SYSTEM STANDARD
CDC-STD 1.10.020

CONTROL DATA
CORPORATION

MODE 4C DATA COMMUNICATION CONTROL PROCEDURE

JANUARY 1975

MODE 4C

DATA COMMUNICATION CONTROL PROCEDURE

Approved by: R. L. Perkins
R. L. Perkins
Vice-President & Senior Staff Officer
Engineering and Software

TABLE OF CONTENTS

TABLE OF CONTENTS (CONT)

APPENDICES

EXPOSITORY REMARKS

1.0 SCOPE

1.1 Purpose - A data communication network, in general, consists of one or more data communication links, each of which interconnects a control station with one or more terminal stations. Each terminal station may be connected to one or more devices and the communication facility over which data is exchanged with a control station. Figure 1 shows typical communication links.

This standard specifies procedures for the control of information as it flows over a data link to a station and eventually to a specific device. Additionally, it defines the various data elements and message components and describes the ways they may be structured in the various message types. This standard provides a framework within which products with similar features can co-exist on the same data communication link and maintain consistent operation. The terminals supported by this standard have a wide range of capabilities and features. For each system, support will be as defined by the applicable product documentation.

1.2 Applicability - This standard applies to new hardware and/or software products and improvements to existing products where Mode 4 support is intended. Mode 4 is best utilized over voice grade lines in a synchronous mode. Because it is a two-way alternate protocol it becomes inefficient at speeds high enough to preclude the use of voice grade lines.

It should be understood that although this standard goes beyond what traditionally is placed in a standard protocol definition in terms of specific delineation, there are additional system characteristics that must be considered for complete compatibility. Examples are:

a. Signalling rate.
b. Mode of data transmission.
c. Communication network interface.
d. Unique device control codes.
e. Exact timing relations, if appropriate.
f. Link establishment for switched lines.

The establishment of connections of either switched or private lines is not a part of this standard. The control station establishes the logical link by the use of the poll or other output message sequences. The control station is always in control of the network. Information exchanges are initiated by the control station. Satisfactory receipt of a response or a system timeout terminates the information exchange. Clearing of a switched connection may be initiated by the network disconnect (DLE EOT). The accomplishment of this function is the responsibility of the switched network facility.

This standard describes a procedure referred to as Mode 4c which is to be used when controlling the flow of information over a link. It should be realized that past implementations of Mode 4 vary, and one can expect varying degrees of compatibility as highlighted in Paragraphs 1.2.1 through 1.2.3.

1.2.1 Mode 4a - Is a de facto standard pertaining to the exchange of data between a CDC 200 UT and a control station. There are several variations to this standard. The unique feature is the selection of peripherals with predefined codes (E codes) contained at the end of the message text.

1.2.2 Mode 4b - Is a de facto standard with several variations that pertain to the exchange of data between CDC 216-type products and a control station. The distinguishing feature of the protocol used is that devices have a unique address.

1.2.3 Mode 4c - The standard protocol being defined in this document is Mode 4c. It differs from 4a and 4b in that it eliminates major conflicts with ANSI standards and eliminates conflicts between similar product features; it defines areas previously undefined and has expanded capabilities to cover the foreseeable requirements in the near term (2 to 3 years).

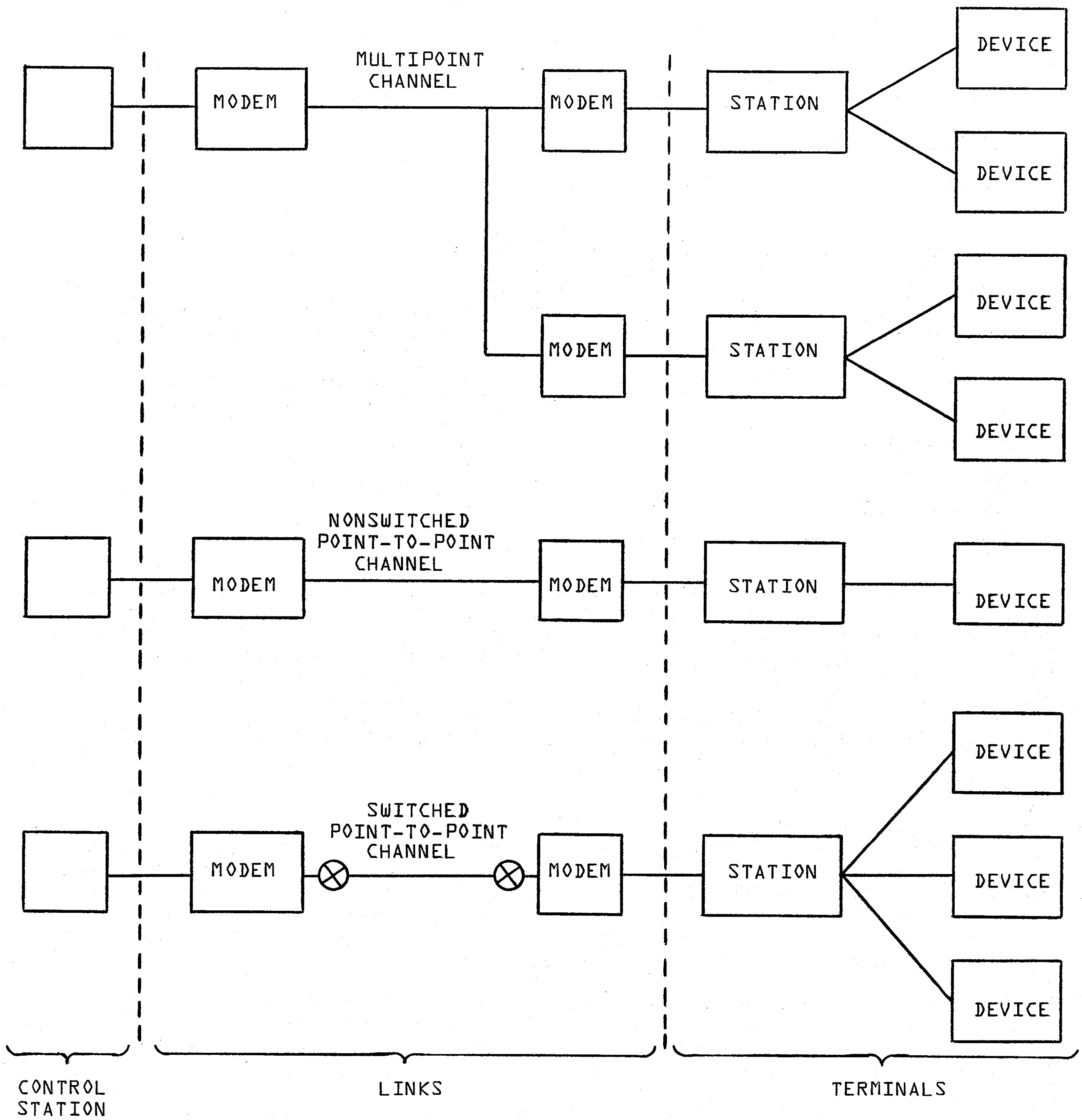


Figure 1 - TYPES OF LINKS SUPPORTED

1.3 Effectivity - This standard shall become effective upon the date of its release and shall remain in effect until it is displaced by a bit-oriented protocol currently under development.

1.4 Authority - The enforcement of this standard is in accordance with CDC Policy 10:04:00. Waivers from this standard must have review and approval via the controlling product document as specified by CDC Policy 10:04:30. The interpreting authority for the standard is the Manager, Systems Standards.

2.0 APPLICABLE DOCUMENTS

2.1 Referenced Documents

CDC-Policy 10:04:00	CDC Technical Standards
CDC-Policy 10:04:30	Deviations or Waivers from CDC Technical Standards
ANSI X3.4-1968	Code for Information Interchange
ANSI X3.12-1970	Vocabulary for Information Processing
ANSI X3.15-1966	Bit Sequencing of ASCII in Serial-by-Bit Data Transmission
ANSI X3.16-1966	Character Structure and Character Parity Sense for Serial-by-Bit Data Communication in ASCII
ANSI X3.28-1971	Procedures for the Use of the Communication Control Characters of ASCII in Specified Data Communication Links

2.2 Related Documents

CDC-STD 1.10.002	Data Communication Control Procedures Compatible with ASCII
CDC-STD 1.10.003	Control Data Subset of ASCII

3.0 GLOSSARY

When interpreting this standard, the reader should refer to the glossary in Appendix A. A number of terms previously used within CDC have been changed to reflect current ANSI usage. By way of examples, what was called a ▿station▿ in the old terminology is now called a ▿device.▿ Similarly, what was called a ▿controller▿ or ▿site▿ in the old terminology is now called a ▿station.▿ The previous ▿data source▿ is now defined as ▿control station.▿ For a more comprehensive list of terms and their definitions, the reader is directed to the American National Standard Vocabulary for Information Processing, X3.12-1970.

4.0 REQUIREMENTS

4.1 Basic Requirements - This standard conforms to the character usage requirements of the American National Standard Code for Information Interchange, X3.4-1968, which is commonly known as ASCII. The data communication procedures described herein can handle all 128 ASCII characters including the 96- and 64-character graphic subsets. Six-bit data is mapped into the 96-character subset. Operation is two-way alternate (half duplex), which means that the terminal can transmit or receive, but not both simultaneously.

4.2 Bytes - Bits are grouped into 8-bit bytes, and all transmissions are multiples of 8 bits. Bytes may contain coded ASCII characters or noncoded binary information. In the first case they are referred to as character bytes and in the latter as binary bytes.

The ASCII character set is shown in the upper chart of Figure 2. Future references in this standard to the ASCII character set are by column/row designation; e.g., 4/13 = symbol M. For readers who are more familiar with octal or hexadecimal codes, the lower chart of Figure 2 is included. ASCII column and row designations appear in decimal as well as hexadecimal. The intersection of a row and column shows the octal equivalent; e.g., column 4, row 13 = octal 115 or, referring to the upper chart, 4/13 = symbol M. ASCII control characters may also be referred to by their alpha symbol; e.g., SOH.

b7 b6 b5 →					0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b4	b3	b2	b1	COLUMN / ROW	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	`	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	¦
1	1	0	1	13	CR	GS	–	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	_	o	DEL

ASCII-CODED CHARACTER SET

ROW \ COL	0	1	2	3	4	5	6	7
0	000	020	040	060	100	120	140	160
1	001	021	041	061	101	121	141	161
2	002	022	042	062	102	122	142	162
3	003	023	043	063	103	123	143	163
4	004	024	044	064	104	124	144	164
5	005	025	045	065	105	125	145	165
6	006	026	046	066	106	126	146	166
7	007	027	047	067	107	127	147	167
8	010	030	050	070	110	130	150	170
9	011	031	051	071	111	131	151	171
10(A)	012	032	052	072	112	132	152	172
11(B)	013	033	053	073	113	133	153	173
12(C)	014	034	054	074	114	134	154	174
13(D)	015	035	055	075	115	135	155	175
14(E)	016	036	056	076	116	136	156	176
15(F)	017	037	057	077	117	137	157	177

COLUMN/ROW TO OCTAL CONVERSION

Figure 2 - ASCII CHARACTER CROSS-REFERENCE TABLES

4.2.1 Bit Sequencing - Bit sequencing for serial transmission of characters is in the order of increasing bit significance with the least significant bit {b1} transmitted first and the most significant bit {b7} transmitted last. The character parity bit follows the most significant bit of the character to which it applies. This procedure conforms to the provisions of the American National Standard for Bit Sequencing of ASCII in Serial-by-Bit Data Transmission, X3.15-1966.

It is noted that the ASCII environment as defined by X3.4-1968 does not specify an eighth bit. Throughout this standard "b8" will be used to represent the eighth or high-order bit of a byte.

4.2.2 Character Parity - All ASCII characters will have parity in bit position eight {b8}. Character parity is such that an odd number of "1" bits are transmitted for each character. This conforms to the American National Standard for Character Structure and Character Parity Sense for Serial-by-Bit Data Communication in ASCII, X3.16-1966, as related to synchronous data.

4.3 Message Format - The basic unit of information is called a message. A message is a single block of data suitable for transmission. The protocol is not cognizant of the use of multiple blocks; however several message blocks can be used to transfer a logical unit of data or to complete a logical operation.

Messages transmitted and received must contain frame control characters in a defined sequence. Any deviations from the established sequence can result in an aborted operation depending upon the point in the message frame at which the deviation occurred. Figure 3 shows the general message frame. A description of message frame components follows.

SYN	SYN	SYN	SYN	SOH	STA	DVA	MTI	Text	ETX	LPC	PAD

SYN =	Synchronization 1/6 {See Paragraphs 4.3.1 and 4.3.8}
SOH =	Start of Heading 0/1 {See Paragraph 4.3.2}
STA =	Station Address 2/0 - 7/14 {See Paragraph 4.3.3}
DVA =	Device Address 2/0 - 7/14 {See Paragraph 4.3.4}
MTI =	Message Type Indicator xxx {See Paragraph 4.3.5}
Text=	Message Text 0/0 - 7/15 {See Paragraph 4.3.6}
ETX =	End of Text 0/3 {See Paragraph 4.3.7}
LPC =	Longitudinal Parity Check xxx {See Paragraph 4.3.8}
PAD =	An Additional Character {See Paragraph 4.3.9}

Figure 3 - GENERAL MESSAGE FRAME

4.3.1 Synchronous Idle (SYN) - The SYN character (1/6) is required to establish synchronization at the beginning of a message (four SYN characters minimum are transmitted, two must be received). They may also be used as a time fill in the message text. The SYN character is required for synchronous transmission. Use of SYN characters is not required for asynchronous transmission; however the system specification for such applications must be consulted to determine whether this convention has been carried forward to maintain consistency between products supporting both synchronous and asynchronous transmission. The SYN character, when used, shall have no other meaning.

When a SYN character is used as time fill during synchronous transmission of message text, it may appear anywhere in the transmission except:

a. between SOH and a message type indicator.
b. between ETX and the following longitudinal parity check (LPC).

When used, the SYN is never considered in the computation of the longitudinal parity check (LPC); see Paragraph 4.3.8.

4.3.2 Start of Heading (SOH) - The SOH (0/1) is used at the beginning of a sequence of ASCII-coded characters which constitute a machine readable address or routing information. SOH is the only character permitted to immediately follow the SYN characters preceding the message. There must be no delay between the SYN characters and the SOH in synchronous transmission.

4.3.3 Station Address (STA) - Station addresses may be designated by 95 ASCII-coded characters (columns 2 through 7 exclusive of the DEL character). However, physical and electrical limitations normally preclude the use of more than 16 addresses per line. See Figure 4 for recommended addressing assignments. Refer to the appropriate system documentation for any restrictions.

4.3.4 Device Address (DVA) - Device addresses may be designated from 94 ASCII-coded characters (columns 2 through 7 exclusive of the DEL character and 6/15). However, they must be assigned in pairs allowing a total of 47 addresses. An address pair consists of an ASCII character from an even-numbered column and the same row character from the next higher odd-numbered column (See Figure 4). Only one device address pair may be assigned to a physical unit. Device addresses are paired to allow message sequencing modulo 2 within the device address character. A detailed description of this procedure, also known as redundant response, is further described under "Output Message Sequence Numbering", Paragraph 4.4.1.2.

4.3.4.1 General Device Address - Of the 47 available address pairs, one pair must be reserved for a general device address for messages addressed to a station or all devices services by a station. The general device address pair is designated as 6/0 and 7/0.

4.3.5 Message Type Indicator (MTI) - A message type indicator must follow the device address in all messages transmitted or received. This character defines the function and characteristics of the message and the type of response expected.

The presently defined message type indicators are from the ASCII character set, columns 0 and 1.

ASCII-defined communication control characters may not be used except as defined in ANSI X3.4-1968. These are:

SOH	ETX	ACK	DLE	SYN
STX	EOT	ENQ	NAK	ETB.

In addition the following control characters are reserved to avoid conflict with common usage of these characters:

NUL SUB ESC DEL.

4.3.5 (Continued)

ASCII communication control characters and assigned message type indicators are shown in Table 1 and Appendix B. The functions, characteristics, format, and expected responses to the specific message types are described in Paragraph 4.4.2 and subparagraphs.

4.3.6 Message Text - In general, the message text can consist of the 96-character graphic subset of ASCII (columns 2 through 7) and selected device control characters from columns 0 and 1. Station control codes (E codes) and special device control codes (escape sequences) are also to be found in the message text. In addition, binary data in 6-bit format is supported. A detailed description of message text codes is found under "Components of Message Text", Paragraph 4.4.3.

4.3.7 End of Text (ETX) - The ETX character (0/3) is a delimiter which defines the end of message text or it may immediately follow the message type indicator in messages which do not include text. Regardless of message type, the ETX character must be included in every message immediately preceding the longitudinal parity check (LPC). Although ETX may not be recognized or received by the end device, it must be passed over the communications link.

4.3.8 Longitudinal Parity Check (LPC) - The LPC code must be included as the last byte of every message. The check tests whether the number of ones in the array of binary digits for each bit position is odd or even. If the number of ones is odd, a zero is inserted for that bit position in the LPC byte; if the number of ones is even, a one is inserted for that bit position in the LPC byte. Thus it ensures that an odd number of ones are transmitted in each bit position. Bit b8 of the LPC byte, however, contains character parity (odd) for the LPC code. It is recommended that a pad character follow the LPC to ensure that all bits of the LPC are transmitted through the modem. It is recognized that this method of calculating LPC does not coincide with the ASCII definition of longitudinal redundancy check.

Longitudinal parity checking for any message begins with and includes the SOH character and extends through the ETX character but excludes all SYN characters regardless of their position in the message frame.

4.3.9 PAD - A character following LPC inserted, when needed, to insure that all bits of the LPC are passed through the transmitting modem and that all received characters are passed to memory.

4.4 Message Frame Detail - Several message frame items require extensive descriptions of their variable properties and their receive/transmit interaction. These are station/device addresses, message types generated as a result of the message type indicator, and message text components. The SYN, SOH, and ETX characters and the LPC code are nonvariables and do not require further explanation.

4.4.1 Message Addressing - A single-character station address and a single-character device address are required message frame components. These two characters occupy the two character positions between the SOH and message type indicator. These characters are required in all message formats with the exception of the Network Disconnect message (see Paragraph 4.4.2.1.10) where they are omitted to be consistent with the American National Standard Procedure for the Use of the Communication Control Characters in Specified Data Links, X3.28-1971.

							ALLOWABLE DEVICE ADDRESS PAIRS		ALLOWABLE DEVICE ADDRESS PAIRS		PREFERRED DEVICE ADDRESS PAIRS	
BITS				b_7 b_6 b_5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4	b_3	b_2	b_1	COL / ROW	0	1	2	3	4	5	6	7
0	0	0	0	0							GENERAL DEVICE ADDRESS PAIR	
0	0	0	1	1								
0	0	1	0	2								
0	0	1	1	3								
0	1	0	0	4								
0	1	0	1	5								
0	1	1	0	6								
0	1	1	1	7								
1	0	0	0	8								
1	0	0	1	9								
1	0	1	0	10(A)								
1	0	1	1	11(B)								
1	1	0	0	12(C)								
1	1	0	1	13(D)								
1	1	1	0	14(E)								
1	1	1	1	15(F)								RE-SERVED
							ALLOWABLE STATION ADDRESSES					PREFERRED STATION ADDRESSES

Figure 4 - ASCII CODE MATRIX SHOWING LOCATION OF STATION AND DEVICE ADDRESSES

Table 1 - MESSAGE TYPE INDICATORS AND CONTROL CHARACTERS

COLUMN/ROW	CONTROL CHARACTER	ASCII MEANING	CDC MEANING
0/0	NUL*	Null	Reserved
0/1	SOH*	Start of Heading	Start of Heading
0/2	STX*	Start of Text	Reserved
0/3	ETX*	End of Text	End of Text
0/4	EOT*	End of Transmission	Disconnect Message
0/5	ENQ*	Enquiry	Poll Message
0/6	ACK*	Acknowledge	Acknowledge Message
0/7	BEL	Bell	Alert Message
0/8	BS	Backspace	Backspace Device Control
0/9	HT	Horizontal Tab	Horizontal Tab Device Control
0/10	LF	New Line	New Line Device Control
0/11	VT	Vertical Tab	Vertical Tab Device Control
0/12	FF	Form Feed	Reset Write Message/Form Feed Device Control
0/13	CR	Carriage Return	Carriage Return Device Control
0/14	SO	Shift Out	Unassigned
0/15	SI	Shift In	Unassigned
1/0	DLE*	Data Link Escape	Data Link Escape
1/1	DC1	Device Control 1	Write Message
1/2	DC2	Device Control 2	Clear Write Message
1/3	DC3	Device Control 3	Read Message
1/4	DC4	Device Control 4	Diagnostic Write Message
1/5	NAK*	Negative Acknowledge	Error Message
1/6	SYN*	Synchronous Idle	Synchronous Idle
1/7	ETB*	End of Transmission Block	Reserved
1/8	CAN	Cancel	Reject Message
1/9	EM	End of Medium	Configuration Request Message
1/10	SUB*	Substitute	Reserved
1/11	ESC*	Escape	Escape
1/12	FS	File Separator	Status Request Message
1/13	GS	Group Separator	Initialize Command/Initialize Request
1/14	RS	Record Separator	Unassigned
1/15	US	Unit Separator	Unassigned
7/15	DEL*	Delete	Reserved

*Reserved for ASCII-defined function. The DEL character, in the strict sense, is not a control character

4.4.1.1 Device Address Pairs - The device address is defined as one of a pair of numbers selected from adjacent columns (2 and 3, 4 and 5, or 6 and 7) in the ASCII character set. This is to allow for message sequence numbering (modulo 2) to detect lost data and avoid message duplication. The method is described following; however the reader is cautioned that he should consult the applicable systems documentation to ascertain whether this feature is supported.

4.4.1.2 Output Message Sequence Numbering - Device address pairs are elements of the same row within adjacent columns of the ASCII character set. Their binary value differs only in the contents of bit b5. This addressing technique allows the control station to sequentially number (modulo 2) successive write messages addressed to the same station. To count modulo 2 in this case, add one without carry to bit b5 in the device address word or simply alternate the device address from one column to another. Note that only one sequence number is maintained by each station regardless of the number of devices connected to the station.

The terminal station will store the received sequence bit of the device address character of a write message only if the message was correctly received and acknowledged. The acknowledge message contains the bit just stored. Under any other conditions, the sequence bit will not be stored. This is effective only for the four write messages. In the event an acknowledge or error response is lost, the control station can issue an alert, device poll, status request, or configuration request to obtain a response with the stored device address sequence bit of the last correctly received write. Thus, the control station, by alternately sending ones and zeros in bit b5 of the device address, can determine whether a write message was correctly received and processed by the terminal. If the sequence bit of the reply is the same as the sequence bit of the last write message, no action is necessary. If the sequence bits are different the last write message must be retransmitted.

4.4.2 Message Types - Message type indicators are used to define the message content, transmission direction, and expected response. This standard describes a two-way alternate message exchange procedure between the control station and terminals (subordinate stations). The control station polls the terminals to solicit input messages and selects terminals when and as it has output for them. The terminals, therefore, to avoid contention with the control station and each other, never transmit until they have received a message from the control station. Terminals do not communicate directly with one another; their messages are routed to and from the control station only.

To maintain an orderly and logical flow of information with defined error detection and recovery procedures, the possible responses to any message type must be predictable. Figure 5 shows message types and possible responses. The message types and conditions causing the various responses are discussed in following paragraphs. Error conditions are discussed in Appendix B. Device responses to certain message types and data codes will be discussed as generally applicable. The system/device reference manuals should be consulted for specific details.

4.4.2.1 Message Types - Control Station to Terminal

1. Poll (ENQ - 0/5) Message - The poll message requests the station or a device to transmit a read message if the terminal or device has data ready to be sent to the control station. The poll message may be a station poll (device address is 6/0 or 7/0), or it may be addressed to a specific device.

   When a poll message is addressed to the station (station poll), the station checks its devices for active read requests. If one or more read request(s) is active, the station selects the device for which it will

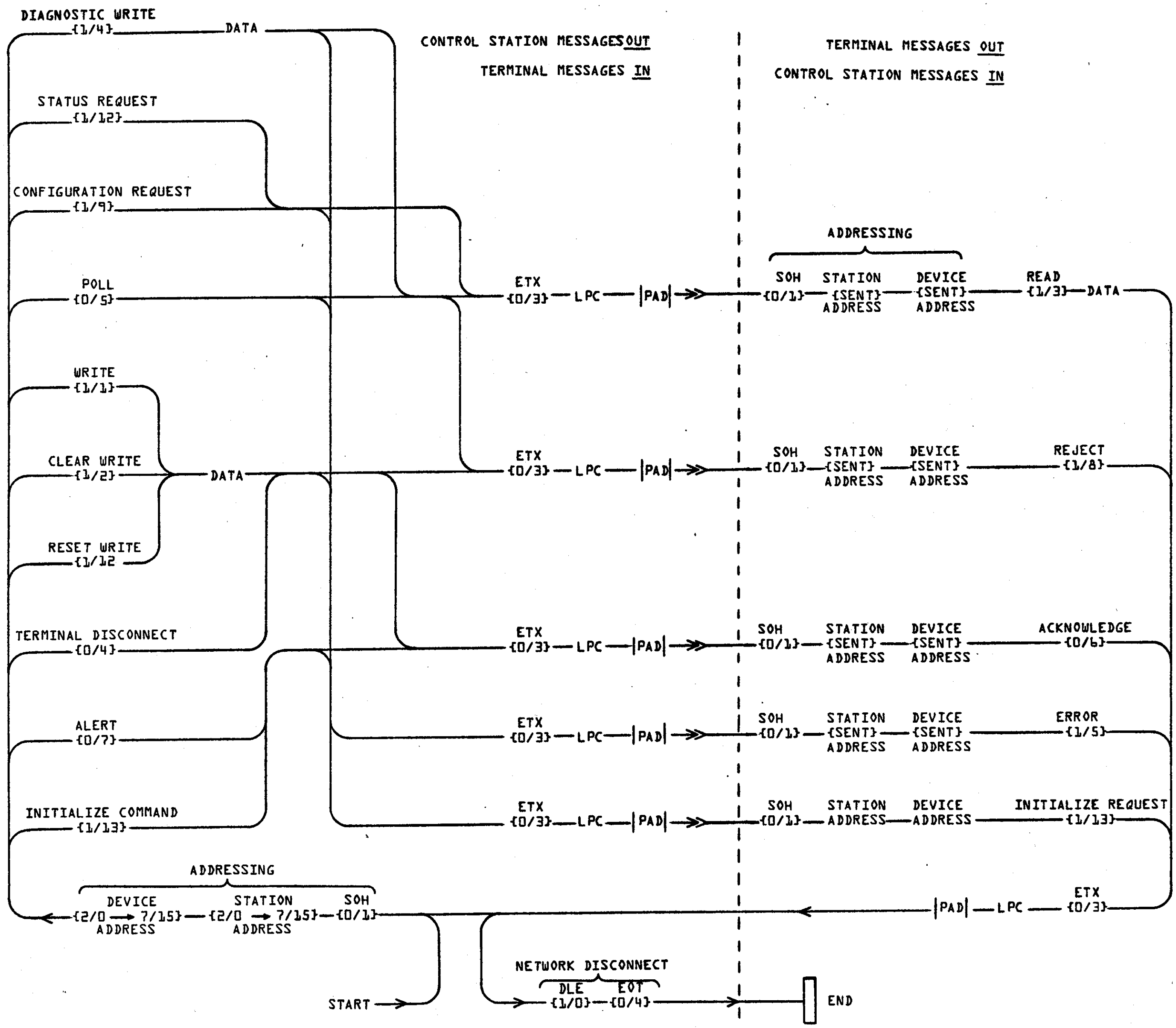


Figure 5 - MESSAGE TYPES AND RESPONSES

4.4.2.1 (Continued)

respond according to its predetermined priority. The normal response is a read message containing the address of, and data from, the device selected which had an active read request. If no devices had an active read request, the station's response to the station poll is a reject message.

A device poll is similar to a station poll except the station responds only for the device addressed in the device poll. If the device has an active read request, the response will be a read message containing the address of, and data from, that device. If the device does not have an active read request, the response is a reject message.

Two other responses to a poll message are possible; see initialize request and error message. The format of the poll message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, ENQ, ETX, LPC, PAD.

2. Alert (BEL - 0/7) Message - The alert message is a means of notifying a specific remote device that the control station has a message for that device. When a device has common usage by the operator and the control station, the alert is used to request assignment and reservation of that device by the control station without interrupting the operation in process.

   The normal response to an alert is an acknowledge message by the terminal that received the alert. The control station may then poll or use an alternate method to determine if and when the request is honored by the device. See Paragraph 4.5.2 for details on the alert sequence.

   To prevent lost or garbled data which could result from unsolicited write messages to specific devices, any type of write message may be preceded by a properly executed alert sequence. It is the user's responsibility to carefully consider the consequences of sending unsolicited write messages. The format of the alert message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, BEL, ETX, LPC, PAD.

3. Write (DC1 - 1/1) Message - The write message is used to transfer data from the control station to a terminal. The write message text may contain any characters that are appropriate to the station and device addressed: graphics, station control codes (E codes), device control codes, coded binary data, and compressed data. The specific function of the write message relative to device output is station and device dependent. (See applicable systems documentation for specific device function.)

   The normal response to a write message is an acknowledge message. The fact that a message has been received and acknowledged by the terminal does not mean it has necessarily been processed correctly by the device. A poll or status request message may be used after receipt of the acknowledge to determine whether the device has successfully completed message processing. If the addressed station or device has no buffer available upon receipt of a write message, it is not processed. The response will be a reject message. (See Paragraph 4.5 and Figure 6 for further explanation of the write message sequence.) The format of the write message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, DC1, TEXT, ESC, E-CODE, ETX, LPC, PAD. Note that the write message need not always contain text.

4.4.2.1 (Continued)

4. Reset Write (FF - 0/12) Message - The reset write message is used to designate that a reset medium function is to be performed by the addressed device and to transfer data from the control station to that device. Other characteristics of the reset write message are the same as those described for a write message in Paragraph 4.4.2.1 no. 3. The format of the Reset Write Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, FF, TEXT, ESC, E-CODE, ETX, LPC, PAD.

5. Clear Write (DC2 - 1/2) Message - The clear write message is used to designate that clear and reset medium functions are to be performed by the addressed device and to transfer data from the control station to that device. Other characteristics of the clear write message are the same as those described for a write message in Paragraph 4.4.2.1 no. 3. The format for the Clear Write Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, DC2, TEXT, ESC, E-CODE, ETX, LPC, PAD.

6. Diagnostic Write (DC4 - 1/4) Message - The diagnostic write message is used to test the terminal and communication link. The message designates that a reset medium function is to be performed and that the data contained in the text of the message is to be stored in the addressed device memory. The terminal's response to the diagnostic write is a read message instead of the normal acknowledge message. The read message is a retransmission of the data just received by the terminal in the diagnostic write message. A Poll Message may be used to cause retransmission of the Read Message sent in response to the Diagnostic Write Message. Three other responses to a diagnostic write message are possible: these are a reject message if the terminal station or addressed device is busy, an error message (See Appendix B) or an Initialize Request (See Paragraph 4.4.2.2 no. 3). The format of the Diagnostic Write Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, DC4, TEXT, ETX, LPC, PAD.

7. Status Request (FS - 1/12) Message - The status request message is generated by the control station to request status information from a terminal or a device. The response to a status request message is a read message containing coded binary data reflecting the status of the terminal or the device.

   If the general device address is specified in the status request message, the response will be status information for the station followed by the status of each device in ascending device address order, one per logical device address. Status data is included for each device address defined for the maximum configuration possible for the terminal whether or not the unit is connected.

   If the status request message contains a specific device address, the response will be status information for that device only. The format of the Status Request Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, FS, ETX, LPC, PAD.

   To obtain single-level status, defined as one byte per station and device, the text portion of the status request message contains no data. To obtain extended status information, i.e., multiple bytes for the station or device addressed, the text portion of the status request message contains ESC, @ (4/0). The format of the Extended Status Request Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, FS, ESC, @, ETX, LPC, PAD.

4.4.2.1 (Continued)

8. Configuration Request (EM - 1/9) Message - The configuration request message allows the control station to determine the configuration of a terminal or a device. The response to a configuration request message is a read message containing coded binary data reflecting the configuration of the terminal or the device. The station configuration byte uniquely identifies the type of terminal.

   If the general device address is specified in the configuration request message, the response will be configuration information for the station followed by data for each device in ascending device address order, one per logical device address. A configuration code is included for each device address defined for the configuration whether or not the unit is connected.

   If the configuration request message contains a specific device address, the response will be configuration data for that device only. The format of the Configuration Request Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, EM, ETX, LPC, PAD.

   To obtain single-level configuration, defined as one byte per station and device, the text portion of the configuration request message contains no data. To obtain extended configuration information, i.e., multiple bytes for the station or device addressed, the text portion of the configuration request message contains ESC, A (4/1). The format of the Extended Configuration Request Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, EM, ESC, A, ETX, LPC, PAD.

9. Terminal Disconnect (EOT - 0/4) Message - The terminal disconnect message indicates that the control station is requesting the terminal logic to disconnect from the link. Upon receipt of the disconnect message the terminal is required to acknowledge the disconnect message and to drop its connection to the link. This applies to switched circuit applications. The control station could then poll to determine if the terminal did drop off the line. A timeout is the expected result of the poll.

   If the terminal is not configured for automatic answer operation (i.e., multipoint private line or leased line), the response to a terminal disconnect message is a reject message. The format of the Terminal Disconnect Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, EOT, ETX, LPC, PAD.

10. Network Disconnect (DLE, EOT - 1/0, 1/4) - The network disconnect is used by the control station to perform a mandatory disconnect of a terminal on a switched circuit. This disconnect function may or may not be part of the terminal control logic. No response is expected and all further communication is suspended until the connection is established again.

    The network disconnect differs from the standard message format in that it does not require the usual SOH, STA, DVA, ETX, or LPC. Network disconnect format is DLE, EOT. For synchronous transmission links this must be preceded by SYN characters.

11. Initialize Command (GS - 1/13) - This message uses a device address of 00 to indicate that the attached information (data) is the station program load or address information. This message type is to be used for transmission of unstructured data of 6 bits, or less, per byte. The data byte must follow the rules for 6-bit binary data. The use of the escape code sequence for identifying the data type is not required.

4.4.2.1 (Continued)

11. (Continued)

If the Initialize Message is not in response to an Initialize Request, it may cause an error response from the terminal the first time it is sent. The format of the Initialize Command is: SYN, SYN, SYN, SYN, SOH, STA, 00, GS, TEXT, ETX, LPC, PAD.

4.4.2.2 Message Types - Terminal to Control Station

1. Acknowledge (ACK - 0/6) Message - The acknowledge message is generated by the terminal as an affirmative response to a correctly received write, reset write, clear write, alert, or terminal disconnect message. It is used only to indicate correct receipt of the control station message. Whether the message has been properly processed can be determined by a subsequent poll or status request. The format of the Acknowledge Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, ACK, ETX, LPC, PAD.

2. Reject (CAN - 1/8) Message - The terminal generates a reject message to indicate rejection of a previously received control station message. The terminal responds with a reject message when: it is polled and no data awaits transfer to the control station; it receives a write, reset write, clear write, or diagnostic write and the station or addressed device is busy; it receives a terminal disconnect message and it is not configured for automatic answering. A reject response to any type of write message means the write was not processed. The format of the Reject Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, CAN, ETX, LPC, PAD.

3. Initialize (GS - 1/13) Request - This message is a request for initial program load. It is generated by the terminal in response to any message when power is applied or the read-write programmable storage requires reloading. The Initialize Request is the result of manual intervention or the receipt by the terminal of a message other than an Initialize Command after initiation of program load. This message may contain a data field. The format of the Initialize Request is: SYN, SYN, SYN, SYN, SOH, STA, 00, GS, ETX, LPC, PAD.

4. Read (DC3 - 1/3) Message - The read message is used to transfer textual data, status, and configuration data from the terminal to the control station. The general format of the Read Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, DC3, TEXT, ETX, LPC, PAD.

   A read message containing textual data may be generated as a result of operator/device interaction in response to a poll or the read message may be the automatic response to a diagnostic write message.

   A read message containing status information is of two distinct types. In response to a status request message a read message containing status data in coded binary form is generated by the terminal as described in Paragraph 4.4.3.5 no. 1 and Appendix B and does not contain an E code. In response to a poll message a read message containing E codes indicating station and device status may be generated by the terminal as described in Paragraph 4.4.3.2 and Appendix B.

   In response to a configuration request message a read message containing configuration data will be generated by the terminal as described in Paragraph 4.4.3.5 no. 2. A Read Message in response to a Configuration Request Message does not contain an E Code.

4.4.2.2 (Continued)

5. Error (NAK - 1/5) Message - The terminal generates an error message to indicate that the control station message just received contained errors after detection of the correct terminal station address. The error message may be returned as a response to all message types except the network disconnect. See Appendix B for a description of conditions under which an error message is returned. The format for the Error Message is: SYN, SYN, SYN, SYN, SOH, STA, DVA, NAK, ETX, LPC, PAD.

4.4.3 Components of Message Text - Message text can be composed of a variety of characters, codes, and code sequences. The parameters of message text defined by this standard include graphics, station control codes (E codes), device control codes (ASCII assigned and escape sequences), coded binary data, and data compression codes. The following paragraphs describe these text components.

4.4.3.1 Graphics - The graphics set used in the message text conforms to the requirements of ANSI X3.4-1968 as shown in Figure 2, columns 2 through 7. It should be noted that the data communications procedures described herein are capable of handling 96 graphic characters, while many existing devices will respond only to a 64-character graphic subset as shown in the upper portion of Figure 2, columns 2 through 5. Refer to the applicable systems documentation. Product documentation should include clarification of problems encountered when 96- and 64-character products are connected to the same link.

The DEL character is a special graphic character. Refer to the applicable device documentation for interpretations of the code by specific terminals.

4.4.3.2 Station Control (E) Codes - E codes are defined as multiple-character station control codes consisting of the ESC character followed by another character from the ASCII character set (See Paragraph 4.4.3.3 no. 2). E codes are unique in their meaning in that they require interpretation by the terminal station but are device dependent in their use. (See example of E code use in Paragraph 4.5.2 and Figure 6). In addition, an E code may have one meaning on output (in a write message) and another meaning on input (in a read message). Interpretation of E codes, however, must be consistent from system to system.

The three defined E codes are:

Code	ASCII
E1	ESC, B (4/2)
E2	ESC, SP (2/0)
E3	ESC, ! (2/1)

Use of these codes may not deviate from the definition prescribed in this standard (See Appendix B). Refer to the applicable systems documentation for E code implementation for a specific device.

4.4.3.3 Device Control Codes - Device control codes are assigned consistent with ANSI X3.4-1968.

1. Single-Character Device Control Codes - The ASCII control code set includes a limited number of device control characters which are primarily oriented to alphanumeric displays and printers.

4.4.3.3 (Continued)

These are:

Code	ASCII	Definition
BS	0/8	Backspace
HT	0/9	Horizontal Tabulation
LF	0/10	Line Feed/New Line*
VT	0/11	Vertical Tabulation
FF	0/12	Form Feed
CR	0/13	Carriage Return

* 1968 ASCII

These characters should be used consistently with the ASCII definition where possible for both input and output. When additional device control codes are required, an escape sequence should be used.

2. Multiple-Character Device Control Codes - The ESC character is defined by ASCII as a means of extending the existing single-character code set to multiple characters for device control. This convention is used to define control functions at two levels, the station (See Paragraph 4.4.3.2) and the device level. Control codes using the ESC convention are called escape sequences. See Appendix B for specific definitions.

   Escape sequences for device control may contain one or more characters after the ESC as required by the device and function to be performed. As an example, the sequence ESC, Z (5/10) is defined as the control sequence to reset the display cursor to the home position. X-Y positioning of the cursor for a display requires a code sequence to identify the required function and additional characters for the coordinate data. Therefore the sequence for X-Y cursor positioning is of the form: ESC, F (4/6), X(x/x), Y (x/x).

3. Restrictions - ASCII control characters defined for communications control are reserved for those functions and are not to be used for device control. The reserved characters are:

Code	ASCII	Definition
SOH	0/1	Start of Heading
STX	0/2	Start of Text
ETX	0/3	End of Text
EOT	0/4	End of Transmission
ENQ	0/5	Enquiry (Poll)
ACK	0/6	Acknowledge
DLE	1/0	Data Link Escape
NAK	1/5	Negative Acknowledge (Error)
SYN	1/6	Synchronous Idle
ETB	1/7	End of Transmission Block

   Device control codes not specifically defined must be treated as time fill and not stored or acted on by the terminal.

4.4.3.4 Binary Data Transmission - Transmission of binary data as defined in this standard provides the facility to transmit information on a communication link. This procedure supports a 6-bit data character for data transmission. Binary transmission is accomplished through the use of an ESC, C {4/3} which indicates that 6-bit binary data follows. Binary data transmission is terminated by use of the ETX. When binary transmission is to the terminal station as opposed to a specific device, the general device address of 6/0 or 7/0 is used. All binary data is packed into the low order bits of the character with bit b7 set to the complement of bit b6.

4.4.3.5 Coded Binary Data

1. Station and Device Status - The status request message provides the control station with a means of obtaining operational status from remote terminal stations and devices. Binary data is transmitted in the received message text with no special control characters. General bit assignments for station and device status are shown in Appendix B; the appendix also lists specific bit assignments for some devices. See the applicable system documentation for specific bit assignments for other devices.

2. Station Configuration - The configuration request message provides the control station with a means of obtaining the operational configuration of the terminal station and devices. The control station sends a configuration request to the terminal station or device which responds with a special read message containing binary data. The binary data represents configuration codes and is not identified by any special control character. A 7-bit configuration code is returned for the device, or the station and each device on that station in logical unit sequence. When a device is not present on a multidevice station, a 4/0 is returned in place of the configuration code.

   Configuration codes assignments are defined in Appendix B. A total of 95 configuration codes are available for use {2/0 to 7/14 of the ASCII character set}. The last 15 configuration codes {7/0 to 7/14} are reserved to fulfill the terminal configuration code requirement for special systems. These codes may be used by any terminal in a special system without requiring reservation of a permanent code in the code assignment table in the appendix.

3. X-Y Position - The ESC, F {4/6} is used for transmitting the X-Y coordinates to a device in the message text. This sequence indicates that the next two characters are the position in the line and the line where the next data character of the message text should be written. The two characters following the ESC, F {4/6} sequence are excess-32 encoded binary characters to avoid conflict with communications control characters.

4.4.3.6 Data Compression - Data compression is defined in Appendix B of this standard for a specific device, the line printer. The described technique or a similar technique may be used by other devices if the code assignments do not conflict with existing device control codes.

4.5 Message Blocking - The control station may have a message for output that exceeds the storage capacity of the addressed device. In this case, the message must be broken into message blocks equal to or less than the physical capacity of the memory of the device.

4.5.1 Block Length - The maximum message length that can be accommodated varies from system to system and is device dependent. It is the user's responsibility to ensure that the message length does not exceed the device capacity. Consult the applicable system documentation to determine maximum message length and associated error conditions.

4.5.2 Multiple Block Processing - Multiple block message processing can be accomplished by reserving the device for the CPU until after the last message block is processed. The operator is locked out during the time that the device is reserved. Devices most likely to be reserved are hard-copy output devices such as lineprinters, character printers, etc. This procedure is shown in Figure 6.

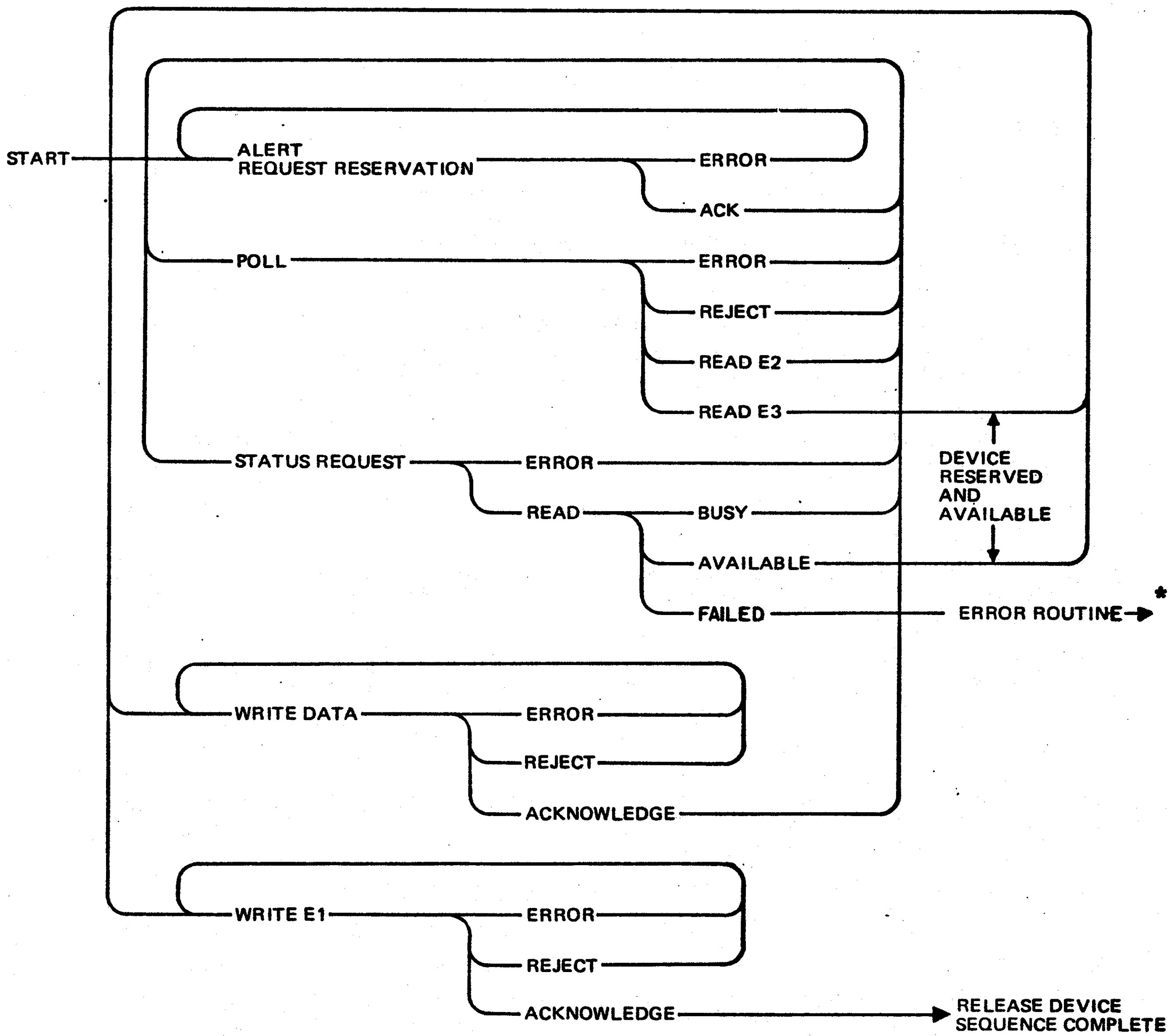
The alert message is sent to request access to a device. The terminal acknowledges proper receipt of the alert. When the device becomes available, it is reserved for us by the control station until further notice. The terminal then sets up a read request to indicate the availability of the output device. After receipt of the acknowledge to the alert the control station may poll the device. A read response to that poll will contain the device availability status. An E1 or E3 response indicates the device is available and reserved for use by the control station. When the response is an E2, the device is not available and a status request may be used to determine the reason. A reject to the poll indicates no read request is active (i.e., device busy). As an alternative, the control station may use the status request to determine when the device is ready to receive data instead of the poll request for status.

When the device is ready to receive data, the control station will send the first (next) message block. When this is received correctly by the terminal it will respond with the acknowledge message and begin processing the data. Since it may be a period of time before the processing has been completed, the procedure in the previous paragraph will be repeated until the device is again available, at which time the cycle is repeated until the logical data transfer is completed. The control station may elect to process any number of messages and/or message blocks during the same device reservation.

When the control station is finished it may release the device by transmitting a write message containing an E1 code. When the terminal acknowledges receipt of the E1 code it will release the prior device reservation. It is recommended that the release request (write E1) be issued after the last data block has been processed to avoid premature release of the device in the event of device failure while processing the last data block.

Device reservation by the control station resolves contention between the control station and terminal device. Resolution of contention for a specific device by application within the control station is the sole responsibility of the control station and is not part of this standard.

Note also that all previously defined message responses and error conditions are still valid during this processing sequence. The device will remain reserved until released by the control station. This should be considered in planning error recovery procedures.



* SYSTEM RECOVERY PROCEDURES SHOULD INCLUDE DEVICE RELEASE.

Figure 6 - MULTIBLOCK PROCESSING WITH DEVICE RESERVATION AND RELEASE

APPENDIX A

GLOSSARY

A1.0 SCOPE

This appendix to the protocol standard contains a glossary as an aid to interpretation of the standard and the related attachments.

A2.0 DEFINITIONS

ACK (Acknowledge) - An ASCII communication control character transmitted by a receiver as an affirmative response to the sender.

ANSI - (American National Standards Institute). In the context of this protocol standard, an American organization concerned with standardization of information communication and processing.

ASCII - (American National Standard Code for Information Interchange, X3.4-1968). The set of 128 control and graphic characters defined as a standard data communications code by the American National Standards Institute.

ASCII Communication Control Characters - The ten communication control characters identified by ANSI X3.4-1968 as being reserved for communication control purposes.

Asynchronous Transmission - A type of transmission in which data characters are not spaced equally in time.

CAN (Cancel) - A communications control character transmitted by a receiver as an indication of rejection of a previously received message. Synonymous with reject.

CDC 200 UT - A remote terminal station including a display with attachable reader and printer. Selection of devices is by predefined codes.

CDC 216 - A remote terminal station with attachable display and character printers. Selection of devices is by unique addresses.

Coded Binary Data - Data bytes having an assigned interpretation of the individual binary bits or a specified method of interpreting the binary code contained in the bits.

Communication Control Procedures - See protocol.

Communication Facility - The medium used to provide the communication paths (e.g. cable, radio).

Communication Network - An integrated system of links, stations, and devices whose operation is controlled according to a set of rules or "protocol".

Control Station - The position on a data link with overall responsibility for the orderly operation of the link. This position has the responsibility to initiate recovery procedures in the event of abnormal conditions on the link. Previously referred to as data source, central processor, higher level processor, etc.

A2.0 {Continued}

Data Compression - A procedure used to increase the effective efficiency of a communication facility. It most commonly replaces a string of repeated character codes with the code that is to be repeated and a count of the number of times it is to be repeated. One or more overhead characters are required for entry to a data compression sequence.

DEL {Delete} - This character is used primarily to "erase" or "obliterate" erroneous or unwanted characters in perforated tape.

Delimiter - A control character used to mark the start or end of a block, record, or message. Delimiters are not required if every character or bit has positional significance in a fixed format.

Device - An equipment connected to the station for the purpose of inputting or outputting information.

Device Poll - See Poll, Device

DLE {Data Link Escape} - An ASCII communication control character which will change the meaning of a limited number of contiguously following characters. It is used exclusively to provide supplementary controls in data communication networks.

DVA {Device Address} - A character used to identify a device with a unique logical address.

E Codes - Three unique two-character codes which are used as station control codes having device dependent interpretation.

ENQ {Enquiry} - An ASCII communication control character used in data communication systems as a request for a response from a remote station. Synonymous with poll.

EOT {End of Transmission} - An ASCII communication control character used to indicate the conclusion of a transmission.

ESC {Escape} - A control character intended to provide code extension {supplementary characters} in general information interchange.

Escape Sequence - A character string of limited length which starts with ESC and is contiguously followed by one or more additional characters. Escape sequences are used in the protocol defined by this standard to supplement the control characters from columns 0 and 1 of the ASCII table.

ETB {End of Transmission Block} - An ASCII communication control character used to indicate the end of a block of data for communication purposes. ETB is used for blocking data where the block structure is not necessarily related to the processing format.

ETX {End of Text} - An ASCII communication control character used to terminate a sequence of characters started with SOH or STX and transmitted as an entity.

A2.0 (Continued)

Excess-32 Notation - A binary-coded decimal notation in which a number is represented by the sum of itself and 32.

Graphic Character - A character from the ASCII code set, columns 2 through 7, normally interpreted as a symbol which can be produced by drawing or printing.

Hexadecimal - A numeration system with a radix of sixteen. In this standard a hexadecimal digit is represented by any of the following characters: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E, or F.

Link - The logical association of two or more stations interconnected by the same data communication circuit including the communication control capability of the interconnected stations. It is frequently referred to as the "link level".

LPC (Longitudinal Parity Check) - An 8-bit character appended to the end of a message immediately following the ETX. The LPC is the complement of a binary add without carries. For a received message without errors, the sum of the LPC and the message characters will be a 7-bit character of all ones. The SYN character(s) are not included in the sum. The sense of the parity bit (b8) is odd like the parity on message characters.

LRC (Longitudinal Redundancy Check) - An 8-bit character which, when added without carries to all message characters between and including SOH and ETX (but exclusing SYN characters), will produce a character of all zeros. Further explanation is available in ANSI standards.

Message - A sequence of characters arranged for the purpose of conveying information from a station to one or more addressed destinations. It contains the information to be conveyed (called text) and may, in addition, contain supplementary information in a heading.

Message Text - That part of a message beginning with the first character following the message type indicator.

Modem - An equipment which provides interface translation between digital terminal and analog communication facilities.

Modulo 2 - A method of preventing message duplication by which a binary 1 is added without carry to bit b5 of the device address in alternate "write" messages.

MTI (Message Type Indicator) - A code which defines the function and characteristics of the message.

Multipoint Link - A data communication link connecting a control station and two or more terminals.

NAK (Negative Acknowledge) - A communications control character transmitted by a receiver as a negative response to the sender. Synonymous with error.

Network Disconnect - The disassociation or release of a switched circuit between two stations.

NUL (Null) - The all-zeros character which may serve to accomplish time fill and media fill.

A2.0 (Continued)

Octal - A numerical representation with a base of eight (8).

Parity - Redundant information used for the purpose of increasing the integrity of data moved from one point to another.

Parity Bit - Bit b8 or the high order bit appended to a 7-bit character code to make the sum of all the binary digits including the parity bit always odd.

Plug Compatible - The ability to substitute one equipment for another at a connector interface point, without modification of any hardware, firmware, or software while maintaining the same functional capability.

Point-to-Point Link - A data communication link connecting a control station to a terminal.

Poll, Device - A poll message containing a specific station address and a specific device address. It is used to determine if a specific device attached to the station has input ready for the control station.

Poll, Station - A poll message containing a specific station address and a general device address. It is used to determine if any device attached to the station has input ready for the control station.

Polling - A technique used for the purpose of inviting a station to respond with a message.

Protocol - Communication control procedures and protocol are used synonymously. It implies a logical method for the establishment, transfer, and termination of data communications between two, or among more than two, station locations.

Protocol Compatible - By definition refers to the ability that permits a control station and a terminal to intercommunicate although they may be physically different equipment not built by the same manufacturer.

Recovery Procedure - The steps taken to restore normal operation to a data communication link after unusual (abnormal) events have occurred, including transmission errors.

SOH (Start of Heading) - A communication control character used at the beginning of a sequence of characters which constitute a machine-sensible address or routing information. Such a sequence is referred to as the "heading".

STA (Station Address) - A character used to identify a station with a unique logical address.

Station - That portion of the terminal serving as the controller and interface between the communications facilities and the devices. It should be noted that existing equipments make very little distinction between what is referred to in the protocol as link, station, and device level control.

Station Poll - See Poll, Station.

STX (Start of Text) - A communications control character which precedes a sequence of characters that is to be treated as an entity and entirely trasmitted through to the ultimate destination. Such a sequence is referred to as "text". STX is not used in Mode 4c protocol.

A2.0 (Continued)

SUB (Substitute) - A character that may be substituted for a character which is determined to be invalid or in error.

Switched Circuit - A circuit which is connected when there is a requirement to exchange information between the control station and the terminal.

SYN (Synchronous Idle) - A communications control character used by a synchronous transmission system in the absence of any other character to provide a signal from which synchronism may be achieved or retained.

Synchronization - The establishment and maintenance of a relative timing relationship between a transmitter and a receiver.

Terminal - A group or cluster of equipment which includes a station and all devices connected to it. It may or may not include the modem, however, this is immaterial from a protocol standpoint since the modem is transparent to the data.

Transparent Data Transmission - A method of framing binary data for transmission in a manner such that the stations on the link are insensitive to the normal interpretation of the bit patterns used for the purpose of link control.

Two-Way Alternate Transmission - The technique used in which messages may be sent in one direction or the other, but not in both directions simultaneously.

Two-Way Simultaneous Transmission - The technique used in which messages may be sent in both directions simultaneously.

APPENDIX B

CODE ASSIGNMENTS AND ERROR CONDITIONS

B1.0 SCOPE

This appendix to the protocol standard establishes initial code assignments, the method of adding new code assignments, and responsibility for control and maintenance of code assignments. The appendix describes these code assignments for communications control, station control, device control, configuration, and status. It does not attempt to define device control codes at the hardware level but applies only to those codes which are transmitted over the communications network. As an aid to the user, this appendix also includes a description of error response conditions.

B2.0 REQUIREMENTS

Paragraph B2.1 and subparagraphs describe the functions of character codes and methods of adding character code assignments. Supporting material is contained in Figures B-1 through B-8 and Tables B-1 through B-6. Paragraph B2.2 contains a brief description of errors which can occur in messages sent to the terminal and the terminal's reaction to these errors.

B2.1 Character Code Assignments

B2.1.1 Communication Control Characters - Figure B-1.

a. Description - Communication control characters are defined by ANSI X3.4-1968 and are used to control or facilitate transmission of information over communication networks. ANSI-defined communication control characters must be used consistent with that definition. Codes that are assigned by ASCII and additional codes reserved to avoid usage conflicts are shown in Figure B-1 of this document and Paragraph 4.3.5 of the protocol standard.

b. Adding Communication Control Characters - Adding message types should be discouraged to minimize the logic within the control station and the terminal station. Rather than adding message types, use of device control characters to accomplish the same effect should be considered.

   Any control character in columns 0 and 1 not defined by ASCII as a communication control character or otherwise reserved may be used as required. When these characters are exhausted, the graphic characters in columns 2 through 7 may be used.

B2.1.2 Station Control {E} Code Assignments - Table B-1.

a. Description - E codes are station control codes the use of which may be device dependent. The interpretation of an E code in a control station-generated message differs from the interpretation of the same E code contained in a terminal-generated message. The three E codes are defined in Table B-1 and in Paragraph 4.4.3.2 of the protocol standard.

b. Adding E Codes - Addition of new E codes is not allowed.

B2.1.3 Device Control Characters - Figures B-2 through B-8 and Table B-1 and B-2.

a. Description - Device control characters are codes that are used for function control of devices associated with the terminal station. They are transmitted on a communication network and are not necessarily the physical codes used to function the hardware device. Device control characters are assigned consistent with ANSI Standard X3.4-1968.

For convenience in listing and describing device control characters the station control (E) codes are included because the use of a specific E code may be device dependent.

b. Adding Device Control Characters - A new device to be added to the system may be similar to previously defined devices and/or may provide similar functions. Whenever possible, previously defined control characters should be used to perform identical functions in the new device. Additional control characters may be selected to perform new functions not already defined. When additional characters are needed, they may be assigned from the available code set as required. Care should be exercised in this selection to avoid forcing duplication or conflict in future code assignments.

When the available control characters for a device type as defined in Figures B-2 through B-8 and described in Table B-2 are exhausted, the code set for the device can be reviewed to determine if characters are reserved for obsolete equipment which can be again made available with a minimum loss of compatibility. Where this is not possible, a multi-level escape sequence may be used to define new device control codes.

B2.1.4 Configuration Code Assignments - Tables B-3 and B-3A.

a. Definition - Configuration codes are used by the control station as a means of identifying the terminal station and devices and any configuration options.

b. Adding Configuration Codes - Single-level configuration code assignments are shown in Tables B-3 and B-3A. Additional codes may be assigned as described in Paragraph 4.4.3.5 no. 2 of the standard and are subject to approval. Through use of the extended configuration sequence, second and subsequent level configuration codes also can be assigned.

B2.1.5 Status Code Bit Assignments - Tables B-4 through B-6.

a. Description - Status codes are used by the control station to determine operating status of a remote terminal. Status assignments are available for the station and its associated devices.

b. Adding Status Bits - Tables B-4 through B-6 show the single-level status bit assignments. Tables B-4 and B-5 show the general assignment of the station and device status bits while Table B-6 shows those specific device status assignments for the device dependent bits (b4 through b6). The format for the second and subsequent level status codes leaves bits b1 through b6 to be assigned as required. Bit b7 is set to the complement of bit b6; b8 is the parity bit.

B2.2 Error Conditions

Error conditions and responses are system and device dependent. The following errors are detected by the terminal which responds as noted. To receive an error message response, transmission of error messages must be enabled by the terminal hardware.

1. Illegal Station Address - No response is given by a terminal unless and until it has recognized its own station address. The control station must perform a timeout to prevent hangups on an illegal station address.

2. Illegal Device Address - If the message is addressed to a nonexistent device, the terminal will abort the receive sequence upon recognition of the ETX character and transmit an error message to the control station.

3. Unrecognized Message Type Indicator - The terminal will abort the receive sequence upon recognition of the ETX character and transmit an error message to the control station.

4. Byte Parity Error - The terminal will store write messages in memory and insert a parity error symbol for the character in error if the message was defined for the device. For all message types it then transmits an error message (except for a diagnostic write sequence), when the ETX character is recognized.

5. LPC Error - The terminal will store write messages in memory and (for all message types) will transmit an error message (except for a diagnostic write sequence).

6. Modem Carrier - If the carrier drops before the terminal detects the ETX character, the terminal will not respond. The control station must perform a time out to prevent hangups.

7. Special Error Conditions - In addition, special error conditions may be detected by the terminal resulting in an error response message, e.g., device data storage exceeded or terminal timeout before receipt of the ETX character. Consult the applicable system documentation.

BITS				b_7 b_6 b_5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4	b_3	b_2	b_1	COL / ROW	0	1	2	3	4	5	6	7
0	0	0	0	0	② NUL	① DLE			③ EXTND STATUS			
0	0	0	1	1	① SOH	WRITE (DC1)			EXTND ③ CONFIG			
0	0	1	0	2	① STX	CLEAR WRITE (DC2)						
0	0	1	1	3	① ETX	READ (DC3)						
0	1	0	0	4	DISCON-NECT ① (EOT)	DIAGT WRITE (DC4)						
0	1	0	1	5	① POLL (ENQ)	① ERROR (NAK)						
0	1	1	0	6	① ACK	① SYN						
0	1	1	1	7	ALERT (BEL)	① ETB						
1	0	0	0	8		REJECT (CAN)						
1	0	0	1	9		CONFIG REQUEST (EM)						
1	0	1	0	10(A)		② SUB						
1	0	1	1	11(B)		② ESC						
1	1	0	0	12(C)	RESET WRITE (FF)	STATUS REQUEST (FS)						
1	1	0	1	13(D)		INITIAL-IZE (GS)						
1	1	1	0	14(E)								
1	1	1	1	15(F)								② DEL

1. ASCII-restricted communication control characters; must be used as defined by ASCII.
2. Reserved for use only as defined by ASCII.
3. Assigned for use with ESC in a status or a configuration request message to obtain extended status or configuration.

Figure B-1 -- ASCII CODE MATRIX SHOWING LOCATION OF MESSAGE TYPE INDICATORS AND ASCII CONTROL CHARACTERS

BITS				b_7 b_6 b_5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4	b_3	b_2	b_1	COL / ROW	0 (1)	1 (1)	2 (2)	3 (2)	4 (2)	5 (2)	6 (2)	7 (2)
0	0	0	0	0				CNTRL KEY		BACK-SPACE (4)		
0	0	0	1	1				FCTN KEY 1	NEW LINE (4)	FWD TAB (4)		
0	0	1	0	2			START INDEX	FCTN KEY 2	E1	LINE SKIP		
0	0	1	1	3			CURSOR DOWN (4)	FCTN KEY 3	PRINT CNTRL	SEL CLEAR		
0	1	0	0	4			CURSOR UP	FCTN KEY 4	SEL PRINT CNTRL	LINE CLEAR		
0	1	0	1	5			RESET ENTRY	FCTN KEY 5	DISABLE DEVICE CNTRL			
0	1	1	0	6				FCTN KEY 6	X-Y POSN			
0	1	1	1	7				RSV (3)		CLR TO END PG		
1	0	0	0	8	BACK-SPACE (4)		LOCK KYBD	RSV (3)		START BLINK		
1	0	0	1	9	HORIZ TAB (4)		RLSE KYBD	RSV (3)		SKIP		
1	0	1	0	10(A)	NEW LINE (4)		INSERT LINE			RESET (4)		
1	0	1	1	11(B)	VERT TAB (4)	ESC	INSERT CHAR IN LINE					
1	1	0	0	12(C)	FORM FEED (4)		DELETE LINE					
1	1	0	1	13(D)			DELETE CHAR IN LINE			PROTECT ACCESS		
1	1	1	0	14(E)			REVERSE TAB			END PROTECT		
1	1	1	1	15(F)						START PROTECT		(1)

1. Columns 0 and 1 are reserved for ASCII-defined functions; 7/15 is also reserved.
2. Control characters from columns 2 through 7 must be preceded by an ESC.
3. E2 and E3 are not used by the display. Codes 3/7 through 3/9 are reserved to avoid conflict with existing display devices.
4. Device controls that can be received by the terminal as either a column 0 code or ESC, x/x but are transmitted by the terminal as a column 0 code.

Figure B-2 -- MATRIX OF KEYBOARD/DISPLAY CONTROL CHARACTERS

BITS				b7 b6 b5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b4	b3	b2	b1	ROW \ COL	0 (1)	1 (1)	2 (2)	3 (2)	4 (2)	5 (2)	6 (2)	7 (2)
0	0	0	0	0			E2			BACK-SPACE (3)		
0	0	0	1	1			E3		NEW LINE (3)			
0	0	1	0	2					E1			
0	0	1	1	3			LINE FEED (3)					
0	1	0	0	4								
0	1	0	1	5								
0	1	1	0	6								
0	1	1	1	7								
1	0	0	0	8	BACK-SPACE (3)							
1	0	0	1	9								
1	0	1	0	10(A)	NEW LINE (3)							
1	0	1	1	11(B)	VERT TAB (3)	ESC						
1	1	0	0	12(C)			CAR-RIAGE RETURN (3)					
1	1	0	1	13(D)	CAR-RIAGE RETURN (3)							
1	1	1	0	14(E)								
1	1	1	1	15(F)								(1)

1. Columns 0 and 1 are reserved for ASCII-defined functions; 7/15 is also reserved.
2. Control characters from columns 2 through 7 must be preceded by an ESC.
3. Device controls that can be received by the terminal as either a column 0 code or ESC, x/x but are transmitted by the terminal as a column 0 code.

Figure B-3 -- MATRIX OF NIP {NONIMPACT} PRINTER CONTROL CHARACTERS

BITS				b7 b6 b5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b4	b3	b2	b1	COL / ROW	0 (1)	1 (1)	2 (2)	3 (2)	4 (2)	5 (2)	6 (2)	7 (2)
0	0	0	0	0			E2					
0	0	0	1	1			E3		NEW LINE (3)			
0	0	1	0	2					E1			
0	0	1	1	3			LINE FEED (3)					
0	1	0	0	4								
0	1	0	1	5								
0	1	1	0	6								
0	1	1	1	7								
1	0	0	0	8								
1	0	0	1	9								
1	0	1	0	10(A)	NEW LINE (3)					TOP OF FORM (3)		
1	0	1	1	11(B)	VERT TAB (3)	ESC						
1	1	0	0	12(C)	FORM FEED (3)		CAR-RIAGE RETURN (3)					
1	1	0	1	13(D)	CAR-RIAGE RETURN (3)							
1	1	1	0	14(E)								
1	1	1	1	15(F)								(1)

1. Columns 0 and 1 are reserved for ASCII-defined functions; 7/15 is also reserved.
2. Control characters from columns 2 through 7 must be preceded by an ESC.
3. Device controls that can be received by the terminal as either a column 0 code or ESC, x/x but are transmitted by the terminal as a column 0 code.

Figure B-4 -- MATRIX OF IMP (IMPACT) PRINTER CONTROL CHARACTERS

BITS				b_7 b_6 b_5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4	b_3	b_2	b_1	COL / ROW	0 ①	1 ①	2 ②	3 ②	4 ②	5 ②	6 ②	7 ②
0	0	0	0	0			E2					
0	0	0	1	1			E3		NEW LINE ③			
0	0	1	0	2					E1			
0	0	1	1	3			LINE FEED ③					
0	1	0	0	4								
0	1	0	1	5								
0	1	1	0	6								
0	1	1	1	7								
1	0	0	0	8								
1	0	0	1	9								
1	0	1	0	10(A)	NEW LINE ③					TOP OF FORM ③		
1	0	1	1	11(B)	VERT TAB ③	ESC						
1	1	0	0	12(C)	FORM FEED ③		CAR-RIAGE RETURN					
1	1	0	1	13(D)	CAR-RIAGE RETURN ③							
1	1	1	0	14(E)								
1	1	1	1	15(F)								①

1. Columns 0 and 1 are reserved for ASCII-defined functions; 7/15 is also reserved.
2. Control characters from columns 2 through 7 must be preceded by an ESC.
3. Device controls that can be received by the terminal as either a column 0 code or ESC, x/x but are transmitted by the terminal as a column 0 code.

Figure B-5 -- MATRIX OF KEYBOARD/PRINTER CONTROL CHARACTERS

BITS				b_7 b_6 b_5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4	b_3	b_2	b_1	COL / ROW	0 ①	1 ①	2 ②	3 ②	4 ②	5 ②	6 ②	7 ②
0	0	0	0	0			E2	CPRS 16 SP	END OF LINE ④	CPRS 16 0's		
0	0	0	1	1			E3	CPRS 17 SP	NEW LINE ③ ④	CPRS 17 0's		
0	0	1	0	2				CPRS 18 SP	E1	CPRS 18 0's		
0	0	1	1	3			CPRS 3 SP	CPRS 19 SP	CPRS 3 0's	CPRS 19 0's		
0	1	0	0	4			CPRS 4 SP	CPRS 20 SP	CPRS 4 0's	CPRS 20 0's		
0	1	0	1	5			CPRS 5 SP	CPRS 21 SP	CPRS 5 0's	CPRS 21 0's		
0	1	1	0	6			CPRS 6 SP	CPRS 22 SP	CPRS 6 0's	CPRS 22 0's		
0	1	1	1	7			CPRS 7 SP	CPRS 23 SP	CPRS 7 0's	CPRS 23 0's		
1	0	0	0	8			CPRS 8 SP	CPRS 24 SP	CPRS 8 0's	CPRS 24 0's		
1	0	0	1	9			CPRS 9 SP	CPRS 25 SP	CPRS 9 0's	CPRS 25 0's		
1	0	1	0	10(A)	NEW LINE ③ ④		CPRS 10 SP	CPRS 26 SP	CPRS 10 0's	CPRS 26 0's		
1	0	1	1	11(B)		ESC	CPRS 11 SP	CPRS 27 SP	CPRS 11 0's	CPRS 27 0's		
1	1	0	0	12(C)			CPRS 12 SP	CPRS 28 SP	CPRS 12 0's	CPRS 28 0's		
1	1	0	1	13(D)			CPRS 13 SP	CPRS 29 SP	CPRS 13 0's	CPRS 29 0's		
1	1	1	0	14(E)			CPRS 14 SP	CPRS 30 SP	CPRS 14 0's	CPRS 30 0's		
1	1	1	1	15(F)			CPRS 15 SP	CPRS 31 SP	CPRS 15 0's	CPRS 31 0's		①

1. Columns 0 and 1 are reserved for ASCII-defined functions; 7/15 is also reserved.
2. Control characters from columns 2 through 7 must be preceded by an ESC.
3. Device control that can be received by the terminal as either a column 0 code or ESC, x/x but is transmitted by the terminal as a column 0 code.
4. The first character following the ESC, 4/0 or ESC, 4/1 code as well as the first character of a write message is interpreted as a format control code. See Table B-2.

Figure B-6 -- MATRIX OF LINEPRINTER CONTROL CHARACTERS

BITS				b_7 b_6 b_5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4	b_3	b_2	b_1	ROW \ COL	0 [1]	1 [1]	2 [2]	3 [2]	4 [2]	5 [2]	6 [2]	7 [2]
0	0	0	0	0			E2					
0	0	0	1	1			E3					
0	0	1	0	2					E1			
0	0	1	1	3					READ BINARY			
0	1	0	0	4								
0	1	0	1	5								
0	1	1	0	6						END OF FILE		
0	1	1	1	7						END OF RECORD		
1	0	0	0	8								
1	0	0	1	9								
1	0	1	0	10(A)								
1	0	1	1	11(B)		ESC						
1	1	0	0	12(C)								
1	1	0	1	13(D)								
1	1	1	0	14(E)								
1	1	1	1	15(F)								[1]

1. Columns 0 and 1 are reserved for ASCII-defined functions; 7/15 is also reserved.
2. Control characters from columns 2 through 7 must be preceded by an ESC.

Figure B-7 -- MATRIX OF CARD READER CONTROL CHARACTERS

BITS				b_7 b_6 b_5	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
b_4	b_3	b_2	b_1	ROW \ COL	0 ①	1 ①	2 ②	3 ②	4 ②	5 ②	6 ②	7 ②
0	0	0	0	0			E2		CARD FEED			
0	0	0	1	1			E3		CARD FEED			
0	0	1	0	2					E1			
0	0	1	1	3					PUNCH BINARY			
0	1	0	0	4					OFFSET			
0	1	0	1	5								
0	1	1	0	6						END OF FILE		
0	1	1	1	7						END OF RECORD		
1	0	0	0	8								
1	0	0	1	9								
1	0	1	0	10(A)								
1	0	1	1	11(B)		ESC						
1	1	0	0	12(C)								
1	1	0	1	13(D)								
1	1	1	0	14(E)								
1	1	1	1	15(F)								①

1. Columns 0 and 1 are reserved for ASCII-defined functions; 7/15 is also reserved.
2. Control characters from columns 2 through 7 must be preceded by an ESC.

Figure B-8 -- MATRIX OF CARD PUNCH CONTROL CHARACTERS

TABLE B-1 - E CODE ASSIGNMENTS

E CODE	ASCII CODE	DEFINITION
E1	ESC 4/2	Write - Any device read active state is reset and the device is no longer reserved {i.e., the device is released}. An E-Code Read Message is not generated* by this Write Message. Read - This code indicates that the message is from a single-record input device or that the data is the last record from a multi-record device {if it is capable of detecting this}.
E2	ESC, 2/0	Write - Any device read active state is reset. If the device was reserved for the control station it remains reserved. It enables the device to generate* another read active state. Read - This code indicates that the device requires operator intervention to continue processing and that data has been lost from all active buffers for this device.
E3	ESC, 2/1	Write - Any device read active state is reset. This code initiates a read on multi-record input devices. Read - This code indicates that a buffer is available for the addressed device.
No. E Code		Write - Any device read active state is reset. If the device was reserved for the control station it remains reserved. An E-code read message is not generated* by this write message. Read - This is illegal.

* The term "generate" as used in various tenses in the above descriptions does not include the actual transmission of the message.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
Backspace	BS (0/8) ESC,P(5/0)	T T	Kybd/Display NIP Kybd/Display NIP	For a keyboard/display, moves the cursor one position to the left without affecting data. If the data protect option is present and the cursor would be placed in a protected field, the cursor will be backspaced through the protected field to one position to the left of the protected field. If the cursor is in the first position of the first line, it is repositioned to the last position of the last line. For a printer, moves the print head left one symbol position.
Back Tab	--	--	--	See Reverse Tab.
Card Feed	ESC,@(4/0) ESC,A(4/1)	T T	Card Punch Card Punch	Causes the card to be moved past the head without being punched. If the card feed code is inserted immediately following the last data word, for that card, the card will be ejected from the punch station and a new card registered. If the card feed code is found within the data stream, the card being punched is fed, without punching (ejected) and the following data is punched on the next card. The preferred card feed code is ESC, 4/0.
Carriage Return	CR(0/13)	T	NIP IMP Kybd/Printer	Controls the movement of the printing position to the first printing position on the same print line.
	ESC,,(2/12)	T	NIP IMP Kybd/Printer	
Clear to End of Page	ESC,W(5/7)	T	Kybd/Display	Causes all data from and including the cursor position to the end of the page to be cleared, except protected data.
CNTRL Key	ESC,0(3/0)	R	Kybd/Display	Pressing the CNTRL key locks the keyboard and initiates transmission of a read message from the keyboard/display in response to a poll message from the control station. CNTRL key depression causes the send symbol (Δ) to be displayed at the current cursor position. Read message transmission begins at the start index symbol (Ξ) or beginning of the display (if no start index is present), and terminates with the control code sequence at the send symbol position. The message ends with the send

*T = transmitted by control station;
R = received by control station.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS {CONT}

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
				code sequence and normal message termination. Protected data in the display memory is not transmitted if the data protect option is present. Retransmission capability and keyboard unlock are handled in the same manner as for the send sequence.
Compress 3 through 31 Spaces	ESC,#{2/3} through ESC,?{3/15}	T	Line Printer	Allows line printer messages to be shortened by use of an escape code sequence in place of the space codes normally required to correctly format a given print line. The character following ESC designates compression of the number of spaces as indicated in Figure B-6.
Compress 3 through 31 Zeros	ESC,C{4/3} through ESC,_{5/15}	T	Line Printer	Allows line printer messages to be shortened by use of an escape sequence in place of the zero codes normally required to correctly format a given print line. The character following ESC designates compression of the number of zeros as indicated in Figure B-6.
Cursor Down	VT{0/11} ESC,#{2/3}	T T	Kybd/Display Kybd/Display	Moves the cursor down one line in the same relative line position without affecting data. If the cursor would be moved into a protected field with the data protect option present, it will be advanced through the protected field to the same relative position on the first unprotected line. If the cursor was in the last line, it will be moved to the first line in the same relative position.
Cursor Up	ESC,$ {2/4}	T	Kybd/Display	Moves the cursor up one line, in the same relative position in the line without affecting data. If the cursor originally was positioned in the first line, it moves to the last line. If the cursor would be moved into a protected area when the data protect option is present, it will be advanced through the protected field to the same relative position on the first line above in which this character position is not protected.
Delete Character in Line	ESC,-{2/13}	T	Kybd/Display	Erases the character above the cursor and moves all data, to the right of the cursor in that line, left one position {except protected data and data to the right of a protected field}. A space is inserted in the last position of the line or in the last position before a protected field. The cursor does not move.

*T = transmitted by control station;
R = received by control station.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS (CONT)

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
Delete Line	ESC,,(2/12)	T	Kybd/Display	Erases the data in the line in which the cursor is positioned and moves all data below up one line. A line of spaces is inserted in the last line. If the cursor is in the last line, all data is erased in that line and the cursor is reset to the first position of the last line. This operation (line delete) is disabled in protect mode with the data protect option.
Disable Device Control	ESC, E(4/5)	T	Kybd/Display	When received as the first data code in a diagnostic write message, the control code interpretation logic is disabled and all subsequent data and normal control codes are stored as data. The disable device control code is not stored in display memory. The subsequent read message will be a retransmission of data just received with the exception that any escape sequence functions duplicated in ASCII column 0 will be returned as the ASCII character. If the disable device control code is used in conjunction with any other write message, it functions the same as in the diagnostic write except that a subsequent read is under operator control.
End of File	ESC,V(5/6)	R T	Card Reader Card Punch	For a card reader, indicates that this is the end of a file of information. For a card punch, indicates that an end of file should be punched.
End of Line	ESC,@(4/0)	T	Line Printer Card Punch	For a line printer, used to terminate the print line and precedes the format control character. New line performs the same function, however, end of line is preferred due to memory characteristics. For a card punch, see Card Feed.
End of Record	ESC,W(5/7)	R T	Card Reader Card Punch	For a card reader, indicates that this is the end of a record of information. For a card punch, indicates that an end of record should be punched.

*T = transmitted by control station;
R = received by control station.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS (CONT)

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
End Protect	ESC,^(5/14)	T	Kybd/Display	Defines the end of the protected area (if the display has the protect option) or indicates a tab set location for use with the forward tab code.
Escape	ESC (1/11)	T/R	All devices	Indicates that the next character is a station or device control code or/and following characters have a special meaning and are not to be interpreted as specified by the ASCII control and graphic character set.
E1	ESC, B(4/2)	T	Kybd/Display NIP IMP Kybd/Printer Line Printer Card Reader Card Punch	Write - Release device: should release any devices from read active state and make available for initiation of next sequence.
		R	Kybd/Display Kybd/Printer Card Reader	Read - This code indicates that the message is from a single-record input device or that the data is the last record from a multi-record device (if it is capable of detecting this).
E2	ESC,SP(2/0)	T	NIP IMP Kybd/Printer Line Printer Card Reader Card Punch	Write - Execute command: on error-free messages should initiate print, punch, or write commands. It enables the device to generate another read active state.
		R	NIP IMP Kybd/Printer Line Printer Card Reader Card Punch	Read - This code indicates that the device requires operator intervention to continue processing and that data has been lost from all active buffers for this device.
E3	ESC,|(2/1)	T	NIP IMP Kybd/Printer Line Printer Card Reader Card Punch	Write - Read device: on multiple-record device, read next block.

*T = transmitted by control station;
R = received by control station.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS (CONT)

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
		R	NIP IMP Kybd/Printer Line Printer Card Reader Card Punch	Read - This code indicates that a buffer is available for the addressed device.
F (Function) Keys	ESC,1(3/1) ESC,2(3/2) ESC,3(3/3) ESC,4(3/4) ESC,5(3/5) ESC,6(3/6)	R R R R R R	Kybd/Display ↓ Keybd/Display	Six function keys are present above alphanumeric keys on the keyboard. The keys are labeled, from left to right, as follows: F1, F2, F3, F4, and F5, and F6. ▽F▽ key operation is the same as CNTRL key operation except that the ESC, ▽CNTRL▽ sequence is replaced with ESC, ▽F▽.
Form Feed	--	--	--	See Reset and Top of Form.
Format Control Double Space Page Eject Single Space Suppress Space	 J(4/10) A(4/1) SP(2/0) 0(3/0)	T	Line Printer	Format control codes are position dependent. The line printer interprets the first character of a write message and the first character following an end of line code or a new line code as a format control code.
Forward Tab	HT(0/9) ESC,Q(5/1)	T T	Kybd/Display Kybd/Display	Moves the cursor forward to one position beyond the first end protect symbol. If no end protect is present between the cursor position and the end of page, the cursor is reset to home position.
Horizontal Tab	--	--	--	See Forward Tab.
Insert Character in Line	ESC,+(2/11)	T	Kybd/Display	Inserts a space at the cursor position and all data, except for protected data and all data to the right of the protected field, from the cursor position is shifted right one position. The last character in the line or the last character before a protected field is lost.
Insert Line	ESC,*(2/10)	T	Kybd/Display	Moves all data in the line in which the cursor is positioned and all data below, down one line. A line of spaces is inserted in the line in which the cursor is located and the cursor is reset to the first position in the line. Data in the last line is lost. This operation (line insert) is disabled in protect mode with the data protect option.

*T = transmitted by control station;
R = received by control station.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS (CONT)

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
Line Clear	ESC,T(5/4)	T	Kybd/Display	Erases all data from the cursor position to the beginning of the first protected field to the right of the cursor on that line if the data protect option is present, or to the end of the line on the display without affecting the cursor position.
Line Feed	VT(0/11) ESC,*(2/3)	T T	NIP IMP Kybd/Printer NIP IMP Kybd/Printer	Advances the paper 1 or 2 lines depending on setting of the carriage feed selector for the NIP. The IMP and keyboard/printer advance one line only. This is the equivalent to Cursor Down for the keyboard/display.
Line Skip	ESC,R(5/2)	T	Kybd/Display	Moves the cursor down one line and resets the cursor to the first unprotected position of the next line. Data is not affected. If the cursor is in the last line, it is reset to the first unprotected position of the display.
Lock Keyboard	ESC,((2/8)	T	Kybd/Display	Locks the keyboard and prevents operator input until a key board release is given.
New Line	LF(0/10) ESC,A(4/1)	T/R T	Kybd/Display Kybd/Display NIP IMP Kybd/Printer Line Printer Card Punch	For a keyboard/display, enters the new line code in memory at the cursor position, displays the new line symbol (¯) at that position, clears the remainder of the line from the right of the cursor position to the beginning of the first protected field on that line if the data protect option is present, or to the end of the line and resets the cursor to the first unprotected position of the next line. If the cursor was located in the last line, it will be reset to the first unprotected position after home. For a NIP, IMP, and keyboard/printer, this code causes the printer to perform a line feed and carriage return. For a line printer, see End of Line. For a card punch, see Card Feed.
Offset	ESC,D(4/4)	T	Card Punch	Causes the card currently being punched to be stacked with its trailing edge offset from the normal card edge registration. The card is fed into the output stacker after column 80 is punched or when a card feed code is encountered.

*T = transmitted by control station;
R = received by control station.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS (CONT)

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
Print Control	ESC,C(4/3)	T	Kybd/Display	When the print control code is received as the first data code in a write message to a keyboard/display, the control code interpretation logic at the display device is disabled and all subsequent data and normal control codes are stored as data. The print control code is not stored in the display memory. The message must terminate with the send symbol. A print request for that keyboard/display device is initiated at the conclusion of the message. Graphic data, protected and unprotected, is printed. Off-line device association rules apply to this operation.
Protect Access	ESC,](5/13)	T	Kybd/Display	Permits the control station to change to mode of operation of the display from protect to access if the data protect option is present so that the control station has access to the protected data on the display. The display is in access mode from the time that the protect access is received until the keyboard release code is received at the display.
Punch Binary	ESC,C(4/3)	T	Card Punch	When sent as the first code in a card frame, causes all data (6-bit binary) following to be punched in binary up to the end of the card or the card feed code. Card data not preceded by a punch binary code causes the card to be punched in Hollerith.
Read Binary	ESC,C(4/3)	T/R	Card Reader	When received in a write message prior to card reading, causes the entire card stack to be read in binary. In a read message, the read binary code will precede the binary data (6-bit) from column 1 for each binary card read. If a 7-9 punch is found in column 1 of a card during Hollerith read, the data is interpreted as binary, preceded by the read binary code, and 160 characters of 6-bit binary data will be read.

*T = transmitted by control station;
 R = received by control station.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS (CONT)

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
Release Keyboard	ESC,)(2/9)	T	Kybd/Display	Unlocks the keyboard. The keyboard is also released by the receipt of the next error-free message (containing an ESC, E1 sequence) or a Master Clear.
Reset	FF(0/12) ESC,Z(5/10)	T T	Kybd/Display Kybd/Display	Positions the cursor to the home position. If protected data is present in the home position, the cursor will be positioned to the first unprotected position. Data is not affected.
Reset Entry	--	--	--	See Reset to Start Index.
Reset to Start Index	ESC,%(2/5)	T	Kybd/Display	Causes the cursor to be reset to the position of the first start index symbol to the left or above the present cursor position. If a start index symbol is not present between the cursor position and home position, the cursor is reset to the home position. Start index symbols located to the right and below the cursor position are disregarded.
Reverse Tab	ESC,.(2/14)	T	Kybd/Display	Moves the cursor backward to one position to the right of the first end protect symbol. If no end protect is present between the cursor position and home position, the cursor is reset to home.
Selective Clear	ESC,S(5/3)	T	Kybd/Display	Erases all data with the exception of protected data from the display and resets the cursor to the home position. If protected data is present in the home position, the cursor moves to the first unprotected position.
Select Print Control	ESC,D(4/4)	T	Kybd/Display	When the select print control code is received as the first data code in a write message to a keyboard/display, the control code interpretation logic at the display device is disabled and all subsequent data and normal control codes are stored as data. The select print control code is not stored in the display memory. The message must terminate with the send symbol and a select print request for that keyboard/display device is initiated at the conclusion of the message. Only unprotected data is printed. Off-line device association rules apply to this operation.

*T = transmitted by control station;
R = received by control station.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS (CONT)

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
Send	--	--	--	See E1. The symbol, if displayed, appears as Δ.
Skip	ESC,Y(5/9)	T	Kybd/Display	Moves the cursor one position to the right without affecting data. If the data protect option is present and the cursor would be placed in a protected field, the cursor advances through the protected field to one position to the right of the field. If the cursor is in the last position of the last line, it repositions to home or the first unprotected position.
Start Blink	ESC,X(5/8)	T	Kybd/Display	Causes a designated area of the display to blink. Blinking continues until either a space code is detected or until the end of line is reached.
Start Index	ESC,"(2/2)	T/R	Kybd/Display	Inserts a start index code in the display memory and displays the symbol (≡) at the cursor position. The start index code marks the beginning of a transmission from the display memory. The code is written in memory. Multiple start index codes may be present on the display. When multiple start index codes are present, data transmission begins with the first start index to the left and/or above the send symbol (Δ).
Start of Entry	--	--	--	See Start Index.
Start Protect	ESC,_(5/15)	T	Kybd/Display	When the terminal is in protect mode, the start protect code must be preceded by a protect access code. Start protect stores a code in memory which indicates the beginning of a protected data field if the data protect option is present.
Tab	--	--	--	See Forward Tab.
Top of Form	FF(0/12) ESC,Z(5/10)	T T	IMP Kybd/Printer IMP Kybd/Printer	Advances the paper to the first position of the first line of a new page of print paper.

*T = transmitted by control station;
R = received by control station.

TABLE B-2 - DEVICE CONTROL CHARACTER DEFINITIONS (CONT)

FUNCTION	ASCII CODE	T/R*	DEVICE	DEFINITION
Vertical Tab	--	--	--	See Line Feed and Cursor Down.
X-Y Position	ESC,F(4/6)	T	Kybd/Display	At any point in the data portion of a write message, this code indicates that the following two characters are X and Y position information for the cursor. The cursor is set to the co-ordinates specified by these characters and further data transfers begin at the new cursor location. If the in-dicated coordinate position is located in a protected field, the cursor is advanced to the first character posi-tion following the protected field.

*T = transmitted by control station;
R = received by control station.

Table B-3 -- DEVICE CONFIGURATION CODE ASSIGNMENTS

COLUMN/ROW	BINARY CODE*	PERIPHERAL ASSIGNMENT **
2/0	P0100000	Card Reader
2/1	P0100001	Card Punch (80-column, low-speed)
2/2	P0100010	Interpreting Card Punch (Univac 1710)
2/3	P0100011	Line Printer (300 LPM)
2/4	P0100100	NA
2/5	P0100101	Line Printer (1200 LPM)
2/6	P0100110	NA
2/7	P0100111	NA
2/8	P0101000	NA
2/9	P0101001	IMP Character Printer (Univac RO)
2/10	P0101010	NIP Character Printer (NCR RO)
2/11	P0101011	Micro Drum
2/12	P0101100	Keyboard/Printer (IMP)
2/13	P0101101	Floppy Disk
2/14	P0101110	IATA Ticket Terminal
2/15	P0101111	NA
3/0	P0110000	Keyboard/Display (CRT)
3/1	P0110001	Keyboard/Display (Plasma)
3/2	P0110010	Keyboard/Display (24 line)
3/3	P0110011	NA
3/4	P0110100	NA
3/5	P0110101	Paper Tape Reader
3/6	P0110110	Paper Tape Punch
3/7	P0110111	Tape Cassette
3/8	P0111000	ICL (low-speed, 9-track)
3/9	P0111001	NA
3/10	P0111010	FDSI Teller Terminal
3/11	P0111011	Cartridge Disk
3/12	P0111100	SDT (ALS)
3/13	P0111101	OCR Reader
3/14	P0111110	NA
3/15	P0111111	NA
4/0	P1000000	Reserved for Device Not Available
4/1 to 6/15	P1000001 to P1101111	NA
7/0 to 7/14	P1100000 to P1111110	Special System Area

* P = parity bit
** NA - not assigned

TABLE B-3A - STATION CONFIGURATION CODE ASSIGNMENTS

COLUMN/ROW	BINARY CODE*	TERMINAL ID
2/0	P0100000	714
4/0	P1000000	Interactive Data Station, Trucking Industry, QSE 12328

TABLE B-4 - STATION STATUS ASSIGNMENTS

BIT	MEANING
b1	Not Assigned
b2	Error Switch in Disable Position
b3	Not Assigned
b4	Not Assigned
b5	Not Assigned
b6	Not Assigned
b7	Complement of Bit 6
b8	Parity

TABLE B-5 - DEVICE STATUS ASSIGNMENTS

BIT	MEANING
b1	Existent Device
b2	Read Request Active
b3	Not Busy
b4	Ready * **
b5	Device Dependent * Error * **
b6	Device Dependent *
b7	Complement of Bit 6
b8	Parity

* See Table B-6 for b4, b5, and b6 assignments which are device dependent.

** Preferred Usage

TABLE B-6 - DEVICE DEPENDENT STATUS BIT ASSIGNMENTS

DEVICE	BIT	MEANING
Keyboard/Display	4 *	Display Mode {0=Access, 1=Protect}
	5 *	Keyboard Mode {0=96 Characters, 1=64 Characters}
	6 *	Off-Line Print
Keyboard/Printer	4	Ready
	5	Paper Malfunction
	6	Print Only
NIP and IMP Printers	4	Ready
	5	Print Error and Lost Data
	6	Off-Line Print
Line Printer	4	Ready
	5	Print Error and Lost Data
	6	Not Assigned
Card Reader	4	Ready
	5	Read Error
	6	Buffer Full
High-Speed Card Punch	4	Ready
	5	Punch Error
	6	Compare Error
Low-Speed Card Punch	4	Ready
	5	Punch Error
	6	On Line
Source Data Terminal	4 *	Read/Write Error
	5 *	Maintenance Switch 0
	6 *	Maintenance Switch 1

* Recommended but not required usage -- deviates from preferred usage.

EXPOSITORY REMARKS

Section 1 - COMPARISON OF EXISTING PRODUCTS WITH THE STANDARD

This section includes a brief listing of code and protocol differences between several of CDC's existing products and the standard.

711-10 FIRMWARE VERSION 5.0

Addressing - 711-10 addressing is a subset of the standard. Redundant addressing or message sequencing modulo 2 is recommended but not supported by the 711-10.

Binary Data - The 711 will not recognize many of the binary data options offered in the standard. These are in addition to 711-10 capabilities.

Message Types -

Network disconnect and terminal disconnect messages are not implemented by the 711-10.

The status request message indicator ETB (1/7) has been changed to FS (1/12). This may impact existing systems (poller).

Extended status and configuration are not implemented by the 711-10.

Reserve Device - When a device reservation is requested with an alert, a read request active status bit is set when the acknowledge message is sent. However, if the device is busy with an off-line function, the response to a poll is a reject until the device completes the off-line operation.

E Code Usage - E code definition for the 711-10 is consistent with the standard except that the 711-10 E4 (send index) is now defined as a device control code, start index, rather than an E code.

Status Codes - Minor differences exist in device status which the user should note.

Keyboard/Display: Fixed bit assignment is consistent; however, b4 through b6 differ from the standard. These are designated for function keys and indicate print or selective print. The standard also assigns bits for display mode and keyboard mode; the 711 does not.

NIP and IMP Printers: Status is consistent for the fixed bits but b4 through b6 are different from the standard. The 711-10 reports paper malfunction, print, and selective print in these bits; the standard defines these bits as ready, paper malfunction, and off-line print.

Device Control Characters - The 711-10 differs from the standard in the following ways.

Keyboard/Display: Several device control characters are assigned to columns 0 and 1 of the ASCII character set and, as defined, are in potential conflict. The standard does not use these codes for device control but uses escape sequences for these functions.

New line as transmitted from the display (ESC, 4/1) is not consistent with the standard.

CR (0/13) performs a line delete function on the display. This is inconsistent with the standard but would not affect the average user.

NIP Printer: The printer functions a line feed for LF (0/10). The standard functions new line. This would not impact the user.

IMP Printer: The VT (0/11) functions a new line rather than a vertical tab. This is inconsistent with the standard but should not impact the user.

Terminology - Several terms used to describe 711-10 features and functions have been changed in the standard for clarity and consistency. The functions performed, however, are consistent. (These terms are defined in Appendix A and under "Device Control Codes" in Appendix B of the standard.)

217-14 (ASCII 200 UT)

Binary Data - The 200 UT does not handle binary data.

Message Types - The 200 UT does not implement the following message types: network disconnect, terminal disconnect, status request, configuration request, and initialize.

SYN Codes in Text - SYN codes must be inserted after a reset write, clear write, or diagnostic write message type indicator; the number depends on the line speed.

Reserve Device - The control station cannot reserve a device.

E Code Usage - The 200 UT uses E codes to select the device to which data is directed. The device address is not used for this purpose.

E1 Code: The E1 code in a read message from a line printer or a card reader signifies that the operator has depressed the "intervene" key to gain access to the system.

E4 Code: The E4 code is used as a start index code and can terminate a message to a display.

216 (ASCII NON-SPEEDEX)

Binary Data - The 216 does not handle binary data.

Message Types -

The 216 does not implement the following message types: network disconnect, terminal disconnect, configuration request, initialize, and extended status.

The 216 uses the ETB (1/7) character as the status request message indicator.

SYN Codes in Text - SYN codes must be inserted after a reset write, clear write, or diagnostic write message type indicator; the number depends on the line speed.

Reserve Device - The control station can reserve a printer using an alert.

E Code Usage -

E1 Code: The E1 code signifies that the message is from a display when encountered in a read message.

E3 Code: When an ESC, E3 code sequence appears in a write message, it is stored as data and has no other meaning.

E4 Code: The E4 code (start index) will unlock the keyboard and can replace the E1 code in a message written to the display.

Status Codes - Differences exist in device status codes. Extended status is not used.

Keyboard/Display: Status bits b3 through b6 are keyboard status switches 1 through 4.

Typewriter: Status bit b2 is not used; b2 through b6 are zero; bit b7 is the complement of b6, a logical 1.

Line printer: Status bits b1 and b2 are consistent with the standard; however, b3 through b5 indicate ready, paper malfunction, and print head malfunction, respectively. Bit b6 is not used and b7 is a logical 1.

215 ASCII POLLER

Addressing -

The poller only generates addresses in the ASCII column 6 range.

The poller does not support redundant addressing.

Message Types -

A network disconnect or terminal disconnect message cannot be passed through the poller.

Configuration request and response message are not supported.

The poller uses ETB {1/7} as a status request message indicator.

Extended status request messages are not generated by the poller.

Codes in Text - The poller will not pass codes in the message text that fall in ASCII columns 0, 1, 6, and 7.

EXPOSITORY REMARKS

Section 2 - GENERAL INFORMATION ON COMMUNICATIONS CONTROL PROCEDURES

Communications Control procedures for data transmission are frequently known as "protocols." A protocol is a logical method by which two or more machines can automatically exchange data, using transmission facilities such as telephone lines and data sets.

It is required of any effective protocol that it provide means to ensure that data are not lost, misdirected, misinterpreted, or mutilated in transmission.

Much confusion surrounds the topic of protocol primarily due to lack of standard definition adhered to by those who voiced comments on the subject. Standards agencies such as the American National Standards Insitute (ANSI) and the International Standards Organization (ISO) have worked for years with members of industry to develop protocols for universal use, but with little success, probably because of rivalry among the various authors, who may represent competing manufacturers or users. Another factor contributing to nonstandardization is the feature differences between competing manufacturers' products. As long as these differences exist, there is little hope one can substitute one for the other on a plug-to-plug basis.

Some protocol publications have appeared which offer character-oriented basic procedures using ASCII codes and a shopping list of formats. It is the shopping list aspect that encourages departures from a given format, usually for reasons of improved efficiency. The designer of a system protocol will frequently argue that efficiency is more important than versatility or vice versa.

Another problem is the choice of the order in which things can happen. No author of a general-purpose control procedure can possibly anticipate all of the uses to which it will be put. Each user adds his own touches to the basic framework with the result that systems using protocols having a common ancestry cannot intercommunicate.

A protocol generated by a manufacturer or user is a de facto standard. To become an effective national standard, a protocol must be capable of precise definition, as by means of a specification, such that intercommunication by users having equipments built by different manufacturers is assured. This definition requires detailed documentation in language that is clear and relevant, now and in future systems applications.

The following is a list of most commonly recognized variables involved in the exchange of information from a control station to a terminal.

1. Method of establishing a link (electrical and logical)
2. Error control techniques
3. Data code set
4. Commands and responses
5. Data formats (both device and application dependent)
6. Timing and sequence considerations.

The following discussions further explain the various functions involved.

LEVELS OF PROCEDURES

Three levels of control procedures are commonly identified in standards, as follows:

1. Link Level
2. Station Level
3. Access Level.

Link level refers to the functional link, irrespective of the physical connection used, between stations. In other words, link level procedures are those used to obtain a logical communications path between sites.

Station level procedures are those related to the functions performed common to all devices; for example, a controller to which a number of peripherals are attached must manage the flow of information between the communications link and the identified peripheral equipment. Such management is station level protcol.

Access level protocol refers to the record handling procedures within peripheral devices or within stand-alone devices. The term ▿access level▿ is preferred to ▿device level▿ in technical standards for protocols. An example of access-level control is remote control of a card reader, in which card movement as well as data movement is governed by a control procedure.

It is not necessary that all three levels exist in a protocol. There are devices such as teletypewriters that may have only access-level control capability.

FORMAT

Protocols control message transmission through the use of data sequences that accompany the message text. These sequences may be in rigid format or may be flexible to conform to changing conditions in a system. When a fixed format is used, the control information frequently has positional significance. For example, a synchronizing sequence might be followed by six bytes signifying start of heading, acknowledgement of previous message, link, station, and device address of the following message, and start of text, in that order. When control information does not have positional significance, extra information must be included in the protocol to identify control and address elements. It is customary, but not required, that protocols utilize bytes or characters of fixed size in a fixed sequence. More efficient control procedures use the minimum number of bits required to convey the information, but may be unreliable because of the higher probability of misinterpretation if bits are altered by noise and if no means are available to validate the bits.

NETWORK RELATED ITEMS

Network parameters must be considered in all communications protocol. The role of the station, type of communications link, method of initiating communication, message structure, and data transfer control are important factors in the design of system protocol.

Dominant/Subordinate Relationship

It is customary to have a dominant station and one or more subordinate stations in a system. Sometimes the dominant station is called the ▿master▿ or the ▿control▿ station. It is possible to delegate dominant status such that a subordinate takes control of a system when instructed to do so. It is this type of situation that complicates protocols.

Point-to-Point Versus Multipoint

When two stations (and only two stations) intercommunicate, it is a point-to-point situation. Either station may be dominant, or the role of dominant station can be alternated between the two stations. Point-to-point operation is possible on either the public switched telephone network or on dedicated leased or private lines. Multipoint operation implies the use of a single transmission facility to link together three or more stations. The facility is almost always a leased private line. Conference calling is not feasible in data transmission on the public switched network.

Contention Versus Selection and Polling

Contention may be illustrated by the attempt by two stations to simultaneously transmit to each other. This is also known as a "head-on collision" or "glare" situation. Another illustration is the attempt by two or more stations on a multipoint facility to simultaneously gain access to another station.

The result of contention is that the transmission is lost or garbled. A procedure is required to resolve contention when it occurs.

Selection is the process whereby a dominant station addresses a subordinate station for the purpose of sending data to the subordinate. Contention is prevented.

Polling is a process in which a dominant station addresses a subordinate station to solicit a response. Contention is prevented.

In a given system the choice of contention or selection/polling as a part of a protocol depends primarily on how frequently traffic must be handled by each station. Polling and selection are generally used in combination with heavily loaded transmission links. Contention is more efficient than polling if stations are inactive, because polling time is not wasted on idle stations. It is possible to use contention until traffic builds up and then automatically switch to polling.

Synchronization and Framing

Asynchronous signalling implies character-by-character transmission with the possibility of pauses between characters. This is sometimes called start-stop or anisochronous transmission. Each character is internally clocked. Only the maximum character rate and the baud rate need be known to the transmitting and receiving terminals. Protocols for asynchronous signalling can be very simple because each character is self-synchronizing and no synchronizing sequences need be considered in the communications protocol.

Synchronous or isochronous signalling implies contiguous bits within each message block, with no pauses and with a minimum of redundancy. A bit stream is required as a prefix or preamble to each new transmission for the purpose of achieving bit-sync and character-sync. This bit stream may consist of alternating one/zero pairs or of repeated SYN characters, or both. The SYN character is frequently used for time fill within or between blocks of data, to avoid resynchronization.

In two-way-alternate synchronous transmission it is usually necessary to provide the prefix ahead of each new transmission. A lot of overhead is incurred as the result, especially if the data block is short. Thus it is apparent that asynchronous transmission is more efficient for conversational transmissions which employ very short transmissions and responses. Some systems may employ a hybrid technique in which short inquiries from keyboards use slow asynchronous signalling, whereas the longer responses from the data processor are sent synchronously at a higher bit rate.

Framing implies a logical method for indicating the limits of a logical entity such as a block or message. A framing character or byte is sometimes called a "flag". Caution must be employed to ensure that a flag is never discovered within a data block, thus causing loss of following data. In some protocols the flag sequence utilizes a distinctive bit pattern that is prevented from appearing in the data stream.

PAD Characters

PAD Characters are inserted after the LPC in a message to insure that all bits of the LPC have passed through the modem. Some implementations have used the SYN character for this purpose in the past. The use of SYN in this manner however can cause trouble in some systems. The implementor should check the equipment specifications before using SYN as a PAD. The "all ones" character could be used.

Control Characters

The ASCII code contains 96 graphic characters, including space and delete (rubout) which are not normally considered printable. The remaining 32 characters are non-printing functions which include ten communications control characters called: ACK,

NAK, DLE, ENQ, SYN, SOH, STX, ETX, ETB, and EOT. These ten must be used only for data transfer and never for the control of data processing functions or for access/device-level functions that do not include communications.

ERROR CONTROL TECHNIQUES

Data is always subject to multilation in transmission. Means are required to detect errors in data and recover from the error situation either by retransmission or by reconstruction of the data.

Error Detection

Means to detect errors usually include redundant information imbedded by the transmitting terminal within the data stream, and logical devices or programs within the receiving terminal that recognize the redundant information. An example of such redundancy is the parity check modulo-2 summation which may be performed on each data byte and on each bit level in order to provide a matrix in the mathematical sense. The parity checking information generated by the source is appended to the data. The receiver compares the parity checks received with those that it computes, and identifies discrepancies as data errors.

Another form of redundancy is the CRC which eliminates the need for parity checking. The CRC or cyclic redundancy check is the mathematical process or result of multiplying or dividing each new data byte by a reference polynomial, using binary arithmetic. Each product or quotient is added to the previous one until the end of the data stream is reached, at which time the summation is appended to the data and is transmitted. The receiver performs an identical operation on the data and then compares the locally generated summation with the one received. Any difference indicates one or more errors in the data or in the CRC.

A 16-bit CRC is effective for all messages of 16 bits or longer. Parity checks are effective only for short messages because of the high probability of compensating bit errors. The CRC provides a fixed overhead independent of message length. Character parity provides overhead proportional to message length.

In general, the CRC is to be preferred to parity checking because it is several orders of magnitude more reliable in detecting multiple-bit errors in noisy transmission media.

Error Recovery

Recovery is the procedure followed after errors are discovered. It usually consists of requesting and obtaining retransmission of data. The procedure used should contain provisions for aborting transmission when retransmission requests are ineffective because transmission quality has deteriorated to the point that no data can be received correctly in several attempts, or if retransmission requests do not reach the station.

Another consideration is the interval between data errors. If the average error rate is greater than the data block transmission rate, virtually no data will ever be received correctly. Under these circumstances, it is desirable for the station to have the ability to dynamically alter the block length to adapt to transmission conditions. Long blocks decrease transmission overhead, but short blocks will less frequently contain errors.

Forward Error Correction

An alternative to retransmission of data containing errors, is the reconstruction of the data by the receiver. If sufficient redundancy is included by the source such that the same information may be found in two or more places in the data stream, and well separated with respect to noise bursts, it is possible for the receiver to correlate redundant elements and to reconstruct errored bits. In general this method is used only when the large overhead can be tolerated and when error correction by retransmission is not feasible.

From the foregoing, it is apparent that CDC does not have a uniform policy toward communications protocols. It is also evident that one must be adopted which specifies in considereable detail how the ASCII code characters must be used. In the existing documents, there are too many opportunities for misinterpretation and for design options which lead to incompatible terminals.

SUPPLEMENTARY INFORMATION

CDC System Standard 1.10.002 provides a two-way alternate protocol. It was implemented in Source Data Collection Equipment and CSPL.

IBM uses three versions of BSC {Binary Synchronous Communications} and a new bit-oriented transparent procedure called SDLC {Synchronous Data Link Control}.

Multinational Data, Project 18, has a number of protocols in development. These are based on ISO proposals and are intended to influence the ISO.

ANSI has published basic mode protocols in publication X3.28 - 1971 and is working on new versions of SDLC called ADCCP {Advanced Digital Communications Control Procedure}.

ECMA has offered a basic mode protocol, ECMA-16, dated May 1968.

ISO has published Recommendation R 1745 dated January 1971, and is working on a version of SDLC/ADCCP. CDC has identified internally sponsored protocol developments called: Mode 4 {two-way alternate, synchronous, like 216 product procedures}. Mode 3 {teletypewriter oriented, asynchronous, two-way alternate}. Mode 2 {two-way-simultaneous, synchronous, noncontinuous}. Mode 1 {two-way-alternate version of Mode 2}.

EXPOSITORY REMARKS

Section 3 - TUTORIAL

PROTOCOL DESCRIPTION

The Mode 4C protocol is a two-way alternate, non-symetric protocol that can be used on switched or dedicated 2 wire or 4 wire point-to-point or multidrop communications links.

By "two-way alternate" is meant that messages can be transmitted and received but not at the same time. A response to each command must be transmitted before another command can be received by the same station. Therefore, after each message has been transmitted to a station the direction of transmission on that link will be reversed and the station will transmit a response. The response received by the control station is thus always in reply to the message immediately preceeding it and is not in response to any other message. If the reply is not received by the control station (because of error bursts on the communications link) the status of the message can be determined by the reply to a subsequent message (as described below in the discussion of error recovery) but the reply is in response to the last message and not the first.

It is necessary that a station always reply to a message before another message can be accepted because the protocol does not have the ability to indicate with each message whether the station is to reply immediately or is to defer a response until a message is received calling for an immediate response.

The term "non-symetric" indicates that the station issuing messages (the control station) has more capability than the station receiving the message. For this reason, only the control station can initiate an exchange of messages, the terminal can only wait for a message and then reply to it. This also implies that all networks on which this protocol is used are polled networks. Contention networks can not be supported.

The sequence of events that occur with this protocol are the same whether the communications link is 2 wire or 4 wire, full duplex or half duplex or point-to-point or multidrop. That is, the terminal always replies to a message before it will accept another message. The time at which the reply will be sent often is dependent on whether the modems used on the links are alternating carrier or constant carrier modems. Alternating carrier modems are commonly found on the following types of voice grade links:

- Two wire, half duplex, switched or dedicated, point-to-point.
- Four wire, half duplex, dedicated, point-to-point.

Constant carrier modems are commonly found on the following types of voice grade links:

- Four wire, full duplex, dedicated, point-to-point.
- Four wire, full duplex, dedicated, multipoint, control station only. (Secondary stations use alternating carrier).

With an alternating carrier modem, the terminal will wait for its received carrier to drop before transmitting the reply. With a constant carrier modem the terminal will transmit its reply as soon as it has received the last character of the message.

The actual amount of time required to complete a message-reply sequence is a function of transmission time, line turnaround time and delay within the modems themselves and is therefore highly variable.

Other than that stated earlier, there is no function performed by timeouts in this protocol. Some other protocols reply on inter-character timeouts to detect the end-of-message condition. This is not the case in Mode 4C protocol since every message is terminated by an ETX code (followed by the LPC character) and this is used to determine end of message.

Certain aspects of the Mode 4C protocol will now be discussed in order to provide the communications system designer (both hardware and software) with a guide to the use of the protocol as well as the intent of the writers of the Mode 4C standard.

DATA PROTECTION

By "data protection" is meant the facility used to ensure that the content of a message received by a station is the same as that which was sent. This is the basis for the error recovery procedures which are discussed below.

In this protocol data protection is provided by character and message parity. Because character parity is used, one bit of each eight bit character is used as a character parity bit and is not available to carry information. Only 128 different characters are possible instead of 256 if all bits were used to transmit information.

Binary data can be transmitted in the information field of a message but not more than 6 bits of each character can be used as data bits. One bit must be used for character parity. Another bit must be used to ensure that a data character does not duplicate a control character.

Of primary concern here is the possibility that an ETX character will be duplicated causing a premature end of message condition, or that a SYN character will be duplicated causing data to be lost or an incorrect message parity character to be generated.

To prevent this condition from occurring bit b-7 is set to the complement of bit b-6 in each binary character, a condition that never occurs in a control character.

CONTROL OF SINGLE AND DOUBLE BUFFERING

The Mode 4C protocol provides for control of single and double buffered output devices. The following definitions and rules apply:

- An E-2 code in a read message indicates that operator intervention is required for continued operation of the device and data has been lost from all buffers assigned to the device.
- An E-3 code in a read message indicates that a buffer is available for the device.
- If a buffer is available, a read message containing an E-3 code is generated when a write message (write, clear write, reset write) containing an E-2 code is received, without error, by the terminal. The next poll message sent to the device will then transfer the read message to the control station.
- If a condition that would generate a read with an E-2 code is present the read message with the E-2 code will be generated following the next write message containing an E-2 code. The read message will be sent to the control station in response to the next poll message directed to the device.
- A dummy write is a write message without data and containing an E-2 code.
- If it is desired that a data not be discarded when a device requires operator intervention and can become operational when the operator intervenes, a read message with an E-2 code will not be generated.

It is possible to control single and double buffered devices:

- using status requests only and ignoring E-code read messages,
- using device polls to obtain E-code read messages,
- using both device status requests and polling for E-code read messages.

All of these procedures are illustrated below for the double buffered case.

It should be noted that single buffereing can be treated as a special case of double buffereing where;

- a buffer will be available when the previous write message has been completely processed,
- there is no initial buffer loading problem (see below).

The control of double buffering can be divided into three phases --

1. Initial buffer loading.
2. Operation after both buffers are loaded and before the final buffer has been transmitted.
3. Obtaining the status of the final buffer.

INITIAL BUFFER LOADING

For a double buffered device, it must be possible to load both buffers without allowing the status of the device to be unknown.

The procedure using device poll messages to obtain E-code read messages indicating the staus of the device is shown below.

	Control Station		Terminal
1.		—— write buffer 1 ——>	
2.		<—— ACK ——	
3.		—— poll ——>	
4.		<—— read E-3 ——	
5.		—— write buffer 2 ——>	
6.		<—— ACK ——	
7.		—— poll ——>	
8.		<—— read E-3 ——	
		<—— reject ——	
		<—— read E-2 ——	

The read E-3 message at line 4 indicates that another write message can be sent to the device because a buffer is available. If an E-2 code had been sent at this time it would indicate that operator intervention is required before more data could be sent and that the previous buffer must be sent again. The E-2 or E-3 read message will be sent immediately in response to the first poll message (line 3) so that the second write message can fill the second buffer (line 5) without fear of destroying the E-code generated by the first write message (line 1).

After the second buffer has been transmitted {line 5} a reject {line 8} in response to the following poll message {line 7} will indicate that the device is still processing the first buffer. A read E-3 message indicates that the first buffer has terminated without error and that another buffer {buffer 1} is available. A read E-2 message {line 8} will indicate that data has been lost from both buffers since the device required operator intervention while processing the first buffer and the operator could not successfully recover from this condition. If the operator could recover and the system was designed to retain buffers under these circumstances, repeated reject messages instead of a read E-2 message would be sent {line 8}. In this case the control station would wait for a read E-2 or E-3 message before more data could be sent to the device.

Note that if a buffer is available after a write message has been sent to a device a read E-3 message will be generated immediately. If the device status changed so that operator intervention is required and the read E-3 message had not been transmitted in response to a poll message the E-3 could be changed to an E-2 read message. Once the read E-3 message has been transmitted it cannot be changed to an E-2 read message.

The E-2 read message will be transmitted in response to the poll message following the next write message containing an E-2 code sent to the device. If this occurs the write message just transmitted will be discarded by the terminal.

The status of the second buffer will be reported in response to a poll following the write message to load the first buffer for the second time. This is shown below.

	Control Station → Terminal	Meaning
9.	——— write buffer 1 ———→	
10.	←——— ACK ———	
11.	——— poll ———→	
12.	←——— read E-3 ———	buffer 2 available
13.	——— write buffer 2 ———→	
14.	←——— ACK ———	
15.	——— poll ———→	
16.	←——— read E-3 ———	buffer 1 available because first message was processed without error.
17.	——— write buffer 1 ———→	
18.	←——— ACK ———	
19.	——— poll ———→	
20.	←——— read E-3 ———	buffer 2 available because second write message processed without error

Thus it can be seen that once both buffers are loaded each read E-3 message indicates that the previous write message has been fully processed, that the other write message is being processed, and that another write message can be sent to this device.

It is also possible to perform this procedure using status requests instead of poll messages as shown below.

Control Station	Terminal	Meaning
21. ——— write buffer 1 ———→		
22. ←——— ACK ———		
23. ——— device status request ———→		
24. ←——— not busy status ———		go to line 25
←——— ready status ———		go to line 25
←——— error, lost data ———		go to line 21
25. ——— write buffer 2 ———→		
26. ←——— ACK ———		
27. ——— device status request ———→		
28. ←——— not busy ———		
←——— ready ———		
←——— error, lost data ———		

If the device status is ▿ready▿ a buffer is available to receive more data. ▿Not busy▿ indicates that the device has completed processing all buffers. ▿Error, lost data▿ indicates that the device requires operator intervention and that all buffers sent have been discarded.

As with read messages, there is a problem of timing in connection with device status. The problem is concerned with the clearing of the ▿error, lost data▿ indication.

If a write message is being processed and an ▿error, lost data▿ condition occurs this condition could go undetected if the control station transmits another write message which clears the ▿lost data▿ condition. The sequence is as follows:

Control Station	Terminal	Meaning
29. ——— write ———→		
30. ←——— ACK ———		
31. ——— device status request ———→		
32. ←——— ready ———		
error, lost data condition occurs		
33. ——— write ———→	{error cleared}	

Control Station	Terminal	Meaning
34. ← ACK		
35. device status request →		
36. ← ready		

To solve this problem it is required that:

- A write message {write, clear write, reset write} containing an E-2 code will not reset the ▽error, lost data▽ indication in the device status byte.
- An alert message sent to the device will clear the ▽error, lost data▽ indicator in the device status byte.

If these rules are followed, the response on line 36 above will be ▽error, lost data▽ and the status will not be lost. To recover, the control station will send an alert message followed by one or both write messages as required.

A third procedure, one that uses both poll messages and device status requests, can also be used as shown below.

Control Station	Terminal	Meaning
37. write →		
38. ← ACK		
39. poll →		
40. ← reject		
41. device status request →		
42. ← status		

Where status can be ▽ready▽, ▽not busy▽ or ▽error-lost data▽ or combinations of these {such as ready and not busy or ready and busy}. This procedure can be used when an expected E-code read message is not transmitted in response to a poll message at the time it is expected. If the device is ▽not ready▽ and ▽busy▽, it may require operator intervention {in the case where the operator can fully recover and data is not discarded}, and an E-2 code is not sent. Note that the E-2 code may eventually be sent if the operator can not recover and is able to indicate this {perhaps via a button on the device that will cause the device status to change to ▽error, lost data▽}.

STATUS OF THE LAST BUFFER

As indicated above the status of each buffer can be obtained from E-code read messages. The E-code read message is enabled by the write message {containing an E-2 code} that follows the write message for which the status is desired. After transmission of the last write message, an E-code read message indicating the status of the last buffer will not be generated. To solve this problem a dummy write message is sent to the device. This will cause the E-code read message to be generated for the last data buffer. The status of the dummy write is not needed.

Notice that a write message not containing an E-2 code will not enable the generation of an E-code read message. This has been done to allow device reservation to be released without generating unwanted E-code read messages.

The status of the last buffer may also be obtained using a device status request, in which case it is immaterial whether or not an E-code read message is generated for the last buffer.

TERMINAL IDENTIFICATION

A communications network can include both switched and dedicated links. The terminals on dedicated links are always known to the communications processor. This is not always the case on switched links. If more than one type of terminal can communicate with the communications processor over a given switched link it may be necessary to identify the type of terminal currently on the line. This is due to differences in terminal capability such as CRT screen size, code set used, buffer capacity, number of buffers per device, etc. Therefore a means of identifying terminals may be needed. For terminals using the mode 4C protocol several methods are available for terminal identification.

A terminal can be identified by its station configuration code sent in response to a configuration request. The mode 4C standard specifies a unique code for each mode 4C terminal. There are several problems with this approach, however. Not all mode 4C terminals developed at the present time implement the configuration code in this manner and cannot use this method of terminal identification. It is also likely that mode 4A and mode 4B terminals could also communicate with the communications processor over the same line used by mode 4C terminals. Since mode 4A terminals do not implement the configuration request and mode 4B terminals do not use the station configuration code for this purpose, the configuration request can not be used to identify these terminals.

A second method of terminal identification makes use of the terminal station address to identify terminals. To do this, each type of terminal would use a unique station address. For example, a 714 could use station address 161, a 711-10 could use 162, a 200 UT could use 160, etc. The communications processor would poll the terminal using a different station address until the terminal responded. The response station address would then identify the terminal. This technique can be used where mode 4A, mode 4B and mode 4C terminals use the same line.

POLLING

Poll messages are used by the control station to obtain read messages from a terminal. These read messages can contain data or status information (E-code read messages). There are several ways in which polling can be implemented by a communications processor.

If the communications processor is designed to read messages from and write messages to terminals only at the request of an application program, the communications processor will use device polls to read data from the terminal. Station polling would not be used because the data obtained would have to be retained until it is requested by an application program. This method of operation provides maximum flexibility to handle every situation but sacrifices performance since idle time polling can not be implemented without storage facilities on the communications processor.

Communications processors can also be designed to search for read messages from the terminals when not processing specific read or write requests from an application program. This is often referred to as idle time polling. When the communications processor obtains a message from a terminal that has not been requested by the application program it will interrupt the application program with a data available status. The application program must then issue a read request so that the communications processor can transfer the message to the application program. Thus the application program must be able to process a message from any terminal whenever it is available. This type of operation increases performance of the communication system by descreasing the overhead of the CPU resident software and overlapping reading and writing of messages to the terminals.

With the mode 4C protocol, idle time polling can be accomplished in several ways. Station status requests can be used to determine which devices of a multi-device station have read requests active. Device poll messages can then be sent to those devices that have active read requests to obtain the messages.

An alternative is to issue station poll messages to obtain any read message that is available. If a given terminal responds with a reject message another terminal can be polled.

Both of these techniques will use the communications link in an efficient manner as compared to polling each device in the poll list to search for data.

ERROR PROCESSING

Error detection in the mode 4C protocol is based on character and message parity. Error recovery is based on retransmission of the message in error. There is no provision for error recovery based on error correction at the receiving station.

Error recovery is always the responsibility of the control station.

The terminal must only react to the commands sent to it by the control station.

Following transmission of a message to a terminal the control station must receive an indication as to the status of this message. This is provided when the terminal transmits an acknowledge, error or reject message in response to each message that it receives. The acknowledge message indicates that the message from the control station has been received without error and will be processed by the terminal. The reject message indicates that the terminal could not accept the message at the time it was transmitted. The most common reason for this is that there was no buffer available to store the message. When this condition clears the terminal will be able to accept messages. The recovery procedure therefore is to resend the rejected message when the terminal is again able to accept messages. This can be determined by retransmitting the message until it is acknowledged or by using a status request to determine when the terminal is ready to accept another message and then resending the message.

The terminal will transmit an error message to the control station for the following reasons:

- A parity error {character or message} was detected after detection of the correct terminal station address. The error message will be transmitted upon detection of the ETX character in the receive message.
- The message contained an illegal device address. The terminal will abort the receive sequence upon recognition of the ETX character and transmit an error message to the control station.
- The message contained an unrecognized Message Type Indicator. The terminal will abort the receive sequence upon recognition of the ETX character and transmit an error message.
- Special error conditions may be detected by the terminal resulting in an error response message, e.g., device data storage exceeded or terminal time-out before receipt of the ETX character.

In addition to receiving an error response the control station is notified of an error condition when it does not receive a response within an expected time interval. The following conditions can result in no response being received by the control station.

- The terminal received a message that did not contain its station address.
- The response sent by the terminal was destroyed by noise on the communications link.
- The carrier dropped before the terminal detected the ETX character.

Upon receipt of an error response the control station can recover by correcting any invalid conditions that existed, if any, and retransmitting the message in error.

This is not the case, however, when no reply is received. If the message was received without error and the acknowledge reply was not received, retransmission will result in a duplicated message. This situation is allowable if the device in question is a CRT but is not desirable in the case of a hard copy device such as a line printer. If the message is not retransmitted and the reply that was lost was an error response then data is lost.

To recover from this situation a message sequence number is generally provided, to be used in conjunction with a command that will cause the terminal to transmit the status of the command whose sequence number was indicated.

In the mode 4C protocol, however, each response indicates only the status of the command immediately preceeding it with its Message Type Indicator but can provide the desired status indication of a previous write command via its sequence number. This sequence number is contained in the device address. Bit b-5 of this character is used as a two valued sequence number in the following way:

- Whenever the control station transmits a write, clear write, or reset write to a given terminal it will alternate the value of bit b-5 between 0 and 1.
- Whenever a terminal receives a write, clear write or reset write message without error (sends an acknowledge reply) it will store the value of bit b-5 of the device address and use it in all subsequent replies.
- When a reply is not received by the control station in response to a write, clear write or reset write message it will issue a poll message (or status request message) to the terminal. Bit b-5 of the device address of the reply can then be examined to determine the status of the message in doubt. If the value of this bit is the same as that sent in the message in question, the message was received without error and acknowledged. A different value indicates that the message was received with an error and must be retransmitted.

Several points should be noted here. When a reply to a write clear write or reset write message is not received no write, clear write or reset write message should be sent to that terminal until the status of the unacknowledged message is determined. The reason for this is that another write message could change the value of the sequence number stored by the terminal and thus prevent proper error recovery. Other operations such as polling can be performed with the terminal before error recovery is performed.

When a reply to a message is not received and error correction is performed immediately, performance will not be affected in any detectable manner. This is due to the very low frequency of the short reply messages being destroyed by noise bursts on the line. The probability of a message being affected by noise increases with increased message size. Since messages that include data are usually 100 times longer than response messages the frequency of lost replies is correspondingly less than the occurrence of errors in data messages, say one occurrence per terminal per day or less.

Another problem, also to a great extent negated by low frequency of occurrence, is the ambiguous state of the terminal which would result if the response to the first write message sent to a terminal that had just been powered up were lost.

The terminal could be built so that the toggle bit (sequence number) were always set to a certain value upon being powered up, but the ambiguity would still exist under other circumstances. If another application program began using the terminal after a previous application terminated and each application program maintained the sequence number the condition discussed above could exist. The same is true if the sequence number were maintained by the control station operating system and it were reloaded.

Fortunately there are several techniques that can be used to solve this problem. The control station can issue a poll message or status request message to the terminal. The response to this message will indicate (by bit b-5 of the device address) the setting of the sequence number in the terminal.

Another method would be for the control station to transmit a write, clear write or reset write message to a CRT device on the terminal if one exists. If the message was not acknowledged it would be retransmitted until the acknowledgement was received. In this way the control station can set the sequence number to the desired value.

ERROR INDICATION TO TERMINAL

We have just discussed the error indication required or available to the control station. Error indicators are also required by the terminal. The reason for this is that a read message transmitted to the control station in response to a poll message can contain errors and must be retransmitted. The terminal must retain the message until it has been received by the control station without error. Therefore the control station must be able to notify the terminal when the read condition can be reset. This is done by transmitting a write, reset write or clear write message to the terminal.