Burroughs

B 1700 Systems
Data Communications

INFORMATION MANUAL

RELATIVE TO MARK V.0 RELEASE

PRICED ITEM

Printed in U.S.A.

December, 1975

1089992

Burroughs

B 1700 Systems
Data Communications

INFORMATION MANUAL

RELATIVE TO MARK V.0 RELEASE

PRICED ITEM

Printed in U.S.A. December, 1975 1089992

Burroughs believes that the information described in this manual is accurate and reliable, and much care has been taken in its preparation. However, no responsibility, financial or otherwise, is accepted for any consequences arising out of the use of this material. The information contained herein is subject to change. Revisions may be issued to advise of such changes and/or additions.

Correspondence regarding this document should be forwarded using the Remarks Form at the back of the manual, or may be addressed directly to Publications Department, Technical Information Organization, TIO-West, Burroughs Corporation, 9451 Telstar Avenue, El Monte, California 91731.

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

GENERAL

The purpose of this manual is to provide a functional description of the Burroughs B 1700 Data Communications (Data Comm) System.

It is assumed that the reader has a working knowledge of the basic hardware and software functions of the B 1700 Data Processing System.

NOTE

The Data Comm system is composed of both hardware and software components.

Appendix A is a glossary of terms used in this manual for the B 1700 Data Comm system

RELATED DOCUMENTS

Some of the details of the B 1700 Data Comm system and its composite elements are contained in other B 1700 Systems manuals. The following is a list of related B 1700 manuals referenced in this manual.

B 1700 Systems Network Definition Language (NDL) Reference Manual, Form Number 1073715.

B 1700 Systems Software Operational Guide, Form Number 1068731.

B 1700 Systems Reference Manual, Form Number 1057155.

B 1700 Systems User Programming Language (UPL) Reference Manual, Form Number 1067170.

B 1700 Systems Report Program Generator (RPG) Reference Manual, Form Number 1057189.

B 1700 Systems COBOL Reference Manual, Form Number 1057197.

2. DATA COMMUNICATIONS CONCEPTS

GENERAL

The term "data communications" (Data Comm) as used in this manual denotes the bidirectional transfer of data between a central computer and one or more terminals. Data Comm is the interfacing of remote users and devices (terminals) with on-site functions performed by the central computer. The B 1700 Data Processing System can function as the central system for a network of remote devices (terminals) or as a satellite computer to a large system. Figure 2-1 illustrates a typical Data Comm network.

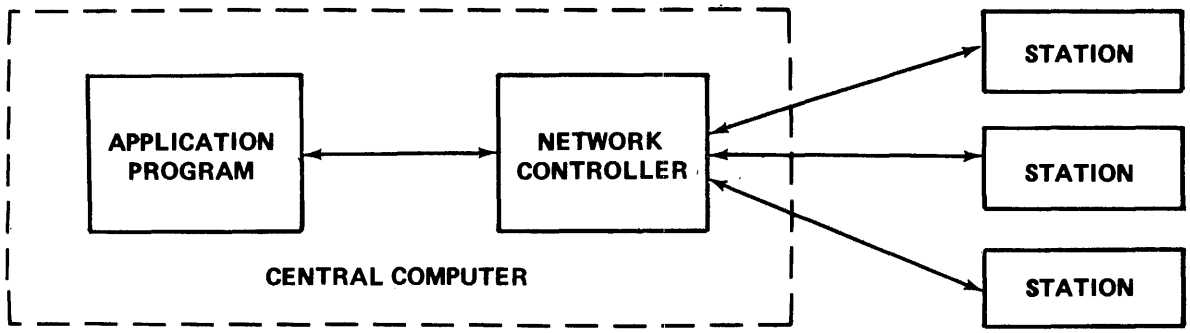


Figure 2-1. A Typical Data Communications Network

A Data Comm network includes a central computer and the remote devices which are attached to it (i.e., lines, adapters, terminals). Terminals vary in how they communicate with the system. The specific type of terminal which can range from a "Teletypewriter" unit to another computer, the distance of the terminal from the central computer, and the transmission lines used vary among Data Comm networks.

SYSTEM REQUIREMENTS

The configuration of a Data Comm system is determined by the functions it is intended to perform. Some considerations in designing an efficient Data Comm system are:

a. The volume of information to be transmitted over the network, including the number of messages and their lengths.

b. How often messages will be transmitted, at what time intervals, and at what time of day.

c. At how many points data will originate or need to be distributed.

d. Whether speed of data transmission (rate) and elimination of delay are important elements of the system design.

e. How reliable the system must be.

Other factors require care in planning, such as capability of system expansion, increased volume of message traffic, programming requirements, and personnel requirements.

DATA COMM APPLICATIONS

The applications of Data Comm systems are numerous and varied in function, and the hardware and the computer programs being used are also varied.

The major applications of Data Comm systems are briefly described below.

a. Inquiry processing or on-line processing makes use of central data files that can be accessed by inquiries from remote locations. A prompt response can be provided for each inquiry. The data transmission is bidirectional and can be combined with real-time file updating, where a transaction causes the central data files to be immediately updated to show the current status of the appropriate accounts.

b. Data collection involves the collection of data by unidirectional data flow from remote terminals to a central processing point.

c. Data distribution uses unidirectional data flow to transmit data that has been generated at a central processing point to terminals. Data collection and data distribution are combined in many Data Comm systems.

d. Message switching is a method of handling large amounts of data to be dispersed among widely separated locations. The data flow is bidirectional between a central switching center and the terminals connected to it. The central switching center may perform processing of the data or any code conversion that may be required.

e. Computer time sharing allows users who are at remote locations from a central computer to use it simultaneously. The data flow is bidirectional, with data and the instructions for processing transmitted from the remote terminals to the central computer. After the desired processing is performed, the results are transmitted back to the terminal. The main advantage of time sharing is that the user at a remote location has fast access to the computer.

DESIGN CONSIDERATIONS

The design of the B 1700 System Data Communications was partly influenced by the considerations described in the following paragraphs of this section of the manual. Changes in the Data Comm network can be accommodated both quickly and easily by the user of the Data Comm system. The system configuration may require changes necessitated by changes in the user's application. Typical examples include changing the number and connection of lines and stations to serve geographic changes in the network, or expanding use of the network. Different types of terminals usually require different line discipline routines. New requirements imposed on the system may require changes in the communication techniques used with certain stations.

FILE-HANDLING TECHNIQUES

File-handling techniques are the method by which application programs communicate with stations or sets of stations (families). Symbolic file names, together with the appropriate OPEN, CLOSE, READ, or WRITE commands are used for most communication with terminals. The use of file-handling techniques is preferred for the following reasons:

a. Application programs usually do not have to be rewritten when system software, hardware, or system configurations are changed.

b. Application programmers need not be data communications experts in order to utilize data communications networks.

USER CONTROL

The user of a Data Comm system should be allowed, when desired, to control the Data Comm network at a fundamental level. Such control is necessary because a generalized system software package cannot efficiently control a Data Comm network without considering the total user environment. A Data Comm system cannot anticipate the error recovery and auditing processes required and used within each user's environment. Another reason for allowing user control is that a Data Comm system cannot anticipate all special communications or formatting techniques to be used with particular remote devices.

If the user were not allowed some control of the Data Comm network, the result could be a limitation of the use of the system by its users, compromises unacceptable to most users, and a tailored Data Comm system that would be difficult to maintain.

DATA COMM SYSTEM INDEPENDENCE

The system design also established that the Master Control Program (MCP), which is the B 1700 Operating System, should be as independent of the Data Comm system as possible. This independence allows the operating system to be smaller in size than it would be otherwise, without penalizing users not having a Data Comm system. The functions of the Data Comm system and the B 1700 operating system are separate and distinct, and maintaining them as separate entities is advantageous to both.

The Single Line Control (SLC) and Multi-Line Control (MLC) handle communication overhead through terminals rather than by the central processor. (Refer to Section 3.)

3. B 1700 SYSTEM DATA COMMUNICATIONS HARDWARE

The B 1700 Data Communications System is a combination of special hardware and system software that performs data communications. The special Data Comm hardware consists of terminals, transmission lines, data sets, adapters, and I/O controls, all of which are described in this section of this manual. Control codes and I/O descriptors, although not hardware, are also described in this section because of their relationship to the special hardware of the Data Comm system.

TERMINALS

Terminals are the electronic or electromechanical devices that can be attached to the transmission lines of a Data Comm system, providing a man/machine interface between the user of the network and the network itself.

When information is sent from a terminal to the central system, the terminal transmits the bit representation of the character through the transmission line. A B 1700 Data Processing System can serve as a terminal itself. It may be connected to a Data Comm system that has another computer functioning as the central system.

The following definitions of terminal and station are provided to aid the reader in understanding the subject of data communications.

Definition of the Term "Terminal"

A terminal is defined as any device that can transmit or receive data through the communications lines of a Data Comm system.

Definition of the Term "Station"

A station is a logical concept, unlike a terminal which is a physical concept. Every station associated with a Data Comm system must be described in the Network Definition Language (NDL) source program. A station is "described" to the Data Comm system by assigning a set of attributes to it. Every station must be given a name (i.e., a "station identifier" attribute) and a terminal species attribute. The station identifier is a logical attribute and the terminal is a physical attribute. Other logical and physical attributes are assigned to each station to define how the Data Comm system should interact with each station.

A station can be associated with a Data Comm line (explained later in this section), but it need not have such an association. However, until such an association is made, the B 1700 Data Processing System cannot communicate with a station. The line to which the station is associated (if any) is not a station attribute. A station may be moved dynamically (i.e., during the continuous operation of the Data Comm system) from one line to another (or from one line to no other line) and still be considered the same station.

A "multiple station line" is a Data Comm line that can have more than one station associated with it at any instant in time. Every station that can be associated with a multiple station line must have a "station address" attribute assigned to the station. Every station assigned to a multiple station line must have identical station address formats (e.g., one-character address, two-character address, or three-character address). All of the stations associated with a multiple station line, at any instant in time, must have unique station addresses.

The B 1700 System Data Communications can communicate with a particular station attached to a multiple station line, but only if the station is associated with the line and only if the logical station address and the physical terminal address are identical. The station address is determined by the field engineer or, if the terminal is programmable, by what the programmer has specified the address to be. The Data Comm system does not attempt to communicate with a terminal, but does attempt to communicate with a station.

TRANSMISSION LINES

The physical connection of the hardware components of a Data Comm system is usually accomplished by transmission lines; for example, telephone lines. These transmission lines may be either normal switched lines or special leased lines.

Switched lines use the switching equipment of telephone exchanges for message routing; consequently, the physical path of the transmission can vary widely. Leased lines are permanent or semipermanent connections and are advantageous in that they usually permit data transmission at a higher speed than switched lines permit.

For data transmission over short distances, as when terminals are in the same building, ordinary twisted-wire pairs can be used. This type of connection is referred to as direct-connect.

DATA SETS

In most Data Comm systems, data sets must be used to transform digital signals, from a computer or terminal, to analog signals suitable for transmission. This transformation is referred to as modulation/demodulation and has no overall effect on the system.

ADAPTERS

An adapter is an electronic interface between a data transmission line and a part of the Data Comm system. Adapters perform character assembly, timing comparison, block parity checking, block parity generation, and other logic operations. All adapters operate in half-duplex mode and present the same interface to the Single Line Control or Multi-Line Control.

The adapters are designed to operate over the communication facilities defined in table 3-1.

Table 3-1. Transmission Facility Classification

Designation	Bandwidth	Transmission Rate	Remarks
Narrowband	Variable. Usually up to 300 Hz.	150–300 bits/second.	Usually a leased line.
Voiceband	Nominal 4 kHz.	600/1200/1800/2000/ 2400/4800/9600 bits per second.	Leased and switched lines.

All adapters accumulate one character of data before notifying the I/O control. The three types of standard adapters are described in the following subsections.

Standard Synchronous Adapter

The Standard Synchronous Adapter provides for a connection of terminals via the following services and Burroughs data sets. Data sets equivalent to those listed below can also be used. Both an RS232C and a CCITT interface are provided. The clock pulse is supplied by the data set. Table 3-2 indicates the transmission speeds and line service applicable for several Burroughs data sets.

Table 3-2. Standard Synchronous Adapter Data Set Interface

Transmission Speed (BPS)	Service	Burroughs Data Set
2000	Switched	TA 734-24
2400	Leased	TA 734-24
4800	Leased	TA 733-48

Standard Asynchronous Adapter

The Standard Asynchronous Adapter provides for a connection of the terminals by the services and Burroughs data sets listed in table 3-3. Data sets equivalent to those listed can also be used. Both an RS232C and a CCITT interface are provided. The adapter is "strapped" (hardwire-wired) by a field engineer to accommodate any one of the listed transmission speeds. Table 3-3 indicates the transmission speeds and line service applicable for several Burroughs data sets.

Table 3-3. Standard Asynchronous Adapter Data Set Interface

Transmission Speed (BPS)	Common Carrier Service	Burroughs Data Sets
150	Switched	TA 714
150	Leased	TA 713/753
300	Switched	TA 714
300	Leased	TA 713/753
600	Switched	TA 714
600	Leased	TA 713/753
1200	Switched	TA 714
1200	Leased	TA 713/753
1800	Leased	UA 783

Standard Direct Asynchronous Adapter

The Standard Direct Asynchronous Adapter provides for a connection of terminals by a two-wire direct interface. The adapter is wired by a field engineer to accommodate any one of the transmission speeds shown in table 3-4.

Table 3-4. Standard Direct Asynchronous Adapter – Cable Lengths

Transmission Speed (BPS)	Cable Length
150	Up to 1000 feet
300	Up to 1000 feet
1200	Up to 1000 feet
1800	Up to 1000 feet
2400	Up to 1000 feet
4800	Up to 1000 feet
9600	Up to 1000 feet

In all cases the transmission code is ASCII-7 1967. The seven data bits are transmitted least-significant bit first, followed by one parity bit. The character parity is odd for synchronous operation and even for asynchronous operation.

Control Codes

Control codes are recognized by the adapters and are used in the Data Comm system for character synchronization, to indicate where a message begins and ends, and as a response to indicate the success or failure of the last transmissions.

The adapters recognize the control codes listed in table 3-5. The corresponding function of each control code is listed in the table.

Table 3-5. Control Codes for Standard Adapters

Function	Control Code
Synchronization	SYN
Start	SOH, STX
Ending	ETX, ETB
Positive Response	ACK, ENQ, BEL, ETX, ETB
Negative Response	NAK, EOT (optional), TIMEOUT (optional)

The following is an explanation of control code functions.

SYNCHRONIZATION CONTROL CODE

Synchronization control codes (SYN) are not stored in memory. Two SYN control codes must be received before a Standard Synchronous Adapter will recognize any other control code. After receipt of the last leading SYN control code, an adapter will store all subsequent control codes in memory, except SYN, up to but not including a Block Check Character (BCC). A synchronization code is not included in a BCC sum. At least four synchronization control codes must be transmitted prior to data transmission. These codes are automatically generated by the adapter prior to the data transmission.

START CONTROL CODES

The start control code (SOH or STX) is used to start the accumulation of a BCC with the next non-synchronization character received by the adapter. After the receipt of a start control code, only those control codes classified as ending control codes terminate the operation.

ENDING CONTROL CODES

The ending control code (ETX or ETB) is used to finish the transmission of the message and end the accumulation of the BCC. Following the ending control code, the BCC will be the next character transmitted if the I/O operation is a write. If the I/O operation is a read, the BCC will be the next character expected after the ending control code is received.

POSITIVE AND NEGATIVE RESPONSE CONTROL CODES

A positive response control code, unless also classified as an ending control code, does not terminate an I/O operation if the positive response code is preceded by a start control code.

A control code that is classified as a positive response or a negative response is defined to be an ending control code, but only if it is not preceded by a start control code.

An option is provided to ignore EOT (End of Transmission) as a negative response control code for read and write operations. On read operations this option allows the I/O control and the system to which it is attached to act as a terminal, and to receive a polling sequence from other data communication systems. No hardware recognition of addresses is provided. On write operations this variant allows the I/O control to ignore EOT in order to transmit polling sequences and other sequences which use EOT as the first control code in a message as a line-clearing code.

Block Check Character (BCC)

A Block Check Character is normally transmitted following all messages that contain a start control code. The BCC is formed by taking the modulo 2 sum for each data bit following a start control code, up to and including an ending control code. The BCC is transmitted after the ending control code with appropriate character parity. A field engineer can wire the adapter to exclude the generation and checking of a BCC.

I/O DESCRIPTORS

The I/O descriptor is the basic communications link between the B 1700 system software and hardware. Figure 3-1 illustrates the I/O descriptor format used in Data Comm; all fields are 24 bits in length.

E FIELD = ENDING ADDRESS
RS FIELD = RESULT STATUS INFORMATION
L FIELD = LINK ADDRESS POINTING TO THE RS FIELD OF THE NEXT I/O DESCRIPTOR IN THE LIST
OP FIELD = OPERATION, VARIANTS, AND ADAPTER NUMBER
A FIELD = STARTING ADDRESS OF DATA
B FIELD = ENDING ADDRESS OF DATA

Figure 3-1. I/O Descriptor Format

I/O Descriptor Field Description

Some of the fields of the I/O descriptor are set up or used by B 1700 system software or B 1700 hardware; some of the fields are used by both software and hardware.

OP FIELD

The operation requested, any variants to the operation, and the adapter identification number (if applicable) are contained in this field. The OP field must be set up by the system software before an I/O operation is initiated.

The high-order (leftmost) three bits of the OP field contain the operation code for either a READ, WRITE, BREAK, PAUSE, STOP, or TEST. The next 17 bits are used for variants of the operation. For example, a write operation and a write with translation operation would have the same operation code, but the write with translation operation would have a variant bit set to cause the operation to be modified.

The low-order (rightmost) four bits of the OP field identify the unique adapter connected to the I/O control for this operation.

A FIELD

When the operation specified in the OP field requires character transfers to or from memory, the A field must contain the starting address in memory where the I/O control can fetch or store characters. When the A field is required, it must be set up by the systems software before it initiates an I/O operation. The A field must define a non-zero, integer number of eight-bit bytes.

B FIELD

The B field is required whenever the A field is required. The B field contains the address of the last character position in memory that the I/O control can access. The B-field contents (ending address) must be greater than the A-field contents (beginning address). On read operations the B field must be large enough to provide space for all characters expected to be read. When the character specified by the B-field address has been transferred, the I/O control terminates the operation.

L FIELD

The L field holds the address of the RS field of an I/O descriptor used for I/O descriptor linking. Linking is a technique used to execute two or more I/O operations consecutively without intervening system software execution. However, the link could be to the RS field of the same descriptor. Both the SLC and MLC (explained later in this section) require that all I/O descriptors be linked, except when the stop operation is used. Upon completion of the I/O operation (if the operation was not a stop operation or if no error occurred), the I/O control fetches the address of the RS field of the next I/O descriptor from the L field. The I/O control then checks the operation complete bit of the next I/O descriptor, and initiates the next operation requested if the bit is FALSE.

Figure 3-2 illustrates an example of I/O descriptor linking.

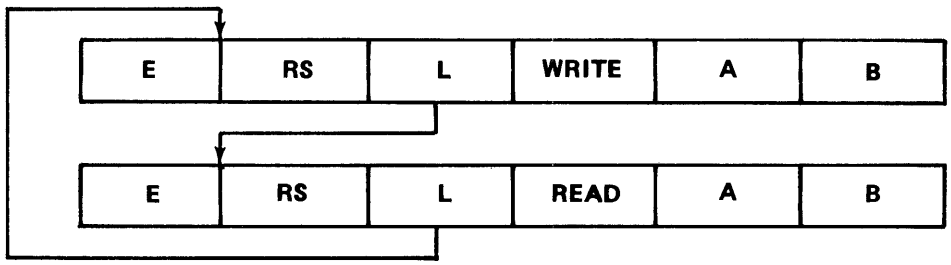


Figure 3-2. I/O Descriptor Linking

A simple write operation linked to a read operation, which is very common in Data Comm operations, is illustrated by figure 3-2. Assume that the system software has cleared the RS field, set up the L, OP, A, and B fields, and stored the address of the RS field of the write descriptor in absolute memory location 0.

Upon receiving an interrupt, the I/O control fetches the RS address of the write I/O descriptor, and the operation requested is initiated, since the operation bit is FALSE.

When the write operation is complete and if no error occurred, the I/O control fetches the address of the read descriptor from the L field of the write descriptor. Since the RS field had been previously cleared, the read operation is initiated.

When the read operation is complete and if no error occurred, the I/O control fetches the address of the RS field of the write descriptor from the L field of the read descriptor. Because the write operation had been completed previously, the I/O control does not wait for the operation complete bit to be reset.

The linked list of I/O descriptors shown in figure 3-3 is similar to those shown in figure 3-2. The read descriptor is linked to the descriptor for a stop operation, which causes the I/O control to stop and not perform further linking.

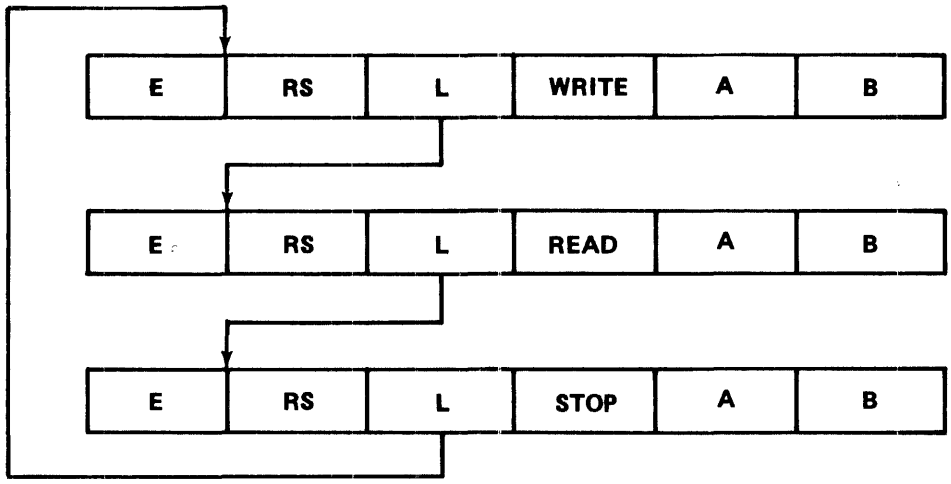


Figure 3-3. Linking to a Stop Operation

Figure 3-4 illustrates an operation using only one descriptor that is linked to itself. As a result, the I/O control pauses after completing the operation.

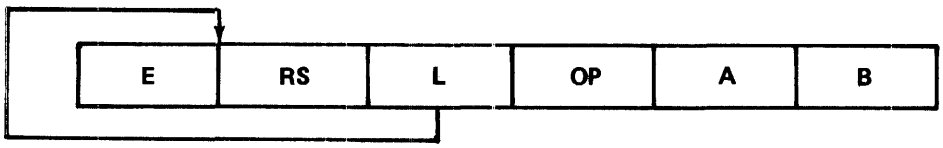


Figure 3-4. Linking to the Same Descriptor

E FIELD

The E field is initialized and used by the hardware during the operation and can be used by the system software upon completion of the operation. For read and write operations, the hardware stores the incremented A-field address in the E field. After a write operation, the E field normally points to the next bit following the ending code. After a read operation, the CSM stores the final incremented A-field address pointing to the memory location where the next character of data would normally be stored.

Upon completion of a read or write operation, the systems software can use the E field for message length computation. For example, the E-field address less the A-field address divided by 8 equals the number of characters read or written.

RS FIELD

The I/O control uses the RS field for three functions:

a. To determine when to initiate the operation.

b. To provide temporary storage during execution of an operation.

c. To provide storage of the result status information after completion of an operation.

Prior to and during an operation, the first 15 bits of the RS field are used to store temporary flags for the CSM, and the last nine bits are used to indicate dynamic interrupt information.

The systems software must clear the result status bits before initiating the I/O operation, because the I/O control will not initiate an I/O operation when the operation complete bit is TRUE (1). The operation complete bit is the first bit of the result status bits and is used by the I/O control in I/O descriptor linking.

After completion of an operation and after storing the actual ending address in the E field, the result status information is stored in the RS field.

The meaning of the individual bits of the result descriptor stored in the RS field of the I/O descriptor is defined in table 3-6.

Table 3-6. Result Status Bits for Standard Adapters

Bit	Meaning
0	Operation complete
1	Exception condition
2	Reserved
3	Parity error
4	Memory access error (Read)
5	Memory access error (Write)
6	TIMEOUT
7	BREAK received
8	No ending control code
9	Chaining terminated
10	Not used
11	Loss of "Data Set Ready"
12	Loss of "Clear to Send" (Write)
	Loss of "Carrier" (Read)
13	Not used
14	Not used
15	Reserved
16	Operation complete
17	Data communications device
18–23	Adapter identification

I/O CONTROLS

Two types of Data Comm controls are used with B 1700 System Data Communications: the Single Line Control (SLC) and the Multi-Line Control (MLC). To the Data Comm system software both I/O controls function the same and have the ability to access the adapters and main memory. Refer to the B 1700 Systems Reference Manual, Form No. 1057155, for information concerning main memory and control memory.) The I/O control itself is sensitive to I/O commands and accumulates characters from adapters to build messages. The I/O control processes the accumulated characters, and initiates various functions as specified by the system software.

The I/O controls can interrupt the system software for notification of I/O completion. After executing an I/O operation, the I/O control stores a 24-bit result descriptor in the Results Status (RS) field of the I/O descriptor that indicates the results of the operation. This I/O descriptor can then be used by the system software to determine what action, if any, should be taken.

On write operations, the I/O controls have the capability to translate internal machine code of the B 1700 to ASCII code, and on read operations, from ASCII code to EBCDIC code.

The I/O controls can execute the AUTOPOLL function, which allows the I/O control to continuously poll (request input from) stations on a line without interrupting the Master Control Program.

The Single Line Control and Multi-Line Control each operate with all defined adapters for the B 1700 system.

Single Line Control

GENERAL

Figure 3-5 shows the relationship of the Single Line Control to pertinent hardware units of the B 1700 system.

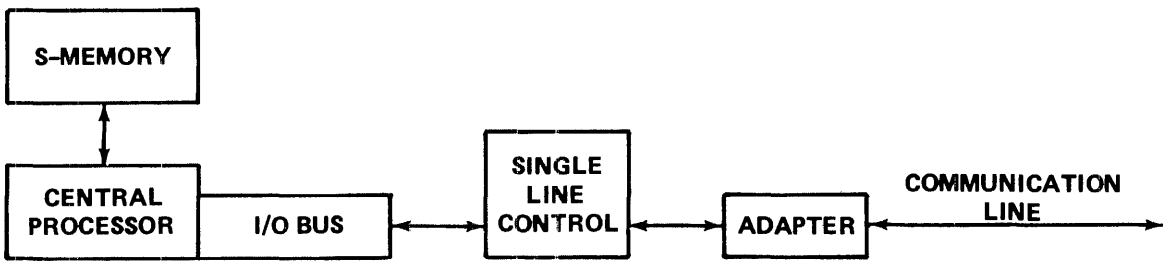


Figure 3-5. B 1700 System Data Communications Hardware Organization – SLC

The Single Line Control and one adapter service one communication line. The SLC provides two buffers of 13 bytes each, which are used in a cyclic manner to receive data during data transfer operations. Any data loss resulting from a buffer being unavailable is flagged as a memory access error.

Data is transferred between the system and the I/O control in 8-bit parallel fashion. The Single Line Control operates with a system clock frequency of 2, 4, or 6 megahertz.

FUNCTIONS OF THE SINGLE LINE CONTROL

The Single Line Control provides the following basic functions for each adapter:

a. Data buffering.

b. Interface to CSM.

c. Reference address storage for an adapter.

d. Initiation of an adapter to an appropriate operating mode.

e. Servicing requests for character transfer.

f. ASCII-7 to/from EBCDIC translation as selected by variant.

g. Result status reporting.

h. AUTOPOLL.

FUNCTIONAL DESCRIPTION OF THE SINGLE LINE CONTROL

The Central Service Module (CSM) assumes that a linked list of I/O descriptors is available to itself. The CSM is initiated by the receipt of a 24-bit address which points to the RS field of an I/O descriptor. The CSM inspects the first two bits of the RS field and, if they are equal to 00, exchanges them with the literal 01 from a storage area. If the exchanged bits are still equal to 00, the CSM initiates the I/O control with the indicated operation. If the first two bits of the RS field are not equal to 00, the CSM initiates a pause operator to the I/O control. After the completion of the pause operator, the CSM repeats the inspection on the same I/O descriptor. After completion of an operation, this procedure is again repeated by using the link address as a new start address.

The following are exceptions to the above.

a. If the result had an exception condition, the CSM forces the I/O control go go idle after returning a result and an interrupt.

b. If the CSM receives a stop operation code, the CSM forces the I/O control to go idle after returning a result, and returns an interrupt if applicable.

c. If the CSM receives a memory parity error signal during the fetch of an I/O descriptor (RS, OP, A, B, C, RS swap, or link fields), or during the fetch of the start address, the CSM sends a special interrupt message to port number 0, channel number 15, consisting of an address pointing near the field in error. The CSM then exits.

At the completion of an operation and after the actual ending data address is stored, the information in the RS field is exchanged with the result status bits from the operation.

The information read from the result status field includes one interrupt request bit, one high-interrupt bit, a 3-bit port number, and a 4-bit channel number. If the interrupt request bit is TRUE, the CSM generates an appropriate interrupt message to the port indicated.

If the interrupt request bit is FALSE, the CSM generates an interrupt message only if the result status bits returned had exception bit number 2 set.

The interrupt message is a 24-bit address pointing between the result status bits and the link address. The channel number contained in the result description status area (normally the channel number of the I/O control which received the I/O descriptor) is also returned to the port indicated.

Figure 3-6 shows the location of the various bits (that are inspected by the CSM) in the result status area prior to storage of the result.

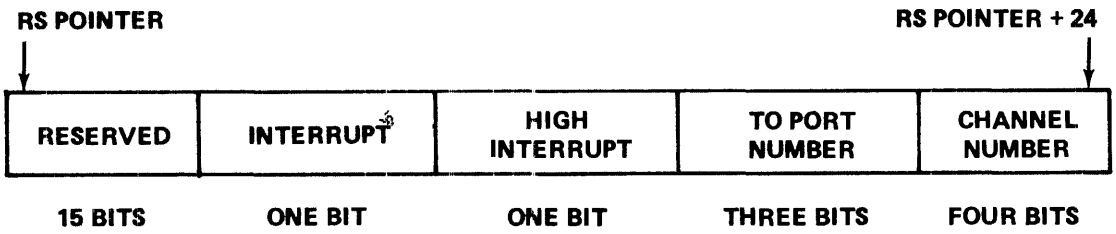


Figure 3-6. Result Descriptor – Single Line Control

When the Single Line Control sequences through the status states, which involve the returning of a result, various bits in the result field are used by the CSM to enable or disable the storage of the result and to take other action. These bit combinations and meanings are as follows.

Bit Combinations			Meaning to CSM
17	23	24	
0	0	0	The CSM does not store a result, but proceeds to the next I/O descriptor. The combination of bits 17, 23, and 24 is used by a read operation (poll) after a negative response with no errors.
0	0	1	The CSM does not store a result, but moves the starting address from the A field into the E field if the final address in the E field is equal to the terminating address in the B field. The CSM then proceeds to the next I/O descriptor. The combination of bits 17, 23, and 24 is used by a write operation (poll) in order to restore the pointer to the beginning of the poll list in the absence of error conditions.

Multi-Line Control (MLC)

GENERAL

The Mulit-Line Control services up to 16 communication lines and requires one adapter per line. The Multi-Line Control can be used only with the B 1720 Series systems. The first eight lines and adapters are serviced by the B 1352 Multi-Line Control, and an additional eight lines and adapters can be serviced by the B 1353 Multi-Line Control Extension, which can be cabled to the B 1352 Multi-Line Control.

Figure 3-7 shows the relationship of the Multi-Line Control to pertinent hardware units.

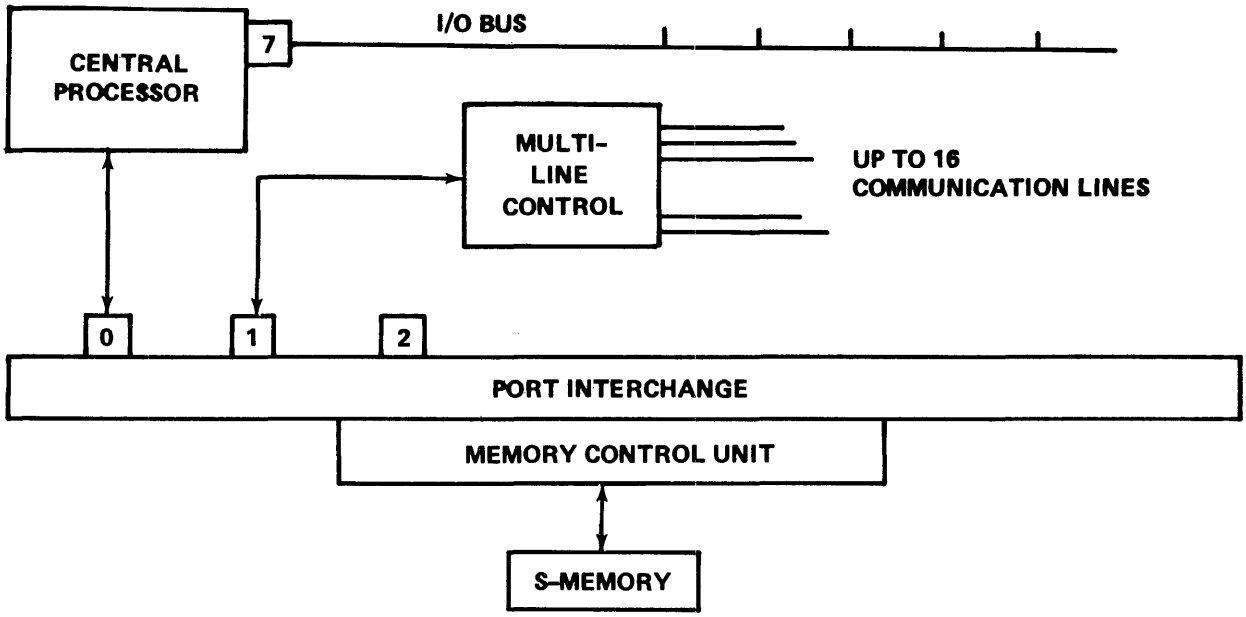


Figure 3-7. B 1700 System Data Communications Hardware Organization – MLC

The Multi-Line Control operates on a 6-megahertz clock frequency. The MLC employs a scanning technique to detect adapters that require servicing. The time required for servicing depends upon the number of adapters simultaneously requesting service and the position of the requesting line relative to the scanning pointer. The scanning is continuous, and requires approximately 32 system clock pulses to scan a complete set of 16 adapters without servicing any of them. Servicing an adapter takes between 2 and 10 system clock pulses. To detect and service an adapter requiring attention takes an average of 24 system clock pulses.

The Multi-Line Control employs scanning in servicing memory accesses for the buffers in a control. Data is stored/fetched, eight bits per memory access, in parallel fahsion.

One character of buffering per adapter is provided in the Multi-Line Control, in addition to the one character accumulated by the adapter.

FUNCTIONS OF THE MULTI-LINE CONTROL

The Multi-Line Control provides the following basic functions for each adapter:

a. I/O initiate handling.

b. I/O descriptor fetch for an adapter.

c. L, A, and B descriptor address storage for an adapter.

d. Adapter initiation to an appropriate operating mode.

e. Servicing requests for character transfer.

f. ASCII-7 to/from EBCDIC translation as selected by variant.

g. Result status and interrupt message-handling.

h. Linking.

i. AUTOPOLL.

FUNCTIONAL DESCRIPTION OF THE MULTI-LINE CONTROL

Input/output is initiated by the receipt of an interrupt signal indicating that a 24-bit message is present starting at absolute memory location 0. The message is an address that points to the RS field of an I/O descriptor. The MLC fetches this address by executing a read and clear operation.

The Multi-Line Control then fetches the descriptor that directs the I/O operation to the adapter specified in the OP field of the I/O descriptor.

The initial I/O descriptor sent to an adapter must be ready for execution. The operation complete bit of the RS field must be set to 0.

Following execution of the initial I/O descriptor, the MLC fetches the link address which points to the RS field of the next I/O descriptor in the chain. The MLC then checks the operation complete bit in the RS field of this descriptor. If OC = 0, the MLC fetches the remainder of the descriptor and performs the operation indicated by the OP field. If OC = 1, a pause and inspection procedure is performed at 8- to 12-millisecond intervals until OC = 0. This procedure is repeated after the completion of each operation.

The following are exceptions to the above conditions and operations.

a. If the result had an exception condition, the MLC for the applicable adapter goes idle, after returning the result status information and an interrupt.

b. If the MLC receives a stop operation code, it goes idle for that adapter, after returning a result and an interrupt if applicable. All adapters must be idle for the MLC to go idle.

c. If the MLC receives a memory parity error signal during the fetch of any I/O descriptor fields or during the read and clear message operation, the MLC sends a special interrupt message to port number 0, channel number 15, consisting of an address pointing 24 bits beyond the field in error.

The MLC accepts all I/O initiate interrupts directed to it, regardless of the state of an adapter. If an I/O initiate is directed to a busy adapter, the MLC forces the adapter to the new operating state, thereby destroying all traces of the previous operation, including a pending interrupt message.

After the completion of an operation and after the actual ending data address has been stored, the information in the RS field of the descriptor is exchanged with the result status bits from the operation, by execution of a memory swap operator.

The information read from the result status field includes one interrupt request bit, one high interrupt bit, one 3-bit port number, and one 4-bit channel number. If the interrupt request bit is TRUE, the MLC generates an appropriate interrupt message to the port indicated.

If the interrupt request bit is FALSE, the MLC generates an interrupt message only if the result status bits returned have exception bit number 2 set.

The interrupt message is a 24-bit address pointing between the result status bits and the link address. The channel number contained in the result description status area is normally the channel number of the MLC which received the I/O descriptor. This channel number is returned in any message generated by the control.

Figure 3-8 shows the location of the various bits in result status area that are inspected by the MLC.

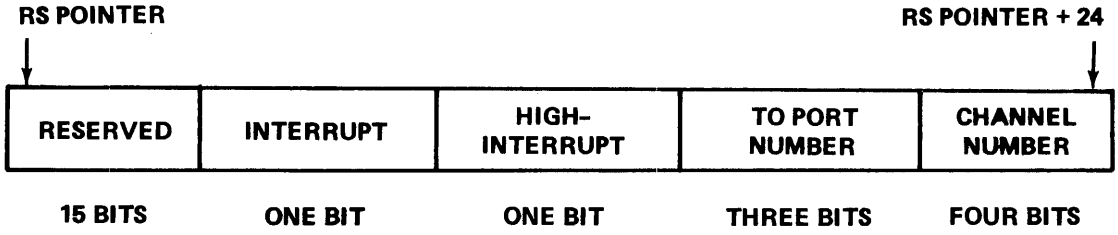


Figure 3-8. Result Descriptor – Multi-Line Control

The Multi-Line Control servicing a write interrupt message has the ability to stop servicing the write interrupt message and to service a normal memory request of a read and clear message operation any time the interrupt register is in use (locked). The MLC is permitted to stop servicing a write interrupt message only during the period that the lockout signal is TRUE, but this must occur, however, before the memory cycle has been granted to the MLC.

4. B 1700 SYSTEM DATA COMMUNICATIONS SOFTWARE DESCRIPTION

The B 1700 Data Communications System (Data Comm) operates asynchronously with the Master Control Program (MCP), providing efficient multiprogramming of data communication functions with on-site production jobs. The Data Comm systems software consists of three different modules or programs: the Network Controller, the Message Control System, and application programs. The Network Definition Language and the Network Controller are described briefly in this section. Refer to the B 1700 Systems Network Definition Language (NDL) Reference Manual, Form Number 1073715, for detailed information concerning this language. The Message Control System, which is optional for a Data Comm system, is described in detail in this section. The remote file syntax for COBOL, RPG, and UPL, and a discussion of station numbering are also included in this section of this manual.

NETWORK DEFINITION LANGUAGE

The B 1700 Network Definition Language (NDL) is classed as both a descriptive language and a programming language. It is a high-level language for data communications and provides a simple means of generating a B 1700 Network Controller (discussed below). The programmer defines a Data Comm network by specifying the network attributes in NDL source code, which includes a description of the physical devices in the network, the line disciplines to be used, the order and priority of the use, and the grouping of stations into files.

If the physical data communications network is reconfigured, the user can easily change the attributes that describe the network in the NDL source program, and recompile to produce the desired Network Controller. To simplify the task of the Data Comm system user, the NDL Compiler maintains a set of standard routines on disk to handle common line disciplines. NDL can also accommodate a user-written Message Control System (MCS) when certain functions and Data Comm decisions need to be controlled.

NETWORK CONTROLLER

The Network Controller (NC) is the heart of the Data Comm system, and its function is to process and supervise the flow of messages between application programs and the remote network. The B 1700 NDL Compiler translates the source code and, from it, produces the Network Controller codefile and the Network Information File (NIF). Figure 4-1 illustrates the NDL generation process.

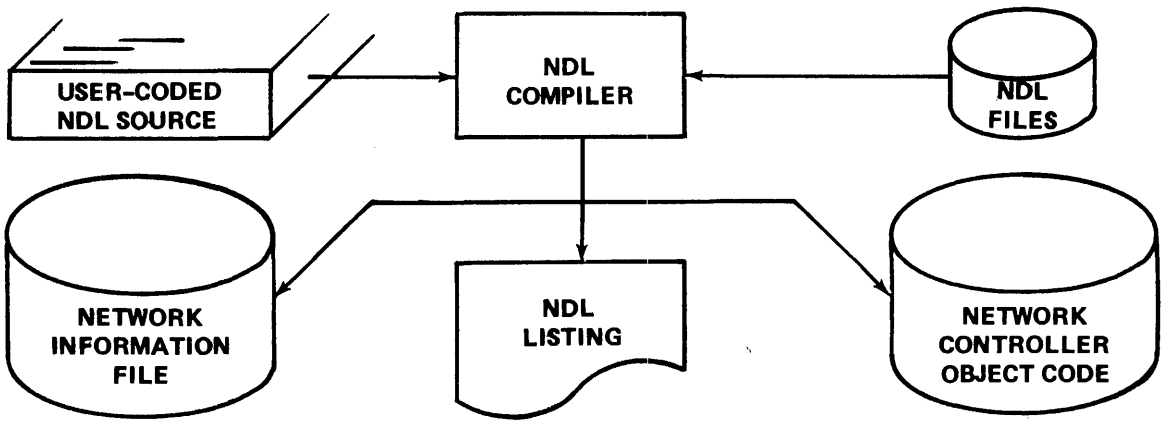


Figure 4-1. NDL Generation Process

The Network Information File (NIF) contains tables that describe the physical and logical attributes of the network. The initialization values for the LINE, STATION, TERMINAL, and FILE tables of the Network Controller (NC) are contained in the Network Information File.

The Network Controller codefile includes REQUEST and CONTROL procedures that are incorporated as subroutines.

The Network Controller is a normal-state program that performs the line discipline, queuing, buffer management, and audit and reconfiguration functions required by Data Comm systems. The Network Controller may be considered to be a user-defined extension of the operating system (MCP). The Data Comm input/output actions are performed directly by the Network Controller and all peripheral input/output functions for all batch programs are performed by the MCP.

The Network Controller performs its functions through the use of user-supplied information and system-supplied routines. The user information, provided by the NDL source program, concerns lines, stations, terminals, line control, line discipline, and interrelated aspects. The system-supplied routeines provide for I/O initiation, I/O completion, entrance to user CONTROL and REQUEST procedures, communications with the application programs and the MCS.

When a message is received from a station, the Network Controller (NC) queues it for the MCS or the application program. When a message is received from an application program or the MCS, the Network Controller initiates the appropriate procedures to transmit the message to the station. The NC also takes care of simultaneous I/O, multiple lines, recovery from line errors, and auditing, if required.

NDL is also used to define other attributes of the B 1700 Data Communications network. Thus, NDL provides a description of the line adapter configuration, the configuration of data sets, each terminal species, each station, and the line disciplines that are used with the lines and stations of the network.

The physical attributes that must be described for each station in the NDL source program are the station terminal species and the station adapter type. The station address is another physical station attribute that can be described.

For additional information concerning the Network Definition Language, refer to the B 1700 Systems Network Definition Language (NDL) Reference Manual, Form Number 1073715.

APPLICATION PROGRAMS

The B 1700 user's application programs for Data Comm may be written in COBOL, RPG, or the User Programming Language (UPL) source code. When the source code is compiled, the resulting application program can process the input/output data in terms of the problem to be solved and does not have to be concerned with the minute details of the data communications network. The function of the application programs is to process the input/output data according to application considerations, as in normal batch processing. The data is contained in the text of messages.

Because of the functions provided in the MCS and the Network Controller, the application program can deal with communication devices in the same manner as with conventional devices such as card readers, line printers, and magnetic storage devices. The Network Controller provides the interface between the application program and the remote network. Application programs are linked to the Network Controller by the file name in the FILE Section of the NDL program.

Figure 4-2 illustrates the relationship of an application program to the Network Controller and the remote devices of a Data Comm network.

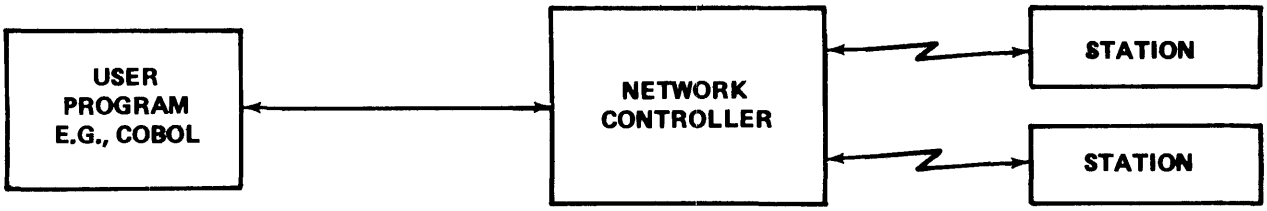


Figure 4-2. Data Comm Program Relationship

An application program can open a remote file through which several stations are accessible. These stations are said to be members of a family of stations and constitute an attribute of the file. If an MCS is present, stations can be added to or deleted from the family while the file is open. The stations of a family need not have the same terminal type. The FAMILY statement in the FILE Section of the NDL program is used to list the names of the stations that are members of the file being described.

Data Comm Remote File Syntax

When using a Data Comm network, the user must declare his file REMOTE. The remote files are used for communication between an application program and the remote terminals by way of NDL-compiled Network Controller. The application program treats the remote files as it does any other file. The Message Control System (MCS) does not use the remote file syntax; instead, the MCS (discussed later in this section) uses the queue file syntax for input and the MCS communicates for output.

The remote file syntax for COBOL, RPG, and UPL is described on the following pages. Following those descriptions is an explanation of station numbering.

Examples of several application programs written in COBOL are included in appendix B of this manual.

COBOL REMOTE FILE SYNTAX

Extensions to B 1700 COBOL permit users to open, close, read, and write remote files. These extensions involve the FILE-CONTROL SECTION of the ENVIRONMENT DIVISION, the FILE SECTION of the DATA DIVISION, and the PROCEDURE DIVISION. The following is the syntax to be observed for each division. The underscored words in uppercase type are required by the COBOL language syntax.

a. FILE-CONTROL SECTION (ENVIRONMENT DIVISION)

SELECT file-name ASSIGN TO REMOTE.
ACTUAL KEY IS data-name. (Optional)

The remote file is identified by the SELECT file-name ASSIGN TO REMOTE statement. When this file is opened in the PROCEDURE DIVISION of the application program, a message is sent by the MCP to the Network Controller, which will attach a family of remote stations to the file. The file-name from the SELECT statement is used unless the VA OF ID (see point b, below) is present in the FILE SECTION of the DATA DIVISION; the file-name must match a family name in the FILE Section of the NDL program.

The ACTUAL KEY option may be used with the remote file to selectively identify one station in the remote file. If the key is specified, it must have the following format:

```
01  REMOTE-KEY
    03  STATION-RSN     PC 9(3).
    03  TEXT-LENGTH     PC 9(4).
    03  MSG-TYPE        PC X(3).
```

STATION-RSN refers to the Relative Station Number of the station in the remote file.

TEXT-LENGTH defines the text size, in characters, and should never be larger than the largest 01 level item declared for the file. If TEXT-LENGTH is larger than the largest 01 level item, data will be truncated from the low-order portion.

MSG-TYPE is set to a 1 by the Network Controller for a read operation. On a write operation, MSG-TYPE must be set to 0 in the application program, unless an MCS is participating and explicitly changes the value.

The level specified for the remote key does not have to be 01, but the group length must be 10. A syntax check is done only on the group length.

All other FILE-CONTROL clauses are ignored for remote files.

b. FILE SECTION (DATA DIVISION)

```
FD
FILE CONTAINS integer    { STATION  }
                         { STATIONS }
VA OF ID    { data-name }
            { literal   }
```

The CONTAINS clause specifies the number of stations and must be specified if the number is greater than 1.

The VA OF ID is used to identify the file and must match a file name in the FILE Section of the NDL program. The VA OF ID statement is optional.

VA OF ID can be used to declare an external file name for remote files, just as it can for other types of hardware, such as magnetic tape units, disk devices, line printers, and card readers.

c. PROCEDURE DIVISION

```
OPEN    { INPUT        }    remote-file-name
        { INPUT-OUTPUT }
        { OUTPUT       }

WRITE   remote-file-record    [FROM       data-name  ]
                              [INVALID KEY statement]
```

READ remote-file [AT END statement]

CLOSE remote-file-name

The OPEN verb opens the named file and causes stations to be attached to it. A remote file can be opened INPUT, OUTPUT, or INPUT-OUTPUT.

WRITE gives text from data-name to the Network Controller for subsequent transmission to the station or stations.

READ waits for the Network Controller to pass any text received from the stations in the remote file. If ACTUAL KEY is used in the FILE-CONTROL SECTION, it will be modified when text is received, to contain information about the station and text. If AT END is present, the AT END statement is executed when directed by the Message Control System, or by the Network Controller at shutdown time when the B 1700 computer operator enters a "QC" keyboard input message from the console printer. If no stations on this file have messages to send, the program is suspended until a message is sent.

NOTE

Exception conditions for the READ verb must be tested by the USE AFTER STANDARD ERROR PROCEDURE ON file-name.

CLOSE detaches the stations and closes the remote file.

RPG REMOTE FILE SYNTAX

In the RPG source program, a Data Comm file specifier is declared in a manner similar to that used for a file that is not a remote file, with the following differences.

a. The word DATACOM must appear starting in column 40 of the File Description Specification card.

b. A Telecommunication Specifications card must follow the File Description Specification card. The Telecommunication Specifications card must contain the following:

Card Column	Contents	Meaning
6	T	For Telecommunication.
7-14	File name	
16	T	Transmit. The file must be an output file or a combined file, and must appear in the output-format specification. Combined files must alternately transmit and receive.
	R	Receive. The file must be an input file or a combined file, and must appear in the input specifications. Combined files must receive on each cycle and optionally transmit.

UPL REMOTE FILE SYNTAX

The UPL remote file syntax, which is somewhat similar to COBOL syntax, includes file declarations and the OPEN, CLOSE, READ, and WRITE verbs.

The FILE statement and the options used for remote files are described below.

```
FILE    internal-file-name        (DEVICE=REMOTE    [,REMOTE.KEY]
                                  [,NUMBER.OF.STATIONS=<NUMBER>])
```

The REMOTE.KEY attribute indicates that a key is present for a read or write operation to that file. If this attribute is not specified, a key cannot be used. Each field of the key is in decimal characters. The key is comprised of 10 characters and has the following format.

```
01  KEY.INFO                 CHARACTER (10),
    02  STATION.NUMBER       CHARACTER  (3),
    02  TEXT.LENGTH          CHARACTER  (4),
    02  MESSAGE.TYPE         CHARACTER  (3),
```

NUMBER.OF.STATIONS specifies the number of stations that can be attached to this file as provided for in the Network Controller.

The following is the syntax for READ and WRITE verbs for use with remote files:

```
READ internal-file-name   [[remote-key]]   (area)   [ON EOF statement;]
                                                    [ON EXCEPTION statement;]

WRITE internal-file-name  [[remote-key]]   (area);
```

Remote-key must be present if REMOTE.KEY is specified in the FILE statement.

On a READ remote, the key contains the data described by the REMOTE.KEY attribute that pertains to the message just read. If the file is empty, the program is suspended until a message is present.

ON EOF is executed when directed by the Message Control System, or by the Network Controller at shutdown time when the B 1700 operator enters the "QC" console input message. If the file is empty, the program is suspended until a message is present.

If the ON EXCEPTION option is specified, it will be executed when an irrecoverable I/O error occurs on a READ.

STATION NUMBERING

There are two types of station numbers used in B 1700 Data Comm: Relative Station Number (RSN) and Logical Station Number (LSN).

Relative Station Numbering

A given remote file may contain one or more stations; reference to these staions is by RSN within the file. The RSN is determined by the order in which the station is defined by the FAMILY statement in the FILE Section of the NDL program, or by the order in which the Message Control System attaches the stations to the file.

Example:

```
FILE TEST:
    FAMILY = S1, T3, S7.
```

Station S1 would be relative station number 1, because it is the first station defined. Station T3 would be relative station number 2, because it is the second station defined for FILE TEST.

Logical Station Numbering

Stations in a Data Comm network may also be identified by LSN within the Message Control System only. Since the MCS has the ability to participate in every message transfer, a file does not have the same meaning to an MCS as it would to an application program. Logical station number has no relationship to file names, but is determined by the order in which a station is defined in the STATION Section of the NDL program.

Examples:

```
STATION DEFAULT TD7:                %   This is a list of station defaults that may be
        MYUSE = INPUT,OUTPUT        %   used in subsequent station definitions.
        RETRY = 5.                  %
        TERMINAL = TD700.

STATION TD8:                        %   This is logical station number 1, because it
        DEFAULT = TD7.              %   is the first station defined.
        ADDRESS = "A3".             %

STATION TD1:                        %   This is logical station number 2, because it
        DEFAULT = TD7.              %   is the second station defined.
        ADDRESS = "A5".             %
```

MESSAGE CONTROL SYSTEM (MCS)

General

The Message Control System is optional for a B 1700 Data Comm System. The MCS, when present, controls message flow between application programs and the Network Controller It is the responsibility of the MCS to process all messages directed to it and then send them to the appropriate stations or programs. The MCS can be potentionally more powerful than a normal application program and operates in conjunction with the Network Controller.

The user must write the MCS in the User Programming Language (UPL).

NEED FOR AN MCS

Users may base their decision on whether or not an MCS is needed to control their B 1700 Data Comm network on the following criteria:

a. Degree of sophistication of the network.

b. Type of line error recovery desired.

c. Need for text editing.

d. Need for message switching.

e. Need for dynamic network reconfiguration.

MCS FUNCTIONS

The functions performed by a Message Control System are at the user's discretion and vary from application to application. Users with Message Control Systems are allowed to participate in, or rigidly control, system functions and decisions, some of which are listed below.

a. Network reconfiguration.

b. Error handling.

c. Message switching.

d. Pre-processing of the message.

MCS/Network Controller Interface

When a user requires an MCS, it must be specified in the NDL source program that an MCS will be present in the Data Comm system. Only one MCS may be present. The MCS cannot have remote file declarations but must have one queue file declaration. All messages coming from stations or sent by application programs are queued in the queue file declared for the MCS. Not all messages receive a response, but those that do cause a message of 27 characters or more to be queued in the MCS queue file.

The MCS/Network Controller interface is message oriented. Both the MCS and the Network Controller use the same message formats.

MCS COMMUNICATES

The MCS communicate is the only UPL construct reserved for Message Control Systems. An MCS can use all UPL constructs and also the MCS communicate, whereas a UPL program cannot use the MCS communicate construct.

The MCS transfers a message to the Network Controller by execution of an MCS communicate construct. The Network Controller transfers messages to the MCS by inserting them into the MCS queue file. Refer to section 5, DATA COMM MESSAGE FLOW, for a detailed explanation of message flow. The MCS communicate construct also can be used for requesting reconfiguration of the Data Comm network, thereby overriding the original NDL declarations.

The syntax of the MCS communicate is as follows.

MCS.COMMUNICATE (result, message).

"Result" represents a data name that is declared in the MCS program.

"Message" represents the message header and message text, explained under the subsection heading, MESSAGE.HEADER, in this section.

If no error is detected by the MCP, "result" will be set to 0. The following two error values may be returned:

"Result" = 75	Indicates an invalid logical station number.
"Result" = 76	Indicates that the text size of the message is greater than 4095 characters, or that the queue number is invalid.

GENERAL MESSAGE FORMAT (MCS)

The following is the message format used for transmission of messages between the Network Controller and the Message Control System. The names of the fields must be coded by the programmer exactly as shown.

```
01  MESSAGE
  02  MESSAGE.HEADER                 CHARACTER (27),
    03  MESSAGE.STANDARD
      04  MESSAGE.TYPE               BIT (8),
      04  MESSAGE.VARIANT            BIT (4),
      04  MESSAGE.FILE.NUMBER        BIT (10),
      04  MESSAGE.LSN                BIT (10),
      04  MESSAGE.TEXT.SIZE          BIT (12),
    03  MESSAGE.VARIABLE
      04  MESSAGE.TIME               BIT (20),
      04  MESSAGE.OLD.TYPE           BIT (8),
      04  MESSAGE.RESULT             BIT (8),
      04  MESSAGE.TRAN.NO            CHARACTER (3),
      04  MESSAGE.ERROR              BIT (24),
      04  MESSAGE.TALLY              CHARACTER (3),
      04  MESSAGE.TOGGLE             BIT (8),
      04  MESSAGE.TERMINAL.TYPE      BIT (6),
    03  FILLER                       BIT (50),
  02  MESSAGE.TEXT                   VARIABLE.
```

The message consists of control information contained in a message header field and data contained in a message text field. These subfields of the message are described below.

MESSAGE HEADER

This 27-character field contains control information and is present in all messages transferred between the Message Control System and the Network Controller. MESSAGE.HEADER contains two major fields:

MESSAGE.STANDARD and MESSAGE.VARIABLE.

MESSAGE.STANDARD. The subfields of MESSAGE.STANDARD are used most often by the Message Control System.

MESSAGE.TYPE is used to identify the type of message and the origin of the message.

MESSAGE.VARIANT allows the MCS programmer to select optional modes of a given MCS communicate. Detailed information about the available options are presented later in this section under the subsection heading MCS-TO-NETWORK CONTROLLER MESSAGE TYPES.

MESSAGE.FILE.NUMBER contains the file number that is used in communicating with an application program.

MESSAGE.LSN contains the logical station number of the station that is currently transmitting or receiving.

MESSAGE.TEXT.SIZE contains the value that indicates the size in characters of the message text.

MESSAGE.VARIABLE.

The subfields of MESSAGE.VARIABLE are supplied most often by the Network Controller in response to messages queued for the MCS.

MESSAGE.TIME contains the time of day the message entered the system.

MESSAGE.OLD.TYPE contains the value of MESSAGE.TYPE that was returned from a previous MCS communicate requiring a reply from the Network Controller.

MESSAGE.RESULT contains error values returned from the Network Controller that indicate that certain fields of a previous MCS communicate were not set correctly.

MESSAGE.TRAN.NO contains the transmission number of the last message received.

MESSAGE.ERROR contains the result descriptor that is set when a message is received in error. The MESSAGE.ERROR field contains 24 bits that indicate the status of a completed operation. The format of the MESSAGE.ERROR field is shown below.

Bit Number	Meaning
0	Parity error.
1	Buffer overflow occurred – Missed memory access.
2	Memory parity error (READ).
3	TIMEOUT (READ or WRITE).
4	BREAK received (WRITE).
5	ENDOFBUFFER. The expected ending control character was not received before the end of the buffer was reached.
6	Loss of Data Set Ready (DSR) – (READ, WRITE, BREAK).
7	Loss of carrier (READ).
8	Address error.
9	Transmission number error.
10	Format error. The received character sequence did not match the sequence requested.
11	Reserved.
12	Exception condition.
13	Reserved.
14-23	Reserved.

MESSAGE.TALLY and MESSAGE.TOGGLE contain the tallies and toggles set by the last REQUEST or CONTROL in the Network Controller that accessed them.

MESSAGE.TERMINAL.TYPE contains the value entered in the TYPE = field of the NDL source program.

MESSAGE.TEXT. This variable-length field contains the actual text of the message. All messages, however, do not contain a text portion.

MCS-TO-NETWORK CONTROLLER MESSAGE TYPES

The following 11 types of MCS communicate messages can be sent by the Message Control System to the Network Controller. The MESSAGE.TYPE field of the message header must be set, for each message type, to a unique value in the MCS.

MESSAGE.TYPE	Message Name
0	WRITE TO STATION
1	WRITE TO APPLICATION PROGRAM
2	MCS PRESENT NOTIFICATION
15	DS THE NETWORK CONTROLLER
16	FILE ATTACH
17	STATION ATTACH/DETACH
18	STOP THE NETWORK CONTROLLER
19	STATUS REQUEST
20	CHANGE REQUEST
22	REMOVE MESSAGE
23	NETWORK SIZE

Additional information concerning the 11 message types follows. Note that some of them require additional fields of the message header to be set in the MCS.

WRITE TO STATION (MESSAGE.TYPE = 0).

This communicate is used to send messages from the MCS to stations. The following subfields of MESSAGE.HEADER must be set in the MCS to the value indicated, or to the appropriate value.

Subfield	Value
MESSAGE.TYPE	0
MESSAGE.VARIANT	0
MESSAGE.LSN	Set to the logical station number of the station.
MESSAGE.TEXT.SIZE	Set to the size (number of characters) of the message to be transmitted.

WRITE TO APPLICATION PROGRAM (MESSAGE.TYPE = 1).

This communicate is used to send messages from the Message Control System to application programs. The following subfields of MESSAGE.HEADER must be set in the MCS to the value indicated, or to the appropriate value.

Subfield	Value
MESSAGE.TYPE	1
MESSAGE.VARIANT	Usually set to 0.
	Set to 1 to make program core-resident.
	Set to 2 to make program disk-resident.
	Set to 3 to cause execution of EOF branch in application programs.
MESSAGE.FILE.NUMBER	Set to the file number obtained from the FILE OPEN NOTIFICATION message.
MESSAGE.LSN	Set to the logical station number of the station that originated the message.
MESSAGE.TEXT.SIZE	Set to the size, in characters, of the message to be transmitted.

MCS PRESENT NOTIFICATION (MESSAGE.TYPE = 2).

This communicate is used to notify the Network Controller that an MCS is present. Before execution of this message, the MCS queue file must be opened INPUT/NEW. After the MCS queue file is opened, the Network Controller waits for the MCS PRESENT NOTIFICATION. While waiting for the MCS PRESENT NOTIFICATION, the only message that the Network Controller will execute is a DS THE NETWORK CONTROLLER (MESSAGE.TYPE = 15). This communicate must precede any other MCS communicate.

The only subfield of MESSAGE.HEADER that must be set for this communicate is:

MESSAGE.TYPE = 2.

No response or error message is returned by the Network Controller immediately following the MCS PRESENT NOTIFICATION communicate.

The following syntax of the MCS PRESENT NOTIFICATION communicate is different from the syntax previously presented in this section under the subsection heading, MCS COMMUNICATES.

MCS.COMMUNICATE (result,message,queue name).

"Result" represents a data name that is declared in the MCS program.

"Message" represents a data name in the MCS.

"Queue name" is the internal file name of the MCS queue file.

DS THE NETWORK CONTROLLER (MESSAGE.TYPE = 15)

This message instructs the Network Controller to immediately cancel all I/O operations, purge all incoming messages from the Data Comm system, and issue stop codes to all Data Comm controls. The use of this message type has the same effect as a "DS" (discontinue program) input message entered from the Console Printer.

The only subfield of MESSAGE.HEADER that must be set for this communicate is:

MESSAGE.TYPE = 15.

FILE ATTACH (MESSAGE.TYPE = 16)

The FILE ATTACH message is sent to the Network Controller (NC) after the MCS has examined the initial FILE OPEN NOTIFICATION (MESSAGE.TYPE = 2) that it received from the Network Controller. In the case of a normal file OPEN, the MCS can approve or disapprove the entire file OPEN, or remove any station from the station list.

The MCS can also participate in input and output, if necessary. If a dummy file is opened, the current station count is equal to 0, and the MCS is expected to insert at least one logical station number in the message text portion of the message.

The message format of the FILE ATTACH message follows. Some of the subfields of MESSAGE.HEADER and MESSAGE.TEXT must be set to the value indicated or to the appropriate value. The subfields of MESSAGE.HEADER and MESSAGE.TEXT that must not be changed are indicated by an asterisk (*) following the field size designation.

Subfield	Length		Value
MESSAGE.HEADER			
MESSAGE.STANDARD			
MESSAGE.TYPE	BIT	(8)	Set to 16.
MESSAGE.VARIANT			
MESSAGE.MCS.APPROVE	BIT	(1)	Set to 0 when MCS does not approve the file OPEN. Set to 1 when MCS approves.
MESSAGE.MCS.PARTICIPATING	BIT	(1)	Set to 1 when MCS participating.
MESSAGE.GOOD.RESULTS	BIT	(1)	Set to 1 when result of output operations are desired.
MESSAGE.KEY.PRESENT	BIT	(1)	MCP sets to 1 when remote key is present.
MESSAGE.FILE.NUMBER	BIT	(10)	*
MESSAGE.MAX.STATION	BIT	(10)	MCS can reduce this value.
MESSAGE.TEXT.SIZE	BIT	(12)	Set to (247 + 10 * MESSAGE.MAX.STATION)/8.

MESSAGE.VARIABLE			
MESSAGE.TIME	BIT	(20)	Supplied by the MCP.
MESSAGE.OPEN.TYPE	BIT	(2)	Set to.0 when invalid OPEN. Set to 1 when opened INPUT. Set to 2 when opened OUTPUT. Set to 3 when opened INPUT-OUTPUT.
MESSAGE.PROG.RESIDENT	BIT	(2)	*
MESSAGE.JOB.NO	BIT	(24)	*
MESSAGE.CUR.STATIONS	BIT	(10)	Set to the number of stations opened.
MESSAGE.FILE.NAME	CHARACTER	(10)	*
FILLER	BIT	(34)	*
MESSAGE.TEXT			
MESSAGE.PROGRAM.NAME	CHARACTER	(30)	*
MESSAGE.LSN.LIST	BIT	(10 * MESSAGE.CUR.STATIONS)	

If the FILE ATTACH message generates no errors, a reply message is not returned to the MCS from the Network Controller. Two error conditions are possible. If either error condition occurs, a NETWORK CONTROLLER REPLY (MESSAGE.TYPE = 4) is sent to the MCS from the Network Controller, with the MESSAGE.RESULT subfield of MESSAGE.VARIABLE containing one of the following values:

a. MESSAGE.RESULT = 2 if the current station count was greater than the MAXSTATION count in the Network Controller tables.

b. MESSAGE.RESULT = 1 when the station list contained stations that were not defined in the Network Controller tables.

STATION ATTACH/DETACH (MESSAGE.TYPE = 17)

This message permits the MCS to attach a station as input to a dummy file, or to detach a station as input from a file. All stations are capable of accepting output from any application program. If the file is marked as MCS participating, all attached stations will have the MCS participating.

The following subfields of MESSAGE.HEADER must be set to the value indicated or to the appropriate value.

Subfield	Value
MESSAGE.TYPE	17.
MESSAGE.VARIANT	Set to 0 for detach. Set to 1 for attach.
MESSAGE.FILE.NUMBER	Set to the number of the file to which the station is attached or detached.
MESSAGE.LSN	Set to the logical station number of the station.

If the message is correct, then no response is received from the Network Controller. An error causes a NETWORK CONTROLLER REPLY (MESSAGE.TYPE = 4) to be queued for the MCS, with the MESSAGE.RESULT field having one of the following values.

Field	Value	Meaning
MESSAGE.RESULT	1	Invalid logical station number.
	2	Station is output only.
	3	MESSAGE.VARIANT was greater than 1.
	4	Queue number not found.

STOP THE NETWORK CONTROLLER (MESSAGE.TYPE = 18)

This communicate causes the Network Controller to come to a controlled stop. This message type has the same effect as a "QC" console input message. Following the controlled stop, all stations are disabled for input.

Without an MCS in the system, End-of-File (EOF) branches are sent to all application programs. The Network Controller continues processing until all output messages have been sent and all remote programs have closed their remote files.

With an MCS in the system, the Network Controller expects that the MCS will ensure that all remote programs receive EOF branches and close their remote files before sending the STOP THE NETWORK CONTROLLER message. If a station is unable to receive its output messages, the MCS must recall all messages for that station in order to stop the Network Controller.

After all activity in the system has stopped, the Network Controller issues stop codes to the Data Comm controls and then goes to End-of-Job (EOJ). If the stop was initiated by the MCS, a NETWORK CONTROLLER REPLY (MESSAGE.TYPE = 4) is queued for the MCS; otherwise, the MCP is notified. In case of error, no error results are returned to the MCS.

The following subfield of MESSAGE.HEADER must be set to the value indicated.

MESSAGE.TYPE = 18.

STATUS REQUEST (MESSAGE.TYPE = 19)

The STATUS REQUEST message requests the Network Controller to return a NETWORK CONTROLLER REPLY message with the pertinent information about a line, station, terminal, or file to the MCS. This information can be used by the MCS to initialize its tables at Beginning-of-Job (BOJ) time or to verify current conditions at run time.

The following subfields of MESSAGE.HEADER in a STATUS REQUEST message must be set to the value indicated or to the appropriate value.

Subfield	Value
MESSAGE.TYPE	19.
MESSAGE.VARIANT	
MESSAGE.NAME.FLAG (this is a 1=bit field)	Set to 1 to request that the appropriate 10-character symbolic name be returned by the NETWORK CONTROLLER REPLY message.
MESSAGE.STATUS.TYPE (this is a 3-bit field)	Set to 0 for terminal status. Set to 1 for station status. Set to 2 for line status. Set to 3 for file status.
MESSAGE.LSN	Set to the number of the station, line, terminal, or file.
MESSAGE.TEXT.SIZE	Set to 13 for terminal. Set to 22 + (10*MAX.STATION.LINE)/8 for line. Set to 34 for station. Set to (90+(10*MAX.STATION.FILE)) for file.

If the requested MESSAGE.TEXT.SIZE is too small, the Network Controller allocates a message area of sufficient length, and corrects MESSAGE.TEXT.SIZE accordingly.

If an error was detected during processing of the STATUS REQUEST message, a NETWORK CONTROLLER REPLY message is queued for the MCS, with one of the following values in MESSAGE.RESULT.

Field	Value	Meaning
MESSAGE.RESULT	0	Indicates a successful STATUS REQUEST.
	1	Indicates invalid logical station number.
	2	Indicates invalid line number.
	3	Indicates invalid terminal number.
	4	Indicates invalid file number.
	5	Indicates invalid MESSAGE.VARIANT field.
	10	Indicates that the requested symbolic name was not found in the Network Information File (NIF).
	11	Indicates that the requested symbolic name was not found in the NIF file.

CHANGE REQUEST (MESSAGE.TYPE = 20)

This communicate allows the MCS to change some of the Data Comm network parameters. Before initiating a CHANGE REQUEST, the MCS must explicitly make the line NOT READY.

Changes that are made by execution of the CHANGE REQUEST are temporary, and are effective until one of the following occurs:

a. The Network Controller goes to End-of-Job (EOJ).

b. A DS of the Network Controller occurs.

c. Another CHANGE REQUEST is executed.

Changes made by execution of the CHANGE REQUEST do not change the values in the NDL Network Information File.

The following subfields of MESSAGE.HEADER must be set to the value indicated, or to the appropriate value.

Subfield	Value
MESSAGE.TYPE	20.
MESSAGE.VARIANT	
MESSAGE.IMPLICIT (this is a 1-bit field)	Set to 0 to keep the line NOT READY. Set to 1 to make the line READY.
MESSAGE.REQUEST.TYPE (this is a 3-bit field)	Set to 1 to change station attribute. Set to 2 to change line attribute.
MESSAGE.LSN	Set to logical line number or logical station number, depending on MESSAGE.REQUEST. TYPE.
MESSAGE.TEXT.SIZE	0.
MESSAGE.TALLY	This field and MESSAGE.TOGGLE must be changed in conjunction with each other. The values to assign to each field to change a given parameter are listed in the following table.
MESSAGE.TOGGLE	See above explanation for MESSAGE.TALLY.

Some of the STATION attributes originally set in the NDL source program can be changed by assigning the desired value to MESSAGE.TALLY and MESSAGE.TOGGLE. The parameters that can be changed and the values to be assigned to them are listed in the following table.

Parameter to be Changed	Value to be Assigned to MESSAGE.TOGGLE	Value to be Assigned to MESSAGE.TALLY
TRAN (RECEIVE)	0	Input transmission number, left justified.
TRAN(TRANSMIT)	1	Output transmission number, left justified.
ADDRESS (RECEIVE)	2	Input address.
ADDRESS (TRANSMIT)	3	Output address.
FREQUENCY (RECEIVE)	4	Integer, right justified.
FREQUENCY (TRANSMIT)	5	Integer, right justified.
RETRY	6	Integer.
ENABLED	7	0, for NOT ENABLED. 1, for ENABLED.
READY	8	0, for NOT READY. 1, for READY.
DIAGNOSTIC	9	0, for non-diagnostic REQUEST. 1, for diagnostic REQUEST.
LOGICALACK	10	0, for NO LOGICALACK. 1, for LOGICALACK.

Some of the LINE attributes can be changed by assigning the values shown in the following table to MESSAGE.TOGGLE and MESSAGE.TALLY.

Parameter to be Changed	Value to be Assigned to MESSAGE.TOGGLE	Value to be Assigned to MESSAGE.TALLY
LOCATION	0	SUBBIT(MESSAGE.TALLY,0,3) = Enter integer for PORT number. SUBBIT(MESSAGE.TALLY,3,4) = Enter integer for CHANNEL number. SUBBIT(MESSAGE.TALLY,7,4) = Enter integer for ADAPTER number.
READY	1	0, for NOT READY. 1, for READY.

NOTE

If any part of the LOCATION parameter is changed, a value for PORT, CHANNEL, and ADAPTER must be assigned to MESSAGE.TALLY.

The following example illustrates the function of the CHANGE REQUEST communicate and the events that take place between the MCS and the Network Controller.

Example:

To change ADDRESS (RECEIVE) of logical station number 13 on line 2 to "A1", the following operations must occur.

a. Line 2 must be made NOT READY. To accomplish this, the following subfields of MESSAGE.HEADER must be set.

Subfield	Value
MESSAGE.TYPE	20
MESSAGE.IMPLICIT	0
MESSAGE.REQUEST.TYPE	2
MESSAGE.LSN	2
MESSAGE.TEXT.SIZE	0
MESSAGE.TOGGLE	8
MESSAGE.TALLY	0

b. An MCS communicate with the following syntax must be executed.

```
MCS.COMMUNICATE (result, MESSAGE.HEADER).
```

c. When line 2 becomes NOT READY, the Network Controller will send a NETWORK CONTROLLER REPLY (MESSAGE.TYPE = 4) to the MCS. This message is identical to the message sent by the MCS, except that the following two fields of MESSAGE.HEADER must be set to the values indicated.

```
MESSAGE.TYPE = 4.
MESSAGE.OLD.TYPE = 20.
```

If an error condition occurred, the MESSAGE.RESULT field contains the error code.

d. With the line in NOT READY state, the MCS must send a CHANGE REQUEST message to the Network Controller to change the station address. The MCS programmer must set the appropriate subfields of MESSAGE.HEADER as follows:

Subfield	Value	Meaning
MESSAGE.TYPE	20	
MESSAGE.IMPLICIT	1	
MESSAGE.REQUEST.TYPE	1	
MESSAGE.LSN	13	Logical station number.
MESSAGE.TEXT.SIZE	0	
MESSAGE.TOGGLE	2	ADDRESS (RECEIVE).
MESSAGE.TALLY	A1	New address.

e. An MCS communicate with the following syntax must be executed.

MCS.COMMUNICATE (result, MESSAGE.HEADER).

f. The Network Controller receives the message from the MCS and takes the appropriate action. When the requested change is completed, a NETWORK CONTROLLER REPLY message is queued for the MCS to inform it of the completion, and line 2 is made READY. The NETWORK CONTROLLER REPLY message to the MCS is identical to the message sent by the MCS, except for the MESSAGE.TYPE and MESSAGE.OLD.TYPE fields of MESSAGE.HEADER.

If an error is detected by the Network Controller in attempting to process a CHANGE REQUEST, the MESSAGE.RESULT field contains one of the values listed below to indicate the error.

Value of MESSAGE.RESULT	Error Indication
0	CHANGE REQUEST completed without error.
1	Requested change not possible.
2	Attempt made to change a READY line.
3	Illegal value in MESSAGE.TOGGLE (greater than 10 for station or greater than 1 for line change).
4	Attempt made to change the location of an invalid line.
5	Attempt made to ready an invalid line.

REMOVE MESSAGE (MESSAGE.TYPE = 22)

The function of this communicate is to remove the last message from the queue of a station. The REMOVE MESSAGE message is always answered by a NETWORK CONTROLLER REPLY (MESSAGE.TYPE = 4).

If there is at least one message queued for the station, the message is removed and the text of the message is queued for the MCS by the NETWORK CONTROLLER REPLY.

The following subfields of MESSAGE.HEADER must be set to the value indicated.

Subfield	Value
MESSAGE.TYPE	22
MESSAGE.VARIANT	0
MESSAGE.LSN	Set to the logical station number.

If the REMOVE MESSAGE communicate is executed with MESSAGE.VARIANT = 1, all messages in the queue of the station are removed, with the exception of the message the Network Controller is currently processing. To remove that message from the queue of the station, the station must first be made NOT READY.

If the removed message was the only message in the queue of the station, or if there were no messages queued for output for the station, MESSAGE.VARIANT is set by the Network Controller as follows.

MESSAGE.VARIANT = 8.

NETWORK SIZE (MESSAGE.TYPE = 23)

This message requests the Network Controller to send information to the MCS that describes the size of the Data Comm network.

The following subfields of MESSAGE.HEADER must be set to the value indicated.

Subfield	Value
MESSAGE.TYPE	23
MESSAGE.TEXT.SIZE	10

After processing the NETWORK SIZE message, the Network Controller queues a NETWORK CONTROLLER REPLY (MESSAGE.TYPE = 4) for the MCS. The format of the NETWORK CONTROLLER REPLY is as follows:

```
02  DUMMY REMAPS MESSAGE.TEXT
    04  MAX.TERMINALS          BIT (8)        Number of terminals declared in NDL.
    04  MAX.STATIONS           BIT (10)       Number of stations declared in NDL.
    04  MAX.LINES              BIT (8)        Number of lines declared in NDL.
    04  MAX.FILES              BIT (8)        Number of files declared in NDL.
    04  MAX.LINE.STATIONS      BIT (10)       Maximum number of stations per line.
    04  MAX.FILE.STATIONS      BIT (10)       Maximum number of stations per file.
    04  SIGNAL.COUNT           BIT (2)        The number of SIGNAL characters.
    04  SIGNAL.CHARACTER       CHARACTER (3)  SIGNAL characters.
```

MESSAGES RECEIVED BY THE MCS

The Network Controller transfers messages to the MCS by inserting them into the MCS queue file. The format of each message must conform to the general message format previously illustrated in this section of the manual. All messages sent to the MCS must contain a 27-byte message header, and message text if appropriate.

The following five types of messages may be received by the MCS. The MESSAGE.TYPE field of the message header must be set, for each message type, to a unique value.

MESSAGE.TYPE	Message Name
0	MESSAGE FROM APPLICATION PROGRAM
1	MESSAGE FROM A STATION
2	FILE OPEN NOTIFICATION
3	FILE CLOSE NOTIFICATION
4	NETWORK CONTROLLER REPLY

Additional information concerning the five message types follows.

MESSAGE FROM APPLICATION PROGRAM (MESSAGE.TYPE = 0)

If the MCS is participating in I/O with a station, all output messages to the station from application programs go to the MCS first. This permits the MCS to do message-switching for the system. The significant fields of the message and the values to which they are set are listed below.

Subfield	Value
MESSAGE.TYPE	0
MESSAGE.FILE.NUMBER	Set by the MCP to the file number of the file to which the station is attached.
MESSAGE.LSN	The key in the WRITE statement of the application program.
MESSAGE.TEXT.SIZE	Set by the application program to the number of bytes of text in this message.
MESSAGE.TEXT	The actual text or data of the message is entered by the application program.

MESSAGE FROM A STATION (MESSAGE.TYPE = 1)

When the MCS is participating in message transfer it will receive all messages from all participating stations. When the MCS is not participating in message transfer it can receive messages, but only those messages whose first character is equal to the MCS SIGNAL character that was declared in the NDL source program.

The values in the following significant fields of MESSAGE.HEADER are set by the Network Controller.

Subfield	Value
MESSAGE.TYPE	1.
MESSAGE.VARIANT	A 0 indicates good input. A 4 indicates exception condition.
MESSAGE.FILE.NUMBER	File number to which station is attached as input.
MESSAGE.LSN	Logical station number.
MESSAGE.TEXT.SIZE	Number of bytes of text in this message.
MESSAGE.TIME	Time of day this message entered the system.
MESSAGE.ERROR	Status indicator.
MESSAGE.TALLY	Station tally.
MESSAGE.TOGGLE	Station toggle.

FILE OPEN NOTIFICATION (MESSAGE.TYPE = 2)

When an application program opens a remote file, the MCP sends a message to the Network Controller. The Network Controller verifies the OPEN, and if the file is not a dummy file, adds the lists of associated logical stations for the file. Then the Network Controller puts the resulting list into the MESSAGE.TEXT part of the message. The FILE OPEN NOTIFICATION message is then sent to the MCS for approval.

The format of the FILE OPEN NOTIFICATION message and the values to which the significant subfields of MESSAGE.HEADER are set by the Network Controller are shown below.

Subfield	Length	Value
MESSAGE.STANDARD		
MESSAGE.TYPE	BIT (8)	
MESSAGE.VARIANT		
MESSAGE.MCS.APPROVE	BIT (1)	0 indicates the OPEN was not approved. 1 indicates the OPEN was approved.
MESSAGE.MCS.PARTICIPATING	BIT (1)	1 indicates the MCS is participating, and information concerning the disposition of output messages is desired.
MESSAGE.GOOD.RESULTS	BIT (1)	
FILLER	BIT (1)	
MESSAGE.FILE.NUMBER	BIT (10)	
MESSAGE.MAX.STATION	BIT (10)	
MESSAGE.TEXT.SIZE	BIT (12)	
MESSAGE.VARIABLE		
MESSAGE.TIME	BIT (20)	
MESSAGE.OPEN.TYPE	BIT (2)	0 indicates invalid OPEN. 1 indicates file opened INPUT. 2 indicates file opened OUTPUT. 3 indicates file opened INPUT/OUTPUT.
MESSAGE.PROG.RESIDENT	BIT (2)	
MESSAGE.JOB.NO	BIT (24)	
MESSAGE.CUR.STATIONS	BIT (10)	
MESSAGE.FILE.NAME	CHARACTER (10)	
FILLER	BIT (10)	
MESSAGE.PROGRAM.NAME	CHARACTER (30)	
MESSAGE.LSN.LIST	BIT (10 * MESSAGE.CUR.STATIONS)	

FILE CLOSE NOTIFICATION (MESSAGE.TYPE = 3)

When a file is closed, the MCP sends a message to the Network Controller. The Network Controller verifies the CLOSE and notifies the MCS by inserting the FILE CLOSE NOTIFICATION message in the MCS queue file. The format of the message is identical to that of the FILE OPEN NOTIFICATION message, except for the following fields and associated values.

MESSAGE.TYPE = 3.

MESSAGE.FILE.NUMBER = Number of the file just closed.

NETWORK CONTROLLER REPLY (MESSAGE.TYPE = 4)

The Network Controller queues a NETWORK CONTROLLER REPLY message for the MCS if the MCS has sent any of the following MCS communicates.

STOP THE NETWORK CONTROLLER (MESSAGE.TYPE = 18).

STATUS REQUEST (MESSAGE.TYPE = 19).

CHANGE REQUEST (MESSAGE.TYPE = 20).

REMOVE MESSAGE (MESSAGE.TYPE = 22).

The message format of the NETWORK CONTROLLER REPLY is identical to the format of the MCS communicate that initiated the reply. The fields of MESSAGE.HEADER that contain different values are as follows:

Subfield	Value
MESSAGE.TYPE	4.
MESSAGE.OLD.TYPE	MESSAGE.TYPE of message initiating the NETWORK CONTROLLER REPLY.
MESSAGE.RESULT	Dependent on which communicate the reply is answering.

If any message text is returned with the 27-byte message header, the MESSAGE.TEXT.SIZE indicates the number of bytes of text.

If an error condition is detected by the Network Controller while processing any message from the MCS, a NETWORK CONTROLLER REPLY is queued for the MCS, with the appropriate error code in the MESSAGE.RESULT field. The following fields of MESSAGE.HEADER contain the values indicated, in order to report the error to the MCS.

Subfield	Value
MESSAGE.TYPE	4.
MESSAGE.OLD.TYPE	MESSAGE.TYPE of message causing the error.
MESSAGE.RESULT	Error code.

Additional information concerning each of the standard NETWORK CONTROLLER REPLY messages and the values that are returned in the significant fields of the message is given in the following paragraphs.

REPLY TO STOP THE NETWORK CONTROLLER. The receipt of this message by the MCS indicates that the Network Controller has stopped. The significant fields of the message and the values to which they have been set are:

MESSAGE.TYPE = 4.

MESSAGE.OLD.TYPE = 18.

MESSAGE.RESULT = 0.

REPLY TO STATUS REQUEST. The status information requested by the MCS is returned by this message from the Network Controller. The significant fields of MESSAGE.HEADER and the values to which they have been set are:

MESSAGE.TYPE = 4.

MESSAGE.TEXT.SIZE = Number of bytes of text that follow.

MESSAGE.OLD.TYPE = 19.

MESSAGE.RESULT = 0.

The subfields of MESSAGE.TEXT are significant for this reply message. They are formatted differently and contain different values, depending upon whether the STATUS REQUEST message from the MCS was for terminal, station, line, or file status.

When terminal status information is requested by the STATUS REQUEST message from the MCS, the MESSAGE.TEXT field of the REPLY TO STATUS REQUEST message has the following format. The appropriate values for each subfield are entered by the Network Controller.

Subfield	Length	Value
M.TERMINAL.NAME	CHARACTER (10)	Symbolic name of the terminal.
M.TERMINAL.TYPE	BIT (6)	Value of the TYPE field in the NDL TERMINAL Section.
M.TERMINAL.MAXINPUT	BIT (12)	Maximum input buffer size for the terminal.

When station status information is requested by the STATUS REQUEST message from the MCS, the MESSAGE.TEXT field of the REPLY TO STATUS REQUEST message has the following format. The appropriate values for each subfield are entered by the Network Controller.

Subfield	Length	Value
M.STATION.NAME	CHARACTER (10)	Symbolic name of station.
M.STATION.READY	BIT (1)	0 – Station is NOT READY. 1 – Station is READY.
M.STATION.ENABLED	BIT (1)	1 – Station is ENABLED INPUT.
M.STATION.MYUSE	BIT (2)	0 – Not used. 1 – Station is INPUT. 2 – Station is OUTPUT. 3 – Station is INPUT/OUTPUT.
M.STATION.TRAN.NO.SIZE	BIT (2)	Number of characters in transmission number.
M.STATION.TRAN.RECEIVE	CHARACTER (3)	Input transmission number.
M.STATION.TRAN.TRANSMIT	CHARACTER (3)	Output transmission number.
M.STATION.I.ADR.SIZE	BIT (2)	Input address size in characters.
M.STATION.O.ADR.SIZE	BIT (2)	Output address size in characters.
M.STATION.ADDR.RECEIVE	CHARACTER (3)	Input address.
M.STATION.ADDR.TRANSMIT	CHARACTER (3)	Output address.
M.STATION.MAX.RETRY	BIT (8)	Maximum retry count for station.
M.STATION.PRIORITY.RECEIVE	BIT (8)	Input priority of station.
M.STATION.PRIORITY.TRANSMIT	BIT (8)	Output priority of station.
M.STATION.LINE	BIT (8)	Line number to which this station is attached.
M.STATION.QUEUED	BIT (12)	Number of messages queued for this station.
M.TERMINAL.LINK	BIT (8)	Terminal number of stations.
M.LOGICALACK	BIT (1)	LOGICALACK is true for this station.
M.STATION.DIAGNOSTIC.REQUEST	BIT (1)	0 – Use normal REQUEST. 1 – Use diagnostic REQUEST.
M.STATION.TALLY.TOGGLE	BIT (32)	The station tallies (3) and toggles (8).

When line status information is requested by the STATUS REQUEST message from the MCS, the MESSAGE.TEXT field of the REPLY TO STATUS REQUEST message has the following format. The appropriate values for each subfield are entered by the Network Controller.

Subfield	Length	Value
M.LINE.NAME	CHARACTER (10)	Symbolic line name.
FILLER	BIT (2)	
M.LINE.PORT	BIT (3)	Port number of line.
M.LINE.CHANNEL	BIT (4)	Channel number of line.
M.LINE.ADAPTER	BIT (4)	Adapter number of line.
M.LINE.ADAPTER.TYPE	BIT (5)	Adapter type.
M.LINE.QUEUED	BIT (16)	Number of messages queued for stations on this line.
M.LINE.STAT.ATTACH	BIT (8)	The number of stations in the station list specified in the LINE Section of the NDL program.
M.LINE.STAT.READY	BIT (8)	The number of stations in use on this line.
M.LINE.MAX.STATIONS	BIT (8)	The value specified in the MAXSTATIONS statement in the LINE Section of the NDL program.
M.LINE.TYPE	BIT (5)	1 - Switched line. 2 - Leased line. 3 - Direct-connect line.
M.LINE.POLL.LGTH	BIT (10)	Value that was entered for the AUTOPOLL statement in the LINE Section of the NDL program.
M.LINE.TALLY	CHARACTER (2)	The current value in LINE TALLY.
M.LINE.TOG	BIT (2)	The current value in LINE TOG.
M.LINE.STATUS	BIT (6)	The current value of LINE STATUS.

Value	Meaning
0	Adapter not present and NOT READY.
1	Adapter not present but line is marked are READY.
2	Test I/O operation in progress.
4	Line is disconnected but READY.
5	Line is ringing.
21	Line is idle.
22	No I/O in progress and line NOT READY.
23	No I/O in progress.
24	Line control procedure performed INITIATE INPUT.
25	Line control procedure performed INITIATE OUTPUT.
26	Line control procedure performed INITIATE INPUT-OUTPUT.
27	Line control procedure performed INITIATE OUTPUT-INPUT.
38	Request performed TERMINATE LOGICALACK.
39	MCS acknowledged LOGICALACK.
40	Request performed FINISH TRANSMIT.
41	Request performed INITIATE RECEIVE.
42	Request performed FINISH TRANSMIT-INITIATE RECEIVE.
43	Control performed INITIATE AUTOPOLL.
44	Request performed INITIATE RECEIVE (NO TIMEOUT).
45	Request performed FINISH TRANSMIT-INITIATE RECEIVE (NO TIMEOUT).
46	Control performed INITIATE CANCEL while line was in RECEIVE.
47	Control performed INITIATE CANCEL while AUTOPOLL in progress.

When file status information is requested by the STATUS REQUEST message from the MCS, the MESSAGE.TEXT field of the REPLY TO STATUS REQUEST message has the following format. The appropriate values for each subfield are entered by the Network Controller.

Subfield	Length	Value
M.FILE.NAME	CHARACTER (10)	Symbolic name of file.
M.MAX.STATIONS	BIT (10)	Number of stations in family.
M.FILE.STATION.LIST(X)	BIT (10)	List(s) of 10-bit logical station numbers of the stations attached to this file.

NOTE

(X) = MAX.STATIONS in the above structure.

REPLY TO CHANGE REQUEST. This message is sent to the MCS following the processing of a CHANGE REQUEST message by the Network Controller. The value in the MESSAGE.RESULT field indicates the results of the requested changes.

Value of MESSAGE.RESULT	Error Indication
0	Changes were completed without error.
1	The requested change is not possible.
2	An attempt was made to change a READY line.
3	Illegal value MESSAGE.TOGGLE (greater than 10 for station or greater than one for line change).
4	An attempt was made to change the location of an invalid line.
5	An attempt was made to READY an invalid line.

REPLY TO REMOVE MESSAGE. If a message is removed from a station queue by the REMOVE MESSAGE MCS communicate, the Network Controller returns the removed message to the MCS in the form of a NETWORK CONTROLLER REPLY message. The only field of the MESSAGE.HEADER that is always changed is:

MESSAGE.TYPE = 4.

If the message removed was the last message in the station queue, or if there were no messages queued for output, MESSAGE.VARIANT is set as follows:

MESSAGE.VARIANT = 8.

5. DATA COMM MESSAGE FLOW

GENERAL

The Network Controller processes and supervises the flow of messages between application programs and the remote stations of the Data Comm network. If a Message Control System (MCS) is present in the network, the MCS controls the message flow between the application programs and the Network Controller.

The data to be transferred to or from the remote devices is contained in the text of messages. A message header always precedes the text of the messages. Messages can vary in size from a single character to any continuous character string that can be accommodated by the equipment. The length and content of the message text depend upon the application.

The first 27 bytes of the message are referred to as the message header and contain control information in a fixed format, so that the text of the message begins with byte 28 of the message. The Network Controller and the Message Control System are the only programs that can examine the message header. The application programs have access to message text only.

The Network Controller has a queue array which has one entry for each station in the network, and one additional entry for special use, which is the first entry. For example, when the application program opens the remote file, a file open indicator is placed in the first entry (entry 0) of the Network Controller queue array.

DATA MOVEMENT

Data movement in an NDL-compiled Data Comm network occurs whenever an application program or an MCS does a read or write operation. In the figures used to illustrate message flow in this section of the manual, the place at which data movement occurs is indicated by the letters DM and a numeral (e.g., DM2) adjacent to an arrow. The numerals show the order in which data movement occurs.

NOTE

> Data movement from the input buffer area does not occur as a message is linked into the remote file queue of a program. Such data movement occurs only when the application program executes a read operation to bring the message into its base-to-limit area.

MESSAGE FLOW – NO MCS PRESENT

Figure 5-1 illustrates Data Comm message flow when no Message Control System is present in the system, and shows the relationship of an application program, the Network Controller, the Network Controller queue array, and the remote file queue for the application program.

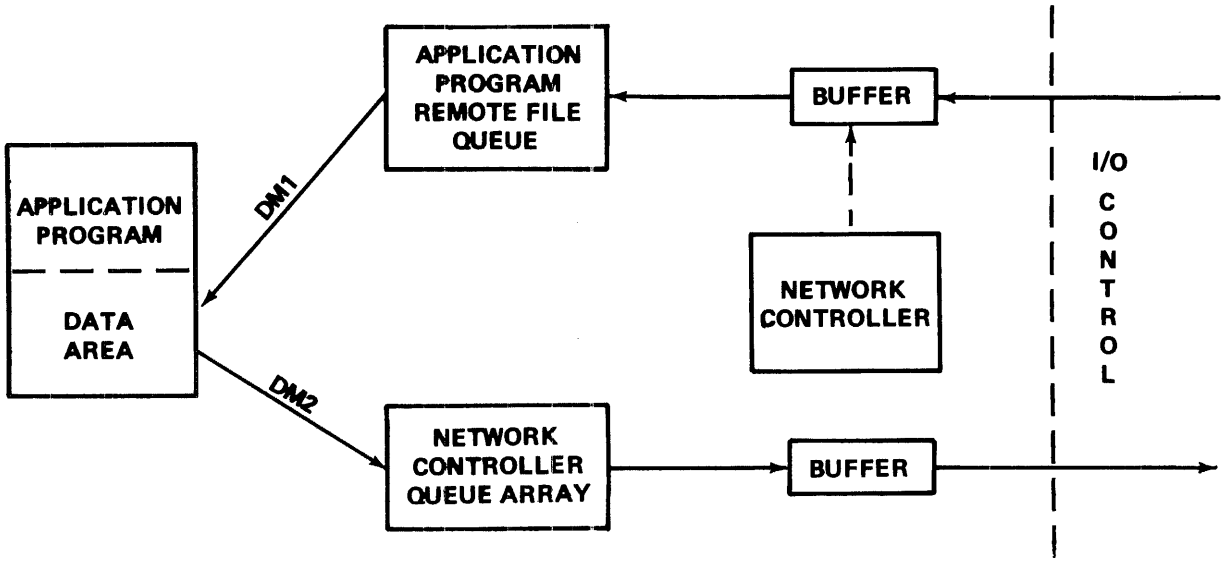


Figure 5-1. Data Comm Message Flow – No MCS

The I/O control collects the input message character(s) in a buffer. The Multi-Line Control has a one-character buffer, and the Single Line Control has two 13-character buffers. The I/O control transfers the character(s) to a buffer in memory that is large enough to hold the entire message. When the application program executes a read operation on its remote file, the MCP transfers the first input message to the record area of the application program. DM1 in figure 5-1 indicates that the first data movement occurred during the message transfer. The application program processes the message and executes a write operation in reply to the remote station. The MCP transfers the reply from the record area of the application program to the queue array of the Network Controller; this is DM2 in figure 5-1. The MCP builds a unique list of messages for each different application program file. All reply messages are sent to the Network Controller queue array.

When an application program executes a file open request operation on a remote file, the following actions take place.

a. The MCP queues a remote file open message to the Network Controller through entry 0 of the Network Controller queue array.

b. The MCP constructs a File Information Block (FIB). The disposition of the remote file open request is not known at that time, and the application program may be suspended (rolled out of main memory) to await permission to open the remote file.

c. The Network Controller grants the file open request, and attaches the station list to the message it just received from the MCP.

d. The Network Controller executes a communicate to the MCP to inform it of the approval of the file open request.

e. The MCP uses the job number to find the suspended program.

f. The MCP attaches the station list to the object program FIB and discards the message that the Network Controller sent.

g. The MCP makes an entry in its queue directory when the remote file is opened.

h. The MCP does a cause operation on the Run Structure Nucleus of the suspended program.

NOTE

If a file open request is executed by an application program with fewer stations declared in the application program than have been specified in the NDL source program, not all of the stations will be marked "READY" by the Network Controller. Only the stations that have the lowest relative station number (in the FAMILY statement of the FILE Section of the NDL program) will be marked "READY." For example, if five stations are declared in the application program, and eight stations are specified in the NDL FAMILY statement, only the first five stations specified will be marked "READY."

A file open request will be disapproved by the Network Controller for any one of the following reasons.

a. The MYUSE declaration in NDL and the file open attempt do not agree on the functional use of a station. For example, MYUSE = INPUT was specified in the NDL source program, but the application program attempted to open the file as an output file.

b. The current file open attempt would result in the openeing of a file not declared in the NDL program.

c. FAMILY = DUMMY was specified in the FILE Section of the NDL program and no MCS is present in the Data Comm system.

The MCP causes a DS or DP condition (discontinue program) after the file open attempt is disapproved.

MESSAGE FLOW – MCS PRESENT BUT NOT PARTICIPATING

Figure 5-2 illustrates a data communication system that has a Message Control System present. The message flow is identical to the case illustrated by figure 5-1, except that the MCS is notified of several events for which it must make a decision.

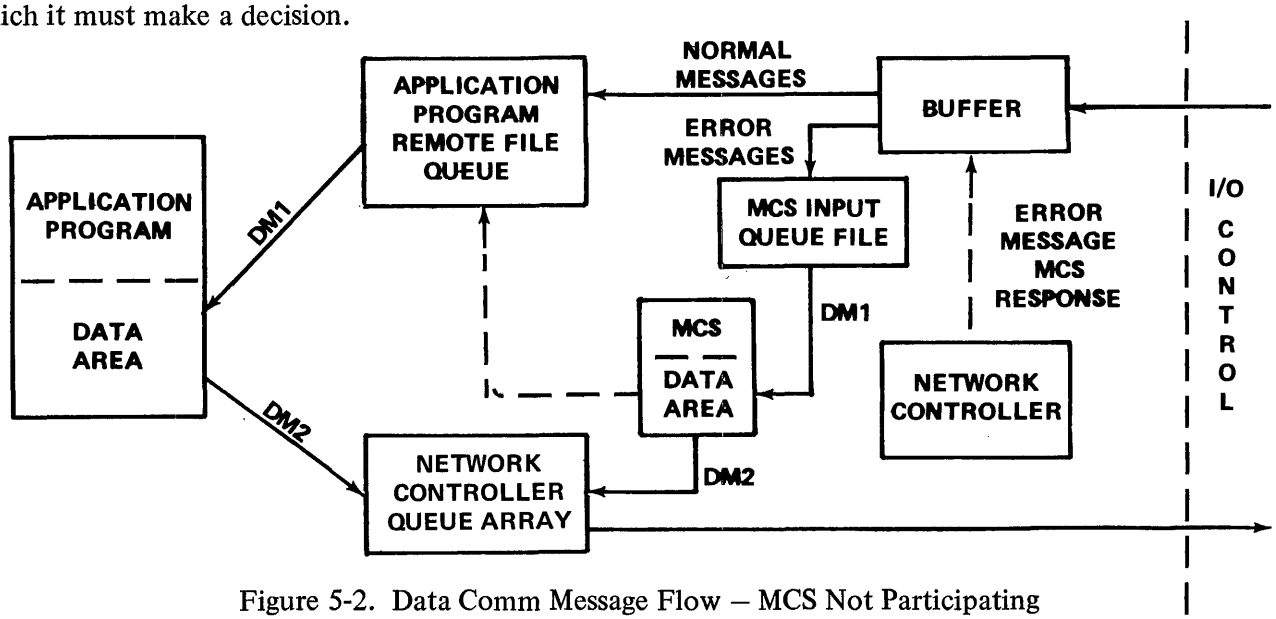


Figure 5-2. Data Comm Message Flow – MCS Not Participating

The MCS is notified of all file open and file close operations executed by the application programs, and the MCS may approve or disapprove. Data communications errors are reported to the MCS by the Network Controller.

The MCS may communicate directly to an application program, but the application program cannot communicate directly to the MCS. The MCS is able to monitor and control system events that affect the application programs. The MCS may alter station/line priority, job priorities, or send special instructions to remote stations through MCS communicate instructions.

MESSAGE FLOW – MCS PRESENT AND PARTICIPATING

When the MCS is present and participating in the Data Comm network, the message flow and data movement are considerably different from the case presented in figure 5-2. Figure 5-3 illustrates these differences.

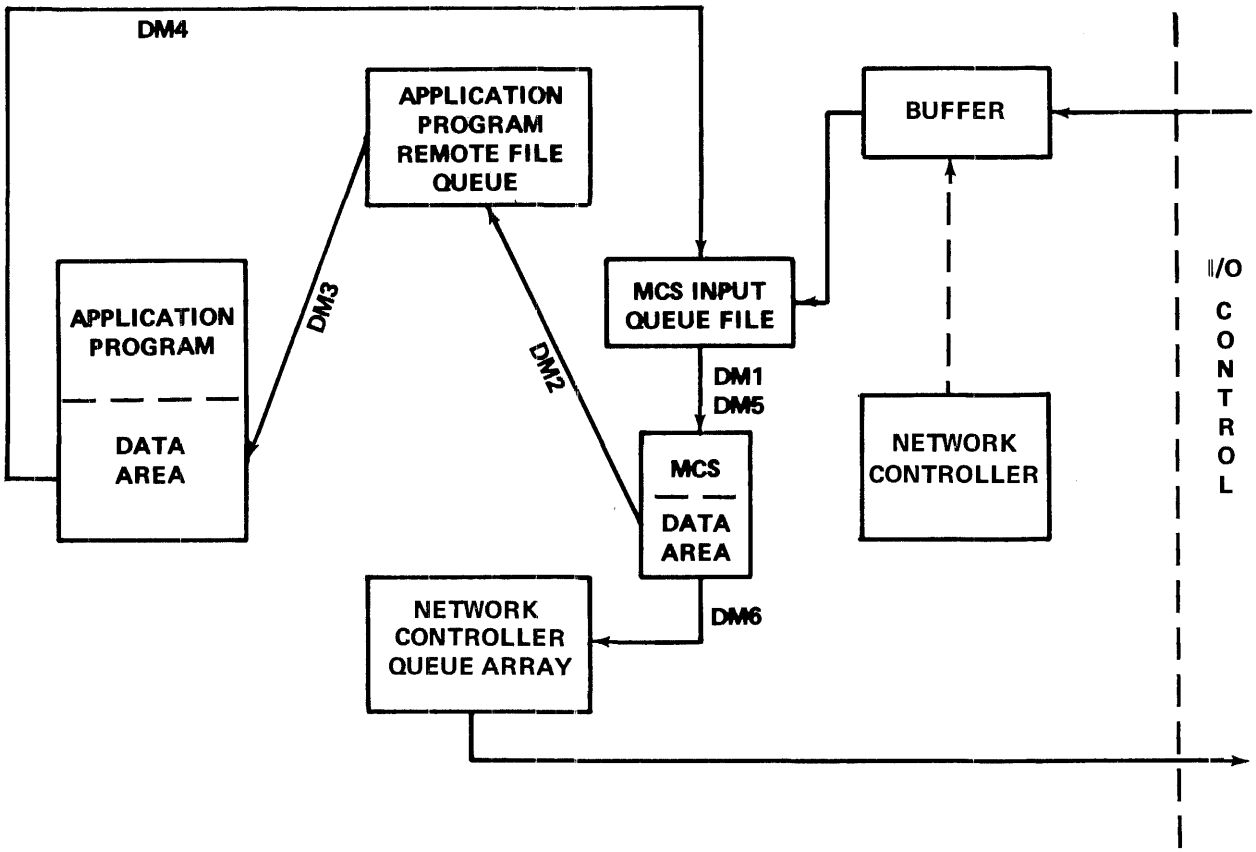


Figure 5-3. Data Comm Message Flow – MCS Participating

In the normal course of the MCS receiving a message from the remote network, processing the message, and sending a reply to the remote network, the following actions occur. Assume that the Network Controller, Message Control System, and application programs have been executed and are at Beginning of Job (BOJ).

a. The MCS does an OPEN on its queue file and the application program does an OPEN on its remote file.

b. The MCS executes a read operation on the queue file, which contains the file open request of the application program.

c. If the remote file was declared as DUMMY in the NDL program, the FILE ATTACH MCS communicate attaches the stations, if any. In the case of a normal remote file open request, the MCS can choose to approve or disapprove the entire file open.

d. The MCS does a read operation on the MCS queue file and the application program does a read operation on the remote file.

e. The message is transmitted from the I/O control to an input buffer.

f. The message is moved from the input buffer to the MCS input queue.

g. The MCS does a READ from its input queue. The first data movement is indicated by DM1 in figure 5-3.

h. By means of an MCS communicate, the MCS does a write operation to the remote file queue of the application program. This is shown by DM2 in figure 5-3. The MCS then performs the action specified in step d, and waits for the completion of a READ of its input queue file.

i. The application program does a read operation on its remote file queue. The data movement is indicated by DM3 in figure 5-3.

j. Processing of the message is carried out by the application program.

k. The application program does a write operation on its remote file. DM4 shows the data movement. The application program performs the action specified in step d.

Appendix A. GLOSSARY

ACK

A control character that is used as an affirmative response to a normal selection (indicating "ready-to-receive") or a transmission (indicating "message accepted"). As an affirmative response to a selection, ACK may optionally be preceded by station identification, AD1, AD2, or other information such as a reply.

ACK0, ACK1

These replies, when in proper sequence, indicate that the previous block received was accepted without error, and that the receiver is ready to accept the next block of the transmission. ACK0 is the positive response to selection (multipoint), or line bid (point-to-point). The alternate use of ACL0 and ACK1 is used for affirmative replies. The use of ACK0 and ACK1 provides a sequential checking control for a series of replies. Thus, it is possible to maintain a continuous check to ensure that each reply corresponds to the immediately proceding message block. The affirmative response to a poll is the transmission of a message.

ACK0 is represented by a DLE character followed by a hexadecimal 70.
ACK1 is represented by a DLE character followed by a hexadecimal 61.

AD1, AD2

A two-character address, established as the address of a device at a terminal. These two characters are used to address a terminal in polling or selection, and used in the message header to identify the terminal from which a message is transmitted. This address may also be used as an identification prefix to acknowledge (ACK) that a terminal is ready to receive a message or to identify NAK. On receipt of a message, the receiving station may use AD1-AD2 to verify that the message originated at the polled terminal. For group addressing of broadcast to all terminals, AD1-AD2 indicates the terminal that will acknowledge receipt of the message. In systems which preclude "downstream" (terminal-to-terminal) communication, AD1-AD2 in the header, transmitted by the central computer, may be defined to represent the terminal address and is used for address checking.

ASCII (American Standard Code for Information Interchange)

This code, established as an American standard by the American Standards Association, defines codes for a set of characters to be employed in the interchange of information between business equipment over telephone and telegraph circuits. The code consists of 128 control and graphic characters.

Application Program

A program that processes data and is usually unique to one type of application.

Automatic Calling Unit (ACU)

A device that may be furnished by a communications carrier allowing a business machine to automatically establish a dialed link over the communications network.

Autopoll

A technique used by the central computer to interrogate terminals to determine if they have anything to transmit without interrupting the Network Controller for the initiation of each station poll.

Asynchronous transmission

A means of transmission which requires that there always be an integral number of unit intervals within a character between any two significant instants in time. However, this need not be so between two significant instants in time in an integral number of unit intervals in different characters.

Bandwidth

An expression of Hertz measurement which states the difference between the high and low frequencies in a communications channel.

Baud

A unit of signaling speed equal to the number of discrete conditions or signal events per second.

Baudot code

A code used in the transmission of data in which five bits represent one character. Named for Emile Baudot, a pioneer in printing telegraphy, it is sometimes referred as five-bit code, five-channel code, five-unit code, "Teletype" code, although a "Teletype" unit is no longer limited to its use.

Binary Synchronous (BSC)

A method of data transmission which allows sending or receiving of data streams. The data streams may contain bit patterns which would normally be detected as control character sequences. Therefore, data streams containing object program code may be transmitted. The transmission is always synchronous.

Bit rate

The speed, usually expressed in bits per second (BPS), with which bits are transmitted over a communication channel.

Bit synchronization

The process by which the transmitting and receiving bit frequencies are made substantially the same and are maintained, by means of correction if necessary, in a desired phase relationship.

Block

A group of characters or bits sent as an integral unit. Usually an error-checking procedure is applied over a block, for control and recovery purposes.

Block Check Character (BCC)

The Block Check Character is a redundant character added to the end of a transmission block for the purpose of error detection.

Block number (BL#)

An option which may be used when data must be subdivided into separate units for transmission. The block number consists of a two-character number identifying the sequential block number in a blocked message. The first character is always a DLE character.

BREAK

The facility to enable a receiving device to interrupt the transmitting device and, by doing so, to be able to take control of the circuit.

Broadcast

The simultaneous sending of a message to several stations of a network.

Broadcast Select (BSL)

The Broadcast Select control code is used to indicate "This is a broadcast message" to all stations. In the broadcast sequence, AD1-AD2 identify the station which will acknowledge receipt of the message. Broadcast Select is followed immediately by a transmission block, without requiring acknowledgement of the selection.

Carrier system

A means of obtaining a number of channels over a single path by modulating each channel upon a different carrier frequency and demodulating at the receiving point to restore the signals to their original form.

Cathode Ray Tube (CRT)

A large electron tube used to emit electrons onto a phosphor screen, thus creating a visual display.

Central office

The place where a common carrier has equipment which interconnects customer transmission lines.

Centralized operation

The control discipline used in a multipoint data communication network in which all message transfers must involve the control station. Transmissions directly between tributary stations are not allowed.

Channel

The communication path used to transmit signals between two or more points. Often referred to as a link, circuit, line, or path.

Character

A set of elements arranged in orderly groups to represent digits, symbols, or letters. Represented in two forms: (1) for use by computers, business machines, communications facilities, etc., usually in groups of binary bits; and (2) for use by man in conveying an understandable form of decimal digits, alphabetic characters, punctuation, or other special symbols. Characters may be represented using groups of bits, commonly five, six, seven, or eight bits.

Character syncronization

A process in which the character frequencies of the transmitting and receiving ends of a transmission circuit are maintained in a phase relationship in order that the receiver can derive the transmitted characters from the signals received.

Circuit

The configuration of equipment used in transmitting data from one location to another. A circuit may involve more than one type of facility.

Circuit assurance

The function of verifying the existence or the operational state of the communication channel between stations.

Code

A system of symbols and rules for use in representing information.

Common carrier

A company that provides communication service for public hire.

Communication link

The connection of two or more stations by the same communication channel. This link includes the communication control capability of the stations connected in the link.

Communication system

The combination of all the links, link interface equipment, application and systems software, including control procedures, that are required to effect the transmission of coded information between stations in the system.

Conditioning

Private and leased transmission lines can be conditioned to reduce distortion and thereby provide data transmission at lower error rates.

Connection

The established path between two or more terminal installations. A permanent connection is established by using switching facilities.

CCITT (Consultative Committee International Telegraphic and Telephonic)

An international committee that defines telegraphic and telephonic interconnections and switching standards.

Contention (CON)

The contention character instructs all terminals receiving this character to go to the contention mode. The contention mode is an operational condition on a data communication link in which no station is designated as a master station. With contention mode, each station on the link must monitor the signals on the link and wait for a quiescent condition before initiating a bid for master status.

Continuous operation

A type of message transmission in which the master station need not stop for a reply after transmitting each acknowledgement unit, but may continue transmission with the next acknowledgement unit.

Control character, data communications

A functional character intended to control or facilitate transmission of information over communication networks. The major communication control characters provided are ACK, BEL, CAN, CR, DC1, DC2, DC3, DC4, DEL, DLE, ENQ, EOT, ESC, ETX, FIB, FF, FS, FSL, GS, GSL, HT, LF, NAK, NUL, POL, RS, SI, SO, SHO, STX, SYN, US, and VT.

Control procedure, data communications

The means used to provide for the orderly communication of information between the stations on a data communications link.

Control state

An operational state on a data communications link in which characters, other than communication controls, may be given control interpretations. A data link is in the control state when message transfer is not in progress.

Control station

A permanently designated (unaffected by link control procedures) station on a data link with the overall responsibility for the orderly operation of the link. A control station generally has additional control capabilities (e.g., the capability to poll other stations) not provided at other stations, and is designated to initiate error recovery procedures in the event of certain abnormal conditions on the link.

Conversational mode

An operational mode in which message information is used in lieu of, or in addition to, control characters as replies for message information.

Cyclic Redundancy Check (CRC-16)

A 16-bit redundant character added to the end of a transmission block for the purpose of error detection and control. All characters following STX (or SOH) except SYN and the first DLE of a DLE DLE sequence are included in the CRC accumulation. A cyclic redundancy check is a division, performed by both the transmitting and receiving stations, using the numeric binary value of the message as a dividend, which is divided by a fixed polynomial. The most common polynomial used is $X^{16}+X^{15}+X^{n}+1$. The quotient is discarded, and the remainder serves as the check character, which is then transmitted as the Block Check Character (BCC) immediately following a check-point character (ITB, ETB, or ETX). The receiving station compares the transmitted remainder to its own computed remainder, and finds no error if they are equal.

Data communications system

One or more data links, each of which may be operating in the same or a different mode.

Data communications

The transfer of encoded information by means of electrical transmission systems.

Data link

This is an ensemble of terminal installations and the interconnecting network, together with controlling procedures operating in a particular mode, that permits information to be exchanged between locations.

Data Link Excape (DLE)

Data Link Escape is a control character that is used to change the meaning of a limited number of contiguously following characters, which then become supplementary data transmission control functions. Only graphics and transmission control characters can be used in DLE sequences.

The Data Link Escape character combinations are used primarily for binary synchronous operation. The Data Link Escape character is always the first character of the two character combinations. The DLE combination characters are listed below:

ACK0 **ACK1** **RVI** **WACK**	These characters are used only for binary synchronous operation. Refer to this Glossary for a detailed description of each of these characters.
DLE STX	Used to start transparent mode in BSC operation.
DLE DLE **DLE EOT** **DLE ENQ** **DLE SYN**	Used as control characters during transparent mode of operation. The mnemonic for DLE EOT is DEOT.
DLE ETB **DLE ETX** **DLE ITB**	These characters are used to end transparent mode of operation.

Data set

A circuit termination device used to provide an interface between a data communication circuit and a data terminal. A modulation and/or demodulation function is typically performed in a data set.

Data terminal

That part of a station concerned with the functions of generating data and/or recording or displaying of data, together with the control equipment or system software necessary to control these functions.

Dedicated channel

A communication channel provided for the exclusive use of a specific subscriber on a contract basis. (See Private line.)

Delimiters

Control signals used to define the extent of a particular sequence of characters.

Demodulation

The process of retrieving an original signal from a modulated carrier wave. This technique is used in data sets to make communication signals compatible with business machine signals (contrast with definition of Modulation).

DEOT

The mnemonic for the DLE EOT communication control sequence, used to signal that a disconnect of a switched circuit must be initiated.

Dial-up

The use of a dial or pushbutton telephone to establish a station-to-station telephone contact.

Disconnect (DEOT)

A Disconnect is a communication control sequence consisting of DLE followed by EOT, used to signal that a disconnect of a switched circuit must be initiated.

Display unit

A device which provides a visual representation of data (see Cathode ray tube).

Duplex

In communications, pertains to the capability of simultaneous two-way transmission.

End of Test (ETX)

The End-of-Text control character is used to indicate the end of the last block of text.

End of Transmission (EOT)

The End-of-Transmission control character is used to indicate the conclusion of a communication sequence. Receipt of an EOT will set the terminal in a control state, waiting for a polling, selection, or contention sequence. EOT may be transmitted by a master station to abort a transmission sequence. To insure that terminals are in a control state, EOT precedes a communication control sequence. EOT is transmitted by a remote terminal as a "no traffic" response to a poll.

End of Transmission Block (ETB)

The End-of-Transmission Block control character is optionally used when messages are of sufficient length to warrant their being broken into smaller transmission blocks. ETB indicates the end of a block of data, either in a heading or in the message text. The heading or text is resumed after transmission of a block number and SOH or STX.

Enquiry (ENQ)

The Enquiry control character is used as a request for a response of station status or for a retransmission of control characters. This character is also used as the final character of a poll or in a selection when response from the other station is required.

Error

Any received character or sequence of characters that does not conform to those transmitted.

Error control

A system that detects the present of errors. In some systems, refinements are added that correct some of the errors, either by operations on the received data or by retransmission of the data from the source.

Error recovery procedure

Data communications control procedures used to restore normal operation to a data link after unusual (abnormal) events have occurred.

Exchange

A defined area, served by a communications carrier, within which the carrier furnishes service.

Facility

The type of communication medium used to provide communication circuits (e.g., cable, radio, open-wire).

Facsimile (FAX)

Transmission of pictures, maps, diagrams, etc. The image is scanned at the transmitter, reconstructed at the receiving station, and duplicated on some form of paper.

Family

A multistation file.

Family population

The number of active stations in the family of the file. Family size is the number of stations assigned to the file.

Fast Select (FSL)

The Fast Select character is used to indicate "This is a fast select," in a selection sequence transmitted by the central computer. Fast Select is followed immediately by a transmission block, without requiring acknowledgement of the selection.

Filler

A character used as a time-fill or space-fill when a block of a specified size is required and the heading and/or text characters are of insufficient length for the requirement.

Format

The predetermined arrangement of characters, field, lines, page numbers, punctuation marks, etc., used to transfer data from one location to another. Refers to input, output and files.

Full-duplex channel

A communication channel where the signaling may be in both directions simultaneously. The signaling speeds used for the two directions of transmission on a full-duplex channel need not be the same.

Full-duplex transmission

A type of transmission where information is sent in both directions simultaneously.

Group Select (GSL)

The Group Select character is used to indicate "This is a message for a group of stations." In the group select sequence, AD1-AD2 identifies the station which will acknowledge receipt of the message. Group Select is followed immediately by a transmission block, without requiring acknowledgement of the selection.

Half-duplex channel

A communication channel where the signaling may be in either direction, but not in both directions at the same time.

Half-duplex transmission

A type of transmission where information is sent in one direction or the other direction, but not in both directions simultaneously. (Refer to two-way alternate transmission.)

Handshaking

This is the exchange of control signals between data sets when the connection is established.

Header

A sequence of characters that may precede the text of a message to provide the information necessary to route the message to its ultimate destination(s). A message header may also contain communications relating information other than routine instructions. A message header is preceded by an SOH character and is ended by an STX character.

Hertz (Hz)

A measure of frequency or bandwidth. The same as cycles per second.

Holding time

The length of time a communication channel is in use for each transmission. Includes both message time and operating time.

Identification

A sequence of characters used during the establishment of a connection and subsequently, if needed, to identify a station.

Identifier

A sequence of one or more characters transmitted by a station in order to identify itself.

Information block

A sequence of characters of fixed or variable length which is a subdivision of an information message formed for the purpose of meeting transmission requirements.

Information message

A sequence of characters conveying the text. It may also convey supplementary information forming a heading.

In-plant system

A data-handling system confined to one building or a number of buildings in one locality.

Intermediate Transmission Block End (ITB)

The ITB control character is used for binary synchronous operations to divide a message (heading or text) for error-checking purposes, without causing a reversal of transmission direction. The block-check character immediately follows ITB and resets the block-check count. After the first intermediate block, successive intermediate blocks need not be preceded by STX or SOH. (For transparent data, which may contain control character sequences, each successive intermediate block must begin with DLE STX). If one intermediate block is heading and the next is heading and the next intermediate block is text, STX must begin the text block.

Normal line turnaround occurs after the last intermediate block, which is terminated by ETB or ETX (DLE ETB or DLE ETX for transparency). When one of these ending characters is received, the receiving station responds to the entire transmission. If a block-check error is detected for any of the intermediate blocks, a negative reply is sent, which requires retransmission of all intermediate blocks.

All bisynchronous stations must have the ability to receive ITB and its attendant BCC. The ability to transmit the ITB character is a station option.

Line adapter

A line adapter consists of a logic card used to interface a data set or communications line into a Multi-Line or Single-Line I/O Control.

Line switching

A switching technique that temporarily connects two lines together so that two stations can directly exchange information.

Link, data communication

The logical association of two or more stations interconnected by the same communication channel. A data link includes the communication control capability of the connected stations.

Local channel

A channel connecting a communications subscriber to a central office.

Master station

A station temporarily designated (can be changed by the link control procedures) to have control of the data link at a given instant. Master status is normally conferred upon a station so that it may transmit a message.

Message

A sequence of characters arranged in a form suitable for the purpose of conveying information from an originator to one or more destinations (or addresses). It contains the information to be conveyed (called the text) and may, in addition, contain communication information to aid in the routing of handling of the message (called the header).

Message Control System (MCS)

A Message Control System is an optional program that controls message flow between application programs and the Network Controller.

Message format

Rules for the placement of such portions of a message as message header, address, text, and end of message.

Message numbering

The identification of each message within a communications system by the assignment of a sequential number.

Message switching

The technique of receiving a message, storing it until the proper outgoing circuit and station are available, and then retransmitting toward its destination.

Microwave

All electromagnetic waves in the radio frequency spectrum above 890 megahertz.

Modem

See definition of Data set.

Modulation

The process by which some characteristic of one wave is varied in accordance with another wave. This technique is used in data sets to make business machine signals compatible with communication facilities (contrast with definition of Demodulation).

Most Significant Bit (MSB)

The most significant bit is the bit of a binary number which is assigned the greatest numerical value. It is generally assigned to the most significant bit position, as in the decimal numbering system.

Multidrop

See Multipoint network.

Multi-Line Control (MLC)

The Multi-Line Control is an I/O control that provides the functional control between the computer system and the line adapters. An integral part of a multi-line data communications subsystem.

Multiple address message

A message to be delivered to more than one destination.

Multiplexing

The division of a transmission facility into two or more channels.

Multipoint circuit

A circuit interconnecting several stations (see definition of Multipoint network).

Multipoint network

(1) A data communication line connecting three or more stations; (2) a data communication link with the control capability necessary to interconnect three or more stations.

NAK

The Negative Acknowledgement control character is used as a negative response to a normal selection (indicating not ready to receive) or a transmission (indicating character parity failure for any character of a block or, in a message, a failure of the BCC, or block number sequence check failure). NAK may optionally be preceded by station identification AD1, AD1, or other information.

Network

The ensemble of equipment through which connections are made between terminal installations. These equipments operate in real time and do not introduce, store, or forward delays. A switched telephone network is the network of telephone lines normally used for dialed telephone calls. A private network is a network of communication channels reserved for the use of one customer.

Network Controller (NC)

The Network Controller interfaces with the hardware of the Data Comm network and is responsible for direct communication with the network.

Network Definition Language (NDL)

A descriptive, free-form language for defining and implementing a data communications network. The NDL compiler analyzes the input statements and generates a custom Network Controller.

Network Information File (NIF)

Contains tables that describe the physical and logical attributes of the network. The initialization values for the LINE, STATION, TERMINAL, and FILE tables of the Network Controller are contained in the NIF.

Non-centralized operation

A control discipline for multipoint data communication links in which transmissions may be between tributary stations or between the control station and a tributary station(s).

Off line

Used to describe terminal equipment which is not connected to a transmission line. Can also describe other devices not in direct communication with the central processing unit.

On line

Used to describe terminal equipment which is connected to a transmission line. Can also describe other devices in a direct communication with the central processing unit.

Pad character (PAD)

To insure that the first and last characters of a transmission are properly transmitted by the data set, all binary synchronous stations add a PAD character before and after each transmission. The leading PAD character consists of a hexadecimal "55" character SYN character, the trailing PAD character consists of all "1" bits (hexadecimal "F-F" character). Although the PAD character is comprised of eight bits, the receiver generally only checks the first four bit positions.

Parallel transmission

The simultaneous transmission of a certain number of signal elements constituting the same telegraph or data signal. For example, all bits of a character may be sent simultaneously in parallel transmission. (Contrast with definition of Serial transmission).

Parity bit

A bit associated with a character or block for the purpose of checking for the presence of error within that character or block. This bit is chosen to make the modulo 2 sum of the bits (including the parity bit) in the character or block a zero (even) or a one (odd) as required by the system.

Parity check

A check that tests whether the number of ones (or zeros) in an array of binary digits is odd or even.

Passive station

A station on a data link that is (temporarily) neither a master station nor a slave station.

Point-to-point connection

A configuration in which a connection is established between two, and only two, terminal installations. The connection may include switching facilities.

Poll (POL)

The Poll character is used to indicate "this is a poll," preceding ENQ in a polling sequence.

Polling

A technique for assignment of master status to a particular station on a Data Comm link. Polling is centrally controlled in order to maintain a strict discipline over the operation of a number of points.

Polling supervisory sequence

A supervisory sequence that performs a polling function.

Prefix

A sequence of characters (other than communication controls) used in a supervisory sequence to define or qualify the meaning of the supervisory sequence.

Private line

A channel or circuit furnished a subscriber for his exclusive use (see definition of Dedicated channel).

Queue

A linear list for which all insertions are usually made at one end of the list and all deletions and other accesses are made at the other end.

Recovery procedure

A process by which a responsible station within the network attempts to resolve either conflicting or erroneous conditions arising in the communication process. The control or master station is responsible for this procedure.

Redundancy

The portion of the total information contained in a message which can be eliminated without loss of essential information.

Remote Job Entry (RJE)

A computer communications system in which a central computer executes programs that were sent to it by other computers that appear as terminals.

Reverse channel

A communication channel between a slave station and a master station; used exclusively for control signals.

Reverse Interrupt (RVI)

The reverse interrupt control character is used only for binary synchronous operation. The RVI control sequence is a positive response used in place of the ACK0 or ACK1 positive acknowledgement. RVI is transmitted by a receiving station to request termination of the current transmission for one of the following reasons: (1) The receiving station must transmit a high-priority message to the sending station; (2) The control station, when in a multipoint environment, acts as a receiver and wants to communicate with another station on the line. Successive Reverse Interrupt control characters cannot be transmitted, except in response to ENQ.

The sending station treats the RVI as a positive acknowledgement and responds by transmitting all data that prevents it from becoming a receiving station. More than one block transmission may be required to empty the buffers of the sending station.

The ability to receive RVI is mandatory for all binary synchronous stations, but the ability to transmit RVI is optional. RVI is represented by a DLE character followed by a hexadecimal 6B.

Recommended Standard 232C (RS232C)

This standard, provided by the Electronic Industries Association, is the recommended interface between data communications terminal equipment and data communications equipment using serial binary data interchange.

Select (SEL)

The Select character is used to indicate "this is a normal select," when it precedes ENQ in a selection sequence.

Selection

A technique for assignment of slave status to a particular station on a Data Comm link.

Selection supervisory sequence

A supervisory sequence that performs a selection function.

Selective calling

The ability of a transmitting station to specify which station or stations on the same line are to receive a message.

Sequential select (SEQ)

The sequential select character is used to indicate that a group of remote terminals is being selected to receive a message addressed to that group. The last terminal selected in the group will acknowledge receipt of sequential select. Sequential select must be followed by AD1-AD2 of another terminal.

Serial transmission

Transmission, at successive intervals, of the signal elements constituting the same telegraph or data signal. The sequential elements may be transmitted with or without interruption provided that they are not transmitted simultaneously. (Contrast with definition of Parallel transmission.)

Shutdown time

The time at which the Network Controller performs several functions necessary to the operation of the Data Comm network. The major functions are the emptying of all output queues, disabling any station input in progress, and issuing termination notification to all executing application programs or Message Control Systems. These functions must be performed before the Network Controller terminates itself.

Signal-conversion equipment

That part of the terminal installation belonging to the data channel, comprising at least one modulator or one demodulator. This equipment provides modulation according to the signals to be transmitted, and/or demodulation of the signals received.

Simplex channel

A communication channel where the signaling may be in one direction only.

Single Line Control (SLC)

The Single Line Control is an I/O control that provides the functional control between a computer system and a line adapter. The SLC is an integral part of a single line data communications subsystem.

Single-address message

A message to be delivered to only one destination.

Slave station

A station that has indicated its readiness to receive a message. The assignment of slave status is temporary and continues for the duration of a transmission.

Start Of Header (SOH)

The Start-Of-Header control character is required only when a meassage header is to be sent with a transmission. When used, SOH is the first of a sequence of characters used for the message header. The header also may contain terminal identification AD1, AD2, and may, under definition for the specific application, contain other information pertinent to the transmission number (Xm#). A header is ended by STX.

Start of Text (STX)

The Start-Of-Text control character precedes a sequence of characters which form the text of the transmission. STX terminates a header.

Start/Stop transmission

Start/Stop transmission is an asynchronous transmission in which a group of code elements corresponding to a character signal is preceded by a start signal. The start signal prepares the receiving mechanism for the reception and registration of a character and is followed by a stop signal. The stop signal brings the receiving mechanism to rest in preparation for the reception of the next character.

Station

The aggregate of the terminal equipment (and system software), and communication control equipment (and system software) attached to a particular line adapter for and SLC or MLC. Used for the input or output of information for the communications system of which it is a part.

Stop element (in a Start/Stop system)

Binary element that indicates to the terminal installation the completion of receipt of a character to bring the receiving mechanism to rest in preparation for receipt of the next character.

Store-and-forward

Process of message handling used in a message-switching system.

Stunt box

A device to control the non-printing functions of a teleprinter terminal. Control characters can be sent to the stunt box over the communications channel.

Supervisory sequence

A sequence of communication control and, possibly, non-communication control characters that perform a defined control function. A supervisory sequence consists of a prefix together with one or more control character delimiters.

Switching

Operations involved in interconnecting circuits in order to establish a temporary communication channel between two or more stations.

Synchronous idle (SYN)

The synchronous idle control character is used only by a synchronous transmission system in the absence of any other character to provide a signal for establishing and retaining a synchronism. On initiation of synchronous transmission, a number of SYN characters are transmitted prior to transmission of any other character to permit the receiving station to acquire character synchronization.

SYN is also used as a "time fill" when no other characters are available for transmission at any point in a character sequence except between ETB or ETX and the next following BCC. SYN is purged at the receiving terminal and is not included in the summation for BCC.

Synchronous transmission

A transmission process in which there is always an integral number of unit intervals between any two significant instants.

Telegraphy

A system of communication for the transmission of graphic symbols, usually letters or numerals, by use of a signal code,.

Telegraphy, printing

A method of telegraph operation in which the received signals are automatically recorded in printed characters.

"Teleprinter"

Term used to refer to the equipment used in a printing telegraph system. A "teletypewriter".

Transparent mode

An operational mode in which all coded combinations of eight-bit characters are allowed as message characters within a message text.

Tributary station

All stations, other than the control station, which are on a non-switched multipoint network are called tributary stations.

Turnaround Time

This is the time required by data sets when switching between transmit and receive modes.

Two-Way Simultaneous Transmission

A type of transmission where information is sent in both directions simultaneously. (See Full-duplex.)

Two-Way Alternate Transmission

A type of transmission where information is sent in one direction or the other direction, but not both directions simultaneously. (See Half-duplex.)

Unattended operation

The automatic features of a station's operation that permit the transmission and reception of messages without a computer operator being in attendance.

USASCII (United States of America Standard Code for Information Interchange)

The USA standard code for information interchange. (See definition of ASCII.)

User Programming Language (UPL)

A compiler-level language used primarily for writing Message Control Systems (MCS) for B 1700 Data Communications Systems.

Vertical Redundancy Checking (VRC)

Vertical redundancy checking consists of generating an odd-parity as each character is received. This technique is used only in the ASCII normal mode (not with EBCDIC). The test is performed on every character, including the LRC.

Voice-grade channel

A channel suitable for transmission of speech, digital, or analog data, or facsimile, generally with a frequency range of about 300 to 3000 Hertz.

Wait-before transmit positive acknowledgement (WACK)

The WACK character (used only for binary synchronous operation) allows a receiving station to indicate a "temporarily-not-ready-to-receive" condition to the transmitting station. This character can be sent as a response to a text or heading block, selection sequence (multipoint), line bid (point-to-point with contention), or as an ID (identification) line bid sequence (switched network). WACK is a positive acknowledgement to the received data block or to selection.

The normal transmitting station response to WACK is ENQ, but EOT and DLE EOT are also valid responses. When ENQ is received, the station will continue. The ability to receive WACK is mandatory for all binary synchronous stations, but the capability is optional.

The WACK character is represented by a DLE character followed by a hexadecimal 7C.

Wideband channel

A channel wider in bandwidth than a voice-grade channel.

Appendix B. PROGRAM EXAMPLES

CUSTOMER–INQUIRY

```
BURROUGHS B1700 COBOL COMPILER
INQUIRY
1
                IDENTIFICATION DIVISION.
                PROGRAM-ID. CUSTOMER-INQUIRY.
                REMARKS. THIS PROGRAM ALLOWS INQUIRY INTO THE CUSTOMER MASTER
                    FILE BY RELATIVE RECORD NUMBER.
                ENVIRONMENT DIVISION.
                CONFIGURATION SECTION.
                SOURCE-COMPUTER. B-1700.
                OBJECT-COMPUTER. B-1700.
                INPUT-OUTPUT SECTION.
                FILE-CONTROL.
                    SELECT CUSMAS ASSIGN TO DISK
                        FILE-LIMIT IS 1 THRU 26
                        ACCESS MODE IS RANDOM
                        ACTUAL KEY IS RECORD-NUMBER.
                    SELECT REMOTE1
                        ASSIGN TO REMOTE
                          ACTUAL KEY WS-KEY.
                DATA DIVISION.
                FILE SECTION.
                FD  REMOTE1
                    FILE CONTAINS 2 STATIONS
                      VALUE OF ID IS "REMOTE1".
                01  AMT-MSG2   PC X(256).                                  [0001] 000,0000120
                01  RMT-MSG.                                                      000,0000120
                    03  RMT-TEXT           PIC X(2).                       [0002] 000,0000120
                FD  CUSMAS
                    FILE  CONTAINS 5 BY 100 RECORDS
                    VALUE OF ID IS "MASTER"/"FILE".
                01  CUSTOMER-RECORD.                                              000,0000632
                    02  FILLER PC X(31).                                          000,0000632
                    02  CURBAL  PC S9(7)V99 USAGE CMP.                            000,0000694
                    02  DAYS30  PC S9(7)V99 USAGE CMP.                            000,0000704
                    02  DAYS60  PC S9(7)V99 USAGE CMP.                            000,0000714
                    02  DAYS90  PC S9(7)V99 USAGE CMP.                            000,0000724
                    02  FILLER  PC X(3).                                          000,0000734
                    02  GRPYTD  PC S9(5) USAGE CMP.                               000,0000740
                    02  SLSYTD  PC S9(7) USAGE CMP.                               000,0000746
                    02  CUSNAM  PC X(24).                                         000,0000754
                WORKING-STORAGE SECTION.
                77  RECORD-NUMBER   PC  9(8)    CMP.                              000,0000802
                77  PAST-DUE        PC  9(8)V99 CMP.                       [0003] 000,0000810
                77  RELREC          PC  99.                                [0004] 000,0000820
                77  NO-OF-STATIONS PIC 99 VA 02.                           [0005] 000,0000824
                01 START-HANDLER PIC X(15) VA "EX DEMO/HANDLER".                   000,0000828
                01 SEND-HANDLER-TO-EOJ PIC X(2), VA "QC".                          000,0000858
                01  WS-KEY.                                                       000,0000862
                    03  FILLER  PIC 9.                                            000,0000862
                    03  STAT-RSN  PIC  99 VA 1.                           [0006] 000,0000864
                    03  TEST-LNGTH   PIC 9999.                            [0007] 000,0000868
                    03  MSG-TYPE     PIC XXX.                             [0008] 000,0000876
                01  FIRST-MSG.                                                    000,0000882
                    03  FILLER PIC 9(14) CMP VA @0C000000000000@.                 000,0000882
                    03  FILLER PC X(22) VA "ENTER CUSTOMER NUMBER ".              000,0000896
                    03  FILLER PIC 9(12)  CMP VA @1F4040401E12@.                  000,0000940
                01  SECOND-MSG.                                                   000,0000952
                    03 FILLER PC 9(14) CMP VA @0C000000000000@.                 000,0000952
                    03  NAME PC X(24).                                            000,0000966
                    03 FILLER PIC 9(6) CMP VA @0D0000@.                         000,0001014
                    03 FILLER PIC 9(6) CMP VA @0D0000@.                         000,0001020
                    03  FILLER  PC X(16) VA "CURRENT BALANCE ".                   000,0001026
                    03  BALCUR    PC Z,ZZZ,ZZZ9.99-.                              000,0001058
                    03 FILLER PIC 9(6) CMP VA @0D0000@.                         000,0001086
                    03  FILLER  PC X(17)   VA "PAST DUE BALANCE ".                000,0001092
                    03  BALPST   PC Z,ZZZ,ZZ9.99-.                                000,0001126
                    03 FILLER PIC 9(6) CMP VA @0D0000@.                         000,0001152
                    03  FILLER   PC X(10)  VA "YTD SALES ".                       000,0001158
                    03  FILLER PC X(10) VA SPACES.                                000,0001178
                    03  YTDSLS  PC Z,ZZZ,Z79-.                                    000,0001198
                    03 FILLER PIC 9(6) CMP VA @0D0000@.                         000,0001218
                    03  FILLER PIC X(11) VA "YTD PROFIT ".                        000,0001224
                    03  FILLER PC X(13) VA SPACES.                                000,0001246
```

```
                03  YTDGRP     PC ZZZZ9-.                                                                    000,0001272
                03 FILLER PIC 9(6) CMP VA @0D0000@.                                                        000,0001284
                03  FILLER PIC  X(22) VA "ENTER CUSTOMER NUMBER ".                                          000,0001290
                03  FILLER PIC 9(12)  CMP VA @1F4040401E12@.                                             000,0001334
                03  FILLER PIC 9(12)  CMP VA @000000000000@.                                             000,0001346
            PROCEDURE DIVISION.
            DECLARATIVES.                                                                                 000,0000000
            ALL-USE  SECTION.                                                                             000,0000000
                USE AFTER STANDARD ERROR PROCEDURE ON REMOTE1.                                            000,0000000
            LABEL-1.                                                                                      000,0000000
                DISPLAY "ERROR - RSN = " STAT-RSN.                                                        000,0000000
                MOVE 1 TO SW1.                                                                            000,0000326
            ALL-EXIT.                                                                                     000,0000367
                EXIT.                                                                                     000,0000367
            END DECLARATIVES.                                                                             000,0000367
            MAIN-LOOP SECTION.                                                                            000,0000376
            DUMMY-NAME.                                                                                   000,0000376
                ZIP START-HANDLER.                                                                        000,0000376
                OPEN INPUT CUSMAS.                                                                        000,0000506
                OPEN I-O REMOTE1.                                                                         000,0000629
                IF NO-OF-STATIONS = 1 GO TO PARA1.                                                        000,0000711
                PERFORM PARA1 VARYING STAT-RSN FROM 1 BY 1 UNTIL STAT-RSN                                 000,0000744
                GREATER THAN                                                                              000,0000761
                NO-OF-STATIONS.                                                                           000,0000761
                GO TO BIG-LOOP.                                                                           000,0000842
            PARA1.                                                                                        000,0000858
                MOVE "000" TO MSG-TYPE.                                                                   000,0000858
                MOVE 35 TO TEST-LNGTH.                                                                    000,0000895
                WRITE AMT-MSG2 FROM FIRST-MSG.                                                            000,0000916
            BIG-LOOP.                                                                                     000,0001253
                READ REMOTE1 AT END GO TO END-RTN.                                                        000,0001270
                IF SW1 = 1 MOVE 0 TO SW1 GO TO BIG-LOOP.                                                  000,0001673
                IF RMT-TEXT = "99" GO TO END-ONE.                                                         000,0001783
                IF RMT-TEXT = "00" GO TO PARA1.                                                           000,0001828
                MOVE RMT-TEXT TO RELREC.                                                                  000,0001873
                ADD 1 TO RELREC.                                                                          000,0001885
                MOVE RELREC TO RECORD-NUMBER.                                                             000,0001898
                IF RELREC IS GREATER THAN 26 GO TO PARA1.                                                 000,0001934
            READ-RTN.                                                                                     000,0001971
                READ CUSMAS INVALID KEY GO TO ERROR-RTN.                                                  000,0001971
            INITIALIZATION-RTN.                                                                           000,0002422
                ADD DAYS30 DAYS60 DAYS90 GIVING PAST-DUE.                                                 000,0002422
                MOVE CURBAL TO BALCUR.                                                                    000,0002612
                MOVE GRPYTD TO YTDGRP.                                                                    000,0002689
                MOVE SLSYTD TO YTDSLS.                                                                    000,0002766
                MOVE CUSNAM TO NAME.                                                                      000,0002843
                MOVE PAST-DUE TO BALPST.                                                                  000,0002903
WARNING...(252) RECEIVING FIELD TRUNCATION
                MOVE "000" TO MSG-TYPE.                                                                   000,0002956
                MOVE 203 TO TEST-LNGTH.                                                                   000,0002993
                WRITE AMT-MSG2 FROM SECOND-MSG.                                                           000,0003018
                GO TO BIG-LOOP.                                                                           000,0003355
            ERROR-RTN.                                                                                    000,0003371
                GO TO PARA1.                                                                              000,0003371
            END-ONE.                                                                                      000,0003387
                ZIP SEND-HANDLER-TO-EOJ.                                                                  000,0003387
                GO TO BIG-LOOP.                                                                           000,0003517
            END-RTN.                                                                                      000,0003533
                CLOSE REMOTE1.                                                                            000,0003533
                CLOSE CUSMAS.                                                                             000,0003615
                STOP RUN.                                                                                 000,0003734
            END-OF-JOB.                                                                                   000,0003772
```

```
  C O D E     D I C T I O N A R Y
      BYTE LENGTH   CODEFILE RELATIVE DISK ADR
      ...........   ..........................
  000 00000472      000010
  001 00000070      000013
      00000542      TOTAL CODE

  D A T A     D I C T I O N A R Y
      BYTE LENGTH   CODEFILE RELATIVE DISK ADR
      ...........   ..........................
  000 00000927      000003

  S - M A C H I N E    P A R A M E T E R S    ( S C R A T C H P A D )

  LENB=9, SEGB=0, DISPB=11, COPXB=4, COPB=24, D.E.F=354, BDISPB=12
  BASE RELATIVE ADDRESSES:
    DATA-SEGMENT-0=400, COP-TABLE=40, STACK=6413 (BIT LENGTH=1000)

  P R O G R A M    P A R A M E T E R    B L O C K

  FIRST-EXECUTABLE-INSTRUCTION=0,376
  INTERPRETER-NAME=COBOL      /INTERP
  STATIC-CORE=7416 BITS
  DYNAMIC-CORE=0 BITS
  DATA DICTIONARY STARTS AT CODEFILE SEGMENT 2, 1 ENTRY
  CODE DICTIONARY STARTS AT CODEFILE SEGMENT 9, 2 ENTRIES
  FILE PARAMETER BLOCKS START AT CODEFILE SEGMENT 14, 2 ENTRIES

LAST ERROR AT SEQUENCE NUMBER       . 1 WARNING
***** COMPILATION COMPLETE
ELAPSED TIME : 01 MINUTE, 15 SECONDS
PROGRAM REQUIRES 16 DISK SEGMENTS OF 180 BYTES EACH.
MEMORY REQUIREMENTS
  0000472 BYTES = LARGEST CODE SEGMENT
  0000927 BYTES = BASE-TO-LIMIT AREA
  0000313 BYTES = DICTIONARIES AND RUN STRUCTURE
  0000689 BYTES = FILE BUFFERS & FILE INFO AREAS - INCLUDES 129 BYTES (+72 TO 540 IF DISK) FOR EACH FILE
  0002401 BYTES = ESTIMATED MEMORY REQUIRED TO RUN IF ALL FILES OPEN
132 SYMBOLIC RECORDS COMPILED AT 105.600 RECORDS PER MINUTE.
```

GAME

```
BURROUGHS B1700 COBOL COMPILER
NUMBER    /GAME

           IDENTIFICATION DIVISION.
           PROGRAM-ID. GAME.
           DATE-COMPILED.
                        TIME 07:56 DATE 10/16/74.
           ENVIRONMENT DIVISION.
           INPUT-OUTPUT SECTION.
           FILE-CONTROL.
               SELECT TDFILE2 ASSIGN TO REMOTE.
           DATA DIVISION.
           FILE SECTION.
           FD  TDFILE2.
           01  R-RECORD.                                                    [0001] 000,0000066
               03  INPUT-NO PC 999.                                         [0002] 000,0000066
               03  FILLER  PC X(253).                                             000,0000072
           WORKING-STORAGE SECTION.
           77 WAIT-NO PIC 9(5).                                                   000,0000578
           77  WAIT-SCREEN  PIC 9 VA 0.                                     [0003] 000,0000588
           01  NUMBERS.                                                           000,0000590
               03  INPUT-NUMBER PIC 999.                                    [0004] 000,0000590
               03  SECRET-NUMBER  PC 999.                                   [0005] 000,0000596
               03  LOW-NO   PC 9999.                                        [0006] 000,0000602
               03  HIGH-NO  PC 9999.                                        [0007] 000,0000610
           01  SCREEN-1.                                                    [0008] 000,0000618
               03  FILLER PIC 9(14) CMP VA  @3C000000000000@.                     000,0000618
               03  FILLER PIC X(21) VA "          NUMBERS GAME".                  000,0000632
               03  FILLER PIC 9(6) CMP VA @0D0000@.                               000,0000674
               03  FILLER PIC X(18) VA "               RANGE".                    000,0000680
               03  FILLER PIC 9(6) CMP VA @0D0000@.                               000,0000716
               03  FILLER PIC X(25) VA "       LOW            HIGH".              000,0000722
               03  FILLER PIC 9(6) CMP VA @0D0000@.                               000,0000772
               03  FILLER PIC X(6) VA SPACES.                                     000,0000778
               03  LOWVALUE PIC ZZZ9.                                       [0009] 000,0000790
               03   FILLER PC XXX VA SPACES.                                       000,0000798
               03  FILLER PC 99 CMP VA @1F@.                                      000,0000804
               03  MIDVALUE PC 999 VA SPACES.                                     000,0000806
               03  FILLER PC 99 CMP VA @1E@.                                      000,0000812
               03  FILLER PC XXX VA SPACES.                                       000,0000814
               03 HIVALUE   PC Z(4) VA 1000.                               [0010] 000,0000820
               03  FILLER PC X(36) VA SPACES.                                     000,0000828
               03  FILLER PC 99 CMP VA @12@.                                      000,0000900
               03  FILLER PIC 9(12) CMP VA @000000000000@.                        000,0000902
               03  FILLER PIC 99 CMP VA @03@.                                     000,0000914
           01  SCREEN-2.                                                          000,0000916
               03  FILLER PIC 9(14) CMP VA @0C000000000000@.                      000,0000916
               03  FILLER PC X(21) VA "         ************".                   000,0000930
               03  FILLER PIC 9(6) CMP VA @0D0000@.                               000,0000972
               03  FILLER PC X(23) VA "         *           ***".                 000,0000978
               03  FILLER PIC 9(6) CMP VA @0D0000@.                               000,0001024
               03  FILLER PC X(23) VA "         *           * *".                 000,0001030
               03  FILLER PIC 9(6) CMP VA @0D0000@.                               000,0001076
               03  FILLER PC X(23) VA "         *           ***".                 000,0001082
               03  FILLER PIC 9(6) CMP VA @0D0000@.                               000,0001128
               03  FILLER PC X(19) VA "           *       *".                     000,0001134
               03  FILLER PIC 9(6) CMP VA @0D0000@.                               000,0001172
               03  FILLER PC X(18) VA "            *******".                      000,0001178
               03  FILLER PIC 9(6) CMP VA @0D0000@.                               000,0001214
               03  FILLER PC X(32) VA SPACES.                                     000,0001220
               03   FILLER PC X(32) VA "   YOU LOSE??? BUY THE COFFEE   ".         000,0001284
               03  FILLER PIC 99 CMP VA @03@.                                     000,0001348
           01  CLEAR-SCREEN.                                                [0011] 000,0001350
               03  FILLER PIC 9(20) CMP VA @0C00000000000000011103@.              000,0001350
           01  DUMMY-LINE.                                                        000,0001370
               03  FILLER PIC 9(14) CMP VA @0C000000000000@.                      000,0001370
               03  FILLER PC 9999 CMP VA ZEROES.                                  000,0001384
               03  FILLER PC X(70) VA "**DUMMY** YOU WERE NOT IN THE RANGE".      000,0001388
               03  FILLER PIC 99 CMP VA @03@.                                     000,0001528
           01  GOOD-BYE.                                                          000,0001530
               03  FILLER PIC 9(14) CMP VA @0C000000000000@.                      000,0001530
               03  FILLER  PIC X(8) VA "GOOD-BYE".                                000,0001544
               03  FILLER PIC 9(4) CMP VA @1103@.                                 000,0001560
           PROCEDURE DIVISION.
           DECLARATIVES.                                                         000,0000000
           ALL-USE  SECTION.                                                     000,0000000
               USE AFTER STANDARD ERROR PROCEDURE ON TDFILE2.                    000,0000000
           LABEL-1.                                                              000,0000000
               MOVE 1 TO SW1.                                                    000,0000000
           ALL-EXIT.                                                             000,0000042
               EXIT.                                                             000,0000042
           END DECLARATIVES.                                                     000,0000042
           BEGIN.                                                                000,0000051
```

```
                    OPEN INPUT-OUTPUT TDFILE2.                                                  000,0000051
                PLAY-GAME.                                                                      000,0000135
                    MOVE ZEROES TO LOW-NO.                                                      000,0000135
                    MOVE 1000 TO HIGH-NO.                                                       000,0000149
                    MOVE TIME TO SECRET-NUMBER.                                                 000,0000179
WARNING...(252) RECEIVING FIELD TRUNCATION
                DDISPLAY.                                                                       000,0000325
                    MOVE LOW-NO TO LOWVALUE.                                                    000,0000325
                    MOVE HIGH-NO TO HIVALUE.                                                    000,0000356
                    WRITE R-RECORD FROM SCREEN-1.                                               000,0000387
                AGAIN.                                                                          000,0000657
                    READ TDFILE2 AT END CLOSE TDFILE2 STOP RUN.                                 000,0000657
                    IF SW1 = 1 MOVE 0 TO SW1 GO TO AGAIN.                                       000,0001123
                    IF INPUT-NO = "BYE" GO TO END-JOB.                                          000,0001235
                    GO TO YOU-LOSE-2, WAIT-FOR-A-WHILE-LONGER, DEPENDING ON                     000,0001289
                    WAIT-SCREEN.                                                                000,0001373
                    MOVE INPUT-NO TO INPUT-NUMBER.                                              000,0001373
                    IF SECRET-NUMBER = INPUT-NUMBER                                             000,0001387
                        THEN GO TO YOU-LOSE.                                                    000,0001387
                    IF INPUT-NUMBER  NOT > LOW-NO                                               000,0001417
                                OR INPUT-NUMBER NOT < HIGH-NO                                   000,0001417
                            GO TO WRITE-DUMMY.                                                  000,0001447
                    IF SECRET-NUMBER < INPUT-NUMBER                                             000,0001493
                            MOVE INPUT-NUMBER TO HIGH-NO                                        000,0001493
                    ELSE MOVE INPUT-NUMBER TO LOW-NO.                                           000,0001523
                WRITE-USUAL.                                                                    000,0001567
                    MOVE LOW-NO TO LOWVALUE.                                                    000,0001567
                    MOVE HIGH-NO     TO HIVALUE.                                                000,0001598
                    WRITE R-RECORD FROM SCREEN-1.                                               000,0001629
                    GO TO AGAIN.                                                                000,0001899
                YOU-LOSE.                                                                       000,0001915
                    WRITE R-RECORD FROM SCREEN-2.                                               000,0001915
                    MOVE 1 TO WAIT-SCREEN.                                                      000,0002209
                    GO TO AGAIN.                                                                000,0002227
                YOU-LOSE-2.                                                                     000,0002243
                    MOVE 0 TO WAIT-SCREEN.                                                      000,0002243
                    WRITE R-RECORD FROM CLEAR-SCREEN.                                           000,0002257
                    GO TO PLAY-GAME.                                                            000,0002527
                WRITE-DUMMY.                                                                    000,0002543
                    WRITE R-RECORD FROM DUMMY-LINE.                                             000,0002543
                    MOVE 2 TO WAIT-SCREEN.                                                      000,0002837
                    GO TO AGAIN.                                                                000,0002855
                WAIT-FOR-A-WHILE-LONGER.                                                        000,0002871
                    MOVE 0 TO WAIT-SCREEN.                                                      000,0002871
                WRITE R-RECORD FROM CLEAR-SCREEN.                                               000,0002885
                    GO TO DDISPLAY.                                                             000,0003155
                END-JOB.                                                                        000,0003171
                    WRITE R-RECORD FROM GOOD-BYE.                                               000,0003171
                    STOP RUN.                                                                   000,0003465
                END-OF-JOB.                                                                     000,0003505
```

```
  C O D E     D I C T I O N A R Y
      BYTE LENGTH   CODEFILE RELATIVE DISK ADR
      ...........   ..........................
  000 00000439      000010
      00000439      TOTAL CODE

  D A T A     D I C T I O N A R Y
      BYTE LENGTH   CODEFILE RELATIVE DISK ADR
      ...........   ..........................
  000 00000998      000003

  S - M A C H I N E     P A R A M E T E R S     ( S C R A T C H P A D )

  LENB=9, SEGB=0, DISPB=11, COPXB=5, COPB=24, D.E.F=354, BDISPB=12
  BASE RELATIVE ADDRESSES:
    DATA-SEGMENT-0=472, COP-TABLE=40, STACK=6981 (BIT LENGTH=1000)

  P R O G R A M     P A R A M E T E R     B L O C K

  FIRST-EXECUTABLE-INSTRUCTION=0,51
  INTERPRETER-NAME=COBOL      /INTERP
  STATIC-CORE=7984 BITS
  DYNAMIC-CORE=0 BITS
  DATA DICTIONARY STARTS AT CODEFILE SEGMENT 2, 1 ENTRY
  CODE DICTIONARY STARTS AT CODEFILE SEGMENT 9, 1 ENTRY

LAST ERROR AT SEQUENCE NUMBER       . 1 WARNING
***** COMPILATION COMPLETE
ELAPSED TIME : 01 MINUTE, 37 SECONDS
PROGRAM REQUIRES 14 DISK SEGMENTS OF 180 BYTES EACH.
MEMORY REQUIREMENTS
  0000439 BYTES = LARGEST CODE SEGMENT
  0000998 BYTES = BASE-TO-LIMIT AREA
  0000296 BYTES = DICTIONARIES AND RUN STRUCTURE
  0000402 BYTES = FILE BUFFERS & FILE INFO AREAS - INCLUDES 129 BYTES (+72 TO 540 IF DISK) FOR EACH FILE
  0002135 BYTES = ESTIMATED MEMORY REQUIRED TO RUN IF ALL FILES OPEN
129 SYMBOLIC RECORDS COMPILED AT 79.740 RECORDS PER MINUTE.
```

cut along dotted line

cut along dotted line

BURROUGHS CORPORATION
DATA PROCESSING PUBLICATIONS
REMARKS FORM

TITLE: B 1700 SYSTEMS
DATA COMMUNICATIONS
Information Manual

FORM: 1089992
DATE: December, 1975

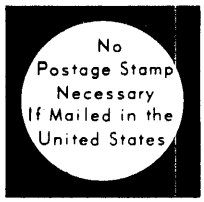
CHECK TYPE OF SUGGESTION:

☐ ADDITION ☐ DELETION ☐ REVISION ☐ ERROR

GENERAL COMMENTS AND/OR SUGGESTIONS FOR IMPROVEMENT OF PUBLICATION:

FROM: NAME _____ DATE _____
TITLE _____
COMPANY _____
ADDRESS _____
_____

FOLD DOWN SECOND FOLD DOWN



BUSINESS REPLY MAIL
First Class Permit No. 1009; El Monte, CA. 91731

Burroughs Corporation
P. O. Box 142
El Monte, CA. 91734

attn: Publications Department
Technical Information Organization, TIO – West

FOLD UP FIRST FOLD UP

Printed in U.S.A. December, 1975 1089992