22294200

CONTROL DATA CORPORATION

CONTROL DATA®
BISYNC COMMUNICATIONS

SOFTWARE REFERENCE MANUAL

REVISION RECORD

REVISION	DESCRIPTION
A	
12/30/75	Manual released.

Publication No.
22294200

© 1975
by Control Data Corporation
Printed in the United States of America

Address comments concerning this manual to:

Control Data Corporation
Publications and Graphics Department
4455 Eastgate Mall
La Jolla, California 92037

or use Comment Sheet in the back of this manual.

Preface

This manual is the reference manual for the Control Data Corporation Bisync Communication Software Package. It explains the mode of operation and gives operating instructions for the software.

Other publications which may be of interest and which support the operation of the software are:

Name	Publication
MSOS 4 Reference Manual	60361500
MSOS 4 Installation Handbook	60361600
CYBERDATA® Operating System Reference Manual	22263100
Binary Synchronous Communication Controller Reference Manual	98862300

The manual is divided into three sections which are:

General description - section 1

Operator interface - section 2

Bisync communications monitor and driver - section 3

Contents

Figures

Tables

SECTION 1

GENERAL DESCRIPTION

General Description

INTRODUCTION

The CDC® Bisync Communication Software Package is designed to simulate the IBM 3780 data communication terminal protocol. It is a modular system, which enables large volumes of batched input data or source programming data to be transmitted at line speed with printed output for remote data transmission applications. It drives the CDC 970-21 Binary Synchronous Communications Controller (BSCC) and executes its control in the CDC 970-2 computer.

The following features are incorporated in the communications package:

- The ability to perform batch transmission
- The ability to operate with either the ASCII code set or the EBCDIC code set
- Two variable-length buffers that alternate in input and in output
- Space compression and expansion
- Data-controlled software horizontal tab control
- Data-controlled vertical forms control
- EBCDIC transparency
- Point-to-point communication facility on leased or private lines

OPERATING CONFIGURATION

The binary synchronous communications controller consists of two printed circuit boards that plug directly into the computer mainframe. The controller occupies one AQ channel slot and one unused slot. An internal cable and an external cable are supplied with the controller to interconnect the two boards and to link them to the local data set modem. The BSCC uses the common RS-232-C interface.

The communications software runs under the MSOS 4 operating system or MSOS-compatible operating system. Its operation is initiated by operator commands given through the operator's console.

Data for transmission can be accepted from a card reader, the operator's console (teletype), or a magnetic tape. Data which is received can be output to a line printer, the operator's console, or a magnetic tape. The operator can change the assignment of devices easily by using the DE (define) command.

The software is installed with the operating system on the disk. It is loaded into memory and placed into execution by operator action via the operator's console. The software uses the operator's console to provide an interface between itself and the operator.

Figure 1-1 illustrates a system configuration of the communication devices.

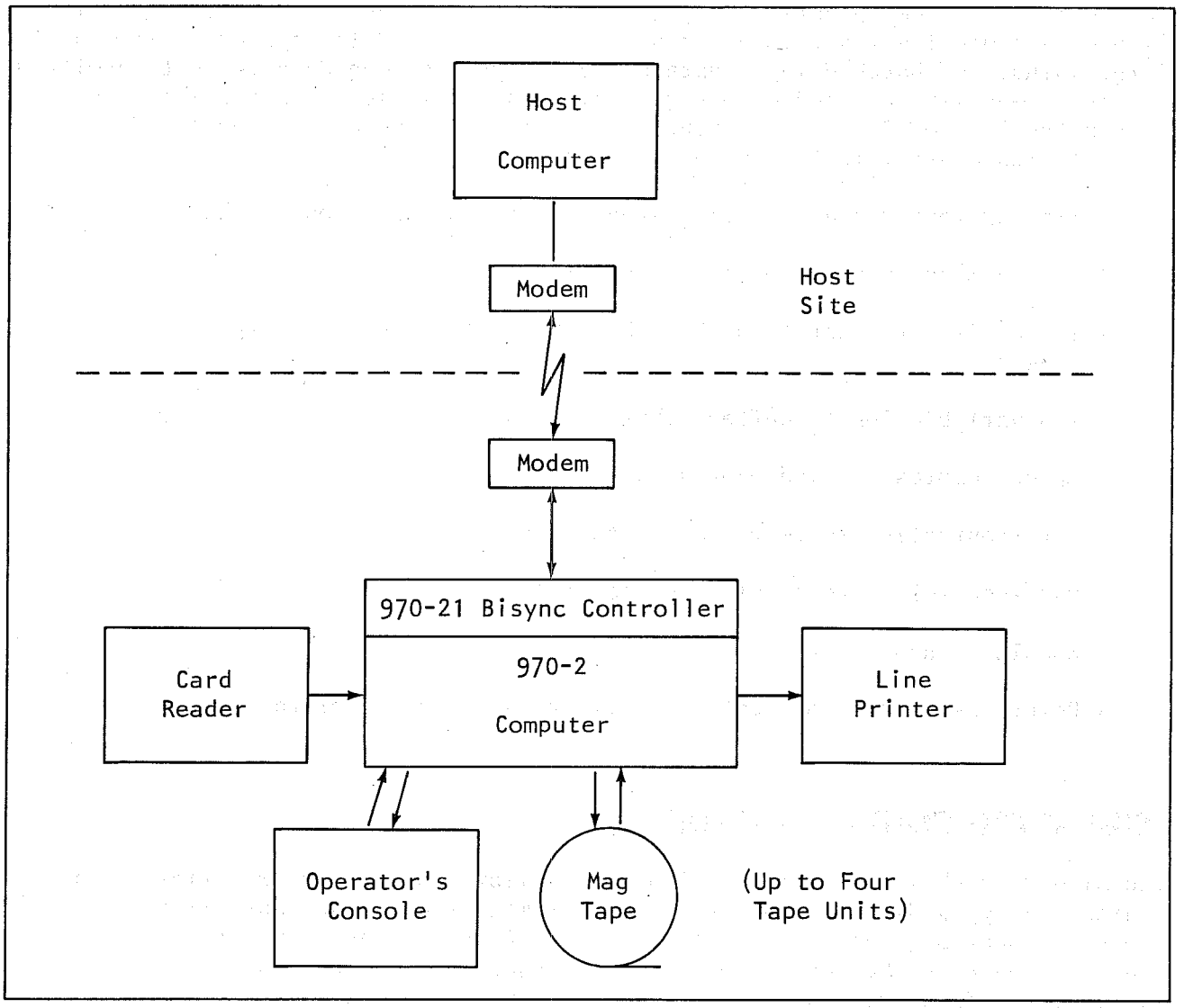


Figure 1-1. Sample System Communications

MODE OF OPERATION

The bisync communication software package is designed so the computer and its peripheral equipment need not be dedicated solely to the task of communications. Depending on the needs of the site, the package may be used as a communications terminal only; run concurrently with other tasks such as key-to-disk, bank entry, etc.; or be called into use only at certain times during a processing day (without interrupting other activities).

Automatic Program Loading

One of the features of the communication software is that after initial use during the processing day, the software can be loaded automatically if the host computer transmits a RVI (reverse interrupt) message. This feature is controlled in the following manner.

One of the operator commands is a line connect command (the operator commands are discussed in section 2). The format of the command is:

LI,{C / D} [CR]

Where:

C - connect the communications unit

D - disconnect the communications unit

The automatic receive function is on when the communication software package is loaded from the disk. If the operator terminates the software without turning off the automatic receive function, the resident portion of the software monitors the line for an RVI signal from the host processor. If an RVI signal occurs, the communications software is loaded from the disk, and the receive function begins.

There could be times when this feature could interfere with another operation of the system. If this occurs, all that needs to be done to discontinue the interference is to have the operator turn this feature off. See section 2 for a complete discussion of the operator commands.

When the automatic receive and program loading feature is used, the list unit assigned for the communication package must be idle; i.e., it cannot be used by another software package. The automatic receive feature assumes that all the units assigned to it are free. If the units are not free, interference results when the units are used by another package.

Likewise, if the communications package is run concurrently with another software package, care must be exercised when equipment is being assigned to each package. The same unit should not be assigned to or used by both packages simultaneously.

Feature and Option Selection

Instead of using physical switches to select features, the mode of operation, and any options, the communication software package uses software switches. These software switches are available to the operator via the operator's console. An operator command (PR) lists the switch settings on the operator's console in a formatted message which is referred to as the communication package parameters. These parameters are discussed in the following paragraphs.

COMMUNICATION PRIORITY

This is item 1 in the parameter list. Two priority choices are offered; primary and secondary. If the primary priority is chosen, the line bid timeout interval is 1 second. If the secondary priority is chosen, the line bit timeout is 3 seconds.

SUPPRESS MESSAGE

Item 2 in the parameter list, the suppress message, offers the operator the option of suppressing the informative messages such as TRANSMIT INITIATED and nnnnn BLOCKS TRANSMITTED. It does not inhibit error messages to the operator.

AUTOMATIC RESTART

This is item 3 in the parameter list. The host processor has the ability to interrupt the data transmission from the terminal to the host by using a reverse interrupt. When a reverse interrupt occurs, the terminal interrupts its transmission of data and automatically goes into a receive mode. If AUTO RESTART is not on, the operator must reinitiate the send function when the data has been received from the host. If AUTO RESTART is on, the send function automatically continues when all the data has been received from the host processor.

TRANSPARENCY

The transparency switch is item 4 in the parameters list. If this software switch is on, any combination of bit settings can be received or transmitted as data; i.e., a binary code which would normally be interpreted as a control character is treated as data. The necessary control characters, which are needed for the communications protocol and which are supplied by the software, are preceded by a DLE character. If an 8-bit combination of data is the same as the code for a DLE character, another DLE character is inserted. When either the terminal or the host detect two DLE characters in sequence, therefore, it is understood that the combination is not a control character, but one 8-bit combination of data.

If the TRANSPCY switch is off, only the character code set of EBCDIC or ASCII is transmitted or received as data. The control codes are detected and accomplish their assigned functions.

MULTIPLE CARD BUFFER

This is item 5 and is called MULTI CARDS in the parameter list. It has an application only if the software switch TRANSPCY is on. If the MULTI CARDS switch and the TRANSPCY switch are both on, six 80-column cards (480 characters) are loaded in one transmission buffer. When the transparent data is received and printed on a line printer, this feature prints three full lines of data plus 72 characters on the fourth line per transmission of a block of data. If the MULTI CARDS switch is off, only 80 characters of data are included in a transmission block.

INHIBIT IRS

This switch is item 6 in the parameter list. If it is turned on, the interchange record separator (IRS) is not inserted at the end of each record on a send function. The IRS is ignored on a receive function when the INHIBIT IRS switch is on. Normally, the IRS provides a new line function when the printer is used in the Receive mode.

If the software switch is off, the IRS is inserted at the end of each record in a send function. Detection of an IRS causes a new line function to be performed when operating in the Receive mode.

SPACE COMPRESSION

This is item 7 in the parameter list. There are three options to this switch: no space compression, compression of trailing spaces only, and compression of all spaces.

If the software switch is set to none, data is transmitted as it is read from the input device and written as it is received on the output device.

If the software switch is set to trailing, spaces are compressed when they are detected at the end of the record (Send mode). When data is being received and the trailing compression option is in effect, the trailing compressed spaces are expanded.

When the full compression option is chosen, spaces are compressed whenever two or more spaces are detected. The space codes may be located anywhere in the record. The software compresses the space codes during the send function and expands them during the receive function.

The same compression and expansion scheme is followed for both compression options. Each group of two or more consecutive space characters (up to and including 63) are replaced by an IGS (EBCDIC) or a GS (ASCII) character followed by a space-count character which defines the number of spaces removed. For a group of 64 or more spaces, an additional IGS or GS character and a space-count character are inserted.

The space-count character is a binary number with the following assignments:

Number of Spaces	*EBCDIC (Bit Position)*	*ASCII (Bit Position)*
	01234567	7654321
2	01000010	1000010
3	01000011	1000011
.	.	.
.	.	.
.	.	.
15	01001111	1001111

Number of Spaces	*EBCDIC (Bit Positions)*	*ASCII (Bit Positions)*
31	01011111	1011111
47	01101111	1101111
63	01111111	1111111

These compression characters are generated during a send function and replace the space characters in the buffer. During the receive operation, the special characters are removed, and the correct number of space characters are inserted into the buffer.

Space codes in an input record that happen to coincide with the last two positions of the send buffer are not compressed. When this happens with the last two characters of an input record, the space codes are removed, and an IRS character is inserted after the last character of compressed space.

NUMBER OF RETRIES

Number of retries is item 8 in the parameter list. It specifies how many times a block of data should be retransmitted or requested when a transmission error occurs. If the block cannot be transmitted error-free in the specified number of retries, the send or receive operation is aborted. Only two different numbers can be specified: 3 or 15.

DEVICE ASSIGNMENT

Items 9, 10, 11, and 12 of the parameter list contain the logical units assigned to the line, input, list, and punch units.* Assignment of the logical unit numbers must be coordinated with device usage by the operating system and the application package which prepares the data for the communication software.

Included with the logical unit designation is the designation of the code set used by the unit. The code can be EBCDIC, ASCII, or BCD. Usually the code used for line transmission is EBCDIC and should be so designated. The input and list devices are usually ASCII devices and, if so, should be designated as ASCII.

LINE BUFFER SIZE

Item 13 of the parameter list designates the transmission block size in number of characters. This size should be coordinated with the host computer up to a maximum of 1,024 characters.

*Use of the punch is not implemented in this release.

INPUT RECORD LENGTH

Item 14 of the parameter specifies the record length of the input device. If 80-column cards are the source of data, the length should be specified as 80. If the input unit is a magnetic tape, the length of the data records should be designated.

LIST UNIT SIZE

Item 15 of the parameter list is used to specify the number of characters per line on the list unit. The number of characters is limited to 132.

PRINTER CONTROL CODES

Several codes contained in the data control the printer. The printer must see these characters as ASCII code. If the data transmitted over the line is in EBCDIC code, the communications software will convert the code to ASCII.

NL — New Line Code

The NL code defines the end of a print line. When the code is detected, the transfer of data to the printer is stopped, and printing is initiated. After printing is completed, the printer advances one line space or causes any escape sequence (carriage or tab function) to be executed.

IRS — Interchange Record Separator Code

The IRS code causes the new line function to occur. It does not, however, terminate a horizontal tab format message. If the INHIBIT IRS software switch is on, the IRS character is ignored.

VT — Vertical Tab Code

The VT code terminates the printer buffer loading and initiates a print cycle. After printing is completed, the printer moves paper until a hole in channel 2 is sensed in the carriage control tape.

FF — Form Feed Code

The FF code terminates the printer buffer loading and initiates a print cycle. After printing is completed, the printer moves paper until a hole in channel 1 is sensed in the carriage control tape.

LF — Line Feed Code

The LF code terminates the printer buffer loading and initiates a print cycle. After printing is completed, the paper is advanced one line.

Carriage Functions

Paper movement can be controlled by using any of several escape sequences. If an escape character (ESC) is followed by one of the characters listed in table 1-1, the paper form will be line-spaced as shown in the carriage operation column. Both ASCII and EBCDIC codes are given. If the line is transmitting EBCDIC, the proper EBCDIC code must be used. ASCII code must be used for ASCII transmission.

Table 1-1. Carriage Functions

USASCII	EBCDIC	Carriage Operation
ESC Q	ESC /	Single line
ESC R	ESC S	Double line
ESC S	ESC T	Double line*
ESC A	ESC A	Skip to channel 1
ESC B	ESC B	Skip to channel 2
ESC C	ESC C	Skip to channel 3
ESC D	ESC D	Skip to channel 4
ESC E	ESC E	Skip to channel 5
ESC F	ESC F	Skip to channel 6
ESC G	ESC G	Skip to channel 7
ESC H	ESC H	Skip to channel 8
ESC I	ESC I	Skip to channel 9
ESC J	ESC J	Skip to channel 10
ESC K	ESC K	Skip to channel 11
ESC L	ESC L	Skip to channel 12
ESC M	ESC M	No line feed

*No triple-line spacing is available.

HT — Horizontal Tab Code

Horizontal tabs for the list unit are provided by the bisync communication software through use of a bit table. This table contains a bit for every character position on the printer. If a bit is set, it means that the corresponding character position on the printer is the beginning of a tab field.

The bit table is preset for tabs in increments of eight beginning at the eighth character position. It may be set or changed by an escape sequence, which must be the first transmission of a new job by the host computer. The following rules must be observed:

- The escape sequence must be the first data received for a job.
- The sequence begins with the ESC character.
- An HT code will set a tab bit for the corresponding position; i.e., an HT code in the tenth character position of the sequence will set a tab bit for the tenth print position.
- IRS code is ignored and does not count as a bit position.
- NL code terminates the sequence.
- All other codes will not set a tab bit (space code is recommended for the nontab character).
- The sequence length must be equal to or smaller than the amount of print positions.
- If the sequence is shorter than the amount of character positions, any previous setting in the table is maintained for those bit positions not included.

TRANSMISSION CONTROL CHARACTERS

The following paragraphs discuss the control characters which are used by the communication software for communication between the host processor and terminal. These control characters are supplied by the software, not by the user.

ENQ Character

The ENQ control character is sent only when the equipment is in the Receive mode and when one of the following conditions occurs:

- The hand-shake procedure is to be initiated
- A response does not match the odd-even block count

- The response is not understood
- No response is received within 3 seconds

ACK0 and ACK1 Characters

The ACK0 and ACK1 control characters are sent as an affirmative acknowledgement to the following items:

- A line bid
- A transmission block received and validated

STX Character

The STX control character signifies the start of text. The character is provided by the software when data is transmitted to the host. When data is received, the STX is included in the data buffers to provide meaning for the list unit driver.

SOH Character

The SOH control character signifies header start. It is not used in transmissions to the host. When it is received with data, it is recognized and treated similarly to the STX character.

NAK Character

The NAK control character is a negative acknowledgement. It is sent to the host for one of the following reasons:

- When a parity error is detected while receiving
- After a premature ending of the test by the transmitter
- When the receive is not enabled

If the NAK character is received by the software, the data block that was transmitted last is transmitted again. The transmission is retried until it has been transmitted correctly or the specified number of retries has been exceeded.

TTD Character

The TTD control character signifies temporary text delay. If a data transmission has been initiated and for some reason (probably a problem in the input device) there is no new data block, the TTD character is transmitted in intervals of 2 seconds. The TTD transmission ceases when a data block is available or the transmission is ended by the operator.

If the TTD is received, a NAK character is transmitted in response.

WACK Character

The WACK control character signifies wait before transmit. It is sent at 2-second intervals when the entire communication package is not loaded and a link has been established in Receive mode. It is transmitted until the package is called in and normal receive operations can begin. A WACK character is also sent if the software is unable to receive because the list unit is not ready.

ETB Character

The ETB control character signifies the end of text block. It is provided by the software when the Send mode is in use and is the last character in the data block transmitted. When it is received, it has the same meaning as it does in the Send mode. It is discarded and causes no action on the list unit.

ETX Character

The ETX control character signifies the end of text. It is provided by the software when in the Send mode is in use. This occurs when the end of file or the end of job is detected. When an ETX is received, it has the same meaning as it does in the Send mode. It is discarded and causes no action on the list unit.

RVI Character

The RVI control character signifies a reverse interrupt. It is received, but it is not sent. When the RVI character is detected, it is treated as a positive acknowledge for the block which was just sent, the send function is suspended, and the receive function is initiated.

EOT Character

The EOT control character signifies the end of transmission. It is sent by the software under any of the following conditions:

- The operator has terminated the transmission with a LI command
- When the number of error retries has been exceeded
- 20 seconds have elapsed in the Receive mode with no activity

CONTROL SEQUENCES

The control sequences and communications protocol are illustrated in the appendixes. Included in the appendixes are the EBCDIC and ASCII character sets.

SECTION 2

OPERATOR INTERFACE

Operator Interface

INTRODUCTION

Section 2 covers operator control of the bisync communications software and messages from the software to the operator. Each command is explained in general terms so the operator may have a feeling for the operations that result from his input.

CONVENTIONS IN ILLUSTRATING STATEMENTS

The conventions used in illustrating the operator statements are as follows:

- An [MI] means the operator is to press the MANUAL INTERRUPT button.
- A [CR] means the operator is to press the RETURN key.
- Braces { } signify that a choice must be made between the options presented.
- Brackets [ ] signify that the portion of the statement enclosed in the marks is optional. If the option is taken, the input must be given exactly as shown. If the option is not taken, the entire portion must be deleted; for example:

  DE,{R / C} [,n1 [,n2]] [CR]

  This statement may take the following forms:

  - DE,R [CR]
  - DE,C [CR]
  - DE,R,3,6 [CR]
  - DE,C,4 [CR]

  The following statement would be rejected because the last comma implies the n2 parameter is to be included:

  DE,C,5 [CR]

- Numeric and alphabetic variables are expressed as nn and aa, respectively. Three n's imply that three digits are expected.

READY (RDY) MESSAGE

The software outputs the message RDY whenever an operation is complete and a new command can be accepted. Only an emergency termination command can be input by the operator before RDY is output.

LOADING AND EXECUTING THE COMMUNICATIONS SOFTWARE

The bisync communication software is loaded and initialized by the following command:

- Operator input

  MI

- System response

  MI

- Operator input

  CM CR

- System response

```
COMMUNICATION PACKAGE n.n IN
RDY
```

TERMINATING THE COMMUNICATIONS SOFTWARE

Terminating the communications software execution can be done in two ways. The first is the normal termination when the transmission of data is successful and the job is complete. When the last operation is complete, the software outputs RDY. The operator then proceeds to key the following:

- Operator input

  EX CR

- System response

```
COMMUNICATION PACKAGE OUT
```

If an operation is in process and for any reason the operator needs to terminate processing, the following sequence should be used:

- Operator input

  MI

- System response

  MI

- Operator input

  CX CR

- System response

  nnnnn RECORDS TRANSFERRED
  nnnnn FILES TRANSFERRED
  RDY

The operation is terminated. The operator can then terminate the execution of the communication software by using the EX command.

PRINT PARAMETERS (PR) COMMAND

This command lists the parameter list of the communication software. The list contains the chosen operating options and assignment of devices to the program. The options and device assignments may be changed by the DE command which is discussed on page 2-5.

The format of the command and an example of the system reply is as follows:

- Operator input

  PR [CR]

- System response

```
COMMUNICATION PACKAGE PARAMETERS
 1 COMM. PRIORITY - PRIMARY
 2 SUPPRESS MESSAGES - NO
 3 AUTO RESTART - NO
 4 TRANSPCY - NO
 5 MULTI CARDS - REGULAR
 6 INHIBIT IRS - NO
 7 SPACE COMPRESSION - TRAILING
 8 NO. OF RETRIES - 3
 9 LINE UNIT - 12 , CODE - EBCDIC
10 INPUT UNIT - 5 , CODE - ASCII
11 LIST UNIT -  9 , CODE - ASCII
12 PUNCH UNIT - 6,  CODE - ASCII
13 BUFFER SIZE - 512 CH
14 INPUT RECORD LENGTH -    80 CH
15 CM LIST UNIT SIZE - 132 CH
```

This system reply to the PR command is an example. Possible parameters for the first eight items of the list are:

1. Primary or secondary
2. Yes or no
3. Yes or no
4. Yes or no
5. Multi or regular
6. Yes or no
7. Full, trailing, or none
8. 3 or 15

Items 9 through 12 of the list show the logical unit assignment for the package. It also shows which code set is used for each device. The unit number of the line unit cannot be changed. The punch unit is not used by the software.

The last three items of the list show the size (in characters) of the line buffer, input record, and list unit record.

DEFINE PARAMETER LIST (DE) COMMAND

The DE command is used to change a part or all of the parameter list. The change may be permanent; i.e., the change is recorded on the disk; or it may be temporary.

The operator format is as follows:

DE, $\left\{\begin{matrix}R\\C\end{matrix}\right\}$[,n1[,n2]] CR

Where:

R - means change the entry for this run only

The next time the bisync software is loaded and executed, the parameter list will be the same as before the change.

C - means to make the changes permanent (constant)

With the changes permanent, when the software is next loaded and executed, the change is still in effect.

NOTE

Items 8 and 9 (number of retries and line unit) can only be changed permanently (C option).

n1 - if specified, is the item number of the list which is to be listed first

If n2 is not specified, only the item identified by n1 will be listed.

n2 - if specified, is the item number of the last parameter to be listed and changed

If neither n1 or n2 is specified, the command allows the whole parameter list to be changed.

The software outputs the list as specified by the use of the n1 and n2 parameters. Each item in the list requires an operator response before the next item is output. If no change is desired for the item, the operator needs only to press RETURN on the operator's console.

If all necessary changes have been made before the entire list is output, the operator may use the EX command to terminate the DE command. The use of the EX command does not terminate the software in this case, but it is used as an exit from the DE command.

Each item in the list is shown in table 2-1. The left column contains the software output in order of occurrence. The right column contains the operator responses and their meanings. Refer to section 1 for a detailed description of each parameter.

Table 2-1. DE Command Operator Responses

Software Message	Operator Response
1 COMMUNICATION PRIORITY (P/S)	If no change, respond with: [CR] If change, respond with: P or S [CR]
2 SUPPRESS MESSAGE (Y/N)	If no change, respond with: [CR] If change, respond with: Y or N [CR]
3 AUTO RESTART (Y/N)	If no change, respond with: [CR] If change, respond with: Y or N [CR]
4 TRANSPCY (Y/N)	If no change, respond with: [CR] If change, respond with: Y or N [CR]
5 MULTI CARDS (Y/N)	If no change, respond with: [CR] If change, respond with: Y or N [CR]
6 INHIBIT IRS (Y/N)	If no change, respond with: [CR] If change, respond with: Y or N [CR]
7 SPACE COMPRESSION (Y/N)	If no change, respond with: [CR] If change, respond with: N or T or F [CR]

Table 2-1. DE Command Operator Responses (cont)

Software Message	Operator Response
8 NUMBER OF RETRIES (3/15)	If no change, respond with: CR If change, respond with: 3 or 15 CR
9 CM LINE UNIT AND CODE	If no change, respond with: CR If change, respond with: input logical unit number, a comma, and either A or E. CR
10 CM INPUT LU AND CODE	If no change, respond with: CR If change, respond with: input logical unit number, a comma, and either A, E, or B CR
11 CM LIST LU AND CODE	If no change, respond with: CR If change, respond with: input logical unit number, a comma, and either A, E, or B CR
12 CM PUNCH LU AND CODE	Not used. No response needed
13 CM BUFFER SIZE	If no change, respond with: CR If change, specify the line buffer size in number of characters CR Length must be less than 1,024 characters and greater than input or output records.

Table 2-1. DE Command Operator Responses (cont)

Software Message	Operator Response
14 CM RECORD LENGTH	If no change, respond with: CR If change, specify the input record length in number of characters CR Length must be 16 characters or more but less than buffer size.
15 CM LIST UNIT SIZE	If no change, respond with: CR If change, specify the list unit line size in number of characters CR Length must be 132 characters or less.

Key

P - primary	N - none	A - ASCII
S - secondary	T - trailing	E - EBCDIC
Y - yes	F - full	B - BCD
N - no		

MOTION (MO) COMMAND

The MO command is used to position a magnetic tape used for input or output. The format is as follows:

$$MO,\left\{\begin{matrix}AF\\AR\\BF\\BR\\DN\\EF\\RW\\UL\end{matrix}\right\},\ \{LU\}\ [,n\ ]\ \boxed{CR}$$

Where:

AF - advance the tape n files

AR - advance the tape n files

BF - backspace the tape n files

BR - backspace the tape n records

DN - set the tape density to n (556, 800, or 1,600 bpi)

EF - write end of file n times

RW - rewind the tape to the loadpoint

UL - rewind and unload the tape

LU - logical unit designation of the tape

n - number of files, records, or density selection

If n is needed but is not specified, the default value is 1. Maximum value of n is 4095.

When the command is complete, the following message is output to the operator:

RDY

TRANSFER (TR) COMMAND

The TR command is the home mode command. It enables the transfer of data between the input unit and the list unit, both of which are defined in the parameter list. No data communication takes place over the transmission line.

The format of the operator command and the system response are as follows:

TR[{F/R}[,n]] CR

Where:

F - transfer files until n number of files has been transferred

R - transfer records until n number of records has been transferred

NOTE

If neither F nor R is specified, data is transferred until the input device is empty or fails and the operator responds CU to an operating system error message. A response of RP will continue the transfer.

If TRANSPCY mode is selected, this will enable binary transfers (no ASCII/EBCDIC code conversion is used), for example:

DE,R,4 - Set yes (TRANSPCY on)

DE,R,10,11 - Set 6,A and 7,A (tape input and output)

DE,R,15 - Set list size same as input record length

n - number of files or records to be transferred

If n is not specified and F or R is specified, the default value is 1.

When the operation is complete, the system responds:

```
nnnnn RECORDS TRANSFERRED
nnnnn FILES TRANSFERRED
RDY
```

LINE (LI) COMMAND

The LI command is used to connect or disconnect the communications unit. The communications unit needs to be connected after the software is loaded in order to start the communication process. When using the PR, DE, or TR commands, the communication unit should be disconnected so the process of the command(s) will not be interrupted by the host computer.

The format of the command is:

LI,{C / D} [CR]

Where:

C - connect the communications unit

D - disconnect the communications unit

Examples

The following example would be used to connect the communications unit after the communications software has been loaded and any necessary changes to the parameter list have been made.

LI,C [CR]

The following example will disconnect the line so the parameter list can be changed or a TR command can be executed without interruption from the host computer.

LI,D [CR]

When the command is complete the following message is output to the operator:

RDY

RECEIVE (RE) COMMAND

The RE command is intended for diagnostic purposes only. Because the software automatically initiates a receive command, there is no need for the operator to initiate the command.

The format of the command is:

RE [CR]

When data is being received from the host computer, the software notifies the operator with the following message (this message is given for the automatic receive also):

RX

NOTE

Each receive (RX) function is terminated by the host computer sending an end of transmission (EOT) message; therefore, several receive functions (RX) may be executed successively, for example:

```
RX
nnnn BLOCKS RECEIVED
nnnn RECORDS TRANSFERRED
RDY
 .
 .
 .
RX
nnnn BLOCKS RECEIVED
nnnn RECORDS TRANSFERRED
RDY
 .
 .
 .
RDY
```

Data is received from the communication line and passed onto the output device. That device is presumed to be a line printer, and the data is deblocked accordingly, even if the operator has defined a magnetic tape as the output device.

When the receive function is complete, the software outputs the following optional message to the operator:

```
nnnnn BLOCKS RECEIVED
nnnnn RECORDS TRANSFERRED
RDY
```

The operator may initiate another command or function.

SEND (SE) COMMAND

The SE command is used to initiate the transmission of data from the input unit to the communications line. If the software detects an RVI (reverse interrupt) from the host computer and auto receive is enabled, the send function is suspended while the data is being received from the host. If the AUTO RESTART option has been chosen, the transmission of data to the host resumes after the receive function is complete. If the option was not chosen, the operator must reinitiate the send function by using another SE command.

The format of the command is as follows:

$$SE\left[\ ,\left\{\begin{matrix}F\\R\end{matrix}\right\}\ [,n]\right]\ \boxed{CR}$$

Where:

F - transmit files until n number of files has been transmitted

R - transmit records until n number of records has been transmitted

NOTE

If neither F nor R is specified, data will be transmitted until the input device is empty or fails and the operator responds CU to an operating system error message.

n - number of files or records to be transmitted

If n is not specified and F or R is, the default value is 1.

When the command has been accepted, the software outputs the following message:

```
TX
```

When the operation is complete, the following message is output to the operator:

```
nnnnn BLOCKS TRANSMITTED
RDY
```

The operator may then initiate another operation.

COMMAND ERROR DETECTION

Each operator-initiated command is analyzed for errors before it is placed in execution. The analysis looks at two items; first to see if the command is a legitimate one, and next to see if the parameters are within range.

If the operator makes an error in naming the command (such as AE for SE) and types in the parameters correctly, the following message will be output to the operator:

```
CM INVALID REQUEST
```

The operator must retype the command and its parameters so the function can be initiated.

If an error was made in the parameters portion of the command, the following message is output to the operator:

```
CM FORMAT INCORRECT
```

This message implies that the command was correct or legitimate, but something is wrong with the rest of the statement. It could be an incorrect value or a misplaced or missing comma. The operator must retype the entire command.

OPERATIONAL ERROR DETECTION

The communications software takes advantage of the operating system features to provide error detection and recovery procedures.

Insufficient Memory

If the operating system does not have sufficient memory for execution of a request, the following message is output to the operator:

```
CM NO CORE AVAILABLE
```

The operator must repeat the command once more or until memory is available.

Device Error

If either the input or the output device fails during the operation of a request, the following message is given to the operator:

```
L,nn  FAILED ec
ACTION
```

Where:

nn - the unit number of the failed device

ec - operating system error codes

See the appropriate operating system manual for the code meaning and action to be taken.

The operator must take appropriate action on the designated device and reply with one of the following:

RP [CR]

This means to repeat the I/O command to the device and resume the operation.

CU [CR]

This indicates the operation is to be terminated.

If the operation is terminated by the operator by the CU [CR] reply, the normal operation summary for the function is output, and then the statement CM I/O ERROR is output. If during an SE (send) command the card reader fails and it is fatal, for example, the messages on the operator's console would look like figure 2-1.

```
SE,F,3 [CR]
TX
L,5  FAILED 8
ACTION
CU [CR]
nnnnn BLOCKS TERMINATED
CM I/O ERROR
RDY
```

Figure 2-1. SE Error Termination

End of Tape

When a magnetic tape is used as the input device, the communications data transmission is ended after an end-of-tape mark has been detected. Tape reading terminates when the end of tape has been sensed.

When a magnetic tape is used as the list unit, the communications software completes the record which is being written when the end of tape is sensed.

When the conditions have been met and completed for the end of tape (either reading or writing), the following message is output to the operator:

```
END OF TAPE
CM I/O ERROR
RDY
```

Line Error

When an irrecoverable error occurs on the line, the operator is not given an option to repeat the attempt or to cancel. Rather, the transmission is stopped immediately, and the following message is given to the operator:

```
LL,nn  FAILED ec
```

Where:

nn - communications number

ec - 00 system time out
     05 internal reject
     06 external reject
     07 illegal request
     10 line not connected
     11 improper request (READ during Send mode, WRITE during Receive mode)
     12 EOT received
     13 EOT transmitted
     17 transparent mode receive error
     19 buffer overflow

After this message is output, the normal end of transmission messages are given.

SECTION 3

BISYNC COMMUNICATIONS MONITOR AND DRIVER

Bisync Communications Monitor and Driver

GENERAL

The Bisync Communications Software package is composed of two separate programs; a driver and monitor. Each program is designed to *stand alone* and thereby offer the modularity and ease of maintenance necessary for today's market.

Figure 3-1 is a block diagram which illustrates the interaction of the two programs and the operating system.

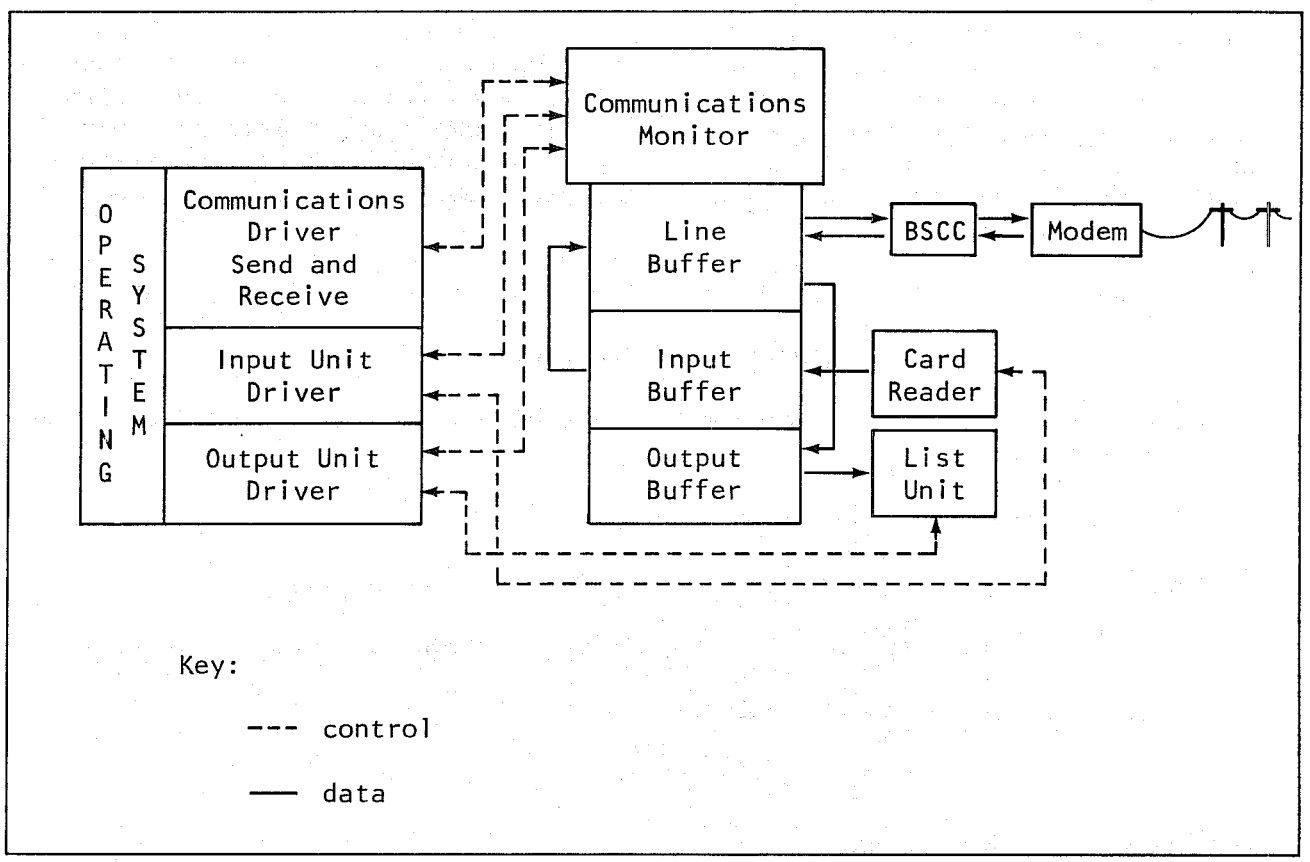


Figure 3-1. Block Diagram

MONITOR GENERAL DESCRIPTION

The Bisync Communications Monitor is a standard software program which is designed to interface with the standard driver and provide a complete bisync communications package. It provides the operator the capability to direct the operation of the package. The monitor provides the linkage and control of the input and output

devices assigned to it. Features of the package which are not related to the actual communications protocol are provided by the monitor; such as, space compression, list unit formatting, EBCDIC/ASCII conversion so data may be read on an ASCII code device and transmitted in EBCDIC, etc.

The monitor provides the external features of the bisync communications software; and therefore, the general description of the monitor is described in section 1. All features described in section 1, except those dealing with the communications protocol, are provided by the monitor.

DRIVER GENERAL DESCRIPTION

The driver is designed to fit into the operating system to drive the CDC binary synchronous communication controller for the implementation of the IBM binary synchronous communications procedure. It is intended to be used with a standard communications monitor to simulate the IBM 3780 Communication Terminal, but since it is a standalone package, it can be used independently of the standard monitor for general binary synchronous communication. It incorporates the following features:

- Standard COS or MSOS operating system read, write, and motion requests
- EBCDIC, ASCII, and transparent EBCDIC code sets
- Linkage to an external program to treat unsolicited data from the communications line
- Reverse interrupt (RVI) processing
- Wait before transmit (WACK) and temporary text delay (TTD)
- Independent time out based on the BSCC's 1-second time-out interrupts
- Comprehensive set of error codes

CONTROL CHARACTERS

The driver provides and detects all control characters used in the communications protocol. It adds the necessary characters such as STX and ETX to the data which is transmitted to the host processor. It does not eliminate such characters from the data received. The monitor must edit all control characters contained in the data.

A list of control characters and their meaning is included in the appendixes. Examples of the communications protocol provided by the driver are also included in the appendixes.

DRIVER CALLING SEQUENCES

The communications driver conforms to standard operating system format and responds to standard READ, WRITE, FWRITE, and MOTION requests. Some nonstandard usage of the request parameters has been necessary to fully implement the communications procedure.

Motion Requests

The motion requests shown in table 3-1 are handled by the driver.

Table 3-1. Communications Driver Motion Requests

Request	Code	Mnemonic
Transmit EOT	02	EOF
Connect controller	03	REW
Disconnect controller	04	UNL
Enable receiver	06	ERC
Disable receiver	07	DRC

Figure 3-2 illustrates the calling sequence for the motion requests.

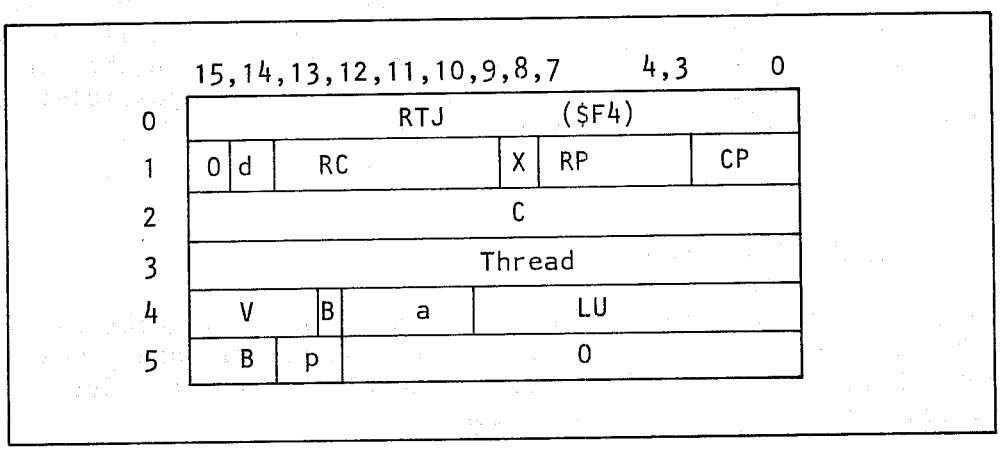


Figure 3-2. Motion Request Calling Sequence

In figure 3-2, the following parameters are used:

d - part 1 indicator

RC - request code

X - absolute/indirect indicator

RP - request priority

CP - completion priority

C - completion address

V - error code setting

a - logical unit absolute/indirect indicator

LU - logical unit

p - motion code

B - either 0 or 1

TRANSMIT EOT (EOF) REQUEST

The EOF motion request is used after data blocks are sent and no more data is to be sent. The EOT sequence; that is, SYN SYN SYN EOT pad, is transmitted to terminate the data link. The controller remains connected and is in the Receive mode after the request is completed. Two error conditions are possible; link was not established, or the controller was not connected.

CONNECT CONTROLLER (REW) COMMAND

The connect command is used to initialize the controller after going on-line. It initiates the time-out interrupts, connects the controller, and places the controller in the Receive mode. The command is complete after the driver receives the first time-out interrupt (1 second).

DISCONNECT CONTROLLER (UNL) COMMAND

The UNL command disconnects and clears the controller.

ENABLE RECEIVE (ERC) COMMAND

The enable receive command sets a bit in an internal table. When the bit is set, a WACK control character is sent as an answer when the station is selected. This allows the automatic receive to be initiated.

DISABLE RECEIVE (DRC) COMMAND

The disable receive command clears a bit in an internal table. When the bit is cleared, a NAK control character is sent as an answer when the station is selected.

Data Transfer Requests

Data transfer across the communications line is accomplished by using the standard READ/WRITE requests of the operating system. The calling sequence for the requests is shown in figure 3-3.

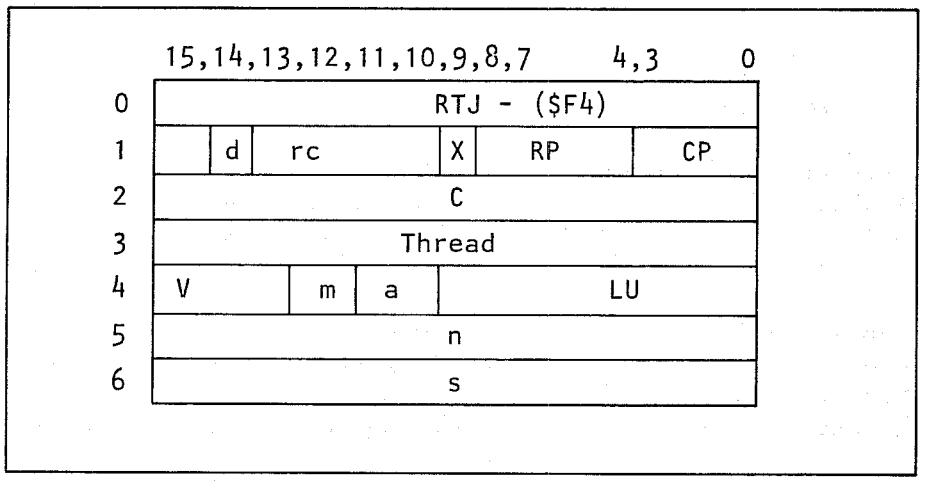


Figure 3-3. Data Transfer Calling Sequence

In figure 3-3, the following parameters are used:

d - part 1 indicator

rc - request code

X - absolute/indirect indicator

RP - request priority

CP - completion priority

C - completion address

V - V-field

Bit 15 - set for error indicator
Bit 14 - set for short read
Bit 13 - set for controller ready when error is set

m - data transfer mode for

WRITE/FWRITE EBCDIC

m - 0 for transparent
m - 1 for nontransparent
m - no meaning during READ or in ASCII

a - absolute/indirect indicator for logical unit

LU - logical unit

n - number of words to transfer

Number of characters is 2n. For Send mode, the number does not include the control characters (STX, ETB or ETX).

s - starting address of the buffer

WRITE AND FWRITE COMMANDS

These commands are used to transmit or to send a block of data. If the data link has not been established, the driver performs the handshake procedure before sending the first data block. If a WRITE or FWRITE call is issued while data is being received, the call is returned with an improper-request error indicated.

Each WRITE or FWRITE call causes one data block to be transmitted. The appropriate control characters are added to the data by the driver. The WRITE and FWRITE calls differ only in that the WRITE request places an ETB character at the end of each data block, and the FWRITE places an ETX character at the end of the block.

Only after the correct positive acknowledge is received with the request be completed with no error. If a NAK or no answer is received, the driver sends the data again until the specified number of retries is reached or until a NAK is received. If the specified number of retries does not result in a NAK, an EOT sequence is sent to terminate the transmission, and control is returned with an EOT error indication.

The number of words to transmit (indicated in word 5 of the calling sequence) does not include the control characters. If the number is zero, the driver performs the hand-shake procedure and then returns control.

READ COMMAND

This command is used to receive data. The FREAD command may be used, but it affects no different operation than the READ command. If the command is issued while transmitting data, the call is returned with an improper-request error indication.

If the data link had not been previously established, the driver goes into Receive mode and waits for an ENQ control character. If the data link was established previously, the driver sends the appropriate positive acknowledge. This acknowledges the previous data block. It then goes into the Receive mode.

The data (including the control characters) which is received is stored in the input buffer. The driver stops receiving data when an ETB, an ETX, or an ENQ character is received in nontransparent mode. It also stops receiving data when the controller goes from transparent mode to nontransparent mode. After the entire block has been received and validated, the request is completed. The positive acknowledge is sent only when the next READ request is initiated, which assures the availability of buffer space for the next block.

When an EOT sequence is received, the request is completed with the EOT which was received being indicated in the error field. If no data is received after a 20-second interval, the EOT sequence is sent, and the request is completed with EOT transmitted in the error field.

APPENDIX A

DATA LINK CONTROL CHARACTERS USED BY THE COMMUNICATIONS DRIVER

Data Link Control Characters Used by the Communications Driver

INTRODUCTION

The characters are listed in table A-1.

Table A-1. Data Link Control Characters

Vocabulary Character	Character Name	Character Meaning		Character Structure	
		Control State	Message-Transfer State	EBCDIC	ASCII
ENQ*	Enquiry	Can you accept transmission (point-to-point)? Respond to your address (multipoint)?	Between blocks: Please respond or repeat last response. Terminating a block: Discard this block and respond with NAK acknowledgment.	ENQ	ENQ
ACK 0*	Even affirmative acknowledgment	I can accept transmission.	Even block received and validated.	DLE (Hex '70')	DLE 0
ACK 1*	Odd affirmative acknowledgment	None	Odd block received and validated.	DLE /	DLE 1
STX	Start of text	Change to message-transfer state and start computing check value (3780 treats SOH as STX, normally sends only STX).	Clear check circuits and start computing new check value (3780 treats SOH as STX, normally sends only STX).	STX	STX
NAK*	Negative acknowledgment	I cannot accept transmission.	Block not validated, can accept retransmission.	NAK	NAK
TTD*	Temporary text delay	Transmission will begin presently. Respond NAK and wait.	Transmission will continue presently. Respond NAK and wait.	STX ENQ	STX ENQ
WACK*	Wait before transmit	Enquire again later, and delay transmission until an affirmative acknowledgment is received.	Enquire again later, and delay further transmission until an affirmative acknowledgment is received. Block received and validated.	DLE,	DLE;
ETB*	End of text block	None	Check value follows, then turnaround and response. Another text block to follow.	ETB	ETB
ETX*	End of text	None	Check value follows, then turnaround and response. This completes the text of a 3780 job but does not release the data link.	ETX	ETX
RVI*	Reverse interrupt	None	Affirmative acknowledgment and signal that processor slave station wants master 3780 to relinquish the line.	DLE @	DLE<
EOT	End of transmission	Drop synchronism and return to control state.	Drop synchronism and return to control state. Not valid in text.	EOT	EOT
PAD	Leading pad Trailing pad	Establish bit synchronism. Turnaround time.	Establish bit synchronism. Turnaround time.	Hex 'AA' All ones (Hex 'FF')	Alternating bits (01010101) All 1's
SYN	Synchronous idle	Establish or assure character synchronism, or time-fill. Discard character.	Establish or assure character synchronism, or time-fill. Discard character.	SYN	SYN

*Line turnaround

Note: These characters are for communications control only. They are added to input data and deleted from output data.

APPENDIX B

CONTROL SEQUENCES USED BY THE COMMUNICATIONS DRIVER

Control Sequences Used by the Communications Driver

INTRODUCTION

The following control sequences are used by the communications driver.

Normal Message Transmission

The transmitting station sends:

	ODD		EVEN		ODD		EVEN	
ENQ	STX (TEXT) ETB	ENQ	STX (TEXT) ETB	ENQ	STX (TEXT) ETB	ENQ	STX (TEXT) ETX	EOT

The receiving station sends:

ACK0	ACK1	ACK0	ACK1	ACK0

Contention for Master Status

The primary station sends:

		ODD		ODD	
ENQ (1-sec bid timeout)	ENQ	STX (TEXT) ETX	EOT	ACK0	ACK1

The secondary station sends:

ENQ (3-sec bid timeout not completed)	ACK0	ACK1	ENQ	STX (TEXT) ETX	EOT

NOTE

Initially, the calling station is conventionally the master station on switched networks; however, because of switched-network discipline, contention should not occur.

Unanswered Line Bid

The transmitting station sends:

E (1-sec or 3-sec	E (1-sec or 3-sec	E (1-sec or 3-sec	E
N bid timeout)	N bid timeout)	N bid timeout)	O
Q	Q	Q	T

NOTE

Primary stations wait 1 second; secondary stations wait 3 seconds. Specify either three tries or 15 tries.

Terminal-to-Station Retransmission Accepted

The transmitting station sends:

	ODD		ODD		EVEN		
E	S	E	S	E	S	E	E
N	T (TEXT-A)	T	T (TEXT-A)	T	T (TEXT-B)	T	O
Q	X	B	X	B	X	X	T

The receiving station sends:

A	N	A	A
C0	A	C1	C0
K	K	K	K

Terminal-to-Station Retransmission Rejected

	ODD		ODD		ODD		ODD		
E	S	E	S	E	S	E	S	E	E
N	T (TEXT-A)	T	T (TEXT-A)	T	T (TEXT-A)	T	T (TEXT-A)	T	O
Q	X	B	X	B	X	B	X	B	T

The receiving station sends:

A	N	N	N	N
C0	A	A	A	A
K	K	K	K	K

Receive-Initiated Transmission Delay

The transmitting station sends:

```
          ODD                 EVEN                                        EVEN
E    S          E    S          E    S          E          E          E  S       E
N    T (TEXT)   T    T (TEXT)   T    T (TEXT)   T          N          N  T(TEXT)T
Q    X          B    X          B    X          B          Q          Q  X       B
```

The receiving station sends:

```
A          A          A          (2-sec W  (2-sec W A
C0         C1         C0         inter- A  inter- A C1
K          K          K          val)   C  val)   C K
                                         K        K
```

NOTE

This allows the receiver to clear buffer block. ENQ is an immediate response to WACK. WACK-ENQ sequences are not counted by driver.

Line bid acknowledgments are always even, the first text block is always odd.

Transmitter-Initiated Transmission Delay

The transmitting station sends:

```
E   S       E   S       E  (2-sec T (2-sec T (2-sec  T S       E  E
N   T(TEXT)T   T(TEXT)T   inter- T inter- T  inter-  T T(TEXT)T  O
Q   X       B  X       B  val)   D val)   D val)    D X       X  T
```

The receiving station sends:

```
A          A          A                              N          A
C0         C1         C0                             A          C1
K          K          K                              K          K
```

NOTE

This allows the transmitter to fill the buffer block. TTD-NAK sequences are not counted by the driver. If the transmitter is unable to continue, it sends an EOT.

STX Format Error, Data Ignored by Slave Station

The transmitting station sends:

	ODD		EVEN			EVEN		ODD		
E	S	E		E	(3-sec	E	S	E	S	E	E
N	T(TEXT-A)	T	(TEXT-B)	T	response	N	T(TEXT-B)	T	T(TEXT-C)	T	O
Q	X	B		B	timeout)	Q	X	B	X	X	T

The receiving station sends:

A	A		A	A	A	
C0	C1	(No re-	C1	C0	C1	
K	K	sponse)	K	K	K	

NOTE

The receiver did not synchronize nor did it receive TEXT-B the first time. The transmitter retransmits.

Response Not Matched to Odd-Even Block Count

The transmitting station sends:

	ODD		EVEN						
E	S	E	S		E	E	E	E	E
N	T (TEXT-A)	T	T	(TEXT-B)	T	N	N	N	O
Q	X	B	X		B	Q	Q	Q	T

The receiving station sends:

A	A	A	A	A	A	
C0	C1	C1	C1	C1	C1	
K	K	K	K	K	K	

NOTE

Specify three tries or 15 tries.

Data Line Aborted on No-Response From Receiver

The transmitting station sends:

E	S		E	(3-sec re-	E	(3-sec	E	(3-sec	E	(3-sec	E
N	T	(TEXT)	T	sponse time-	N	ENQ	N	ENQ	N	ENQ	O
Q	X		B	out)	Q	timeout)	Q	timeout)	Q	timeout)	T

The receiving station sends:

A	(No re-	(No re-	(No re-	(No re-
C0	sponse)	sponse)	sponse)	sponse)
K				

NOTE

Specify three tries or 15 tries.

Data Link Stalemated on No-Continuation by Transmitter

The transmitting station sends:

E	S		E		
N	T	(TEXT)	T		
Q	X		B	(No continuation	(20-sec timeout)

The receiving station sends:

A	A	E
C0	C1	O
K	K	T

APPENDIX C

SUMMARY OF LEGAL COMBINATIONS OF UNIT AND CODE

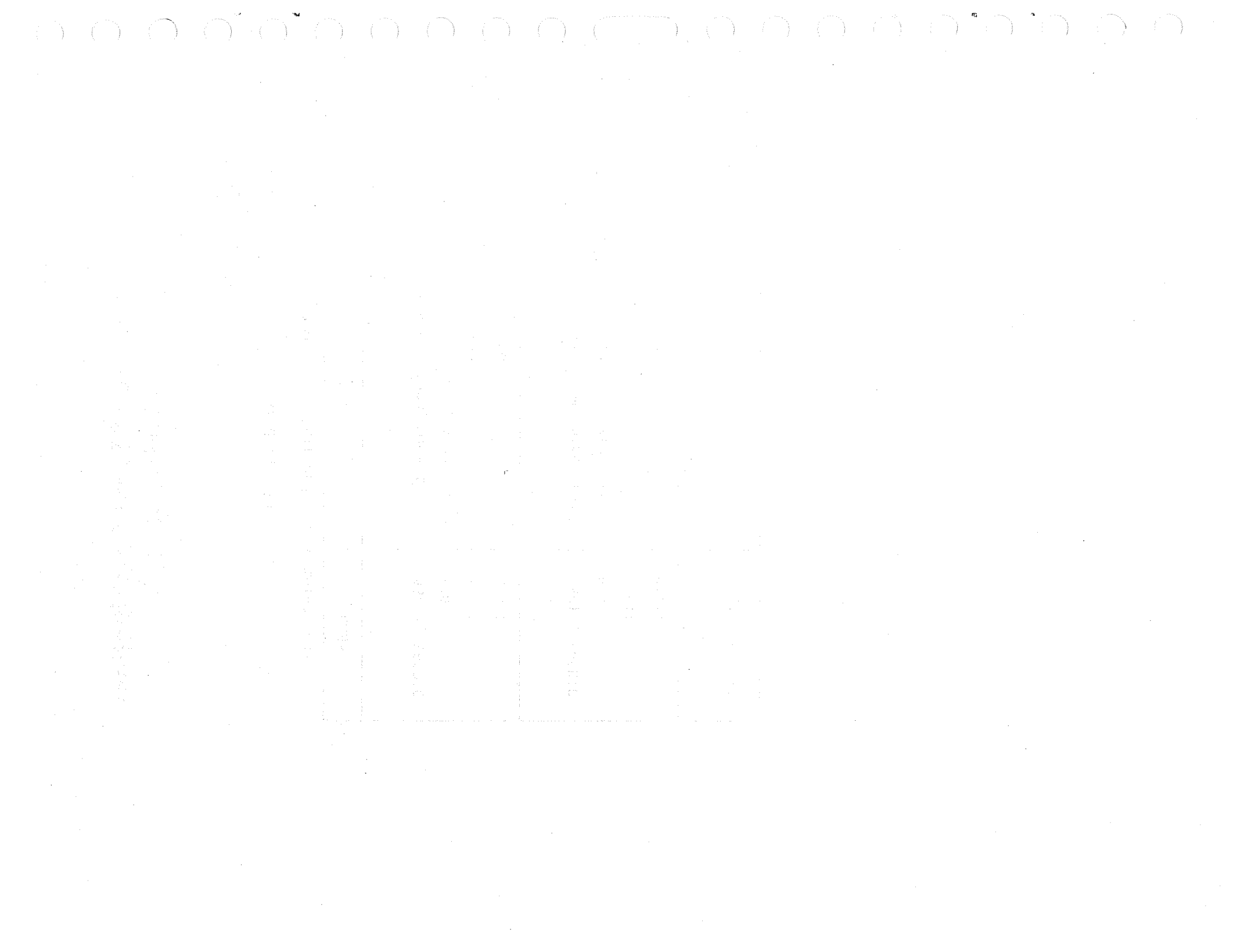
Summary of Legal Combinations of Unit and Code

INTRODUCTION

The legal combinations are listed in table C-1.

Table C-1. Unit and Code Combinations

Unit	Code
Input:	
Magnetic tape, nine track	ASCII, EBCDIC
Magnetic tape, seven track	BCD
Card reader	ASCII
Teletype	ASCII
List:	
Magnetic tape, nine track	ASCII, EBCDIC
Magnetic tape, seven track	BCD
Line printer	ASCII
Teletype	ASCII
LINE:	
BSCC	ASCII, EBCDIC



APPENDIX D

EBCDIC CODE SET

EBCDIC Code Set

INTRODUCTION

Refer to table D-1.

Table D-1. EBCDIC Codes

Char	Card Code	Hex	Char	Card Code	Hex
NUL	12-0-1-8-9	00	NL	11-5-9	15
SOH	12-1-9	01	BS	11-6-9	16
STX	12-2-9	02	IL	11-7-9	17
ETX	12-3-9	03	CAN	11-8-9	18
PF	12-4-9	04	EM	11-1-8-9	19
HT	12-5-9	05	CC	11-2-8-9	1A
LC	12-6-9	06	CU1	11-3-8-9	1B
DEL	12-7-9	07	IFS	11-4-8-9	1C
	12-8-9	08	IGS	11-5-8-9	1D
RLF	12-1-8-9	09	IRS	11-6-8-9	1E
SMM	12-2-8-9	0A	IUS	11-7-8-9	1F
VT	12-3-8-9	0B	DS	11-0-1-8-9	20
FF	12-4-8-9	0C	SOS	0-1-9	21
CR	12-5-8-9	0D	FS	0-2-9	22
SO	12-6-8-9	0E		0-3-9	23
SI	12-7-8-9	0F	BYP	0-4-9	24
DLE	12-11-1-8-9	10	LF	0-5-9	25
DC1	11-1-9	11	ETB(EOB)	0-6-9	26
DC2	11-2-9	12	ESC(PRE)	0-7-9	27
DC3(TM)	11-3-9	13		0-8-9	28
RES	11-4-9	14		0-1-8-9	29

Table D-1. EBCDIC Codes (cont)

Char	Card Code	Hex	Char	Card Code	Hex
SM	0-2-8-9	2A		12-0-3-9	43
CU2	0-3-8-9	2B		12-0-4-9	44
	0-4-8-9	2C		12-0-5-9	45
ENQ	0-5-8-9	2D		12-0-6-9	46
ACK	0-6-8-9	2E		12-0-7-9	47
BEL	0-7-8-9	2F		12-0-8-9	48
	12-11-0-1-8-9	30		12-1-8	49
	1-9	31	¢	12-2-8	4A
SYN	2-9	32	.	12-3-8	4B
	3-9	33	<	12-4-8	4C
PN	4-9	34	(	12-5-8	4D
RS	5-9	35	+	12-6-8	4E
UC	6-9	36	|	12-7-8	4F
EOT	7-9	37	&	12	50
	8-9	38		12-11-1-9	51
	1-8-9	39		12-11-2-9	52
	2-8-9	3A		12-11-3-9	53
CU3	3-8-9	3B		12-11-4-9	54
DC4	4-8-9	3C		12-11-5-9	55
NAK	5-8-9	3D		12-11-6-9	56
	6-8-9	3E		12-11-7-9	57
SUB	7-8-9	3F		12-11-8-9	58
SPACE	NO PUNCHING	40		11-1-8	59
	12-0-1-9	41	!	11-2-8	5A
	12-0-2-9	42	$	11-3-8	5B

Table D-1. EBCDIC Codes (cont)

Char	Card Code	Hex	Char	Card Code	Hex
*	11-4-8	5C	F5	12-11-0-5-9	75
) (Note)	11-5-8	5D	F6	12-11-0-6-9	76
;	11-6-8	5E	F7	12-11-0-7-9	77
¬	11-7-8	5F	F8	12-11-0-8-9	78
-	11	60		1-8	79
/	0-1	61	:	2-8	7A
	11-0-2-9	62	#	3-8	7B
	11-0-3-9	63	@	4-8	7C
	11-0-4-9	64	'	5-8	7D
	11-0-5-9	65	=	6-8	7E
	11-0-6-9	66	"	7-8	7F
	11-0-7-9	67		12-0-1-8	80
	11-0-8-9	68	a	12-0-1	81
	0-1-8	69	b	12-0-2	82
	12-11	6A	c	12-0-3	83
,	0-3-8	6B	d	12-0-4	84
%	0-4-8	6C	e	12-0-5	85
—	0-5-8	6D	f	12-0-6	86
>	0-6-8	6E	g	12-0-7	87
?	0-7-8	6F	h	12-0-8	88
	12-11-0	70	i	12-0-9	89
F1	12-11-0-1-9	71		12-0-2-8	8A
F2	12-11-0-2-9	72		12-0-3-8	8B
F3	12-11-0-3-9	73		12-0-4-8	8C
F4	12-11-0-4-9	74		12-0-5-8	8D

Table D-1. EBCDIC Codes (cont)

Char	Card Code	Hex	Char	Card Code	Hex
	12-0-6-8	8E	x	11-0-7	A7
	12-0-7-8	8F	y	11-0-8	A8
	12-11-1-8	90	z	11-0-9	A9
j	12-11-1	91		11-0-2-8	AA
k	12-11-2	92		11-0-3-8	AB
l	12-11-3	93		11-0-4-8	AC
m	12-11-4	94		11-0-5-8	AD
n	12-11-5	95		11-0-6-8	AE
o	12-11-6	96		11-0-7-8	AF
p	12-11-7	97		12-11-0-1-8	B0
q	12-11-8	98		12-11-0-1	B1
r	12-11-9	99		12-11-0-2	B2
	12-11-2-8	9A		12-11-0-3	B3
	12-11-3-8	9B		12-11-0-4	B4
	12-11-4-8	9C		12-11-0-5	B5
	12-11-5-8	9D		12-11-0-6	B6
	12-11-6-8	9E		12-11-0-7	B7
	12-11-7-8	9F		12-11-0-8	B8
	11-0-1-8	A0		12-11-0-9	B9
	11-0-1	A1		12-11-0-2-8	BA
s	11-0-2	A2		12-11-0-3-8	BB
t	11-0-3	A3		12-11-0-4-8	BC
u	11-0-4	A4		12-11-0-5-8	BD
v	11-0-5	A5		12-11-0-6-8	BE
w	11-0-6	A6		12-11-0-7-8	BF

Table D-1. EBCDIC Codes (cont)

Char	Card Code	Hex	Char	Card Code	Hex
	12-0	C0	R	11-9	D9
A	12-1	C1		12-11-2-8-9	DA
B	12-2	C2		12-11-3-8-9	DB
C	12-3	C3		12-11-4-8-9	DC
D	12-4	C4		12-11-5-8-9	DD
E	12-5	C5		12-11-6-8-9	DE
F	12-6	C6		12-11-7-8-9	DF
G	12-7	C7	\	0-2-8	E0
H	12-8	C8		11-0-1-9	E1
I	12-9	C9	S	0-2	E2
	12-0-2-8-9	CA	T	0-3	E3
	12-0-3-8-9	CB	U	0-4	E4
⌡	12-0-4-8-9	CC	V	0-5	E5
	12-0-5-8-9	CD	W	0-6	E6
⅂	12-0-6-8-9	CE	X	0-7	E7
	12-0-7-8-9	CF	Y	0-8	E8
	11-0	D0	Z	0-9	E9
J	11-1	D1		11-0-2-8-9	EA
K	11-2	D2		11-0-3-8-9	EB
L	11-3	D3	⌐	11-0-4-8-9	EC
M	11-4	D4		11-0-5-8-9	ED
N	11-5	D5		11-0-6-8-9	EE
O	11-6	D6		11-0-7-8-9	EF
P	11-7	D7	0	0	F0
Q	11-8	D8	1	1	F1

Table D-1. EBCDIC Codes (cont)

Char	Card Code	Hex	Char	Card Code	Hex
2	2	F2	9	9	F9
3	3	F3		12-11-0-2-8-9	FA
4	4	F4		12-11-0-3-8-9	FB
5	5	F5		12-11-0-4-8-9	FC
6	6	F6		12-11-0-5-8-9	FD
7	7	F7		12-11-0-6-8-9	FE
8	8	F8		12-11-0-7-8-9	FF

System/360 byte 0 1 2 3 4 5 6 7

First bit transmitted on the communications line

APPENDIX E

ASCII CODE SET

ASCII Code Set

INTRODUCTION

The codes are listed in table E-1.

Table E-1. ASCII Codes

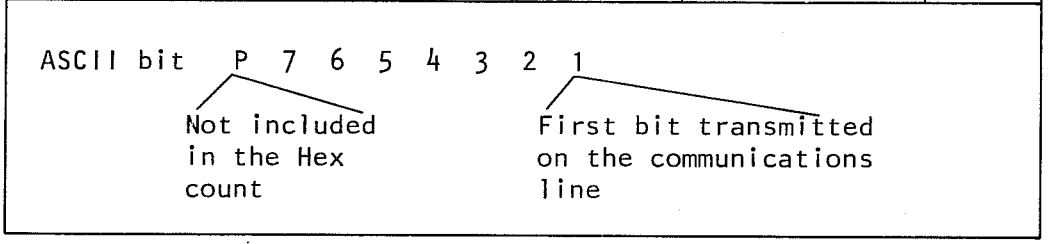
Char	Card Code	Hex	Char	Card Code	Hex
NUL	12-0-9-8-1	00	NAK	9-8-5	15
SOH	12-9-1	01	SYN	9-2	16
STX	12-9-2	02	ETB	0-9-6	17
ETX	12-9-3	03	CAN	11-9-8	18
EOT	9-7	04	EM	11-9-8-1	19
ENQ	0-9-8-5	05	SUB	9-8-7	1A
ACK	0-9-8-6	06	ESC	0-9-7	1B
BEL	0-9-8-7	07	FS	11-9-8-4	1C
BS	11-9-6	08	GS	11-9-8-5	1D
HT	12-9-5	09	RS	11-9-8-6	1E
LF	0-9-5	0A	US	11-9-8-7	1F
VT	12-9-8-3	0B	SPACE	No punches	20
FF	12-9-8-4	0C	|	12-8-7	21
CR	12-9-8-5	0D	"	8-7	22
SO	12-9-8-6	0E	#	8-3	23
SI	12-9-8-7	0F	$	11-8-3	24
DLE	12-11-9-8-1	10	%	0-8-4	25
DC1	11-9-1	11	&	12	26
DC2	11-9-2	12	'	8-5	27
DC3	11-9-3	13	(	12-8-5	28
DC4	4-8-9	14	)	11-8-5	29

Table E-1. ASCII Codes (cont)

Char	Card Code	Hex	Char	Card Code	Hex
*	11-8-4	2A	C	12-3	43
+	12-8-6	2B	D	12-4	44
,	0-8-3	2C	E	12-5	45
-	11	2D	F	12-6	46
.	12-8-3	2E	G	12-7	47
/	0-1	2F	H	12-8	48
0	0	30	I	12-9	49
1	1	31	J	11-1	4A
2	2	32	K	11-2	4B
3	3	33	L	11-3	4C
4	4	34	M	11-4	4D
5	5	35	N	11-5	4E
6	6	36	O	11-6	4F
7	7	37	P	11-7	50
8	8	38	Q	11-8	51
9	9	39	R	11-9	52
:	8-2	3A	S	0-2	53
;	11-8-6	3B	T	0-3	54
<	12-8-4	3C	U	0-4	55
=	8-6	3D	V	0-5	56
>	0-8-6	3E	W	0-6	57
?	0-8-7	3F	X	0-7	58
	8-4	40	Y	0-8	59
A	12-1	41	Z	0-9	5A
B	12-2	42	[	12-8-2	5B

Table E-1. ASCII Codes (cont)

Char	Card Code	Hex	Char	Card Code	Hex
\	0-8-2	5C	n	12-11-5	6E
]	11-8-2	5D	o	12-11-6	6F
¬	11-8-7	5E	p	12-11-7	70
—	0-8-5	5F	q	12-11-8	71
`	8-1	60	r	12-11-9	72
a	12-0-1	61	s	11-0-2	73
b	12-0-2	62	t	11-0-3	74
c	12-0-3	63	u	11-0-4	75
d	12-0-4	64	v	11-0-5	76
e	12-0-5	65	w	11-0-6	77
f	12-0-6	66	x	11-0-7	78
g	12-0-7	67	y	11-0-8	79
h	12-0-8	68	z	11-0-9	7A
i	12-0-9	69	{	12-0	7B
j	12-11-1	6A	¦	12-11	7C
k	12-11-2	6B	}	11-0	7D
l	12-11-3	6C	~	11-0-1	7E
m	12-11-4	6D	DEL	12-9-7	7F



APPENDIX F

BCD CODE SET

BCD Code Set

INTRODUCTION

Refer to table F-1.

Table F-1. BCD Codes

026 Char	026 Punches	029 Char	029 Punches	6-Bit Ext BCD
Space	No punch	Space	No punch	20_8
!	11-8-2	!	12-8-7	52
"	8-7	"	8-7	17
#	12-8-7	#	8-3	77
$	11-8-3	$	11-8-3	53
%	0-8-5	%	0-8-4	35
&	8-2	&	12	00
'	8-4	'	8-5	14
(	0-8-4	(	12-8-5	34
)	12-8-4	)	11-8-5	74
*	11-8-4	*	11-8-4	54
+	12	+	12-8-6	60
,	0-8-3	,	0-8-3	33
-	11	-	11	40
.	12-8-3	.	12-8-3	73
/	0-1	/	0-1	21
0	0	0	0	12
1	1	1	1	01
2	2	2	2	02
3	3	3	3	03

Table F-1. BCD Codes (cont)

026 Char	026 Punches	029 Char	029 Punches	6-Bit Ext BCD
4	4	4	4	04
5	5	5	5	05
6	6	6	6	06
7	7	7	7	07
8	8	8	8	10
9	9	9	9	11
:	8-5	:	8-2	15
;	11-8-6	;	11-8-6	56
<	12-8-6	<	12-8-4	76
=	8-3	=	8-6	13
>	8-6	>	0-8-6	16
?	12-8-2	?	0-8-7	72
@	0-8-7	@	8-4	37_8
A	12-1	A	12-1	61
B	12-2	B	12-2	62
C	12-3	C	12-3	63
D	12-4	D	12-4	64
E	12-5	E	12-5	65
F	12-6	F	12-6	66
G	12-7	G	12-7	67
H	12-8	H	12-8	70
I	12-9	I	12-9	71
J	11-1	J	11-1	41
K	11-2	K	11-2	42

Table F-1. BCD Codes (cont)

026 Char	026 Punches	029 Char	029 Punches	6-Bit Ext BCD
L	11-3	L	11-3	43
M	11-4	M	11-4	44
N	11-5	N	11-5	45
O	11-6	O	11-6	46
P	11-7	P	11-7	47
Q	11-8	Q	11-8	50
R	11-9	R	11-9	51
S	0-2	S	0-2	22
T	0-3	T	0-3	23
U	0-4	U	0-4	24
V	0-5	V	0-5	25
W	0-6	W	0-6	26
X	0-7	X	0-7	27
Y	0-8	Y	0-8	30
Z	0-9	Z	0-9	31
[	12-8-5	[	12-8-2	75
\	0-8-2	\	0-8-2	36
]	11-8-5	]	11-8-2	55
↑	11-8-7	^	11-8-7	57
←	0-8-6	—	0-8-5	32

COMMENT SHEET

MANUAL TITLE CDC® Bisync Communication Software Package

Software Reference Manual

PUBLICATION NO. 22294200 REVISION A

FROM: NAME: _____

BUSINESS ADDRESS: _____

COMMENTS:

This form is not intended to be used as an order blank. Your evaluation of this manual will be welcomed by Control Data Corporation. Any errors, suggested additions or deletions, or general comments may be made below. Please include page number references and fill in publication revision level as shown by the last entry on the Record of Revision page at the front of the manual. Customer engineers are urged to use the TAR.

CUT ALONG LINE

AA3419 REV. 11/69 PRINTED IN U.S.A.

NO POSTAGE STAMP NECESSARY IF MAILED IN U. S. A.

FOLD ON DOTTED LINES AND STAPLE

TAPLE

STAPLE

OLD

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 8241 MINNEAPOLIS, MINN.

POSTAGE WILL BE PAID BY

CONTROL DATA CORPORATION
PUBLICATIONS AND GRAPHICS DIVISION
4455 EASTGATE MALL
LA JOLLA, CALIFORNIA 92037

CUT ALONG LINE

OLD

FOLD