SPERRY RAND

UNIVAC

9000 SERIES

COMPATIBLE DATA COMMUNICATIONS SUBSYSTEM (DCS-1C)

PROGRAMMER REFERENCE

UP-7846

This document contains the latest information available at the time of publication. However, the Univac Division reserves the right to modify or revise its contents. To ensure that you have the most recent information, contact your local Univac Representative.

UNIVAC is a registered trademark of the Sperry Rand Corporation.

PRINTED IN U.S.A.

CONTENTS

1. INTRODUCTION

1.1. GENERAL

The UNIVAC Compatible Data Communications Subsystem (DCS-1C) is essentially a communications adapter that can be used on any UNIVAC 9000 Series system. The subsystem can control one data link and is compatible with currently accepted communications control procedures.

A single DCS-1C (Feature F1357-00) may be mounted within the 9000 Series processor cabinet. Up to four DCS-1C's may be mounted in a freestanding cabinet (Type T-8577-00). The DCS-1C is connected to a user-specified position of the multiplexer channel. The data link itself can be point-to-point private, point-to-point switched, or multipoint private.

This document describes the functions, capabilities, and programming requirements for the DCS-1C. For detailed information on the I/O characteristics of the associated processors, see:

- *UNIVAC 9200/9300 Systems Data Communications Subsystem (DCS) Supplementary Reference, UP-7614* (current version)
- *UNIVAC 9400 System Assembler/Central Processor Unit Programmers Reference, UP-7600* (current version)

It is assumed that the programmer is familiar with data communication terminals, lines, procedures, and other related equipment which comprise a complete data communication system. However, a complete glossary of all communication terms and abbreviations used in this manual is given in Appendix B.

2. BASIC SUBSYSTEM COMPONENTS

2.1. GENERAL

The basic DCS-1C contains a line terminal controller, communication interface, and one line terminal. The line terminal controller handles the "handshaking" required by multiplexer channel logic, validates I/O commands, and passes bit-parallel data between the line terminal and the processor. The communication interface (either standard or wideband) provides the physical, electrical, and logical connection with a synchronous data set. Both the controller and the interface are essentially transparent to the programmer.

2.2. LINE TERMINAL

The basic DCS-1C must include either the nontransparent (F1395-00) or transparent (F1395-01) line terminal. The nontransparent line terminal provides a longitudinal redundancy check on ASCII messages. It is otherwise identical to the transparent line terminal.

NOTE: The 9200 and 9300 systems, under most conditions, receive only bits 4 through 7 of the status byte (see 3.2). Users of these systems cannot distinguish between EBCDIC- and ASCII input. For these systems, the code to be employed on a given data link should be prespecified.

2.2.1. Byte Transfer

Eight-bit bytes are transferred on a bit-parallel basis between a line terminal and the processor. Between the line terminal and the data set, characters are exchanged on a bit-serial basis.

NOTE: Do not be confused by the conflict between byte notation (zero through seven, left to right) and transmission notation (one through parity, right to left). The least significant position, bit 2^0 (byte position 7) is transmitted and received first.

2.2.2. Synchronization

The line terminal precedes each output message with four synchronization (SYN) patterns according to the code specified (EBCDIC or ASCII). During transmission, the terminal inserts a SYN pattern (DLE-SYN in transparent messages) each second. All synchronization patterns are deleted from incoming messages, so that no program involvement is required.

NOTE: The programmer may precede a nontransparent output message with any number of SYN patterns. After either the SOH or STX starting character, however, any SYN patterns inserted by the programmer would be included in the block parity calculation.

2.2.3. Timers

The DCS-1C includes the following hardware timers (clocks) which can be activated as desired at a particular installation:

- Transmit Timer (1 second)

  This timer controls the insertion in the message of the synchronizing character (SYN in nontransparent mode; DLE-SYN in transparent mode) at nominal one-second intervals.

- Receive Timer (3 seconds)

  This timer monitors the receipt of the synchronizing characters and indicates an error condition if the characters are not received within a nominal three-second period. SYN patterns are deleted from input messages whether or not this timer is used. In the case of a dialed access system, this timer should not be active because the normal time required for the establishment of a connection on the switched network would probably exceed three seconds.

- Text Time (0.5, 1, or 5 seconds)

  The time from occurrence of a starting character (SOH or STX) to the next ending character (ETB, ETX, or ENQ) is monitored by this timer to limit message length on a data link. The time can be set to 0.5, 1, or 5 seconds as desired at a particular installation.

2.2.4. Transparency

Both line terminals accommodate EBCDIC- or ASCII-encoded traffic. The F1395-01 Line Terminal also handles transparent blocks. Only text messages may be transmitted under the SEND TRANSPARENT command. Control sequences such as EOT, NAK, or ENQ must be issued with the SEND NORMAL command.

The program establishes an output buffer containing the transparent message. The first character in that buffer must be STX. The last character must be ETX or ETB to avoid a block parity error at the remote terminal.

Once the SEND TRANSPARENT command is accepted, the terminal requests the first output character. If this character is not STX, the command is terminated and a FORMAT error is reported. If the character is STX, the terminal transmits the STX character after a character sequence of SYN SYN SYN SYN DLE. Each byte forwarded from the processor is checked for correspondence with DLE, ETB, ETX, or ENQ. If the character is DLE, a second DLE character is inserted and both are forwarded to the data set. If ETB, ETX, or ENQ is detected, the terminal makes an additional output request to determine whether this is transparent data or a valid ending character. If an additional character is received from storage, the previous pattern is assumed to be data and is transmitted. If the buffer is empty, a valid ending character is assumed. DLE is inserted ahead of this character and the 16-bit cyclic redundancy check (CRC) summation is inserted behind the ending character. All 32 bits are then transmitted and the command is terminated.

An incoming transparent message is recognized by the occurrence of DLE-STX immediately after the last SYN pattern. The terminal deletes the first DLE and begins transferring input characters. As transfers continue, the terminal scans for three sequences: DLE-DLE (the first DLE is discarded), DLE-SYN (both characters are deleted), or DLE followed by ETB, ETX, or ENQ (DLE is discarded and the ending character is transferred). A parity check is then made and normal completion is reported if the parity count calculated matches the CRC pattern received. Parity failure is reported if the check fails or if the ending character was ENQ.

2.2.5. Error Checking

Each block of data transmitted is checked for errors at the receiving station in one of several ways, depending on the code used and the mode of operation. Character parity (odd) is calculated for and checked only on ASCII-coded characters, and only in nontransparent mode. However, this vertical redundancy check (VRC) is employed in both transparent and nontransparent line terminals.

Block parity calculation and checking consist of either a longitudinal redundancy check (LRC) or a cyclic redundancy check (CRC) and begins only after a message starting character has been sensed and forwarded. The calculation continues up to and including a message termination character. Hardware-inserted synchronization characters are not counted.

The LRC and CRC check characters are accumulated at both the sending and receiving terminals during the transmission of a block. The receiving station compares the transmitted character to its own computed character, and reports no error if they are equal.

The type of block parity employed depends upon the code being used (EBCDIC or ASCII), the transmission mode (transparent or nontransparent), and the line terminal selected; see Table 2–1.

CODE	MODE	NONTRANSPARENT LINE TERMINAL CHECKS	TRANSPARENT LINE TERMINAL CHECKS
ASCII	NONTRANSPARENT	VRC and LRC	VRC and CRC-16
	TRANSPARENT	-----	CRC-16
EBCDIC	NONTRANSPARENT	CRC-16	CRC-16
	TRANSPARENT	-----	CRC-16

Table 2–1. Error Checking Application

LRC: This check is used only for nontransparent ASCII-coded text and consists of a modulo-2 noncarry addition of the seven data bits in all characters after the starting character up to and including the termination characterization. An odd VRC check is also applied to the LRC pattern itself. The LRC eight-bit pattern is transmitted directly after the termination character, and on input operations, the pattern is transferred to storage.

CRC-16: This check is used in both nontransparent and transparent line terminals, and consists of a 16-bit cyclic redundancy check which is inserted directly after the termination character. The CRC characters represent the remainder of a division, by a fixed polynomial, of the text which is treated as a binary value. On input operations, the CRC pattern is transferred to storage.

2.2.6. Character Recognition

The line terminal recognizes the following nine control characters in either EBCDIC or ASCII code:

CHARACTER RECOGNIZED	HEXADECIMAL CODE EBCDIC	ASCII
DLE	10	10
ENQ	2D	05
EOT	37	04
ETB	26	17
ETX	03	03
NAK	3D	15
SOH	01	01
STX	02	02
SYN	32	16

On output, the programmed command establishes whether the control character code is to be EBCDIC or ASCII. On input, the terminal makes this determination by examining the SYN patterns.

The terminal's reaction to a given control character is conditioned by its adherence to certain procedural conventions. These conventions are:

- A nontransparent text message begins with either SOH or STX.
- A transparent text message begins with STX.
- Both nontransparent and transparent text messages end with ETB, ETX, or ENQ. ETB or ETX is treated as a normal ending. ENQ implies that the message should be ignored.
- Any input or output message which is not preceded by SOH or STX is a control sequence.
- Control sequences are nontransparent.

The terminal implements these procedural conventions by logically establishing one of two data transfer modes: text or control. The terminal's operating mode is determined by the control characters being transferred from the data link to storage, or from storage to the data link, not by the current I/O command.

Text mode is established when the first non-SYN character from the data line or storage is SOH or STX. Text mode is terminated upon detection of ETB, ETX, or ENQ.

Output control mode is established when the first non-SYN character from storage is neither SOH nor STX. It is terminated upon detection of ENQ, or when the programmed buffer is emptied.

Input control mode is established when the first non-SYN character is other than SOH, STX, or DLE-STX. This mode is terminated upon detection of NAK, DLE, EOT, or ENQ.

2.2.7. Text Mode (Nontransparent)

On output operations, parity accumulation begins with the first character after SOH or STX and continues through the first ending character (ETB, ETX, or ENQ) encountered.

NOTE: An STX character may occur after SOH. If so, the STX character is included in the parity count.

The SYN character inserted by the hardware at one-second intervals is not included in the parity count. Control characters appearing between starting and ending characters are ignored, but are included in the parity count and are transmitted. When one of the three ending characters is detected, it is counted and transmitted. The parity count is then forwarded and the output command is terminated. The terminal can be arranged to transmit a pad character after the parity pattern. (A received pad character is deleted.)

NOTE: The end of the programmed output buffer need not correspond with the ending character, because the command is terminated by the terminal's detection of ETB, ETX, or ENQ. On the other hand, the buffer end may not occur before the ending character – nor can the program expect any characters beyond ETB, ETX, or ENQ to be transmitted.

On input operations, transfers begin with the starting character (SOH or STX), and parity accumulation starts with the next character. SYN characters are deleted and are not counted. Transfers continue until ETB, ETX, or ENQ is detected. The ending character is transferred, followed by the parity bytes, and the input command is terminated. Normal completion is reported if the message ended with ETB or ETX and the parity count was satisfactory. If the message ended in ENQ, or if the received parity count did not match the accumulated count, a block check error is returned.

NOTE: The input buffer must be at least large enough to accommodate all message characters and the parity count (1 byte for ASCII nontransparent, 2 bytes for all CRT patterns). Because the terminal controls termination, the buffer may be of any length beyond the minimum.

2.2.8. Control Mode

On output operations, all control sequences must be nontransparent and may not be preceded by SOH or STX. Odd vertical parity is added to ASCII-coded characters. The command is terminated by the terminal upon detection of ENQ, but the ENQ character is transmitted. If the sequence does not contain ENQ, output continues until the programmed buffer is emptied.

On input operations, all characters of the control sequence are transferred. Vertical parity is not checked. The terminal terminates the input command upon detection of ENQ, NAK, EOT, or DLE. If the detected character is DLE, the DLE and the following character will be transferred to storage.

2.3. OPTIONAL EQUIPMENT

2.3.1. Dialing Adapter

The Dialing Adapter (F1363-00) permits program-controlled establishment of connections on the switched telephone network. The dialing adapter operates in conjunction with an appropriate pulse or tone Bell System Automatic Calling Unit (ACU). The ACU options which should be specified are:

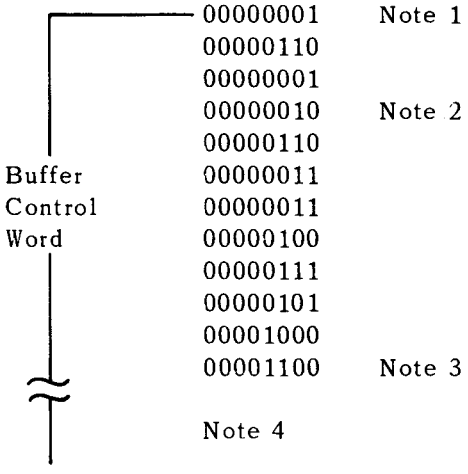
- Detect end-of-answer tone.
- Terminate call, after DSS (data set status) goes on, by way of data set.
- Stop ACR (abandon call and retry) timer, when DSS goes on.

The dialing adapter uses the same address as the output logic of the associated line terminal. A BUSY status would be returned if the program attempted an output data command while automatic dialing was in progress.

To initiate a dial sequence, the program loads a buffer with the telephone number of the station to be called. The dial digits are expressed in binary. For example, the number:

1-612-633-4758

would be buffered, without hyphens, as:



Note 1: In some areas, this digit is required to access the toll networks.

Note 2: The three-digit area code is used only if the called station is outside of the originator's number plan area. If the originator and called station are both in the same area, neither the toll access digit nor the area code must be buffered.

Note 3: This end-of-dial byte tells the dialing adapter that all digits have been presented. This byte is not transferred to the ACU.

Note 4: The dialing adapter does not make additional output requests once the end-of-dial byte is detected. The buffer's end need not necessarily coincide with this last byte. The buffer may not, of course, be terminated before this byte is transferred.

Once the buffer is loaded, the program issues the DIAL command. The dialing adapter accepts the command whether or not the telephone access line is then occupied. It would only reject the DIAL command if an output (data) command was active, or if a previously initiated dial was still in progress. In response to the DIAL command, CRQ (CALL REQUEST) is issued to the ACU and the DATA TERMINAL READY signal is turned on.

Depending upon the ACU employed, issuing the dial digits takes between approximately one (tone) and 15 (pulse) seconds. Depending upon the geographical separation of the originating and called stations, connection requires an additional three to 25 seconds.

The originating station's ACU concludes that the call has been established when it detects an answer tone from the remote station.

NOTE: This answer tone indicates that the call has been completed to a station equipped with a data set. The program should ensure that the correct station was reached by requesting identity.

When the answer tone stops, the ACU transfers control to the data set. The line terminal reports normal (CHANNEL END/DEVICE END status) completion to the program.

The originating station's ACU is equipped with a clock called the abandon call and retry (ACR) timer. The ACR timer is adjustable in increments from seven to 40 seconds. It effectively begins timing the connection after the last dial digit has been passed to the central office. The timer is cleared when the ACU transfers control to the data set. If a timeout occurs, the ACU signals the dialing adapter. The dialing adapter then cancels the attempted call (by dropping CRQ), terminates the command, and notifies the program with a UNIT CHECK status condition.

2.3.2. Polling

The Polling feature (F-1360-00) facilitates use of a 9000 Series processor as a multipoint master station. With this feature, the program is relieved of many of the I/O functions associated with traffic solicitation from tributary stations.

Operation with the polling feature presumes the existence of two or more discretely addressable stations sharing a common data link with the host processor. Upon recognition of the polling sequence, the tributary is expected to respond with a message or a no-traffic indicator. The tributary's message is not preceded by an address and is implicitly directed to the host processor (by definition, a centralized multipoint configuration). The message may be transparent or nontransparent. In lieu of text traffic, the tributary is expected to respond with the no-traffic indicator, EOT.

To initiate a polling cycle, the program loads an output buffer with the polling list. Each list entry identifies one of the multipoint tributaries. Entries are ordered (and repeated, if necessary) to satisfy the scheduling requirements of the system. Entries may be of variable length. Each must contain the tributary station's assigned address, followed by ENQ, followed by an index character. The index character is program-selected to uniquely identify a given polling list entry. For example, assume list entries for stations with the address XZF, GQ, and RKYN. The list, if EBCDIC were specified as the common code for this data link, would be:

	HEXADECIMAL CODE		
Buffer Control Word	E7	X	
	E9	Z	
	C6	F	
	2D	ENQ	
	F1	1	(index)
	C7	G	
	D8	Q	
	2D	ENQ	
	F2	2	(index)
	D9	R	
	D2	K	
	E8	Y	
	D5	N	
	2D	ENQ	
	F3	3	(index)

The program also specifies an input buffer capable of accommodating the anticipated input message, the parity bytes associated with that message, and one byte for the index character.

NOTE: Multipoint systems frequently use EOT to clear the address logic and acknowledgment counters of tributaries. The program should reset all stations, before polling, by issuing an EOT sequence.

The program then issues the POLL command. The line terminal turns on output and input logic, transmits the first list entry (including ENQ), stores the index character, and sets a three-second timer.

- If a no-traffic response (EOT) is received, the next list entry is automatically transmitted. The EOT is not transferred to storage and the program is not notified. The next index character overlays the previous one, and the timer is recycled.

- If text is received, it is transferred to storage according to the conventions for transparent or nontransparent traffic described in 2.2. The first input byte is the index character for that list entry. The POLL command is terminated and the status (normal or error) is reported to the program.

- If a timeout occurs, the command is terminated and a UNIT CHECK status is reported to the program. The index character for that list entry is returned in sense byte 3 (see 3.2).

2.3.3. Station Select

The Station Select feature (F1361-00) enables detection of a single, user-specified character by the line terminal. This character would, presumably, be the first address character of a multipoint tributary station.

To enable the detection logic, the program issues the ADPREP command. Once this command is accepted, the line terminal monitors the multipoint data link looking for a reset character, EOT. When an EOT has been detected, the terminal begins examining traffic for its wired address. The terminal ignores bit 2^5 and the parity bit.

NOTE: Bit 2^5 and the parity bit are ignored because binary synchronous communication procedures dictate that bit 2^5 is program specified, by the host system, to distinguish between polling and selection.

If that station's address is detected, input transfers begin. The first address character, the one recognized by the terminal, is transferred. Input continues until an ending character (ETB, ETX, ENQ) has been detected and transferred. The ADPREP command is then terminated.

Normal termination is reported if the input transfer ended with an ENQ character not preceded by SOH or STX. If SOH or STX follows the station's address, parity accumulation begins. Normal termination (CHANNEL END/DEVICE END status) is reported if this text was found to be error-free and terminated by ETB or ETX. A UNIT CHECK status indicates either parity failure or an ENQ termination of a text message.

If the station senses SOH or STX before its address, it waits for the next EOT sequence before again looking for its address. Under these circumstances the ADPREP command is not terminated.

2.3.4. Application Options

There are three additional DCS-1C features, none of which have any direct effect upon programming:

- Wideband Interface (F1395-01)
- I/O Buffer (F1359-00): temporary storage for four input and four output characters.
- Synchronous Timing Assembly (F1011-00): an internal clock for use in those systems where the data set does not furnish the requisite clock.

3. COMMAND, STATUS, AND SENSE BYTES

3.1. I/O COMMANDS

A total of 16 input/output commands comprise the instruction set to which the DCS-1C responds. Nine of these commands are for use in controlling the input side of the line terminal of the data communications subsystem, while four are concerned with the output side of the line terminal. Three of the commands apply to either input or output operations.

A complete descriptive summary of all the I/O commands is given in Table 3–1. The commands are listed in alphabetical order and indicate which side of the line terminal (input or output) is affected by each one.

COMMAND	DESCRIPTION
Mnemonic: ADPREP EBCDIC Code: 26 ASCII Code: 26 Addressed to: Input Features Required: Station Select (F1361-00)	Enables receipt of addressed control or text messages. See 2.3.3.
Mnemonic: DCS TEST EBCDIC Code: 0B ASCII Code: 0B Addressed to: Input Features Required: None	Disables connection between interface and data set and directs output to input logic of terminal. Input terminal generates CHANNEL END/DEVICE END status.
Mnemonic: DIAL EBCDIC Code: 05 ASCII Code: 05 Addressed to: Output Features Required: Dialing Adapter (F1363-00)	Initiates program-controlled establishment of dialed connections. See 2.3.1.
Mnemonic: DISCONNECT EBCDIC Code: 13 ASCII Code: 13 Addressed to: Input Features Required: None	Terminates (hangs-up) dial-connected call, at DCS-1C end of line, by dropping the DATA TERMINAL READY lead between the interface and the data set.
Mnemonic: END TEST EBCDIC Code: 0F ASCII Code: 0F Addressed to: Input Features Required: None	Terminates DCS TEST or LOCAL TEST command. Input terminal generates CHANNEL END/DEVICE END status.

Table 3–1. I/O Commands (Part 1 of 3)

COMMAND	DESCRIPTION
Mnemonic: LOCAL TEST EBCDIC Code: 07 ASCII Code: 07 Addressed to: Input Features Required: None	Enables loop-back test from output logic, through data set, and back to input. Input terminal generates CHANNEL END/DEVICE END status. (Data set must have loop-back capability.)
Mnemonic: LOOK FOR SYN EBCDIC Code: 06 ASCII Code: 06 Addressed to: Input Features Required: None	Causes the communication interface to turn on the DATA TERMINAL READY signal, after which the data link is monitored for two successive SYN characters. Once detected, input data transfers begin with the first non-SYN character and continue until an ending character (ENQ, ETX, or ETB) is detected and transferred. Block parity characters, if present, are transferred after the ending character and before termination. Data transfer could also begin in the middle of a message, if an embedded SYN SYN sequence was detected.
Mnemonic: NEW SYN EBCDIC Code: 0A ASCII Code: 0A Addressed to: Input Features Required: None	Overrides and effectively recycles a current LOOK FOR SYN command. If another input command was active, a BUSY status is returned. The NEW SYN command causes the station to drop character phase and begin looking for two successive SYN characters, after which the command performs the same functions as the LOOK FOR SYN command.
Mnemonic: POLL EBCDIC Code: 15 ASCII Code: 21 Addressed to: Output Features Required: Polling (F1360-00)	Turns on input and polls multipoint tributary stations. See 2.3.2.
Mnemonic: SEARCH EBCDIC Code: 22 ASCII Code: 22 Addressed to: Input Features Required: Station Select (F1361-00)	Terminal receives all traffic addressed to it, as well as all control (nontext) traffic occurring on a multipoint circuit. Command is terminated upon detection of EOT.

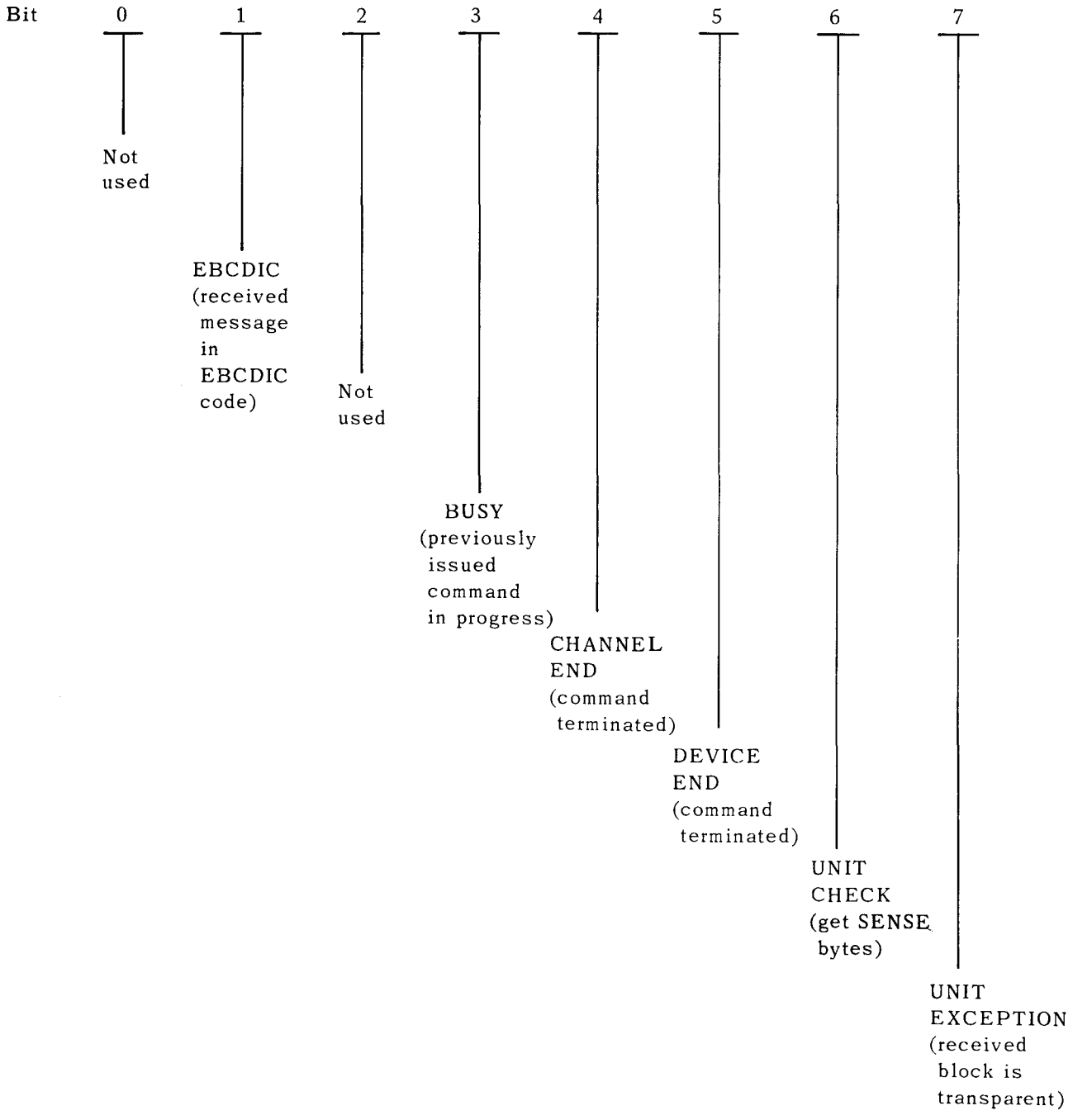
Table 3–1. I/O Commands (Part 2 of 3)

COMMAND	DESCRIPTION
Mnemonic: SEND NORMAL EBCDIC Code: 31 ASCII Code: 01 Addressed to: Output Features Required: None	Used to send nontransparent text and all control sequences. See 2.2.4, and 2.2.7.
Mnemonic: SEND TRANSPARENT EBCDIC Code: 71 ASCII Code: 41 Addressed to: Output Features Required: Transparent terminal	Used to send transparent text. See 2.2.4.
Mnemonic: SENSE EBCDIC Code: 04 ASCII Code: 04 Addressed to: Input or Output Features Required: None	Causes the terminal to return an all-zeroes status byte, normally followed by two sense bytes. (The POLL command timeout causes three sense bytes to be generated.) (Used on 9400 Systems – see TEST I O.)
Mnemonic: SET DTR EBCDIC Code: 10 ASCII Code: 10 Addressed to: Input Features Required: None	Causes the interface to set the DATA TERMINAL READY lead to the data set, in order to receive dialed calls without also having the terminal's input logic on.
Mnemonic: TEST I/O EBCDIC Code: 00 ASCII Code: 00 Addressed to: Input or Output Features Required: None	Causes the addressed terminal to return status information. (Used on 9200 and 9300 Systems.)
Mnemonic: TURN OFF EBCDIC Code: 03 ASCII Code: 03 Addressed to: Input or Output Features Required: None	Causes termination of any active command and generates a CHANNEL END, DEVICE END status.

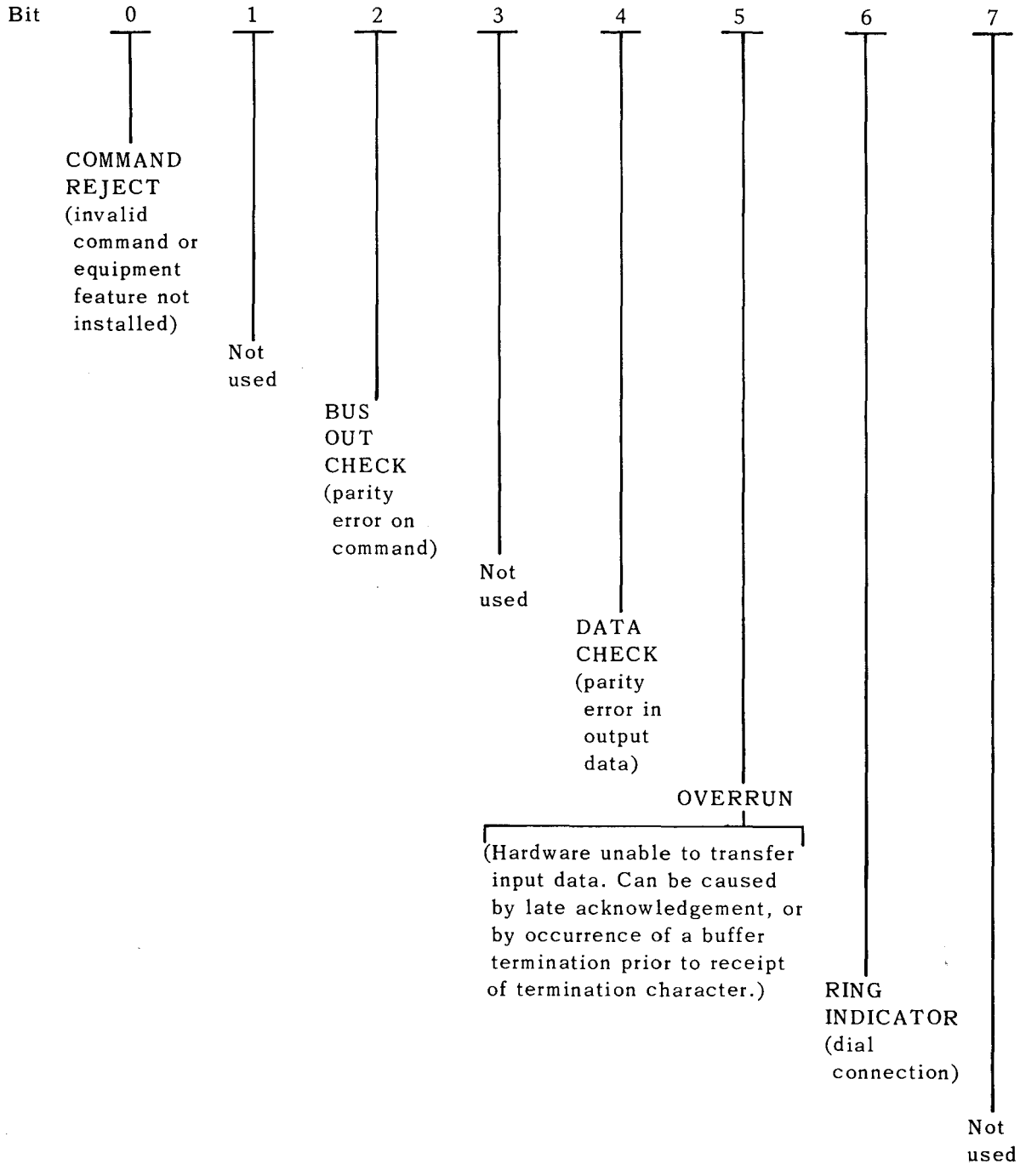
Table 3–1. I/O Commands (Part 3 of 3)

3.2. STATUS AND SENSE BYTES

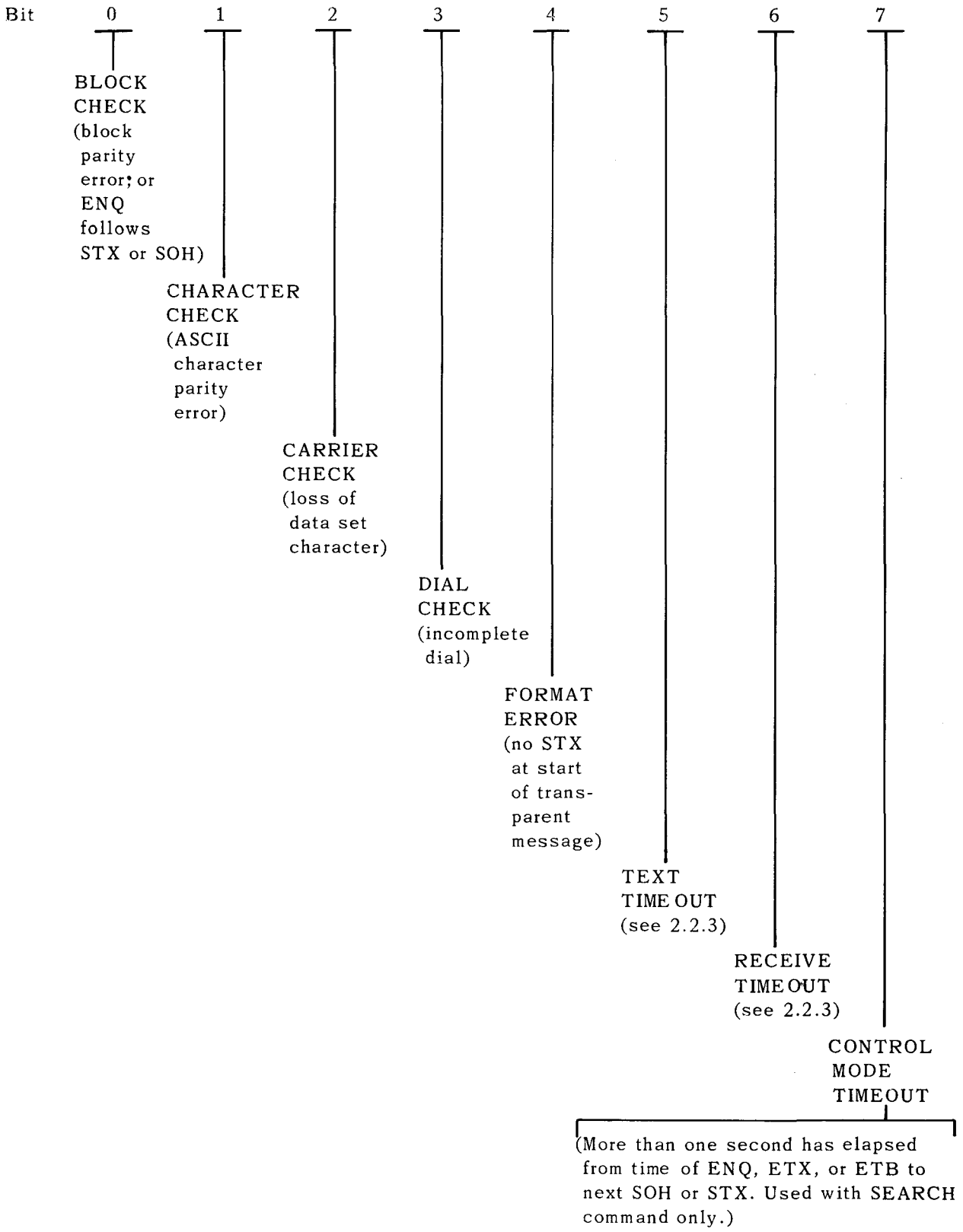
A status byte is sent from the DCS-1C to the processor to furnish information about the acceptance of a command, status of a unit, or performance of a command. The bit positions of the status byte are set as follows:



Two or three sense bytes, which contain information concerning an error condition, are sent to the processor by the DCS-1C after receipt of a SENSE (9400 System) or TEST I/O (9200/9300 System) command from the processor. The bit positions of the first sense byte are set as follows:



The bit positions of the second sense byte are set as follows:



Sense byte 3 is returned only when a timeout occurs in connection with a POLL command (see 2.3.2). In this case, the sense byte contains the index character of the last station polled.

APPENDIX A. CHARACTER CODE CHARTS

In this appendix, two tabulations are given which summarize the various requirements for recognizing control characters (Table A–1), and the EBCDIC and ASCII character assignments (Table A–2).

	HEX CODE		TRANSPARENT		NONTRANSPARENT			
			TEXT		TEXT		CONTROL	
	EBC	ASC	INPUT	OUTPUT	INPUT	OUTPUT	INPUT	OUTPUT
DLE	10	10	Deleted except for second DLE of DLE-DLE transparent sequence	Added in front of STX, DLE, ETX, ETB or ENQ.	Ignored		Terminates. DLE and next input character are transferred.	Ignored (Buffer must terminate.)
ENQ	2D	05	Terminate if found as DLE-ENQ, UNIT CHECK status reported.	Terminates when found as last character in buffer.	Terminates. UNIT CHECK status reported.	Terminates with normal status.		
EOT	37	04	Ignored (Not expected in text messages)				Terminates	Ignored (Buffer must terminate.)
ETB	26	17	Terminates if found as DLE – ETB/X	Terminates when found as last character in buffer.	Terminates		Not permitted in control sequences	
ETX	03	03						
NAK	3D	15	Ignored				Terminates	Ignored (Buffer must terminate.)
SYN	32	16	SYN alone is ignored. DLE-SYN is deleted.	4 SYN sent before text. DLE-SYN inserted each 1 second interval.	All SYN's are deleted.	4 SYN sent before text. SYN-SYN inserted each 1 second interval.	All SYN's are deleted.	4 SYN sent before control sequence.
SOH	01	01	Not permitted as starting character. May occur as transparent text.		Valid starting character. Parity count begins with next character.		Not permitted in control sequences.	
STX	02	02	Mandatory starting character. (Terminal inserts leading DLE.)		Included in parity count if preceded by SOH. If not preceded by SOH, parity count begins with next character.		Not permitted in control sequences.	

Table A–1. Control Character Recognition Summary

MSD / LSD	0		1		2		3		4		5		6		7		8		9		A		B		C		D		E		F	
	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC	ASC	EBC
0	NUL	NUL	DLE	DLE	SP	DS	0		@	SP	P	&	`	-	p																	0
1	SOH	SOH	DC1	DC1	!	SOS	1		A		Q		a		q			a		j						A		J				1
2	STX	STX	DC2	DC2	"	FS	2	SYN	B		R		b		r			b		k		s				B		K		S		2
3	ETX	ETX	DC3	DC3	#		3		C		S		c		s			c		l		t				C		L		T		3
4	EOT	PF	DC4	RES	$	BYP	4	PN	D		T		d		t			d		m		u				D		M		U		4
5	ENQ	HT	NAK	NL	%	LF	5	RS	E		U		e		u			e		n		v				E		N		V		5
6	ACK	LC	SYN	BS	&	ETB	6	UC	F		V		f		v			f		o		w				F		O		W		6
7	BEL	DEL	ETB	IL	'	ESC	7	EOT	G		W		g		w			g		p		x				G		P		X		7
8	BS		CAN	CAN	(		8		H		X		h		x			h		q		y				H		Q		Y		8
9	HT		EM	EM	)		9		I		Y		i		y			i		r		z				I		R		Z		9
A	LF	SMM	SUB	CC	*	SM	:		J	¢	Z	:	j		z	:																
B	VT	VT	ESC		+		;		K	.	[	$	k	,	{	#																
C	FF	FF	FS	IFS	,		<	DC4	L	<	\	*	l	%	¬	@																
D	CR	CR	GS	IGS	-	ENQ	=	NAK	M	(	]	)	m	_	}	'																
E	SO	SO	RS	IRS	.	ACK	>		N	+	^	;	n	>	~	=																
F	SI	SI	US	IUS	/	BEL	?	SUB	O	|	¬	¬	o	?	DEL	"																

Table A–2. EBCDIC and ASCII Character Assignments (Part 1 of 2)

ALPHANUMERICS

	ASC	EBC
A	41	C1
B	42	C2
C	43	C3
D	44	C4
E	45	C5
F	46	C6
G	47	C7
H	48	C8
I	49	C9
J	4A	D1
K	4B	D2
L	4C	D3
M	4D	D4
N	4E	D5
O	4F	D6
P	50	D7
Q	51	D8
R	52	D9
S	53	E2
T	54	E3
U	55	E4
V	56	E5
W	57	E6
X	58	E7
Y	59	E8
Z	5A	E9

	ASC	EBC
a	61	81
b	62	82
c	63	83
d	64	84
e	65	85
f	66	86
g	67	87
h	68	88
i	69	89
j	6A	91
k	6B	92
l	6C	93
m	6D	94
n	6E	95
o	6F	96
p	70	97
q	71	98
r	72	99
s	73	A2
t	74	A3
u	75	A4
v	76	A5
w	77	A6
x	78	A7
y	79	A8
z	7A	A9

	ASC	EBC
0	30	F0
1	31	F1
2	32	F2
3	33	F3
4	34	F4
5	35	F5
6	36	F6
7	37	F7
8	38	F8
9	39	F9

CONTROLS

	ASC	EBC
ACK	06	2E
BEL	07	2F
BS	08	16
BYP	–	24
CAN	18	18
CC	–	1A
CR	0D	0D
DC1	11	11
DC2	12	12
DC3	13	13
DC4	14	3C
DEL	7F	07
DLE	10	10
DS	–	20
EM	19	19
ENQ	05	2D
EOB	–	26
EOT	04	37
ESC	1B	27
ETB	17	26
ETX	03	03
FF	0C	0C
FS	1C	22
(I) FS	1C	1C
(I) GS	1D	1D
HT	09	05

	ASC	EBC
IL	–	17
LC	–	06
LF	0A	25
NAK	15	3D
NL	–	15
NUL	00	00
PF	–	04
PN	–	34
PRE	–	27
RES	–	14
RS	–	35
(I) RS	1E	1E
SI	0F	0F
SM	–	2A
SMM	–	0A
SO	0E	0E
SOH	01	01
SOS	–	21
SP	20	40
STX	02	02
SUB	1A	3F
SYN	16	32
UC	–	36
(I) US	1F	1F
VT	0B	0B

SYMBOLS

	ASC	EBC
!	21	5A
"	22	7F
#	23	7B
$	24	5B
%	25	6C
&	26	50
'	27	7D
(	28	4D
)	29	5D
*	2A	5C
+	2B	4E
,	2C	6B
–	2D	6D
.	2E	4B
/	2F	61
:	3A	7A
;	3B	5E
<	3C	4C
=	3D	7E
>	3E	6E
?	3F	6F
@	40	–
[	5B	–
\	5C	–
]	5D	–
^	5E	–

	ASC	EBC
_	5F	60
`	60	–
{	7B	–
|	7C	5F
}	7D	–
~	7E	–
¢	–	4A
|	–	4F
@	–	7C

Table A–2. EBCDIC and ASCII Character Assignments (Part 2 of 2)

APPENDIX B. GLOSSARY

ACK 0, ACK 1 (affirmative acknowledgment)

These replies (DLE sequences), indicate that the previous transmission block was accepted by the receiver and that it is ready to accept the next block of the transmission. Use of ACK 0 and ACK 1 alternately provides sequential checking control for a series of replies. ACK 0 is also a ready reply to a station selection (multipoint), or to an establishment sequence (line bid) in point-to-point operation.

ASCII (American Standard Code for Information Interchange)

A standardized seven-bit information character code.

Control station

The station in a multipoint system that controls network traffic by means of polling and selection. On a centralized multipoint network, tributary stations can communicate only with the control station when polled or selected by the control station.

DLE (data link escape)

A control character used exclusively to provide supplementary line control signals (control character sequences or DLE sequences). These are two-character sequences where the first character is DLE. The second character varies according to the function desired and the code used.

EBCDIC (Extended Binary Coded Decimal Information Code)

A standardized eight-bit information character code.

ENQ (inquiry)

Used as a request for a response to obtain identification and/or an indication of station status. Transmitted as part of an establishment sequence (line bit) in point-to-point operation, and as the final character of a selection or polling sequence in multipoint operation.

EOT (end of transmission)

Indicates the end of a transmission, which may include one or more messages, and resets all stations on the line to control mode (unless it erroneously occurs within a transmission block). EOT is also transmitted as a negative response to a polling sequence.

ETB (end of transmission block)

Terminates a group of characters (transmission block) started with SOH or STX, and indicates that the message continues with a following block. A message may contain one or more transmission blocks ending with ETB. The block check character (LRC or CRC) is sent immediately following ETB. ETB requires a reply indicating the receiving station's status.

ETX (end of text)

Indicates the end of a message. If multiple transmission blocks are contained in a message, ETX terminates the last block of the message. (ETB is used to terminate preceding blocks.) The block check character is sent immediately following ETX. ETX requires a reply indicating the receiving station's status.

NAK (negative acknowledgment)

Indicates that the previous transmission block was in error and the receiver is ready to accept a retransmission of the erroneous block. NAK is also the "not-ready" reply to a station selection (multipoint) or an establishment sequence (line bid) in point-to-point operation.

Remote station

Synonymous with tributary station on a multipoint network. A station that can be called by the central station, or can call the central station if it has a message to send on a point-to-point switched network.

RVI (reverse interrupt)

A control character sequence (DLE sequence) sent by a receiving station instead of ACK 1 or ACK 0 to request early termination of the transmission in progress.

SOH (start of heading)

Precedes a block of heading characters.

STX (start of text)

Precedes a block of text characters.

SYN (synchronous idle)

Character used as a time fill in the absence of any data or control character to maintain synchronization. The sequence of two contiguous SYN's is used to establish synchronization (character phase) following each line turnaround.

Tributary station

A station, other than the control station, on a centralized multipoint system which can communicate only with the control statement when polled or selected by the control station.

TTD (Temporary text delay)

A control character sequence (STX – ENQ) sent by a transmitting station to either indicate a delay in transmission or to initiate an abort of the transmission in progress.

WACK (wait before transmit positive acknowledgment)

This DLE sequence is sent by a receiving station to indicate that it is temporarily not ready to receive.


UNIVAC
SPERRY RAND