires attention. There are four formats of responses. The response code reflects which type it is.

Normal Unsolicited Response {Type 1}

<table>
<thead>
<tr>
<th>Stream No.</th>
<th>If Applies</th>
<th>Line No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>L06</td>
<td>Response Code</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7</td>
</tr>
</tbody>
</table>
Normal Solicited Response to HLP-LCC Request {Type 2}

<table>
<thead>
<tr>
<th>Field</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination Field</td>
<td>15</td>
</tr>
<tr>
<td>Response Code</td>
<td>1</td>
</tr>
<tr>
<td>Base Address of Command</td>
<td>6</td>
</tr>
</tbody>
</table>

Normal Solicited Response to a Stream Request {Type 3}

<table>
<thead>
<tr>
<th>Field</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination Field</td>
<td>15</td>
</tr>
<tr>
<td>Response Code</td>
<td>1</td>
</tr>
<tr>
<td>Base Address of Command</td>
<td>6</td>
</tr>
<tr>
<td>BYTE Count</td>
<td>1</td>
</tr>
</tbody>
</table>

Abnormal Unsolicited Response {Type 4}

<table>
<thead>
<tr>
<th>Field</th>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line No.</td>
<td>8</td>
</tr>
<tr>
<td>Response Code</td>
<td>1</td>
</tr>
</tbody>
</table>

Byte Count = Number of 8-bit bytes received/stored in data buffer.
5.3.5.1 Response Code Meanings

<table>
<thead>
<tr>
<th>L</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>INT</td>
<td>UNSOL</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>INT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

LOG = Log Flag. When set, this flag indicates conditions which should be logged on an error file.

INT = Interrupt Flag. This bit is set, as is the SAC I/O port interrupt line, when the associated request specified response with interrupt.

Bits 8 to 15 specify the response code which indicates the status of the operation.

UNSOl = Unsolicited response flag. This flag is set with an unsolicited response, that is, a response that cannot be directly associated with a request.
The following responses are returned to the response list upon conclusion of a command associated with a TTY line.

<table>
<thead>
<tr>
<th>Resp. Code</th>
<th>Resp. {Hex} Type</th>
<th>Associated Command</th>
<th>Cause of the Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>All TTY</td>
<td>HLP requested a response upon command initiation.</td>
</tr>
<tr>
<td>1-20</td>
<td></td>
<td></td>
<td>Not Assigned</td>
</tr>
<tr>
<td>21</td>
<td>3</td>
<td>WRITE{fc 60}</td>
<td>The output is complete.</td>
</tr>
<tr>
<td>22</td>
<td>3</td>
<td>WRITE{fc All}, READ {fc 00}, READTI{fc 01}</td>
<td>The associated WRITE command has terminated. This response has no inherent meaning to the LCC for subsequent I/O operations. In the case of output the response implies the entire buffer has been transmitted; in the case of input, the response implies the message terminated with CR.</td>
</tr>
<tr>
<td>23</td>
<td></td>
<td></td>
<td>Not Assigned.</td>
</tr>
<tr>
<td>Resp. Code [Hex]</td>
<td>Resp. Type</td>
<td>Associated Command</td>
<td>Cause of the Response</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------</td>
<td>-------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>TERMINATE</td>
<td>The command to terminate the associated output stream is accepted. When the termination is complete, response 28 is sent.</td>
</tr>
<tr>
<td>25</td>
<td>2</td>
<td>DUMP</td>
<td>The command to dump LCC memory is completed.</td>
</tr>
<tr>
<td>26</td>
<td>2</td>
<td>CLOSE</td>
<td>The CLOSE line command is initiated; when the request is completed, the response 4E is sent. The CLOSE may terminate an active stream command in which case response 28 can also be expected.</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td>Not assigned.</td>
</tr>
<tr>
<td>28</td>
<td>3</td>
<td>TERMINATE CLOSE</td>
<td>This response indicates that the indicated stream request is abnormally terminated. This can result from a TERMINATE or CLOSE command or from a forced closing of the line by the LCC.</td>
</tr>
<tr>
<td>29</td>
<td>3</td>
<td>All</td>
<td>The command is unrecognizable and/or the command contained illegal values.</td>
</tr>
<tr>
<td>2A</td>
<td>3</td>
<td>All</td>
<td>The internal buffer pool in the LCC has reached a threshold level, this request is rejected.</td>
</tr>
<tr>
<td>Resp. Code</td>
<td>Resp. Type</td>
<td>Associated Command</td>
<td>Cause of the Response</td>
</tr>
<tr>
<td>------------</td>
<td>------------</td>
<td>--------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>2C-2D</td>
<td>2</td>
<td>SUSPEND</td>
<td>Not Assigned.</td>
</tr>
<tr>
<td>2E</td>
<td>2</td>
<td>SUSPEND</td>
<td>The command to suspend the LCC has been initiated. It may happen that the LCC sends response 40 during the same interval that the HLP must be aware of this possibility and ignore the response if the last issued READ for the line is incomplete.</td>
</tr>
<tr>
<td>41-43</td>
<td>4</td>
<td>N/A</td>
<td>Not Assigned.</td>
</tr>
<tr>
<td>*45</td>
<td>4</td>
<td>N/A</td>
<td>System had an intermittent SAC reject.</td>
</tr>
<tr>
<td>*46</td>
<td>4</td>
<td>N/A</td>
<td>System buffers are saturated, this line is closing.</td>
</tr>
<tr>
<td>47</td>
<td>1</td>
<td>N/A</td>
<td>Not Assigned.</td>
</tr>
<tr>
<td>*48</td>
<td>1</td>
<td>N/A</td>
<td>The DSA connected to this line is malfunctioning, the line is closing. If a stream is active, response 28 is sent.</td>
</tr>
<tr>
<td>49</td>
<td>1</td>
<td>N/A</td>
<td>The channel flag is cleared. This occurs when the scan of the line list is complete during the open procedure or when the suspend command is cancelled.</td>
</tr>
<tr>
<td>4A-4C</td>
<td>1</td>
<td>N/A</td>
<td>Not Assigned.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>The LCC is in a suspended state.</td>
</tr>
<tr>
<td>Resp. Code</td>
<td>Resp. Type</td>
<td>Associated Command</td>
<td>Cause of the Response</td>
</tr>
<tr>
<td>------------</td>
<td>------------</td>
<td>--------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>4E</td>
<td>1</td>
<td>N/A</td>
<td>The indicated line is closed. This response is returned following either a CLOSE command or forced line closing caused by an abnormal condition. If any stream commands are active when closing the line, they are terminated. Response 28 is returned for each stream so terminated.</td>
</tr>
<tr>
<td>4F</td>
<td>1</td>
<td>N/A</td>
<td>The indicated line is open but a call is not yet answered.</td>
</tr>
<tr>
<td>50</td>
<td>1</td>
<td>N/A</td>
<td>The LCC has answered a call on a TTY line. This response is sent when the Data Set Ready lead from the modem is on and occurs with either a switched or dedicated line. The line is prepared for unsolicited keyboard input.</td>
</tr>
<tr>
<td>51</td>
<td>1</td>
<td>N/A</td>
<td>X-OFF has been received from the paper tape reader and there has been no non-DEL/NULL input for .2 seconds. It is assumed that tape motion has stopped. If a READ command is active and the X-OFF did not occur with data, the READ is not terminated and will be used for subsequent keyboard input.</td>
</tr>
<tr>
<td>Resp. Code [Hex]</td>
<td>Resp. Type</td>
<td>Associated Command</td>
<td>Cause of the Response</td>
</tr>
<tr>
<td>------------------</td>
<td>------------</td>
<td>--------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>52</td>
<td>1</td>
<td>N/A</td>
<td>E0T has been detected; otherwise as above for X-OFF.</td>
</tr>
<tr>
<td>53-57</td>
<td></td>
<td></td>
<td>Not Assigned.</td>
</tr>
<tr>
<td>58</td>
<td>1</td>
<td>N/A</td>
<td>The receive carrier lead from the modem indicates that a carrier signal is not being received. The line is closing. If a stream is active, response 28 is sent. When the close is complete, response 4E is sent.</td>
</tr>
<tr>
<td>59</td>
<td>1</td>
<td>N/A</td>
<td>The Data Set Ready lead from the modem is off - the call has been disconnected and the line is closing. If a stream is active, response 28 is sent. When the close is complete, response 4E is sent.</td>
</tr>
<tr>
<td>Resp. Code (Hex)</td>
<td>Resp. Type</td>
<td>Associated Command</td>
<td>Cause of the Response</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------</td>
<td>--------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>5A</td>
<td>1</td>
<td></td>
<td>One or more characters of the TTY message are lost due to:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1. The HLP did not supply an input buffer in time.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2. Internal LCC timing conflicts - the LOG flag is set in the response word.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This is an informative response; no special recovery actions are required by the HLP as the LCC has initiated re-transmission of the message.</td>
</tr>
<tr>
<td>5B</td>
<td>1</td>
<td></td>
<td>Input is arriving on the indicated TTY line. The HLP must promptly issue a READ request to define the receiving buffer.</td>
</tr>
<tr>
<td>5C</td>
<td>1</td>
<td></td>
<td>Input was detected during an output operation. The active output is terminated. This response is to be interpreted as an interrupt by the remote operator.</td>
</tr>
<tr>
<td>5D</td>
<td>1</td>
<td></td>
<td>The input buffer filled before an ending delimiter {CR or LF} was encountered.</td>
</tr>
<tr>
<td>Resp. Code</td>
<td>Resp. Type</td>
<td>Associated Command</td>
<td>Cause of the Response</td>
</tr>
<tr>
<td>------------</td>
<td>------------</td>
<td>--------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>This is an informative response - no special recovery actions are required by the HLP as the LCC has initiated re-transmission of the message.</td>
</tr>
</tbody>
</table>

Both Solicited and Unsolicited Responses

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>READ {fco0}</th>
<th>The TTY input stream has terminated normally with ending delimiter of LF.</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x61</td>
<td>0x3</td>
<td>READ TI{fco1}</td>
<td></td>
</tr>
</tbody>
</table>

Note: All responses that are starred have the LOG flag set.
5.4 Overview of Mode 3

Although it is not within the scope of this document to describe the precise nature of the HLP software and the terminal operation, this section is included to give an understanding of the functions performed and the general flow of the operation.

The major functions of the HLP can be summarized as follows:

1. LCC Subsystem autoload - cause the LCC software to be established in the LCC memory.
2. Set up the Line Table in the HLP memory.
3. Format Request and Command Table entries into the HLP memory to direct LCC activities.
4. Scan the Response Table for complete operations.
5. Manage the allocation of data buffers in HLP memory.
6. Control the SAC interface.
7. Coordinate activities with the terminal operator.
8. Interfacing with application programs.

On the other side of the Mode 3, the TTY provides a user interface to the system. The operation of the TTY is very simple when compared to that of a remote batch terminal. The TTY basically provides a keyboard input and a graphic output. An optional form of input and output is the TTY paper tape. The TTY provides no buffering or control over the interaction with the HLP.
Establishment of the communication subsystem begins by auto-
loading the LCC. A 64 word autoload/autodump bootstrap is sent
across the CDC compatible trunk from the HLP memory to the
LCC memory. The bootstrap program then loads an initializer
program from the HLP memory. The initialization program in
turn loads the standard software program and constructs the
necessary tables in the LCC. At this point, the LCC has
begun normal operations and is waiting for a channel flag
from the HLP {no communication has started, the LCC is in a
ready state}. Some point in time after the autoload, the HLP will decide to
allow communications with a terminal. For this to be accom-
plished, the HLP must open the line.

To open a line, the HLP constructs a Request List and enters
its pointer into the Line List. It also sets the Channel
Flag. When the LCC detects that the Channel Flag is set, it
searches the Line List to determine which line or lines are
to be opened. It also determines the location of the Request
List. Then it sends the Data Terminal Ready signal to the
modem enabling the modem to automatically answer any incoming
call. The LCC then waits for the Data Set Ready and Carrier
On signals from the modem to indicate that a call has been
answered. Using a Model 33 teletype as an example, these
signals will be generated when the ORG button is pushed and
the number of the local Data Set is dialed. The HLP is notified by a response when the call is answered.

At this point the user is now able to use the resources of the system. The HLP may indicate this to the user by outputting some message to the TTY and positioning the carriage at the beginning of a new line.

As an example, assume the user wishes to send the HLP a data file which is greater than 80 characters in length. The user would input the data file 80 characters at a time until the entire file was inputted. Each 80 characters would be followed by a LF {line feed} or CR {carriage return} to terminate the message segment.

When the LCC detects the LF in the input stream, it would send a response to the HLP which indicates termination with LF. This response may be interpreted by the HLP as it sees fit {e.g. the LF may be used by the processing application to denote that a segment of a larger message has been entered, but that the message is not yet complete}. The LF would not be placed into the HLP buffer.

The LCC would then send a carriage return to the teletype following these messages that terminated with LF. This positions the carriage and informs the TTY operator that the next 80 characters {or less} of the data file can be entered.
If a CR was inputted to terminate a message segment, it would be processed by the LCC in a manner similar to that of the LF. The LCC would send a response to the HLP which indicates termination with CR. Again, this response may be interpreted by the HLP as it sees fit {e.g. the CR may be used to indicate that the final segment of a message has been entered}. The CR would not be put into the HLP buffer and the LCC would send a LF to the teletype.

The HLP may attach special meaning to certain characters. For example, a shift 0 inputted by the user may indicate to the HLP to ignore the preceding character. Or an $X^C$ {control X} may indicate to the HLP to ignore all characters from the preceding LF or CR up to the $X^C$. The LCC is transparent to these characters with special meaning. It just passes them on to the HLP as data characters.

If a problem such a lost data occurs in the system while the user is inputting data, the LCC notifies the HLP via the Response List mechanism.

Now assume the TTY has paper tape capability. As an example, the user could submit a small program to the HLP via the keyboard. He then could request the HLP to compile it and punch it out on paper tape so it can be entered more easily later. The user should turn on the paper tape punch before the LCC sends the program out to be punched. Later, when the
user wishes to resubmit the program via paper tape to the HLP, he would type an appropriate message in to the HLP indicating paper tape mode. The HLP would issue a READTI command and the LCC would output an X-ON to turn the paper tape reader on, and paper tape reading could then begin. When the LCC detects an X-OFF or EOT condition, it times out for .2 seconds. If no data comes in during this time, it assumes that paper tape motion has stopped.

In the above case the user can terminate the paper tape input by turning the paper tape reader off. The LCC will inform the HLP via an unsolicited response.

A user can use paper tape capability in another way. He can create a paper tape off-line and later submit it to the HLP when the line is open. In this case comments above regarding characters with special meaning to the HLP such as shift 0 and X^C would apply here also. In addition, the message segments excluding all control characters {LF, CR, X-OFF, and EOT} and DEL/NUL should not exceed 80 characters. Refer to section 5.3.4.3 for a detailed explanation of how message segments and messages should be terminated on a paper tape.

For both keyboard and paper tape input, LF, CR, X-OFF, EOT, and DEL/NUL characters are not sent on to the HLP.
Line closing can be initiated by either the user or the HLP. The HLP can send a close line request/command to the LCC in reaction to a typed in message from the user, or on its own initiative (e.g., specified time of day). In either case, all input is immediately cut off, and the line is closed and disconnected.
6.0 MODE 4.0 [200UT/216 PROTOCOL] PROCEDURE

The LCC mode 4 software supports terminals using the 200 UT or 216 communication protocols. The terminals supported are listed in Table 6.1. Mode 4 protocol is a two-way alternate, asymmetric protocol having two variants; 200 UT and 216. Mode 4 link control procedures are provided on the following synchronous communication channels:

1. Lines of speed 2000 bps. point-to-point, half duplex, switched.
2. Lines of speed 2400 to 9600 bps. point-to-point, full or half duplex, non-switched.
3. Lines of speed 2400 bps. to 9600 bps. multi-drop, full or half duplex, dedicated leased or private.

The following capability is provided by the mode 4 software:

1. Switched network automatic answer/disconnect for mode 4 terminals.
2. Multi-drop capability on dedicated leased or private, mode 4 links to reduce overall customer cost per line.
3. Code conversion, specified on an individual request basis, for mode 4 terminals.
4. Mode 4 link control procedures which:
   - Insert and delete link control characters that enclose the data blocks.
   - Calculate and check the message parity character.
   - Ensure that the proper sequence of messages are sent to
and received from the remote terminal in order to perform the operation specified by the HLP.

- Provide error detection and recovery procedures. Each message character is checked for proper character parity. Each message block is protected by a "message parity character". When the LCC software senses an error, the proper steps are taken to cause retransmission of the message block in error. The LCC furnishes error status information to the HLP for control and error logging purposes.

<table>
<thead>
<tr>
<th>TERMINAL</th>
<th>PROTOCOL</th>
</tr>
</thead>
<tbody>
<tr>
<td>711 IAT/CRT</td>
<td>216</td>
</tr>
<tr>
<td>241 GRID</td>
<td>200 UT</td>
</tr>
<tr>
<td>200 UT BCD</td>
<td>200 UT</td>
</tr>
<tr>
<td>216 BCD</td>
<td>216</td>
</tr>
<tr>
<td>216 ASCII</td>
<td>216</td>
</tr>
<tr>
<td>731-12 LSBT</td>
<td>200 UT</td>
</tr>
<tr>
<td>732-12 MSBT</td>
<td>200 UT</td>
</tr>
<tr>
<td>200 UT ASCII</td>
<td>200 UT</td>
</tr>
</tbody>
</table>

**TABLE 6.1**
6.1 Mode 4 Software Description

6.1.1 General Description of Mode 4

The mode 4 software is designed to allow full use of the various features found on mode terminals and also provides for future support of additional features with little change to the software.

The mode 4 software has logically two interfaces; one with the HLP and another with the terminal. The interface with the HLP enables the HLP to:

- send messages to a terminal,
- receive messages from a terminal,
- begin and stop communication with terminals,
- determine the cause of errors, and tell the LCC how to perform an operation.

This interface allows the LCC to:

- tell the HLP when it has completed an operation,
- notify the HLP of internal {to LCC} problems,
- report the occurrence of errors, and
- give information to the HLP that allows the HLP to be more efficient.

The following paragraphs present a brief discussion of the interface with the HLP. The HLP communicates with the LCC by issuing "requests" and "commands", as discussed in Section 3.
To send a message to a terminal the HLP sets up a command with one of the following function codes:

- write {80}
- clear write {81}
- reset write {82}
- alert write {83}
- graphic write {85}
- write ASCII and send index {86}

There are several reasons for providing more than one function code to send a message to a terminal rather than only one:

1. Several function codes are recognized by mode 4 terminals and different operations are performed because of them. {Only one function code is allowed per message.}

2. Because of the general nature of the mode 4 software, the HLP is not restricted in the kind of operation it can ask the terminal to perform. In this way the HLP can take full advantage of the features of any mode 4 terminal.

3. Because the mode 4 software performs the duties of a driver rather than the duties of a front end device, the HLP application is transparent to the mode 4 software. The HLP must be aware of the relationship between the messages being received from and sent to a terminal since the mode 4 software makes no assumptions of this nature. The several "write" function codes give the HLP its needed level of control to do this.
Several "read" function codes are also provided. These are:

- read {00}
- read wait {01}
- read all {02}

More than one "read" function code is provided for the following reasons:

1. The read request is provided to allow the HLP to quickly determine whether a particular terminal has data to send to the HLP. If no data is available the HLP will be notified quickly, the read will be terminated by the LCC and the HLP may release the buffer used in the read for another use. If data was available when the read was issued, the data will be transmitted to the HLP.

2. The "read wait" request is provided to relieve the HLP of the burden of continually "polling" a terminal to obtain data from the terminal. The read wait will be satisfied only when the specific terminal referenced has data to send. Unlike the "read" command, the "read wait" command may tie up a buffer for a long time.

3. The "read all" request is provided to relieve the HLP of the burden of polling a group of terminals or stations for data and to lessen the HLP's requirement for buffers. This request will be satisfied when any terminal or station on a given line has data to send.
4. All of these function codes are provided to increase the flexibility of the mode 4 software and thereby allow the HLP to use a variety of techniques to obtain data from a terminal. This is in keeping with the general purpose nature of the mode 4 software.

Another function of the interface with the HLP is to allow the HLP to start and stop communication with the terminals. The following function codes are provided to facilitate this:

1. Close line. This command notifies the mode 4 software that the HLP wishes to discontinue communication with terminals on a certain line.

2. Terminate active request. This command allows the HLP to terminate a request before it has been completely processed by the LCC.

The HLP starts communication with terminals by putting the address of the lines it wishes to begin communication with in the Line List.

The HLP is able to determine the cause of errors by initiating a data set or data set adapter loop test on the line. Command codes are provided for this purpose. This provision is in addition to the information contained in the responses sent to the HLP by the LCC.
Another function of the HLP interface, as stated above, is to enable the HLP to indicate to the LCC how operations are to be performed {rather than what operation to perform}. Thus, the HLP is able to:

- Tell the LCC what type of terminals are allowed to dial into a switched line. This is done at autoload time when the HLP presents Line Parameter Records containing a terminal model selection mask to the LCC for each mode 4 line. The HLP can also reselect the terminal models allowed on a line after the system has been autoloaded by using the "configure line" request to respecify the TM selection mask.

- Tell the LCC how to perform code conversion by sending code conversion tables to the LCC. A command function code is provided for this purpose. By being able to do this the HLP can specify the code to be processed by the HLP; the number of members in the code set eg. 64, 96, 128 or 256; whether the code set is to be folded eg. converting a 96 member code set into one with only 64 members, and change flag codes into other flag codes. In addition, the HLP can specify whether code conversion is to be done at all and which of several code conversion tables are to be used if conversion is to be performed. This is accomplished by a field in the request that is provided for this purpose.
Tell the LCC what action to take after an error has occurred. In some situations buffers could be processed out of sequence when an error occurs. A flag in the request is provided to direct the LCC's actions when this happens.

Determine how the LCC will poll terminals. The mode 4 software is designed to allow any mode 4 terminal to dial into any mode 4 line. To do this, restrictions are placed on the site codes that various terminals can use. However, sometimes it may be advantageous to restrict the terminals in the configuration or to assign certain terminals to certain lines, with the HLP having knowledge of the configuration. A bit in the terminal model site code field is provided to allow the HLP to instruct the LCC to expect any site code configuration without restriction if this is desired.

Select which of the 48 lines are to be assigned to which communication mode. This is done at autoload time by the Line Select Masks in the Autoload Table.

Information also flows from the LCC to the HLP through the interface with the HLP. The LCC notifies the HLP that it has completed a requested operation by writing responses in the
HLP's response list. This type of response is a solicited response (because it is always associated with a request) and provides the following information in addition to flagging the end of an operation:

- The number of errors that occurred while the LCC attempted to perform the operation.
- That the operation was or was not successfully completed.
- The amount of data processed (in bytes).
- That the LCC has begun processing on a request.
- That an exception condition (as distinguished from an error condition) has occurred. An example of this is the notification that the operator has depressed the "intervene" key while reading cards or printing data on the line printer of a 200 UT.

In the same manner the LCC uses responses, both solicited and unsolicited, to report problems internal to the LCC. Some of the internal problems reported by solicited responses (and therefore associated with a particular request) are the following:

- The LCC has reached a point where it has too few internal buffers to continue normal processing and must reject the present request.
The LCC found an illegal parameter in the request or command it is processing and must terminate this request.

The following internal problems, not associated with a particular request are reported by unsolicited responses.

- A fatal error has been detected on a communication link and this line is being closed because it is not possible to do additional processing on it.

- Intermittent errors were detected on the data path between the HLP and LCC.

- A fatal error was detected on a Data Set Adapter. The LCC is closing the line involved because it cannot continue processing on the link.

The LCC also uses the interface with the HLP to report errors. Some of the conditions discussed above also fall in this category. Error conditions reported include the following:

- A peripheral device referenced by a request is in a not-ready state.

- A request could not be successfully completed in the number of error retries allowed.

- A request could not be successfully completed before it was terminated by HLP request.
A fatal error occurred in the data link.

A DSA had a fatal malfunction.

Errors were detected on the data path between the LCC and the HLP.

The final type of information that is reported to the HLP by the LCC through the interface with the HLP is that which allows the HLP to operate more efficiently and thereby increase system throughput. The following procedures are provided to accomplish this:

- When a terminal dials into an opened mode 4 switched line, the mode 4 software will send a response to the HLP identifying the type and configuration of the terminal. This information provides the basis by which the HLP decides to specify code conversion, addresses the various stations on the terminal by the coded "destination" field in the request, controls the reading of cards and printing by the line printer, etc.

- When a terminal or a station on a multi-station terminal has data to send to the HLP for the first time after the line has been opened, the LCC will send an unsolicited response to the HLP to indicate this. This response will be sent only once for each station from which an operator can directly communicate with the system each time the line is opened. This will relieve
the HLP of the burden of polling each inactive terminal or station to determine when it will become active. This capability is especially useful on multi-drop lines and with multi-station terminals such as the 2LB.

- The "read wait" command is provided to relieve the HLP of the burden of polling an active station. Because the ability of the HLP to read data from a CRT is limited by operator response time, many attempts to read may be needed before data is actually obtained.

- The "read all" command will similarly relieve the HLP of the duty of successively polling active stations to obtain data. However, this command is provided essentially to reduce the requirement for HLP resident buffer space by allowing the HLP to read data from any terminal or station on a multi-drop line or any station on a multi-station terminal. The "read all" request is satisfied by any terminal on the line for which it is issued. Without this command several buffers would be allocated to receiving data from several terminals or stations for an indefinite amount of time. The system throughput will be reduced if the HLP runs out of unused buffers.
The second logical interface to be discussed is the LCC's interface with the terminal. Significant aspects of this interface are the following:

- The LCC will identify the terminal that dials into a switched line. This is done without HLP action by sending poll messages to the terminal. These poll messages are generated internally by the LCC.

- In like manner, the LCC will determine initial station activity without any action being required by the HLP to enable the LCC to do this.

- The LCC will add/delete message framing characters in messages exchanged with the terminals. The LCC will also check these for validity.

- The LCC will attempt to recover from errors detected while communicating with terminals. Either a finite or an infinite number of retries will be allowed.

- The LCC will perform all link control procedures needed to communicate with the terminals. Included in this are link initialization and disconnect, message sequence control, code conversion and polling for data.
6.1.2 Configuration

Since mode 4 is a two-way alternate communication scheme, the number of lines supported are: (see Section 3.1.3 for further explanation).

- up to 30 lines at 9600 bps.
- up to 48 lines at 4800 bps, or less

The number of lines serviced as stated above must be reduced by 10% when code conversion capability is used, since code conversion adds 2 microseconds to the processing time for each data character.

A mix line speeds can be attained in the manner described in Section 3.1.3.

The amount of core required for mode 4 software is as follows:

\[ \text{mode 4 words of core} = 2500 + 3BL_4 + 8AL_4 + 1BS_4 + 8Q + \text{TMA} + 0.5\text{CCT} + 8\text{TMP} \]

where,

- \( L_4 \) = the number of mode 4 lines
- \( AL_4 \) = number of active (i.e. opened) mode 4 lines
- \( BS_4 \) = sum of active mode 4 streams
- \( Q \) = sum of the queued "read wait" or "read all" requests for all mode 4 lines
- \( \text{TMA} \) = Terminal Model Area. The size of this area varies with the number and type of TM's in the system from 3 to 448 words.
- \( \text{CCT} \) = the number of code conversion tables.
- \( \text{TMP} \) = the number of terminals on multi-drop lines
6.1.3 Mode 4 Features

6.1.3.1 Terminal Identification

A major problem faced with communications systems in which more than one kind of terminal is supported, is the identification of a terminal as it becomes active with respect to its type or class so that its nature and configuration will be known and it can be properly controlled or driven. The restrictions imposed to accomplish this often result in much equipment being required, the number of different kinds of terminals controlled being severely limited, and inflexible, restrictive rules being imposed.

To help solve this problem, the LCC Version 3.0 software, when concerned with mode 4 terminals, will identify terminals as they dial in on switched lines. This capability will not be provided on non-switched lines because the terminals on such lines are known. The following procedure is used:

1. After the LCC has been autoloaded and initialized, the HLP will open each line with which it wishes to communicate.

2. As a line is opened, the LCC will determine whether a terminal is present on the line and determine its type and configuration, if present.

3. The LCC will send a response to the HLP when the kind of
terminal on the line is known or when all allowable types are tried whichever occurs first. The response will contain a code identifying the terminal or indicating that it can't be identified or that none are dialed in.

4. If the response sent to the HLP indicated that the terminal could not be identified at the time that the line was opened, the LCC will continue to attempt to identify the terminal until it is identified or until the HLP terminates the procedure by issuing a "close line" request. If the terminal can be identified in this manner, an unsolicited response will be sent to the HLP and will contain a code identifying the terminal. If a "close line request is received, an attempt will be made to disconnect the terminal. To be able to identify a terminal in this manner, the following is required:

1. At autoload time the HLP must describe each unique terminal in the system by a Terminal Model {TM}. These terminal models will be read by the System Initializer and stored in LCC memory, unless they are preassembled into LCC memory {see Section 3}.

2. For each line in the system, a 32-bit parameter selecting up to 16 TM's must be sent to the LCC at autoload time, or be assembled into LCC memory.

3. Each of the 16 TM's selected for a given line must contain a site address code different than the site
address code of any other TM selected for that line.

4. Each terminal in the system must have its site address code set equal to the site address code of the TM which uniquely describes it.

The LCC will identify a terminal as follows. The LCC will issue a poll message to the connected terminal. If a reply is received, the LCC matches the site code of the received reply with the site code in a Terminal Model that describes a type of terminal allowed on that line. When a match is made, an identifying response containing the terminal model number is sent to the HLP. The reply from the terminal may be any valid response e.g. READ, REJECT, ERROR.

This capability will allow the following techniques to be used:

1. For all practical purposes, the HLP does not need to restrict a certain kind of terminal to a certain line or group of lines (e.g. BCD 200 UT's, ASCII 200 UT's, 241 graphics terminals and 216's could all dial into the same line.)

2. Only one LCC-11 is needed to control a diverse array of mode 4 terminals.
3. Only one LCC driver in the HLP is required.

4. Unauthorized terminals dialing in are identified. When the "data set ready" signal is present for the dialed line and the terminal does not respond to poll messages in which every valid site address has been sent, the terminal will be considered to be unauthorized and a response to the open line request will be sent indicating this condition.

6.1.3.2 Determination of Initial Activity of Terminals and Stations of Multi-Station Terminals

The LCC will notify the HLP, by an unsolicited response, that an inactive station wishes to become active. In this context, a station is defined to be one of the following:

- A terminal whose station address is not enabled,
- A CRT station on a terminal with the station address enabled. The station can be referenced through its unique site/station address.

An inactive station is one that has not replied to a poll message with a read message since its line was last opened. A response will be sent for each station on a terminal having more than one station. The unsolicited response will be sent only once for each station each time the line is opened.
This procedure will be used on dedicated and switched lines and with terminals utilizing either 200 UT or 216 protocol. The determination of initial activity will be made by sending a poll message, generated internally by the LCC, to the terminal. The poll message may have been sent when attempting to identify the terminal, as described above, and a read reply received at that time. If so, one poll message will serve to identify the terminal and to determine that a single station terminal is ready to become active. If a reject was received when attempting to identify the terminal, or if the terminal uses 216 protocol, additional poll messages will be generated by the LCC to determine initial activity. In any case, no data will be sent to the HLP as a direct result of determining initial activity since the operator of the station or terminal is trying to gain initial access to the communications facility.

This feature is provided to reduce the amount of data sent through the coupler and thus improve performance since the HLP need not issue repeated requests to determine this condition. The HLP is also relieved of the burden of polling, in turn, each inactive station of a multi-station terminal.

6.1.3.3 Read Command Option for Mode 4 Terminals

The LCC will respond to three read commands sent to mode 4 terminals as follows:
1. Read. The Terminal will be polled once for data and an immediate response will be sent upon completion of the read sequence. The response will indicate whether or not data was received and will signal the completion of the request.

2. Read Wait. The terminal will be repeatedly polled by the LCC until a read message is received in response to the poll message. The LCC will then send a response to the HLP indicating completion of the read request with data sent to the HLP. This request will not prevent communication with other stations or terminals on the line. If reject messages are being received from a terminal for which a "read wait" request is being processed the "read wait" request will be queued in LCC memory to await further processing so that requests for other stations or terminals on the line can be processed.

3. Read all. A "read all" request will be satisfied when data is obtained from any station or terminal on the line for which the request was issued. As in the case of the "read wait", this request will not prevent communication with other stations or terminals on the line. The response sent when this request is completed will identify the station and terminal that satisfied the "read all" in the DESTINATION field of the response. The LCC will generate the SITE ID and the STATION ID codes of the
DESTINATION field. The lower 8 bits of this field will be the same as the lower 8 bits of the "read all" request DESTINATION field. The upper 8 bits of the request DESTINATION field will be ignored by the LCC and therefore the "read all" request cannot be used to read data from a card reader. The "read all" request is provided to reduce the buffer memory requirements of the HLP.

If an irrecoverable error is detected on a terminal or station while the LCC is processing a "read all" request, the request will be abnormally terminated and the terminal and station identified in the response. This terminal will be flagged internally to the LCC so that it will not be polled again when trying to satisfy a "read all" request until the LCC clears the flag. The LCC will clear the read all lock out flag when it has successfully completed any other type of read or write operation on the terminal.

When a "read all" request is being processed on a line, the LCC will not initiate another read of any kind on that line until the "read all" is completed.
When a "read wait" request is being processed for a terminal or station on a line, no other read request will be initiated for that station until the "read wait" request is completed. The LCC will not initiate a "read all" request on a line if a "read wait" request is outstanding for any station on the line.

L.1.3.4 Code Conversion

The LCC will provide a limited code conversion facility for mode 4 terminals only. By code conversion is meant the translation of an 8-bit byte into another 8-bit byte. No validity or range checks will be made nor will codes be added or deleted. Code conversion must be specified in each request whenever the capability is desired.

At autoload time, the HLP must specify in the Autoload Table (see Section 3) the number of code conversion tables to be used. If code conversion tables are assembled into LCC memory this value should be set to zero. Sixty-four words will be allocated in LCC memory for each table. If more than one table is allowed, the last word of table N will be immediately followed in LCC memory by the first word of table N+1. Pointers to the first word of each table will be established in the direct core area of LCC memory. An area immediately following the code conversion tables will be allocated to hold a bias for each code conversion table. This bias will be subtracted from the first word address of the
respectively code conversion table to allow for code sets whose
lowest numbered code has a value other than zero.

Each code conversion table record when sent to the LCC will
contain 65 16-bit words. The first word will contain the
code bias and the second through the 65th words contain the
code conversion table that is stored in the LCC.

Each 16-bit word of this table will contain two 8-bit
characters. The high order character will be used in output
{from the LCC to the terminal} operations while the low order
byte will be used in input {to the LCC from the terminal}
operations. Each conversion character must have proper
parity when the table is sent to the LCC. The format of the
table is shown below as it appears in the HLP buffer.

<table>
<thead>
<tr>
<th>Word 0</th>
<th>Word 1</th>
<th>Word 64</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>Output Code Bias</td>
<td>Input Code Bias</td>
<td></td>
</tr>
<tr>
<td>Output Conversion Char.</td>
<td>Input Conversion Char.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Output</td>
<td>Input</td>
</tr>
</tbody>
</table>
It is the responsibility of the HLP to:

1. Transmit the required code conversion table to the LCC prior to the initiation of I/O on a line requiring code conversion, unless the table is assembled into memory.

2. Determine when code conversion is needed on a request basis and notify the LCC by putting the code conversion table number in the request for which code conversion is desired.

3. Determine how many code conversion tables can reside in LCC memory and not transfer more than this number to the LCC. The LCC software does not impose a restriction on the number of tables allowed, but nevertheless there is a restriction imposed by memory availability. Code conversion tables lessen the number of internal (to the LCC) buffers available to the LCC to perform its operations. If these are inadequate, the autoload procedure could be aborted, requests could be abnormally terminated and lines closed.

Normally code conversion will be performed on code sets with 64 members. When code conversion is needed on a terminal and a code set with more than 64 members is involved, the following procedure is used:

1. The first 64 words of the table set sent to the LCC as table N.

2. The second 64 words of the table are sent to the LCC as table N+1.
3. Additional 64 word blocks of the table are sent as table N+2 or N+3.

4. When code conversion is specified, the first 64 word table number is specified.

As an assembly option the code conversion tables and pointers may be assembled into the LCC memory at system generation time. If this is done, the tables need not be sent to the LCC by the HLP.

6.1.3.5 Dynamic Selection of a TM on a Line

A request function code will be provided to allow the HLP to select a different subset of up to 16 TM's from the original 32 TM array for use on a given line, at any time. Once the LCC has been autoloaded, no change to the TM's or to the number of TM's in the system is possible. Selection is on a line for line basis and is made by providing a 32-bit selection parameter.

6.1.3.6 Multi-drop Lines

The LCC will control up to 16 terminals on a dedicated leased or private line. Terminals on a dedicated line must be completely specified by the HLP through proper TM definition and selection as for point-to-point lines. In addition, site codes must be supplied for each terminal at autoload time. Station codes, where applicable, must equal those in the
appropriate TM. Only mode 4 terminals are supported on multi-drop lines. The same mixture of terminal types and code sets are allowed on dedicated lines as are allowed on switched lines.

Responses to indicate initial activity on stations will be sent for all terminals on dedicated lines in the same manner as for dialed lines.

b.1.3.7 Line Disconnect

An attempt will be made to disconnect the terminal from a switched line when a "close line" request is received by the LCC. This capability will enable the HLP to terminate a call when communication with the terminal is complete, e.g. when the terminal operator logs out.

b.1.3.8 Error Handling Option for Mode 4 Terminals

When the LCC completes a stream request the next request on that stream is scheduled to be read from the HLP memory. If an irrecoverable error has occurred on a stream request, reading of the next request queued on that stream could result in buffers being processed out of sequence. This is possible if the HLP sets up the next "write printer" request as soon as the LCC begins to process the current "write printer" request, and the HLP rewrites the buffer if an irrecoverable error occurs, as part of its error recovery process.
To help solve this problem the LCC will have the following capability when controlling mode 4 terminals:

1. Stream requests will have an "error option" flag bit \{E\} having the following meaning:

   \( E=0 \) \quad \text{Process any request in this stream slot if an irrecoverable error has occurred.}

   \( E=1 \) \quad \text{Abnormally terminate all requests present for this stream when the error occurred. When this has been done the LCC will continue to abnormally terminate all requests for this slot until a request is found having its \( E \) flag set to zero.}

This capability will allow the HLP to issue a new request before the previous request for a given slot has been completed to ensure high speed operation. If an irrecoverable error occurs a queued request can be removed and the HLP can reissue the unsatisfied request if that is desired.

6.1.3.9 Control Characters Added/Deleted

For transmission from the LCC to the terminal, the LCC frames all data blocks with a set of control characters. These control characters consist of four sync characters, a start of header code \{SOH\} a site address, a station address, a function code, any sync codes required after the function code, data, an escape code \{ESC\} and E code, an end of text \{ETX\} and a block parity code \{BPC\}. 
The LCC receives from the terminal data blocks which are bounded with similar control characters. The LCC interprets these control characters, ascertains if they are valid and strips them off of the data block before it records the data in the HLP data buffer.

6.1.3.10 Generation and Validation of Messages Between the Terminal and the LCC Needed to Perform the Operation Requested by the HLP

To cause a data buffer to be sent to a terminal, the HLP issues a single "write" command [several varieties] to the LCC. To receive a data block from a terminal, the HLP issues a single "read" command [several varieties] to the LCC.

To perform these operations, data is transmitted between the LCC and the terminal as a series of messages which allow for the transmission of data as well as additional information in the form of parity bits and check messages so that many kinds of errors occurring during transmission can be discovered and corrected. This will result in a sequence of two or four messages being transmitted. Whenever a message is sent to the terminal, a reply is expected from the terminal.

Messages sent by the LCC to the terminal are of two types:
1. The "write" message which is used to send data to the terminal.
2. The "poll" message, which is used to request data from the terminal.
Messages sent by the terminal to the LCC are of four types:

1. The "acknowledge" message to indicate satisfactory receipt of a write message.
2. The "reject" message to indicate that there is no data to send to the LCC.
3. The "error" message to indicate a transmission error on the previous "write" or "poll" message.
4. The "read" message to send data to the LCC.

It is the responsibility of the LCC to initiate and monitor these sequences. The HLP merely specifies that data be written or read without regard to the processes needed to accomplish this.

6.1.3.11 Error Recovery

When the LCC detects an error condition while exchanging data and control messages with a terminal, it initiates error recovery procedures a maximum of N times. A positive value of N is used as a retry count, a negative value is interpreted as an infinite number of retries. When N is positive and recovery procedures have been executed N times, the LCC exercises the Data Set Adaptor {DSA} to see if the condition is caused locally. The following action is then taken:

- The results of the test and the condition of a non-recoverable error are reported to the HLP by an unsolicited response {code 44}.
. The request being processed at the time is abnormally terminated.

. All internally queued {to the LCC} requests for the terminal in question are abnormally terminated.

. If the line is a switched line or a point-to-point direct line, the LCC will process no stream requests until it finds and processes a "close line" request. If the line is a multi-drop line the LCC will process no stream requests until it finds and processes a "terminate stream" or "close line" request.

. When a "close line" request is received, the LCC will send a solicited response indicating that the close line request is initiated {code 26}. All requests in the request list for the line will be abnormally terminated by a response with response code 28 and when this has been done a response with response code 4E will be sent indicating that the line is closed.

. When a "terminate active stream request" request is found, the LCC will send a solicited response with response code 24 to indicate that the request to terminate a stream has been received, any stream request in the Request List for the stream in question will be abnormally terminated by a response with response code 28. If no active stream
request is present in the Request List for the stream in question, a response with response code 27 will be issued to the HLP.

When N is negative and recovery procedures have been executed 10 times, the LCC will read the HLP to LCC request for the line, looking for a "terminate active stream request" request. If this request is not found, error recovery attempts will be executed again as described above. If a "terminate active stream request" request was found, the LCC exercises the DSA and reports the results of the test and the condition of the non-recoverable error to the HLP as described above.

Following this the procedure for abnormally terminating stream requests when a "terminate active stream request" request is received is used as described above.
The conditions which cause the LCC to perform error recovery are:

<table>
<thead>
<tr>
<th>ERROR CONDITION</th>
<th>RECOVERY ACTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. TIME OUT - A valid response to a previous transmission is not received within the allowed time frame.</td>
<td>Terminal status is checked if the last output was a write. The last message will be retransmitted if required.</td>
</tr>
<tr>
<td>2. ERROR response {NAK}. The last message transmitted by the LCC was not correctly received by the terminal. See terminal hardware manual for specific conditions.</td>
<td>The message is retransmitted to the terminal.</td>
</tr>
<tr>
<td>3. Parity error detected in reply to a poll or status message.</td>
<td>The message is retransmitted.</td>
</tr>
<tr>
<td>4. Incorrect MPC detected in reply to a Poll or Status message.</td>
<td>The message is retransmitted.</td>
</tr>
<tr>
<td>5. Parity error or MPC error detected in reply to a Write message.</td>
<td>The LCC transmits a POLL message to the terminal to determine the status of the last Write. The previous message may or may not be retransmitted depending upon status received.</td>
</tr>
<tr>
<td>6. Station address toggle error detected in reply to a write message. Message Parity is correct.</td>
<td>The last Write message is retransmitted.</td>
</tr>
</tbody>
</table>
L.1.3.12 Site Polling Option

An option is provided to allow all possible terminal site codes to be polled on a line. This situation is found in a computer utility environment where a large number of terminals of a single type are found among the user population and it is not desired to place restrictions on the site codes allowed. When this option is used, all possible site addresses will be used while the LCC is trying to identify the terminal on a dialed line provided that the following restriction is met:

Only a single terminal model can be selected on any line using this option. However, if this option is to be used on more than one line, each such line may select a different terminal model.

This option is selected on a line basis by setting the high order bit of the 5 bit site address in the site descriptor of the terminal model selected, to one. It should be noted that the same TM {Terminal Model} cannot be used on a line utilizing the site polling option and one not using the option. When this option is selected, the 4 low order bits of the site address in the TM are not used and can be set to any value.

L.1.4 Interface with the HLP

The interface by which the HLP and the LCC communicate is composed of tables and buffers resident in HLP memory and the set of rules governing their use.
The table portion of this facility is composed of the following parts:

1. Line List
2. Request List
3. Commands
4. Response List

6.1.4.1 Line List

The Line List, whose location in HLP memory is passed to the LCC at autoloader time, contains up to 48 entries, 32 bits in length. They serve to locate the Request List for each line in the system. The Line List is ordered (ascending) according to line number, there being one 32 bit entry for each line attached to the LCC, and each empty port up to the highest numbered port with a line attached. The last entry in the Line List is a 64 bit entry containing the "suspend operation on the LCC" command (command code F3 {hex}). If there is no new activity for the line, the corresponding entry in the Line List {slot} will be negative. Otherwise, a non-negative address of the Request List will be stored. It will be the responsibility of the HLP to set each slot negative for which there is no new activity. The HLP will put a new entry in the Line List for the following reasons:
1. The HLP wishes to suspend the LCC.

2. The HLP wishes to begin processing on a line.

b.1.4.2 Request List

The Request List serves as the primary means of indicating to the LCC that a certain type of action is requested by the HLP. This action is further specified in the command that is associated with the request.

The LCC periodically references the Request List to determine if a new request has been entered. Upon completion of the request, a response is sent to the HLP.

The request list for mode 4 lines contains three (3) request slots. The first slot is reserved for communication between the HLP and the LCC. The second slot is used to indicate an output operation from the LCC to the terminal. The third slot is used to indicate an input operation to the LCC from the terminal.

With this request list format the slot position of the request does not identify the station for which the request was issued as in mode 2 or mode 3. Therefore, mode 4 stations must be identified by the DESTINATION field of the request. This 16 bit field is formatted as follows:

```
0  3 4  7 8  15
DESTINATION  SITE ID  STATION ID  FOR HLP USE
```
SITE ID - This field contains an ordinal from 0 to F {hex} selecting one of 16 terminals on the line. For a point-to-point line where only one terminal is present this value should be zero. For multi-drop lines a code of 0 {zero} will select the terminal described by the first TM/SITE record of the Line Parameter Record for the line sent to the LCC at autoload time. A code of 1 {one} will select the terminal selected by the second TM/SITE record, etc.

STATION ID - This field contains an ordinal from 0 to F {hex} selecting one of several stations on the terminal. A value of zero will select the device described by the first device descriptor of the Terminal Model that describes the terminal referenced. A value of one selects the device described by the second device descriptor etc.

FOR HLP USE - This field is not defined by the LCC but can be used by the HLP, probably to define the line. This field is returned to the HLP in the destination field of the response as are the site and station ID fields.
To allow the LCC to process requests as quickly as possible so as to drive the terminals at the highest speeds possible, the HLP should do the following:

1. Set the I-field of the request (see below) to request that a response be sent to the HLP when the request is initiated.

2. When the response indicating that the LCC has initiated the request is received, the HLP should set up the next request.

It should be noted that in normal high speed operation with a 200 UT, the amount of time from receipt of the "request initiated" response until the LCC reads the next request for the same terminal with 1000 character buffers, is on the order of several seconds. Thus the HLP software will be able to perform the above listed operations at a leisurely pace.

A problem arises with high speed operation of a terminal when requests are queued (as above for instance). When an irrecoverable error occurs, it may be necessary for the HLP to send a message to the terminal to inform the operator that manual intervention is required. After the operator corrects the problem, the HLP may want to retransmit the buffer on which the error was detected. If the LCC has begun to process the next queued request, buffers could be processed
out of sequence. To help prevent this from happening, the following capability is provided:

1. An error handling option flag is provided in the stream request {the E-field}. If set, the LCC will abnormally terminate all queued requests for an unbuffered terminal when an irrecoverable error is found. If the terminal is buffered, requests queued on the station on which the error occurred will be abnormally terminated.

2. After the queued requests have been terminated, the LCC will again process stream requests. {See Section 6.1.3.8}

This will allow the HLP to send a message to the terminal to request operator intervention, reissue the buffer on which the irrecoverable error occurred and reissue all queued requests that were abnormally terminated. This will preclude buffers being processed out of sequence.

6.3.4.2.4 Format of Requests

The format of the HLP to LCC request is shown below:

<table>
<thead>
<tr>
<th>0</th>
<th>15 16 17 18</th>
<th>26 27 28 31</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DESTINATION</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S</td>
<td></td>
</tr>
<tr>
<td></td>
<td>I</td>
<td></td>
</tr>
<tr>
<td></td>
<td>BASE ADDRESS OF COMMAND</td>
<td></td>
</tr>
</tbody>
</table>

Word 1

Word 2
<table>
<thead>
<tr>
<th>Word</th>
<th>Bits</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0-15</td>
<td>Destination field, supplied by the HLP and returned in the solicited response. Used to identify the terminal and station involved in the request.</td>
</tr>
<tr>
<td>16</td>
<td></td>
<td>S-Field: Status of this request slot.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0 - This request is active or has been completed. The LCC clears this bit when it initiates this request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 - This request is ready for initiation. The HLP sets this bit when it enters a new request.</td>
</tr>
<tr>
<td>27</td>
<td></td>
<td>I-Field: Interrupt and response options.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0 - Do not send interrupt with request response.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1 - Send interrupt with request response.</td>
</tr>
</tbody>
</table>
The format of the stream request is shown below:

<table>
<thead>
<tr>
<th>0</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td>DESTINATION</td>
<td>S</td>
<td>E</td>
<td>I</td>
<td>C</td>
<td>BASE ADDRESS OF COMMAND</td>
<td>Word 1</td>
<td>Word 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Word</th>
<th>Bits</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>00-15</td>
<td>Destination Field: Required in all requests. Used to identify the terminal and station involved in the request. It is returned in the solicited response.</td>
</tr>
<tr>
<td>16</td>
<td>S-Field: Status of this request slot.</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>This request is active or has been completed. The LCC clears this bit when it initiates this request.</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>This request is ready for initiation. The HLP sets this bit when it enters a new request.</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>E-Field: Error handling option field.</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>Process the next request upon termination of present request if an error occurs.</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>If this request must be terminated due to an irrecoverable error abnormally terminate all requests for this stream if any are queued. Continue to terminate requests in this request slot until a request is found with its E flag set to zero.</td>
<td></td>
</tr>
</tbody>
</table>
### Word | Bits | Description
--- | --- | ---
| 24-27 | I-Field: Interrupt and Response Options.  
0XXX - Do not send response or interrupt when request is initiated.  
10XX - Send response when request is initiated.  
11XX - Send response and interrupt when request is initiated.  
XX00 - Do not send interrupt when request is completed.  
XX10 - Send interrupt if request terminates abnormally.  
XX11 - Send interrupt when request is completed or terminated. |
| 28-31 | Code conversion table number. |

**6.1.4.3 Commands**

Commands are 64 bits in length and give detailed information to the LCC for proper execution of a request. Only one command per request is allowed. There are two command formats, one for HLP to LCC requests and one for stream requests. The formats are shown in the following two sections.
6.1.4.3.1 Command Format, HLP to LCC

<table>
<thead>
<tr>
<th>FUNCTION CODE</th>
<th>A₁</th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>15</td>
<td>16</td>
<td>31</td>
</tr>
</tbody>
</table>

Word 1

Word 2

The use of fields A₁, A, and B depends on the function code. Generally, field A₁ defines the number of words to transfer. Field A defines the stream number. Field B is the HLP buffer address.

<table>
<thead>
<tr>
<th>Function Code</th>
<th>{Hex}</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>F₁</td>
<td>Close Line: Terminate all requests on the line specified in the request and make the line inactive. If the line is a switched line, the LCC will attempt to disconnect the terminal.</td>
<td></td>
</tr>
<tr>
<td>F₂</td>
<td>Terminate the active request on stream X defined in Field A. Field A contains the DESTINATION field of the stream request to be terminated.</td>
<td></td>
</tr>
<tr>
<td>F₃</td>
<td>Suspend operation on the LCC until the HLP responds to resume operations by setting the channel flag. <strong>NOTE:</strong> This command is found as the last entry in the Line List and is read when a channel flag is set by the HLP.</td>
<td></td>
</tr>
<tr>
<td>Function Code</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td><strong>F4</strong></td>
<td>Transfer the LCC storage to the HLP. Field A defines the beginning LCC address to be transferred. Field A1 defines the number of LCC words to be transferred. This number must be in increments of 4 words. Field B defines the 32-bit base address in the HLP.</td>
<td></td>
</tr>
<tr>
<td><strong>F5</strong></td>
<td>Configuration definition for line X. This request is used to change the TM selection mask for the line. Field B contains a 32 bit selection mask used to select up to 16 TM's on the line. Care should be exercised in the use of this command.</td>
<td></td>
</tr>
<tr>
<td><strong>F6</strong></td>
<td>Data set adapter loop request. This request instructs the LCC to loop the data set adapter; i.e. the data character supplied in the request is transmitted 256 times out of the LCC and is immediately returned on the input side of the communication line. No data is transmitted to the remote terminal. The number of unsuccessful receipts is put into the termination response which is written in the HLP. The line is then closed. It is suggested that the line have no outstanding</td>
<td></td>
</tr>
<tr>
<td>Function Code (Hex)</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>stream requests when this request is issued. This operation is the same operation initiated by the LCC when the link reaches the maximum number of error recovery attempts and loops the line. Bits 24 to 31 of Field B contain the character to be involved in the loop back operation.</td>
<td></td>
</tr>
<tr>
<td>F7</td>
<td>Data set loop request. This request is similar to the one defined above, however, it must be issued to the AT&amp;T 303 series modem or the equivalent. It is suggested that this request not be issued on a line that does not provide this function.</td>
<td></td>
</tr>
<tr>
<td>F8</td>
<td>Read code conversion table X into LCC memory. The code conversion table specified in Field A is to be read into LCC memory. Field B gives the HLP address of the 65 word 16-bit code conversion table.</td>
<td></td>
</tr>
</tbody>
</table>
### Command Format, Stream

<table>
<thead>
<tr>
<th>Function Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>Read (normal). The LCC stores the data it receives from the line into the data buffer. This is the normal read command to be used for CRT's, graphic stations, etc. When a non-error reply to the poll message sent by the LCC is received, an immediate response is sent to terminate this request whether the reply was a reject or a read.</td>
</tr>
<tr>
<td>01</td>
<td>Read wait. Similar to the operation described above except a response will be sent to the HLP only when data has been received from the terminal.</td>
</tr>
<tr>
<td>02</td>
<td>Read All. This request will be satisfied by any station on any terminal on the line for which this was issued. It is provided to reduce the amount of HLP memory that must be devoted to buffers. When data is received from a terminal it is stored in the data buffer provided by the HLP. The terminal and station</td>
</tr>
<tr>
<td>Function Code</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>02 (continued)</td>
<td>that satisfied this request will be identified in the DESTINATION field of the response to the read all. The LCC will generate the SITE ID and the STATION ID. The lower 8 bits of the DESTINATION field will be the same as those of the &quot;read all&quot; request. Data from the card reader cannot be read by the &quot;read all&quot; request.</td>
</tr>
<tr>
<td>80</td>
<td>Write. Write transparent text to the terminal. The byte count field specifies the number of 8-bit data bytes to be processed from the data buffer.</td>
</tr>
<tr>
<td>81</td>
<td>Clear Write. The write described for function code 80 is to be performed with a clear write command code sent to the terminal.</td>
</tr>
<tr>
<td>82</td>
<td>Reset write. The write operation described for function code 80 is to be performed with a reset write command code sent to the terminal.</td>
</tr>
<tr>
<td>83</td>
<td>Alert write. No data will be sent. The alert will be turned on at the terminal.</td>
</tr>
<tr>
<td>84</td>
<td>Diagnostic write. The data in the buffer specified by the command is sent to the terminal with a diagnostic write function code. The data will then be read from the terminal and placed in the buffer specified by the first 32 bits of</td>
</tr>
<tr>
<td>Function Code</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>84 [continued]</td>
<td>the buffer containing the data to be output. The 32 bits of received buffer address is not transmitted to the terminal. See buffer discussion in Section 6.1.4.4.</td>
</tr>
<tr>
<td>85</td>
<td>Graphic write. The write operation described for code 80 is performed with a graphic write command code sent to the terminal.</td>
</tr>
<tr>
<td>86</td>
<td>Write ASCII and send index. The write specified for function code 80 is performed but an E4 code will be sent in place of the normal E1 code at the end of the message. This command will erase from the terminals CRT memory any previously sent E4, and will begin writing the data in the buffer pointed to by this command at the first location after the previous E4 code. The E4 sent with this message will be stored and denotes the beginning point of the next message.</td>
</tr>
</tbody>
</table>
6.1.4.4 Buffers

The responsibility for ensuring that data buffers used with
mode 4 terminals are properly formatted belongs to the HLP and
not the LCC. Mode 4 buffers must have the following character-
istics:

1. Within a single mode 4 buffer there is only one data mode.
   A data mode is a particular format of data according to
   byte size, the arrangement of bytes within a 16 bit word,
   and the code set represented by bits. An example of a
data mode is a buffer of 8 bit ASCII characters packed
   2 per 16 bit word.

2. Data bytes are 8 bit bytes with 2 bytes per 16 bit word.

3. When writing to the CRT, the HLP data buffer must contain
   an escape code {ESC}, carriage return {CR} sequence when-
   ever line truncation is needed.

4. When writing to the line printer, the HLP data buffer must
   contain an escape code, end of line {EOL} or an escape
   code, carriage return sequence whenever line truncation is
   needed.

5. The first character in a buffer and the first character
   following an EOL or CR in a message sent to the line printer
   must be a format control code. Format control codes cause
   the printer to perform single space, double space, page
   eject or suppress space functions.
6. From 3 to 15 zeros may be compressed when writing to the line printer by inserting the ESC, "compression character" sequence in the buffer whenever compression of zeros is required. The "compression character" has a value in the range 03 {hex} to 0F {hex}, indicating the compression of from 3 to 15 zeros respectively.

7. From 3 to 31 blanks {or spaces} may be compressed when writing to the line printer by inserting an ESC, "compression character" sequence in the buffer whenever compression of spaces is desired. The "compression character" has a value in the range 23 {hex} to 3F {hex}, indicating the compression of from 3 to 31 spaces respectively.

8. Buffers must be large enough to allow an entire message from a terminal to be read into a single buffer or an entire message to be written to a terminal from a single buffer. Buffer tanking is not handled by the LCC. If the receiving buffer is not large enough to hold all of the data bytes being read from a terminal, the read request will be terminated with error due to lost data.

9. Buffers must be a multiple of 4 bytes in length. This is true of both input and output buffers.

Buffers for use in a diagnostic write operation constitute a special case. When a data buffer is output to a mode 4 terminal during a diagnostic write operation, the terminal will
immediately transmit the data it received back to the HLP through the LCC. The HLP will then compare the transmitted data with the received data to determine error conditions. Two buffers are needed to do this, although the structure of the command allows only one buffer to be specified. The LCC will handle this problem in the following manner:

1. The "base address of data buffer" field of the "diagnostic write" command will point to the buffer containing the data to be sent to the terminal.

2. The first 32 bits of the buffer containing the data to be sent to the terminal will contain the address of the buffer in HLP memory into which the data received from the terminal in response to a diagnostic write command is to be placed. The 32 bit buffer address is not considered part of the data and will not be sent to the terminal by the LCC.

6.1.4.5 Response List

The response list is a circular list whose entries are responses to requests. The list contains IN and OUT pointers used to enter and remove responses from the list. Responses are used by the LCC to indicate the occurrence of an event such as the initiated or completion of a request or the occurrence of an abnormal condition and to provide information about the event. There are two types of responses, solicited and unsolicited. Solicited responses are in reply to a request while unsolicited responses are initiated by the occurrence of an unexpected
condition on a stream or in the system. Responses are 128 bits in length and have the following formats.

6.1.4.5.1 Normal Unsolicited Response Format - Type 1

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>7</th>
<th>8</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>STREAM NUMBER {IF APPLICABLE}</td>
<td>LINE NUMBER</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>LOG</td>
<td>RESPONSE CODE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>NOT USED</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>FIELD A</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

6.1.4.5.2 Normal Solicited Response - Type 2

This response format is type 2 and is sent in response to HLP to LCC requests and to illegally formatted stream requests.

<table>
<thead>
<tr>
<th>0</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>DESTINATION</td>
</tr>
<tr>
<td>1</td>
<td>RESPONSE CODE</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>BASE ADDRESS OF COMMAND {32 Bits}</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>
6.1.4.5.2 Normal Solicited Response - Type 3

Type 3 sent in response to a stream request.

<table>
<thead>
<tr>
<th>0</th>
<th>7 8</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DESTINATION</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>LOG RESPONSE CODE</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>32 BIT BASE ADDRESS OF COMMAND</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>REXMIT COUNT</td>
<td>LINK STATUS</td>
</tr>
<tr>
<td>7</td>
<td>BYTE COUNT</td>
<td></td>
</tr>
</tbody>
</table>

REXMIT = the number of error recovery attempts on the last block.

BYTE COUNT = the number of 8 bit bytes of data transferred.

6.1.4.5.4 Abnormal Unsolicited Response - Type 4

<table>
<thead>
<tr>
<th>0</th>
<th>7 8</th>
<th>15</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LINE NUMBER</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>LOG RESPONSE CODE</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>FIELD A</td>
<td>FIELD B</td>
</tr>
<tr>
<td>7</td>
<td>FIELD C</td>
<td>FIELD D</td>
</tr>
</tbody>
</table>
There are two type 4 responses. They are response 44 {link has errored out} and 45 {intermittent SAC problems}. Response 44 uses fields B, C, D.

Field B = Link status {8 bits}; B, S, C, NE, N, T0, SAC, NA

  B = illegal data SID, not used with mode 4
  S = out of sequence block, not used with mode 4 terminals
  C = BPC or character parity error
  NE = no ending delimiter or E code
  N = NAK was received
  T0 = link time out on output
  SAC = had intermittent SAC problem
  NA = ending delimiter, not used with mode 4 terminals

  0 - ETB
  1 - ETX

Field C = DSA error results from DSA loop exercise.

  = 0 all attempts were successful
  ≠ 0 number of unsuccessful attempts

Field D = Carrier on/off status

  = 00 carrier is on and data set is ready
  = 01 carrier was on but data set not ready
  = 02 carrier was not on, data set ready
  = 03 neither carrier on nor data set ready

The "intermittent SAC problems" response uses Field A to indicate the number of intermittent SAC rejects in this pass.
### 6.1.4.5.5 Solicited Response Codes

<table>
<thead>
<tr>
<th>Code</th>
<th>Resp. Type</th>
<th>Associated Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>2</td>
<td>any stream</td>
<td>HLP solicited response on initiation of a stream request.</td>
</tr>
<tr>
<td>20</td>
<td>3</td>
<td>00</td>
<td>A not ready condition is present on the card reader (an E2 code has been received). Data from the card reader has been received.</td>
</tr>
<tr>
<td>21</td>
<td>3</td>
<td>80-82</td>
<td>A write to the printer request terminated normally with an E1 code detected. This means that the operator has depressed the &quot;intervene&quot; key and wishes to gain access to the system. The data has been printed.</td>
</tr>
<tr>
<td>22</td>
<td>3</td>
<td>00-02</td>
<td>Normal termination without irrecoverable errors for a stream request.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>80-8b</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>2</td>
<td>F4, F6</td>
<td>Normal termination without irrecoverable errors for an HLP to LCC request.</td>
</tr>
<tr>
<td>23</td>
<td>2</td>
<td>F5</td>
<td>The HLP request to change the TM selection mask {reconfigure line} is complete.</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>F2</td>
<td>The HLP request to terminate a stream has been initiated. Response code 27 or 28 will eventually be given for this stream.</td>
</tr>
<tr>
<td>25</td>
<td>2</td>
<td>F4</td>
<td>Request to transfer LCC memory into the HLP is complete.</td>
</tr>
<tr>
<td>26</td>
<td>2</td>
<td>F1</td>
<td>Close line request is initiated.</td>
</tr>
<tr>
<td>Code</td>
<td>Type</td>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>27</td>
<td>2</td>
<td>F2</td>
<td>Terminate stream request terminated normally. LCC did not find an outstanding stream request.</td>
</tr>
<tr>
<td>28</td>
<td>3</td>
<td>all</td>
<td>Stream request terminated abnormally.</td>
</tr>
<tr>
<td>29</td>
<td>2</td>
<td>all</td>
<td>Request rejected; illegal parameter.</td>
</tr>
<tr>
<td>2A</td>
<td>3</td>
<td>00-02</td>
<td>System is at buffer threshold, this request is rejected.</td>
</tr>
<tr>
<td>2A</td>
<td>3</td>
<td>80-8B</td>
<td></td>
</tr>
<tr>
<td>2B</td>
<td>4</td>
<td>F6</td>
<td>Request to initiate DSA loop is completed. The format of this response is the same as the abnormal unsolicited response number 44, except the base address of the command is in words 4 and 5.</td>
</tr>
<tr>
<td>2C</td>
<td>4</td>
<td>F7</td>
<td>Request to initiate Data Set loop is completed. The format of this response is the same as the abnormal unsolicited response number 44, except the base address of the command is in words 4 and 5. A write request sent to the printer has terminated abnormally. A not ready status has been detected on the line printer. The data sent in the request has not been printed.</td>
</tr>
<tr>
<td>Code</td>
<td>Resp. Type</td>
<td>Associated Command</td>
<td>Description</td>
</tr>
<tr>
<td>------</td>
<td>------------</td>
<td>--------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>2E</td>
<td>2</td>
<td>F3</td>
<td>The HLP request to suspend the system has been initiated.</td>
</tr>
<tr>
<td>2F</td>
<td>3</td>
<td>00</td>
<td>A read request sent to the card reader has terminated abnormally with an El code detected. This means that the operator has depressed the &quot;intervene&quot; key and wishes to gain access to the system. No data has been read from the card reader.</td>
</tr>
</tbody>
</table>
### Unsolicited Response Codes

<table>
<thead>
<tr>
<th>Code</th>
<th>Resp. Type</th>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>1</td>
<td></td>
<td>This stream is becoming active for the first time since the line has been opened. It has data to send to the HLP. The HLP is requested to set up a read request on this stream.</td>
</tr>
<tr>
<td>44</td>
<td>1</td>
<td></td>
<td>Link has errored out, all outstanding requests are being terminated abnormally. If the line is point-to-point it is closing. A multi-drop line remains open.</td>
</tr>
<tr>
<td>45</td>
<td>1</td>
<td></td>
<td>System had intermittent SAC rejects. Field A contains the current count.</td>
</tr>
<tr>
<td>46</td>
<td>1</td>
<td></td>
<td>System buffers are saturated, this line is closing.</td>
</tr>
<tr>
<td>48</td>
<td>1</td>
<td></td>
<td>Had a disastrous DSA malfunction, this line is closing.</td>
</tr>
<tr>
<td>49</td>
<td>1</td>
<td></td>
<td>Channel flag is cleared.</td>
</tr>
<tr>
<td>4A</td>
<td>1</td>
<td></td>
<td>A terminal has dialed into the line which was previously opened. The TM ordinal is in word 2.</td>
</tr>
<tr>
<td>4B</td>
<td>1</td>
<td></td>
<td>An unauthorized terminal has dialed into the line. The line is closed. An attempt has been made to disconnect the terminal.</td>
</tr>
<tr>
<td>4D</td>
<td>1</td>
<td></td>
<td>Unsolicited response, the system is in a</td>
</tr>
</tbody>
</table>
suspended state.

Unsolicited response. The line is closed.

No terminal is dialed in {dialed lines only}. The line is open. The LCC will send an unsolicited response {4A or 4B} which will identify the terminal when a terminal dials in.

Unsolicited response. The line is open.

6.1.4.b Use of the Interface by the HLP

The mode 4 capability of the LCC software has been designed to use the mode 4 terminals at a high level of performance, to take advantage of some of the unique properties of mode 4 terminals to attain this goal, and to provide a software base that will not hinder future feature development and will make such additions easier to implement. Those users who do not want to use some of the features of this product are provided with options whereby certain features can be bypassed.

Examples of this are:

1. Terminal Models and LCA parameters do not have to be passed to the system initializer at autoload time. These, and code conversion tables may be assembled into memory and portions of the SI routine assembled out.
2. Request commands that provide features not desired do not have to be issued. Examples are: suspend operation on the LCC (command function code F3), transfer LCC memory to the HLP (F4), configuration definition for a line (F5), data set loop request (F7), data set adapter loop request (F6), read code conversion table (F8), either read normal (00) read all (02) or read wait (01) can be ignored, alert write (83), diagnostic write (84), graphic write (85).

3. Responses from the LCC can be ignored such as those indicating initial activity on a station. This may result in decreased performance, however.

6.1.4.6.1 HLP's Role in Error Recovery

It is the responsibility of the LCC to detect and correct transmission generated errors. This includes parity errors, both character and message, handling of destroyed data and responses and handling of error replies from the terminal. There is another class of error from which the LCC cannot properly recovery without the assistance of the HLP. The two instances of this are:

1. Printer not ready

The printer will become not ready for a variety of reasons such as: out of paper, power off, gate open. When this occurs, operator intervention is required to clear the error condition. It will be the responsibility of the HLP to
attempt to clear this condition or take other steps to continue processing even though this may be accomplished simply by not sending messages to the printer. The most common method of clearing a printer not ready condition is for the HLP to initiate the writing of a message to the operator informing him of the printer not ready condition. Following this, the HLP should set up a read of the CRT to receive notification from the operator that the not ready condition has been cleared. Normal printer operation can then continue.

The card reader not ready condition is detected by the LCC while processing a "card read" request from the HLP. This condition occurs for various reasons: some of which are:

1. The last card has been read, hopper empty.
2. Stacker full.
3. Card jam in the transport.
4. Power off.
5. Card read error.

The condition that the last card has been read does not necessarily mean that an error has occurred. If the last card read indicates the end of file, there is no error. This condition cannot be determined by the LCC since the card data is essentially transparent to the LCC.

The HLP has the responsibility of determining whether an error has actually occurred by determining whether an end of file
condition exists or not. If an error has occurred, manual intervention by the operator will be required to clear the not ready condition. It will be the responsibility of the HLP to initiate a dialogue with the operator to clear the card reader in the manner discussed above for the line printer.

6.1.4.6.2 Intervene Key Processing

The "intervene" key is a key on the 200 UT keyboard which, when depressed, unlocks the keyboard and allows the operator to depress the "send" key. Depressing the "send" key will cause a "read" message to be transmitted by the terminal when the LCC polls the terminal. In this way the operator is able to indicate to the HLP that he wishes to gain access to the system during continuous card reading or while writing to the line printer. The LCC will detect this condition while processing a card read or write to the printer request. The LCC will notify the HLP that the "intervene" key has been depressed by sending a unique response code in reply to the request for which the condition was noted. When this occurs, the following is true:

1. If a read cards operation was requested it has not been performed and no card data has been transmitted to the HLP.

2. If a request to write data to the line printer was made the operation has been completed without error.

3. The "intervene" key has been depressed.
4. If the operator has entered data before depressing the "send" key, this data has not been transmitted to the HLP.

If the next request processed by the LCC is a read request, the message entered by the operator will be transmitted to the HLP. This information will be lost to the HLP after the LCC processes a write request. For this reason the LCC will not process a write request after an E1 condition is detected until it has processed a read request.

6.1.4.6.3 Termination of Stream Commands by the HLP

A command code {F2} has been provided to allow the HLP to terminate an active stream request. In this way, the HLP will be able to perform the following functions, if desired:

1. Set timeouts on requests and terminate those that are outstanding when a timeout occurs.

2. Terminate requests that are queued by the LCC such as the "read wait" or "read all" request.

The LCC will periodically read the HLP to LCC request slot for each line for which any of the following is true:

1. A "read wait" or "read all" command is being processed but the LCC is receiving rejects from the terminal.

2. Infinite retries on an error condition has been specified and an error condition has occurred.

3. Before reading a stream request.
When a "terminate active stream request" request is found, the following action will be taken:

1. The LCC will issue a solicited response with response code 24 to the HLP to indicate that the LCC has begun to process the terminate request.

2. Any internally queued (to the LCC) requests having a DESTINATION field the same as the DESTINATION field contained in field A of the terminate request, will be abnormally terminated. The response sent will have a response code of 28.

3. Any stream request in the Request List with the specified DESTINATION field will be abnormally terminated and a response with response code 28 will be sent.

4. If no active stream request is present in the Request List or internally queued in the LCC, a response with a response code of 27 will be sent to the HLP.
6.1.4.6.4 Auto-disconnect Requiring Action by the HLP

Because the LCC does not examine the data received or transmitted between the HLP and the terminal, it cannot detect the following conditions:

1. The terminal operator has logged out.
2. The last block of data has been transmitted in a data gathering operation.
3. The last block of data has been sent in the transmission of a file to an unattended terminal.

When conditions such as these occur, the HLP may want to disconnect the terminal for which processing is complete from the line and begin processing with another terminal. If this is true, the HLP must inform the LCC that the terminal is to be disconnected by closing the line. Processing will continue when the line is again opened.
### 6.1.5 Interface with the Terminal

#### 6.1.5.1 Error Correction Procedure with Read Commands

The following procedures will be used when executing a "read" or a "read wait" or "read all" request from the HLP.

<table>
<thead>
<tr>
<th>CONDITION</th>
<th>PROCEDURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. The LCC receives a &quot;reject&quot; response to a poll message while performing a &quot;read&quot; request.</td>
<td>The &quot;read&quot; request will be immediately terminated with an &quot;abnormal stream termination&quot; response indicating that the terminal has no data to send.</td>
</tr>
<tr>
<td>2. The LCC is processing a &quot;read&quot; request. An infinite number of retries is specified if an error occurs. An error message or a message with character or message parity errors is received in response to a poll message.</td>
<td>The terminal will be repolled until a &quot;terminate stream&quot; request is received or until a timeout occurs indicating the loss of carrier or a DSA failure. The DSA loop test will be performed and the request will be terminated with an abnormal termination response. The LCC will search for a &quot;terminate stream&quot; request after every 10 attempts to perform the operation requested. If data is received without error while attempting to recover from an error, the request will be completed with a normal termination response.</td>
</tr>
<tr>
<td>CONDITION</td>
<td>PROCEDURE</td>
</tr>
<tr>
<td>---------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>3. The LCC is processing a &quot;read&quot; request. A finite number of retries is specified if an error occurs. An error message or a message with character or message parity errors is received in response to a poll message.</td>
<td>The poll message will be repeated the number of times specified if errors persist. The DSA loop test will then be performed and the request will be terminated with an abnormal termination response.</td>
</tr>
<tr>
<td>4. The LCC is processing a &quot;read wait&quot; or &quot;read all&quot; request. An infinite number of retries is specified if an error occurs. A reject message or an error message or a message with character or message parity errors is received in response to a poll message.</td>
<td>The poll message will be retried 10 times if data is not received without error. A &quot;terminate stream&quot; request will then be searched for. If not found, the poll message will be resent 10 times. If a &quot;terminate stream&quot; request is found or if a timeout occurs, the DSA loop test will be performed and the request will be terminated with an abnormal termination response.</td>
</tr>
</tbody>
</table>
5. The LCC is processing a "read wait" or "read all" request. A finite number of retries is specified if an error occurs. A reject message or an error message or a message with character or message parity errors is received in response to a poll message.

The poll message will be repeated the number of times specified if a "read" message is not received without error. The DSA loop test will then be performed if an error condition exists. The request will then be completed with an abnormal termination response.
6.1.6 Overview

This section presents a general description of how the system operates.

Before any communications activity can begin, the LCC-11 must be autoloading with the LCC software. Autoloading is done at the instigation of the HLP and is performed under its control. The LCC resident software is loaded into LCC-11 memory through the PL-SAC Coupler. A 64 word bootstrap program is first transferred to the LCC and begins execution. The bootstrap program reads in the LCC software and system initialization routines.

The System Initialization routine (SI) processes records from the HLP containing information about the number of lines in the system, the assignment of lines to communication modes, a description of the terminals in the network in the form of Terminal Models (TM), the assignment of terminals to the lines, and data needed to establish the Line Control Area for each line. When the table area and buffers have been established, the SI will idle waiting for the HLP to set the channel flag to indicate that it has placed entries in the Line List and is ready to open lines. When this occurs, the SI routine gives control to the LCC software which begins execution.

The LCC will read each entry in the Line List, clear the channel flag indicator when the last Line List entry has been read.
and send an unsolicited response to the HLP indicating that this has been done. For each line for which a positive Line List entry was found, the LCC will send an unsolicited response indicating the line is open. If the line is a switched line, the LCC will determine if a terminal has dialed in. If no terminal has dialed into the line at this time, an unsolicited response is sent to the HLP indicating that no terminal is present. The line remains open and the LCC continues to search for the presence of a terminal on the line. When a terminal dials into the line, the LCC will attempt to identify the terminal in the following manner.

The Terminal Model Selection Mask will be used to reference each Terminal Model allowed on the line. One poll message will be sent for each Terminal Model referenced using the site code specified in the Terminal Model. If a reply was received to one of these poll messages, the LCC will complete the "open line" request with a solicited response identifying the terminal. If no reply was received, the LCC will send an unsolicited response indicating that an unauthorized terminal is present. The line is closed and an attempt is made to disconnect the terminal.

When the line is opened and the terminal on a switched line has been identified, the LCC will do the following:

1. Read the Request List for the line and process any HLP to LCC and stream requests found.
2. Determine when a terminal or station of a buffered terminal first becomes ready to transmit data to the HLP and notify the HLP with an unsolicited response.

The determination of initial activity response is sent only once for each station each time the line is opened. If the HLP ignores the response, the LCC will not repeat it. This determination is made by the LCC sending either a poll message {in the case of a terminal with 200 UT protocol} or a station status request message {in the case of a terminal using 216 protocol} to the terminal. The status of the terminal or stations will be known from the reply received. No data is transmitted to the HLP in this process and no data will be lost. When the HLP issues a request to read, the data from the terminal will be transmitted to the HLP. This procedure will be applied to all mode 4 terminals on all lines whether the lines are dedicated or switched, point-to-point or multi-point.

At any time, when it is desirable to stop activity on a line, a request to close a line may be sent from the HLP. Communication activity will be stopped, the line will be set to closed status and a response will be sent to the HLP. When reopened, the terminals on the line will again be identified and have initial activity determined as discussed above.
APPENDIX A

GLOSSARY OF TERMS
GLOSSARY OF TERMS

The following definitions, terms, and abbreviations are used in this document:

ACTIVE LINE - A communication line that has an area in the HLP's memory dedicated for requests.

ACTIVE STREAM - A stream that has a valid request in the request list.

ALPHA {α} Information Trade - See Section 4.2.1 for a complete definition.

BC - See Buffer Controller.

Beta {β} Information Trade - See Section 4.2.1 for a complete description.

Block - A transmission block is a group of bits, characters, or bytes that are transmitted on a link as an entity by use of the appropriate link control procedure.

BPS - Bits Per Second.

Buffer Controller - A 16-bit parallel digital computer with a 200 nsec or 1.1 usec memory cycle time.

Byte - 8 bits.

CHANNEL FLAG - Communication from the HLP to the LCC is initiated by the HLP sending a pulse on the trunk control strobe and by sending a channel flag control code on the control lines.

Closing Delimiter - A character sequence which defines the end of a transmission block.

Correct Transmission - A transmission whose opening delimiter, closing delimiter are recognizable {i.e., valid} and cyclic redundancy check is not in error.

DATA - Data is those characters which make up a message.
Glossary Continued....

Data Framing
Characters - ASCII control characters that delimit the data and indicate its transmitted mode, SOH, STX, TSTX.

Data Set - A piece of hardware which modulates/demodulates data signals in a link.

Data Set Adapter {DSA} - A hardware module that acts as an interface between a data set {modem} and normal channels. During input operations, the DSA assembles the serial bit stream furnished by the data set into 8-bit characters. During the output operation, the reverse disassembly occurs.

DOMINANT - Stations on a link that initiate all activity. The subordinate must always reply to the dominant, the reverse is not always true.

DSA - See Data Set Adapter.

EOM - End of Message.

Full Duplex - A hardware capability of transmitting and receiving on a link simultaneously.

HLP - Higher Level Processor.

Information {Status} Block - A transmission block which contains only stream identifiers and their status {i.e.: SID}.

Interpretable Response - A response which has valid opening and closing delimiters {i.e., can be interpreted}.

INPUT Stream - Input streams are those simplex flow of data which are received by the LCC {i.e., the LCC inputs the data}.

LCC - Local Communication Controller.

LINK - A communication channel and communication control logic in all directly connected stations.
Glossary Continued....

Message  - A logically related group of data elements that must be switched or processed as an entity. In this application, a message is always ended by an ETX or TETX.

Modem  - See Data Set

Non-Continuous  - Every block transmitted must have a response before another block is transmitted in the same direction.

Non-Switched Line  - A communication channel which has both ends connected point-to-point. The line does not go through a telephone switch network.

Off-Line  - A remote station is off-line when it does not service the communication line, but does some outside utility.

Opening Delimiter  - A sequence of characters which define the beginning of a transmission block.

Output Stream  - Output streams are those simplex flow which are transmitted by the LCC (i.e., the LCC outputs the block).

Poll  - To quiz the remote for activity. A poll is a signal from the LCC to the remote, asking the remote if it has anything to send.

Point-To-Point  - Only two data sets are connected on a communication line.

Preemptive Priority  - A request which is to be processed before all others, even those which have been in the system longer.

SID  - Streams Identification with Status.

SINK  - The receiver of information.

SOURCE  - The sender of information.

Station Control Information  - Information about the status or activity condition of a stream.

Stream  - A simplex flow of data.
Glossary Continued....

**Subordinate** - A station on a link that responds to communications and does not initiate communications.

**Text** - That portion of a message that contains user data.

**Transparent Data** - Data which uses all 8-bits of a byte. There is no character parity.

**Two Way Simultaneous** - A mode of operation supporting both an outgoing transmission and an incoming transmission simultaneously. A full duplex line is a requirement for two-way simultaneous operations.

**Valid Transmission** - A transmission whose opening delimiter and closing delimiter are recognizable.

**WORD** - 16-bits of information, two bytes.

**XSN** - Transmission sequence number. ASCII graphic character that forms part of the opening prelimier in Mode 2 procedure.