Honeywell

SERIES 200

HARDWARE

TYPE 281-2BSC COMMUNICATION CONTROL AND TYPE 285-2BSC COMMUNICATION ADAPTER

Honeywell

SERIES 200

TYPE 281-2BSC COMMUNICATION CONTROL AND TYPE 285-2BSC COMMUNICATION ADAPTER

SUBJECT:

Hardware Description, Functional Characteristics, Operation, and Programming Information for the Type 281-2BSC Communication Control and Type 285-2BSC Communication Adapter.

SPECIAL INSTRUCTIONS:

This manual is one of a series of hardware publications for communication controls and adapters. It should be used in conjunction with the manual, *Type 286-6 and 286-7 Message-Mode Multi-Line Communication Controls,* Order No. G86, and the appropriate programmers' reference manual.

DATE:

March 31, 1971

FILE NUMBER:

113.0005.1100.0-F11*

*Underscoring denotes Order Number.

PREFACE

This manual provides the Honeywell Series 200 user with general information, functional characteristics, operating information, and programmimg information concerning the Type 281-2BSC Communication Control and the Type 285-2BSC Communication Adapter. Additional information concerning the central processors and other peripheral devices will be found in other Honeywell Series 200 publications.

#F11

TABLE OF CONTENTS

TABLE OF CONTENTS (cont)

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION I
INTRODUCTION

GENERAL DESCRIPTION

The Honeywell communication devices in this manual allow a Series 200 central processor[1] to communicate with remotely-located terminal equipment using binary synchronous communications (BSC). Data transfer can be initiated by the central processor or by the remote terminal.

The Type 281-2BSC Single-Channel Communication Control controls the flow of data between the central processor and the remote terminal over voice-grade telephone lines. The lines can be private (lines dedicated to a specific application) or switched (a telephone network where the connection between the local and remote terminals is established, as required, by dialing or some other means of switching). The Type 285-2BSC Communication Adapter allows a Type 286-6 or -7 Message-Mode Multi-line Communication Control to be used in lieu of the Type 281-2BSC in applications where a large number of communication channels are required.

Figure 1-1 illustrates typical applications of the Type 281-2BSC and 285-2BSC. As shown, both units are connected to the line by means of internal sync data sets, and the same type of data set is required at both ends of the line. Data communication is in the message block mode.

LINE TRANSMISSION CHARACTERISTICS

Data transmission is two-way, nonsimultaneous (half-duplex). Clock pulses for bit stream synchronization are supplied by the data set. Character synchronization is achieved through recognition of two successive synchronization characters.

INTERFACE

The Type 281-2BSC Communication Control is connected to a standard Series 200 input/output channel by means of standard peripheral interface logic. Each communication control occupies one standard Series 200 logic drawer.

The Type 285-2BSC Communication Adapter is connected to a Type 286-6 or -7 Message-Mode Multi-line Communication Control by way of standard cables and Series 200 interface packages. Two communication adapters occupy one standard Series 200 logic drawer.

---

[1] A Series 200 Control Unit Adapter is required for attachment to a Type 121 Central Processor; these communication devices cannot be used with a Type 111, 121-0, 201, or 201-1 processor.

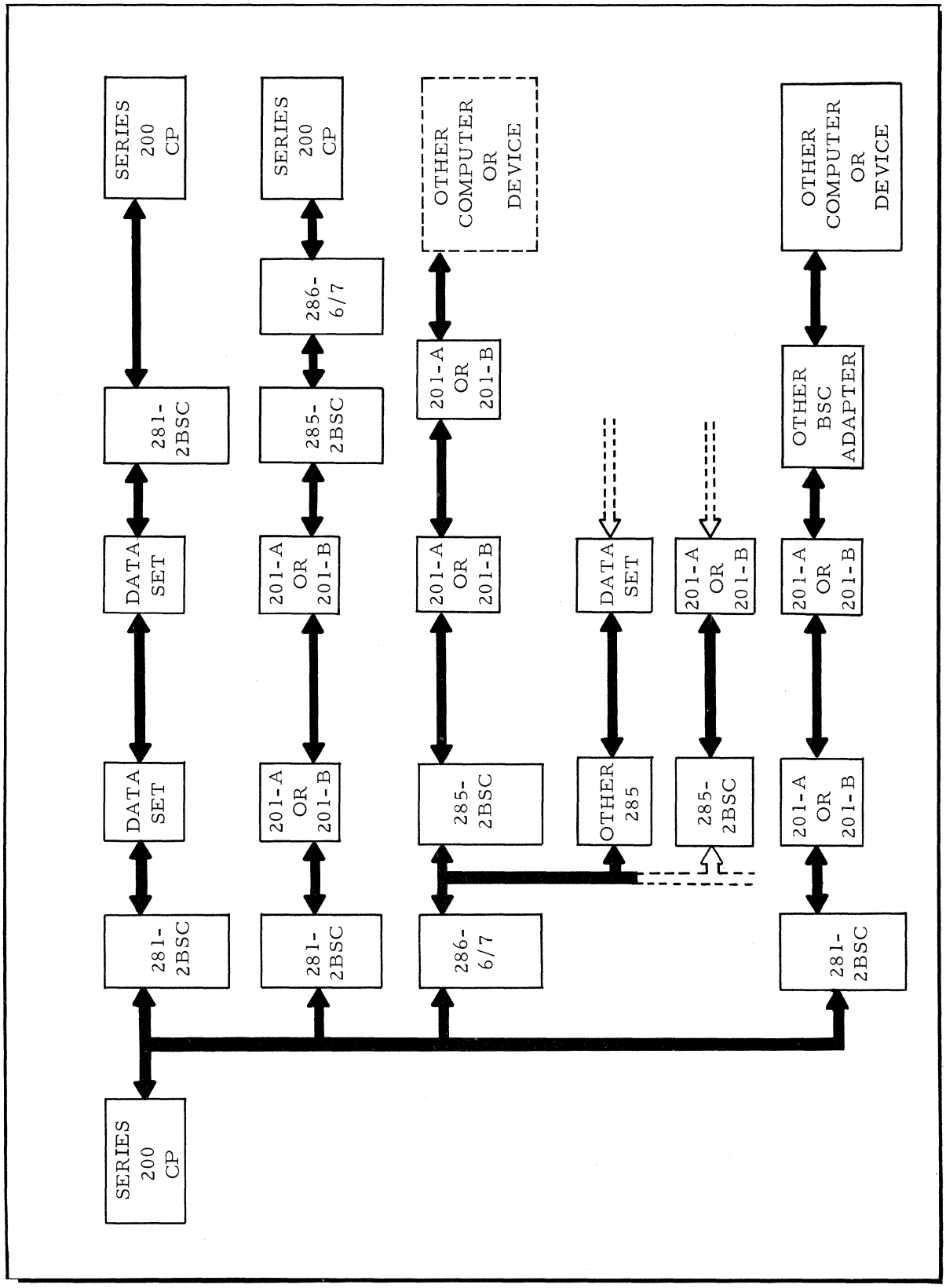


Figure 1-1. System Configuration

The communication controls and communication adapters interface with the data set in accordance with Electronic Industry Association RS-232C.

REMOTE TERMINALS

The Type 281-2BSC and 285-2BSC will operate with three types of remote terminals:

1. Series 200 central processor[1] equipped with a compatible BSC communication control and a compatible data set.
2. Keytape[2] 760/960 High-Speed Communicators.
3. Any computer or device that is equipped with a compatible data set and that satisfies the following requirements:
   a. The same code set must be used at both ends of the transmission line.
   b. BSC message format must be used at both ends of the transmission line.
   c. The same error checking procedures and conventions, as described later in this manual, must be used at both ends of the line.
   d. Operation must be in the continuous bit stream mode, with synchronization established for each text block.

COMMUNICATION CODES

Depending on the intended use established at the time of order, the EBCDIC (Extended Binary Coded Decimal Interchange Code) with 256 character assignments, the ASCII (American Standard Code for Information Interchange) with 128 character assignments, or the Six-Bit Transcode with 64 character assignments, can be selected as the communication code.

Also, on special order, the communication control or adapter can be made to operate with any 6- or 8-bit code. However, if a parity bit is required, bit 8 (the most significant bit) must be used for this purpose, and parity must be odd.

Regardless of the code specified, the communication control or adapter is capable of transmitting and receiving in a code transparency mode to allow the transmission of data that does not fall within the code set (pure binary, machine-language instruction, etc.).

LINE SPEEDS

Private line operation is normally at 2400 bits per second; switched line operation is normally at 2000 bits per second. When the 285-2BSC adapter is equipped with Feature 1072, 1073, 1074, 1075, or 1076, operating speed is increased to 3600, 4800, 5400, 7200, or 9600 bps, respectively. (The data sets and line must be capable of operation at the appropriate speed.)

[1] Interrupt capabilities must be present.

[2] Trademark.

TEST UNITS

All Type 281-2BSC Communication Controls and Type 285-2BSC Communication Adapters are supplied with a test unit, which is housed in the same logic drawer. The test unit can be used for loop-back testing, and is activated by the appropriate PCB and PDT instructions described later in this manual. The test unit simulates the data set timing signals to allow functional testing.

SECTION II
FUNCTIONAL CHARACTERISTICS

TRANSMISSION MODE

The Type 281-2BSC Communication Control and Type 285-2BSC Communication Adapter operate half-duplex in the message block mode. This means that after the initial block of test is transmitted, each ensuing block of text will be transmitted only after a response for the previous block has been received from the distant terminal. The text blocks may be of variable length; the maximum block size is determined only by available memory size and data storage capabilities at the terminals, except when transmitting in the transparent mode; then the block length cannot exceed 4096 line characters. (This restriction applies only to transmission.)

MESSAGE BLOCK FORMAT

The general message block format for BSC communication is as follows:

Synchronization Characters	Start of Text	Data	End of Text	BCC Character

The variety of legal message block formats is shown in Figure 2-1. Table 2-1 shows the bit structure of the BSC line control characters shown in Figure 2-1, as well as the octal representation that is to be stored in memory.

COMMUNICATION CONTROL CHARACTERS

The 281-2BSC and 285-2BSC are capable of character recognition and decoding. The various control characters are summarized in Table 2-2 and described in the following paragraphs.

Synchronization (SYN) Characters

The synchronization (SYN) character is used to establish and maintain synchronism between the two ends of the communication line. Each block of characters must be preceded by at least three SYN characters. The 281-2BSC or 285-2BSC must recognize two successive SYN characters in order to synchronize with the remote terminal. Received SYN characters are not transferred to the central processor, nor are they included in LRC or CRC computation.

During communication the transmitting communication control automatically inserts two successive SYN characters into the data stream at one-second intervals to maintain character synchronization. Also, if the communication control is awaiting data from the central processor

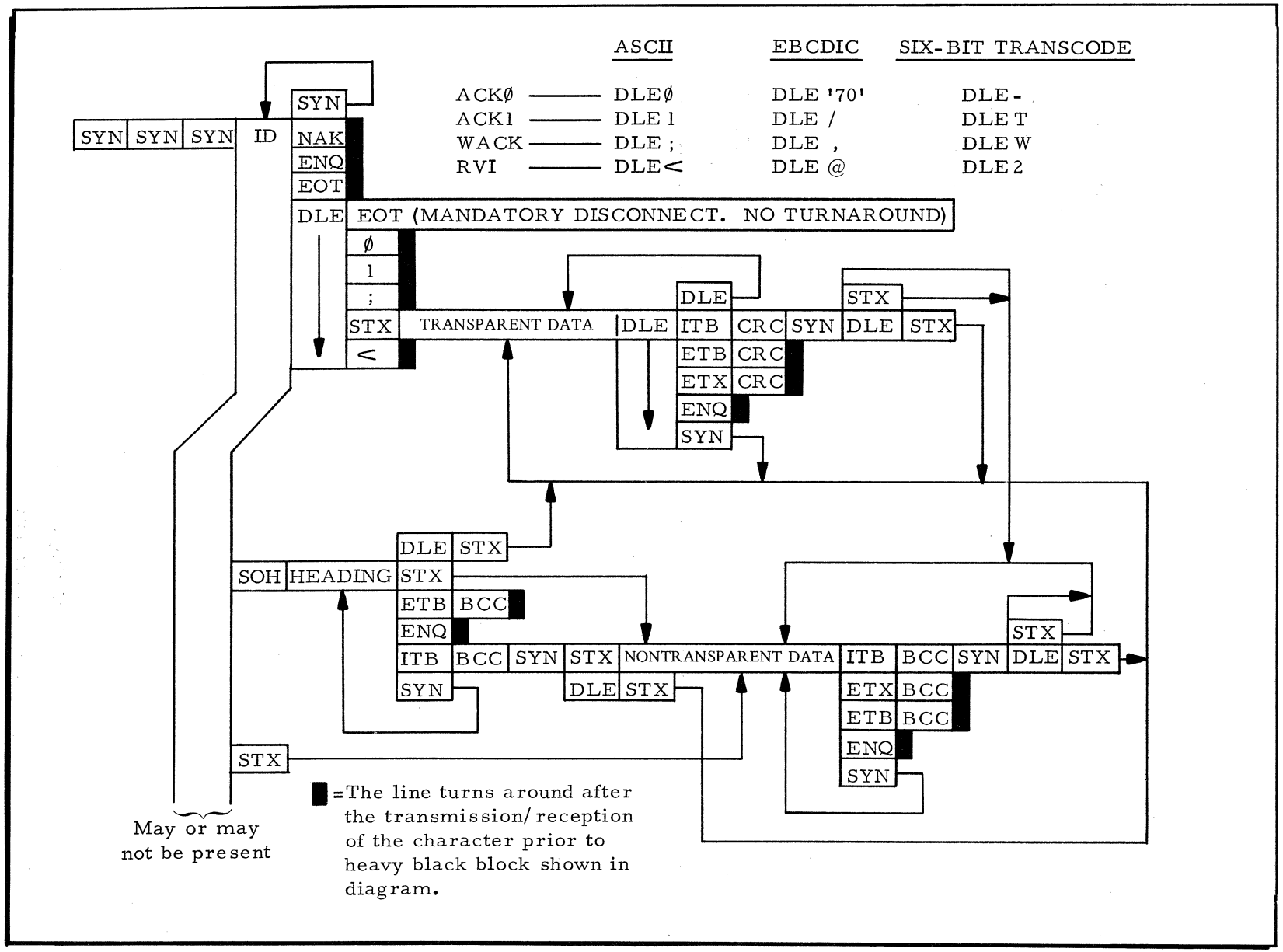


Figure 2-1. Binary Synchronous Communication Formats

Table 2-1. BSC Line Control Characters

Character	ASCII		EBCDIC		Six-Bit Transcode	
	Binary	Octal #1	Binary	Octal #1	Binary	Octal #1
SYN	00010110	0026	00110010	0062	111010	72
NAK	00010101	0025	00111101	0075	111101	75
ENQ	10000101	0205	00101101	0055	101101	55
EOT	00000100	0004	00110111	0067	011110	36
DLE	00010000	0020	00010000	0020	011111	37
Ø #2	10110000	0260				
1 #2	00110001	0061				
; #2	00111011	0073				
'70' #2 #3			01110000	0160		
/ #2			01100001	0141		
- #2					100000	40
T #2					100011	43
W #2					100110	46
< #2	10111100	0274				
@ #2			01111100	0174		
2 #2					110010	62
STX	00000010	0002	00000010	0002	001010	12
ITB	00011111	0037	00011111	0037	011101	35
ETB	10010111	0227	00100110	0046	001111	17
ETX	10000011	0203	00000011	0003	101110	56
SOH	00000001	0001	00000001	0001	000000	00
, #2			01101011	0153		

NOTES: 1. Eight-bit codes are stored in two memory locations and six-bit codes are stored in one memory location. Each memory location contains a two-digit octal number.

2^7	2^0
000/087	654/321
1st memory position	2nd memory position

2. These characters are used only in the two character sequences below:

	ASCII	EBCDIC	Six-Bit Transcode
ACKØ	DLE Ø	DLE '70'	DLE -
ACK1	DLE 1	DLE /	DLE T
WACK	DLE ;	DLE ,	DLE W
RVI	DLE <	DLE @	DLE 2

3. '70' is a hexadecimal number representation of an unassigned EBCDIC code character.

Table 2-2. Control Character Summary

Character	Name	No. Required	Action of Communication Control[1]
SYN	Synchronization Character	3 (minimum)	(R) Used to achieve and maintain synchronization. Not transferred to the central processor and not entered into BCC computation.
SOH	Start-of-Heading	1	(R) If synchronization is established, SOH is tranferred to the central processor with all of the following characters. (T and R) If first SOH or STX of a message, BCC accumulation is reset, and is restarted with the following character.
STX	Start-of-Text	1	(R) This and all following characters are transferred to the central processor, if synchronization is already established. (T and R) BCC accumulation is reset, and is restarted with the following character, if this is the first STX or SOH of the message.
ETX[2]	End-of-Text	1	(T and R) Causes an interrupt. (T) BCC accumulation is transferred immediately following ETX. (R) BCC check is made with character(s) received immediately following ETX.
EOT[2]	End-of-Transmission	1	(T and R) Causes an interrupt. (R) Transferred to the central processor.
ENQ[2]	Enquiry	1	(T and R) Causes an interrupt. (R) Transferred to the central processor.
NAK[2]	Negative Acknowledgment	1	(T and R) Causes an interrupt. (R) Transferred to the central processor.
ACKØ[2] ACK1[2]	Positive Acknowledgment	1	(T and R) Causes an interrupt. (R) Transferred to the central processor.
ITB	Intermediate Block	1	(R) Causes an interrupt in the transparent mode only. Transferred to the central processor.
ETB[2]	End-of-Transmission Block	1	Same as ETX. Both ETX and ETB are transferred to the central processor.

Table 2-2 (cont). Control Character Summary

Character	Name	No. Required	Action of Communication Control[1]
DLE	Data Link Escape	1 or 2	Provides auxiliary control information, depending on the character that follows.
WACK[2]	Wait before Transmission	1	(T and R) Causes an interrupt. (R) Transferred to the central processor.

[1] T = Transmission, R = Reception.

[2] Reception or transmission of these characters causes a line turnaround.

while transmitting, it sends SYN characters as a time-fill to prevent the receiving communication control from interpreting the lack of data as a transmission that has been interrupted prematurely.

Start-of-Heading (SOH) Character

The Start-of-Heading (SOH) character precedes a block of heading characters, and signals its beginning. The heading consists of a sequence of characters which constitutes the auxiliary information (such as routing and priority) necessary for the communication of the text portion of the message. If synchronization has been established, the SOH character and all subsequent heading, control, and data characters are transferred to memory (until an end character is recognized).

Start-of-Text (STX) Character

The Start-of-Text (STX) character precedes a block of text characters and signals the beginning of a data block. The data block constitutes that portion of the message that is treated as an entity, and transmitted through to its ultimate destination without change. If synchronization has been achieved, the STX character and all subsequent data characters are transferred to memory (until the recognition of an end character, which is also transferred to memory). Such a sequence is called "text."

The STX character can also be used to terminate a sequence of heading characters started by an SOH.

End-of-Text (ETX) Character

The End-of-Text (ETX) character terminates a data block. A block check character is transmitted immediately following the ETX character, and a status reply (ACK0, ACK1, NAK, or WACK) is expected from the receiving terminal. The ETX character is transferred to memory.

The ETX character is only used to end the last block of text in a message. All other blocks are ended by an ETB character (see the following paragraph).

End-of-Transmission Block (ETB) Character

The ETB character signals the end of a transmission block (heading or text) to which a status reply (ACKØ, ACK1, NAK, or WACK) is expected from the receiving terminal. The blocking structure is not necessarily related to the processing format. The block check character is sent immediately following the ETB character. The ETB is transferred to memory.

End-of-Transmission (EOT) Character

The EOT character signals the conclusion of a transmission, which may have contained one or more messages, consisting of one or more text blocks and any associated heading and responses. The EOT character is supplemented by a DLE-EOT sequence, known as the "mandatory disconnect," in switched-network operation.

Enquiry (ENQ) Character

The ENQ character is used in requesting a response, such as the identification or status (ACKØ, NAK, etc.), from the receiving remote terminal. The ENQ character can also be used to obtain a repeat transmission of a reply, if transmission was not received when expected or if it was garbled. When used in text, the ENQ character signifies the abnormal end of the text being transmitted. This is known as the abort sequence.

Positive Acknowledgment Sequences (ACKØ and ACK1)

Binary synchronous communication (BSC) employs two positive acknowledgment sequences referred to as ACKØ and ACK1. These are the 2-character sequences defined in note 2 of Table 2-1. The responses are used alternately in response to text transmission. ACK1 is the response to the first text transmission and to successive odd-numbered blocks during the transmission. ACKØ is the response to the second text block and to all even-numbered blocks.

These sequences indicate that the previous block was accepted and that the receiver is ready to accept the next block of transmission. Alternating positive acknowledgments (DLE sequences) must be transmitted.

Negative Acknowledgment (NAK) Character

The NAK character indicates that the previous block was unacceptable (an error was detected) and that the receiver is ready to accept a retransmission of that block. It is also the "not ready" reply to a terminal selection.

Data Link Escape (DLE) Character

The DLE character changes the meaning of a limited number of contiguous following characters. It is used exclusively to provide supplementary line control signals, which include Wait Before Transmitting (WACK), affirmative acknowledgments (ACKØ and ACK1), and start of transparent text (the sequences DLE STX and DLE ETX initiate and terminate transparent text). DLE EOT is used as a mandatory disconnect control sequence (in switched network operation). Other control sequences using DLE are available to provide active control characters within transparent text as required (see Section IV). DLE is normally sent as a single

character; however, in transparent mode it is sent twice when a bit pattern in the data corresponds to the DLE character. In this case, the first DLE is "thrown away" by the control, and the second DLE is considered a data character. This doubling of DLE characters is done automatically by the transmitting control.

End-of-Intermediate-Block (ITB) Character

The ITB character signals (1) the end of an intermediate block of data, and (2) that the next character is the BCC character to be used to check for errors in that block. No response is transmitted to an ITB, but the character is transferred to memory and the BCC character is processed normally for an error check. The result is stored in the communication control or adapter and reported when the entire block of data is received. The central processor will therefore receive only one status report, which will be for the entire block of data just received.

The use of the ITB character allows the program to break up a long block of data into more easily handled intermediate blocks; however, an error in any intermediate block will necessitate the retransmission of the whole block. The US character (in ASCII and Six-Bit Transcode) and the IUS character (in EBCDIC) are used as the ITB character, since the ITB character is not directly provided in any of these codes.

Longitudinal Redundancy Check (LRC) and Cyclic Redundancy Check (CRC) Characters

The LRC or CRC character is transmitted or checked by the communication control immediately following an end character. Their composition and significance are discussed in the paragraph entitled "Error Control" (see page 2-10). They are not transferred to the central processor.

Wait Before Transmitting (WACK) Character

The WACK character allows the receiving terminal to indicate a temporary "not able to receive" condition to the transmitting terminal. It is sent as a positive acknowledgment to a text or heading block. The normal transmitting terminal response to WACK is ENQ; however, EOT and DLE EOT are also valid responses. When ENQ is received, the receiving terminal may continue to respond with WACK until it is ready to receive further data transmissions.

Reverse Interrupt (RVI) Sequence

The RVI control sequence is used in place of the ACKØ or ACK1 positive acknowledgment. RVI is transmitted by a receiving terminal to request termination of the current transmission because of a high priority message which it must transmit to the sending station, or, in a multipoint environment, because it now wishes to communicate with another station on the line. Successive RVI's cannot be transmitted except in response to ENQ.

The transmitting terminal treats RVI as a positive acknowledgment and responds by transmitting all data that prevents it from becoming a receiving station. To accomplish this, more than one block transmission may be required.

Temporary Text Delay (TTD) Sequence

The TTD control sequence (STX ENQ) is sent by a transmitting terminal when it wishes to retain the line but is not ready to transmit. STX ENQ should be sent when the TTD timeout occurs (approximately two seconds after the end of reception of the last reply). This 2-second timeout avoids the nominal 3-second receive timeout at the receiving terminal.

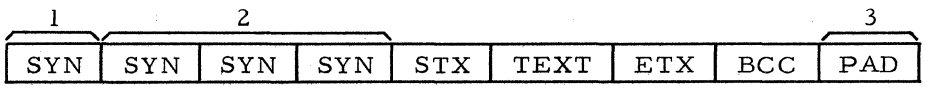
The receiving terminal must respond with a NAK to the TTD sequence, and wait for transmission to begin. If the transmitting terminal is still not ready to transmit, the TTD sequence can be repeated one or more times. It should be noted that any 281-2BSC communication control receiving a TTD sequence will signal the central processor that an abort has been received. Therefore, it is necessary that the program distinguish this sequence from an actual abort.

Pad Characters

To insure that the last character of a transmission is properly transmitted by the data set, all 281-2BSC controls and 285-2BSC adapters add a pad character after each transmission. Since ETB or ETX causes a line turnaround, the pad character follows the BCC. The trailing pad character ensures that the last significant character (for example ETB BCC, ETX BCC, NAK, EOT, ENQ, etc.) is sent before the data set transmitter turns off. The pad character is not transferred to the central processor.

It is suggested that the communications program add a pad character before the beginning synchronization sequence (the three SYN characters) of all transmissions to ensure that the data set is turned on and transmitting. This leading pad character can be another SYN character.

An example of block padding is shown below.



In the example above, the SYN character designated "1" is the leading pad character added by the communications program. The three SYN characters designated "2" comprise the required synchronization pattern provided by the program. The character designated "3" is the trailing pad character added by the communication control.

CHARACTER FORMAT

Line Format

The 281-2BSC and 285-2BSC are capable of transmitting or receiving any 8-bit or 6-bit character, depending on the code chosen prior to the time of installation. In 8-bit operation, the character can be in any code with odd parity in the eighth bit, such as ASCII, or any code without parity, such as EBCDIC. In 6-bit operation, the character can be in any 6-bit code without parity.

Memory Allocation

If 8-bit operation is chosen, each character occupies two positions in main memory; if 6-bit operation is chosen, each character occupies one position. When an 8-bit character is stored, bits 1 through 6 occupy positions 1 through 6 of the second character position. Bits 7 and 8 occupy positions 1 and 2 of the first character position.

Variable Character Recognition

The 281-2BSC and 285-2BSC recognize and react to ten control characters and the twelve 2-character control sequences defined by BSC operating procedures. The control characters are selected prior to communication control installation. The 2-character sequences are comprised of control characters or, in some instances, of a control character and one of four data characters (which are also selected prior to installation).

It is important to note that character recognition cannot be changed by programming. The data characters selected only cause control action when they are immediately preceded by the control character DLE. The bit patterns and octal equivalents for the control characters, as defined for ASCII, EBCDIC, and Six-Bit Transcode, are presented in Table 2-1.

ERROR CONTROL

Error control is maintained by combining error detection with retransmission of messages until they are received error-free. The communication control performs the error detection function while the program must provide the necessary retransmission capabilities. Each block of data transmitted can be error-checked in any of several ways, depending on the codes and functions employed. Three major methods of error detection are employed in BSC: odd parity checking by the vertical redundancy check (VRC); message block checking by the longitudinal redundancy check (LRC); and the cyclic redundancy check (CRC). Checking is always done at the receiving terminal. In VRC checking, this occurs on a character-by-character basis as each character is received.

The block check character (BCC), either LRC or CRC, is accumulated identically at both the transmitting and receiving terminals, for each block of data transmitted. The transmitting communication control transmits the BCC immediately following an ETB, ETX, or ITB character. The receiving control compares its accumulated BCC against the transmitted BCC for an equal condition, which indicates that the previous block was properly received. The results of the comparison are reported to the central processor via an interrupt, but the BCC is not transferred to the central processor. The basic ground rules for accumulating the BCC in any form are:

1. Reset accumulation to all zeros on the first STX or SOH character of the message after line turnaround.
2. Accumulate all characters that follow the first STX or SOH, except SYN characters.
3. In the transparent mode, following DLE STX, do not accumulate the first DLE of any 2-character control sequence.
4. Reset accumulation to all zeros following an ITB BCC, and restart accumulation with the first non-SYN character that follows.

The combinations of the three error checks that are used depend on the code employed and, in the case of ASCII and other similar codes (7 bits plus parity), whether transparent transmissions are to be allowed.

Table 2-3 lists the combinations available in the 281-2BSC and 285-2BSC. After each transmission, the receiving terminal replies with ACKØ or ACK1 (data accepted—continue sending) or NAK (data not accepted—retransmit previous block), depending on the results of the check. Retransmission of a block of data following an initial NAK is usually attempted at least three times. If the transmitting terminal does not receive a reply to a data block, or if the reply is garbled, the transmitting terminal can request retransmission of the reply by sending ENQ.

Table 2-3. Error Checking Capabilities

Transmission Code	Type of Checking		
	Transparency Not Programmed	Transparency Programmed and Operating	Transparency Programmed but Not Operating
EBCDIC	CRC-16	CRC-16	CRC-16
ASCII	VRC/LRC	CRC-16	VRC/CRC-16
Six-Bit Transcode	CRC-12	CRC-12	CRC-12

NOTES:

1. No 281-2BSC communication controls and 285-2BSC communication adapters that work with EBCDIC or Six-Bit Transcode need to be programmed to allow transparent transmissions; ASCII controls and adapters must be programmed to allow transparent transmissions, due to the different error-checking procedures employed.
2. The error checking methods are defined as follows:

VRC	Vertical redundancy check (odd parity on each character)
LRC	Longitudinal redundancy check
CRC-16	Cyclic redundancy check (using 16 bits)
CRC-12	Cyclic redundancy check (using 12 bits)

VRC/LRC Combination

This combination of error checking methods is available only on those communication controls that operate with ASCII or similar codes (7 data bits plus one parity bit) and that are programmed not to handle transparent mode messages.

Vertical redundancy checking (VRC) is an odd-parity check performed on a per-character basis. The 281-2BSC accepts the seven data bits from memory and generates the eighth bit (parity bit) when transmitting. When receiving, the parity check is performed as each character is received; all eight bits are transferred to memory.

The longitudinal redundancy check (LRC) is a Block Check Character (BCC) that is accumulated from all the characters in the message block. The accumulation is performed both by the transmitting and receiving controls by modulo-2 half-adding each data bit position of each character of a message block to the corresponding data bit position of the LRC character. The transmitting control transmits the LRC character immediately following an ETB, ETX, or ITB character. The receiving control compares the transmitted BCC against its accumulated BCC for an equal condition, which indicates that the previous block was received without a block check error.

The transmitting control generates an odd-parity bit for the 7-bit LRC character and the receiving control checks for odd parity in the LRC character. The LRC character is not transferred to memory, but the results of the comparison are reported to the central processor via an interrupt.

The LRC accumulation is reset to zero by the first SOH or STX character received in a transmission. All STX or SOH characters received thereafter, until the next line turnaround, are included in the accumulation. The accumulation is also reset to zero following an ITB BCC.

Cyclic Redundancy Check (CRC-12/CRC-16)

Cyclic redundancy checking (CRC) is a more powerful method of block checking than LRC. Two modes of CRC are employed with BSC. The first, CRC-12, is used for 6-bit transmission codes (such as Six-Bit Transcode); the second, CRC-16, is used for 8-bit transmission codes (EBCDIC and similar codes).

A cyclic redundancy check is a division performed by both the transmitting and receiving controls using the numeric binary value of the message as a dividend, which is divided by a constant. The quotient is discarded, and the remainder serves as the check character, which is then transmitted as the block check character immediately following an ITB, ETB, or ETX character. The receiving control compares the transmitted remainder to its own remainder, and finds no error if they are equal.

The 281-2BSC and the 285-2BSC use the polynomial $X^{16} + X^{15} + X^2 + 1$ for the constant when performing a CRC-16 check. They use the polynomial $X^{12} + X^{11} + X^3 + X^2 + X + 1$ when performing a CRC-12 check.

With these forms of block checking, the block check character actually consists of two characters (i.e., 16 bits for CTC-16 and 12 bits for CRC-12) when it is transmitted on the line.

VRC/CRC-16 Combination

This is a combination of VRC and CRC used by communication controls operating with ASCII and similar codes (7 data bits plus one parity bit) that are programmed to handle transparent mode transmissions. The VRC is inhibited during transparent operation and the communication control accepts all eight bits from memory when transmitting. The 16-bit CRC generator is used in both the normal and transparent modes.

Format Errors

A format error is defined as any sequence of communication control characters that is not defined by BSC procedures. Format errors are checked on a block-by-block basis, but are not checked through a line turnaround. (For example, the communication control does not report an erroneous replay as a format error if the format of the reply follows a correct BSC format.) See Figure 2-1 for legal formats.

Format errors cause an interrupt at the end of a received or transmitted message. If the communication control is transmitting, the "sent illegal message" flag is turned on. If the control is receiving, the "received illegal sequence" flag is turned on.

Overwrite

If a character contained in the Type 281-2BSC Communication Control's receive buffer is overwritten (by another character being received), an appropriate status signal is turned on and, at the end of reception of the data block, the central processor is notified via an interrupt. The Type 285-2BSC Communication Adapter is also subject to character overwrite; if this occurs, the overwrite status will be transferred to the Type 286-6 or -7 multi-channel communication control at the end of reception. The program must always transmit a reply of NAK to any data block received with an overwrite error.

TIMEOUTS

Timeouts are used during the reception and transmission of control (ID) heading and text data to ensure efficient utilization of transmission facilities. They prevent indefinite tie-ups caused by illegal sequences or missed turnaround signals (ETB, ETX, etc.). The timeouts provide a fixed time within which any particular operation must occur. Six timeout functions are provided: transmit, receive, inactivity, underflow, text, and text delay.

Transmit Timeout

This is a 1-second timeout that establishes the rate at which idle SYN characters are inserted automatically into transmitted headings and text data. In normal data transmission, two consecutive SYN characters are inserted for timing purposes only, and they have no effect on the message format. The programmer can insert extra idle characters into normal data, but cannot do so in transparent data.

Receive Timeout

This 3-second timeout limits the turnaround reply interval and is used as follows:

1. It limits the waiting time for a transmitting terminal to receive a reply after a line turnaround.

2. It permits the receiving terminal to check the line for idle characters. These idle characters indicate that the transmission is continuing and that character synchronization has been maintained. Thus, this timeout is restarted each time an idle character is detected. If a timeout occurs, the communication control initiates an interrupt by turning on the "loss of sync" status signal, and starts to scan for SYN characters. An "end-of-order" status signal is also sent to the central processor when this timeout occurs.

Inactivity Timeout

This is a nominal 30-second timeout that protects against prolonged periods of inactivity. The timeout is started when the "interlock" signal from the data set goes on. If, during the 30-second timeout period, no data is received or transmitted, the communication control will notify the central processor via an interrupt.

Underflow Timeout

This is a nominal 30-millisecond timeout that indicates a loss of data from either the central processor or the data set. If the communication control is transmitting and does not receive data from the central processor for a period equal to the length of this timeout, an interrupt occurs to notify the central processor that the communication control has transmitted an abort. If the communication control is receiving and the data set does not indicate that it is receiving a character during the timeout period, character synchronization is dropped, an interrupt occurs, and the 281-2BSC or 285-2BSC starts to scan for synchronization characters.

Text Timeout

This optional timeout is available for use as a "reasonableness" test for the length of heading or text data (i.e., the transmitted or received block of data). It is started when the communication control achieves character synchronization. It runs for a period determined by the user (prior to installation). Honeywell suggests that this period be approximately 15 seconds (equal to the time necessary to transmit 4500 8-bit characters using a Bell System 201B data set). If an end-of-block character (ETB, ETX, etc.) is not received during the timeout period, the communication control or adapter disconnects and initiates an interrupt.

Text Delay Timeout

This is a nominal 2-second timeout associated with the transmission of TTD and WACK sequences. It is used where the speed of input or output devices may cause transmission delays. This timeout starts at the end of a reception and runs until (1) a transmission is started, or (2) it runs out and causes an interrupt.

If the interrupt occurs at the transmitting terminal, it requests that the central processor start sending a message. If the interrupt occurs at the receiving terminal, it requests that the reply that was to be sent be delayed until the ENQ from the transmitting terminal is received.

SECTION III

OPERATION

POINT-TO-POINT OPERATION

A point-to-point data link consists of a communication facility between two terminals. All transmissions over the data link must be between the two terminals operating on that data link. The point-to-point link can be established over leased, nonswitched communication lines or a switched network. On a leased line (permanent-type connection), the transmissions are always between the same two terminals. On a switched network, the data link is disconnected after the two terminals complete their transmission; a new data link is established for subsequent transmissions by standard dialing procedures (either manual or automatic). The new data link can be established with any other terminal in the network.

Point-to-Point Operation with Contention

When operating on a nonswitched, point-to-point network the connection is made by either privately owned lines or leased private lines. When transmission is started, an initialization sequence (SYN SYN SYN ENQ) must be instituted by the terminal attempting to acquire the line. A terminal receiving this sequence (and ready for message reception) replies with an affirmative acknowledgment sequence (SYN SYN SYN ACK0). If the station is not ready for reception, it replies with either of the following: (1) SYN SYN SYN NAK (not ready to receive); or (2) SYN SYN SYN WACK (temporarily unable to receive). The format for the complete initialization phase, including the start of the actual message transmission if shown in Figure 3-1.

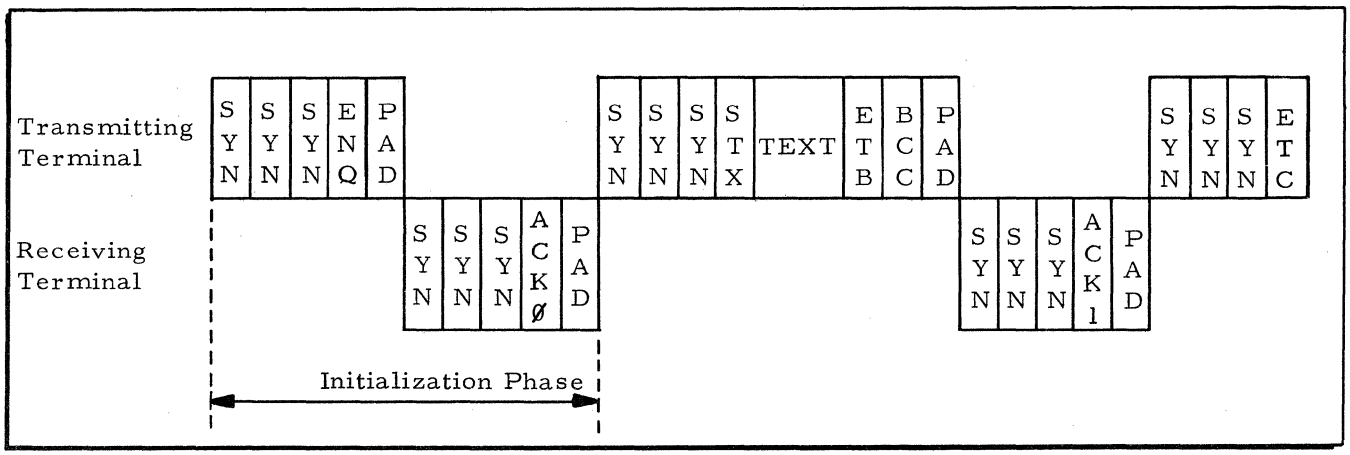


Figure 3-1. Point-to-Point Initialization Phase

To avoid the problems associated with simultaneous transmission requests, each terminal is assigned a primary or secondary priority. The higher priority (primary) terminal sends an ENQ to acquire the idle line; it continues to send ENQ until it receives an affirmative response or until the retry limits of the primary terminal are reached. If the primary terminal receives an ENQ and it has not initiated a request for the line, then it replies with ACKØ (ready to receive), WACK (temporarily unable to receive), or NAK (not ready to receive). Thus the secondary terminal can gain control of the line for transmission only when the line is left free by the primary terminal.

Message transmission is ended and the line is returned to an idle state by the transmission of EOT. The terminal sending EOT must not send an initialization sequence until three seconds have elapsed, thus allowing the other terminal to bid for the line.

Switched Network Operation

When operating in a switched network, the point-to-point connection is established by either manual or automatic means. Dialed connections are operated as a point-to-point line with contention. Both terminals begin in the circuit assurance mode. Once circuit assurance is established and identification (optional) is completed, the terminals use the normal BSC procedures required for operation (switched point-to-point). A disconnect signal is normally sent when both terminals have completed their message transmissions.

The circuit assurance mode is entered when the called terminal goes off-hook. At this time, the calling terminal is notified by a signal from its data set that a connection with another data set has been established. Once this indication is received, the calling station sends either of the following messages:

1. WRU—Who are you? The transmitted sequence is SYN SYN SYN ENQ.
2. IAM/WRU—I am X..., who are you? The transmitted sequence is SYN SYN SYN X... ENQ. "X" indicates the graphics for terminal identification and supplementary control purposes. This message identifies the calling terminal and requests the called terminal to identify itself.

Either message is then followed by an identification message from the called station: ID ACKØ. (The transmitted sequence is SYN SYN SYN X... ACKØ where ID is optional.)

NOTE: If the received ID is unsatisfactory, either terminal can initiate a disconnect sequence.

Additional signals available as a reply to the WRU and IAM/WRU messages are:

1. NAK — This reply indicates that a "not ready" condition exists at the called station.
2. WACK (optional) — This reply indicates that a "temporarily not ready to continue" condition exists at the called station.

BSC communication programs used with the 281-2BSC or 285-2BSC should provide the capability to transmit identification sequences in order to permit several types of BSC terminals to operate on the same switched line. An ID sequence can be from 2 to 15 characters long. The minimum 2-character sequence consists of the same character transmitted twice. All BSC communication programs at receiving terminals should be capable of recognizing control sequences regardless of the presence of leading graphics; the terminal's specifications determine whether the terminal will attach any functional significance to these characters or will have the ability to transmit them.

Both terminals exit from the circuit assurance mode when any of the following sequences are sent or received:

1. SYN SYN SYN EOT — This sequence returns the data link to normal operation (the passive monitoring mode in which either terminal may bid for the line).
2. SYN SYN SYN SOH — This sequence initiates a block of heading data.
3. SYN SYN SYN STX — This sequence initiates a block of text data.
4. SYN SYN SYN DLE STX — This sequence initiates a block of transparent text data.

All signals other than those described are considered to be errors. If a valid reply is not received by the calling station (following either a WRU or an IAM/WRU) within the receive time-out period, the request message can be retransmitted. However, the data link continues in circuit assurance mode until the circuit assurance sequence is satisfactorily completed.

The call between stations can be terminated by the inactivity timeout or by transmission of the disconnect sequence SYN SYN SYN DLE EOT. This sequence may be initiated by either terminal when operating on a switched network basis. When operating with a control terminal, the control terminal normally initiates the disconnect sequence. As this sequence is transmitted and received, each terminal returns to an on-hook condition and the line is dropped.

For switched network operation, procedures should be provided to permit the connection to be broken and re-established (by hanging up and redialing the number) whenever transmission difficulties are experienced. This procedure provides the opportunity for establishing a totally different communication path, which may improve transmission capabilities, when the connection is re-established.

MULTIPOINT OPERATION

A multipoint data link consists of communication facilities between three or more terminals on privately owned or leased private lines. In the multipoint network, one terminal is designated as the control terminal, and the remaining terminals are designated as tributary terminals. All

transmissions for this type of operation are regulated via the control terminal by means of polling or selection. By sequentially polling each tributary terminal, the control terminal directs the incoming message traffic. The outgoing traffic is regulated from the control terminal by selection of the desired tributary terminal, which receives the message. All transactions are between the control terminal and the selected tributary terminal.

The initialization phase for multipoint operation is accomplished by the control station transmitting the following sequence: SYN SYN SYN EOT PAD SYN SYN SYN (polling or selection address) ENQ PAD. This sequence ensures that all monitoring tributary terminals are now in the control mode and are thus prepared to receive either a poll or a selection from the control terminal.

The polling or selection sequence transmitted while in the control mode designates the terminal that is to transmit or receive data. This sequence can also define the specific device required, if the terminal has several available. The recognition of the polling and selection sequences is performed by the program in the central processor. All action taken on the data contained in the selection or polling sequences must be initiated by the central processor. An example of the complete sequence transmitted from the control terminal is shown in Figure 3-2.

The polling and selection sequences, consisting of from one to seven characters, are followed by ENQ. For specific sequences, the appropriate terminal specifications should be considered.

The possible replies from a polled tributary terminal are:

1. Heading data — SYN SYN SYN SOH...
2. Text data — SYN SYN SYN STX ...
3. Transparent text data — SYN SYN SYN DLE STX transparent text...
4. Negative reply when the terminal has nothing to send — SYN SYN SYN EOT.
5. Temporary text delay when the terminal is unable to transmit its initial block within two seconds — SYN SYN SYN STX ENQ.

The possible replies from a selected tributary terminal are:

1. Affirmative reply indicating that the tributary terminal is prepared to receive — SYN SYN SYN ACKØ.
2. Negative reply indicating that the tributary terminal is not ready to receive — SYN SYN SYN NAK.
3. Reply indicating that the tributary terminal is temporarily unable to receive — SYN SYN SYN WACK.

The complete formatting of message traffic for a multipoint operation is shown in Figures 3-2 and 3-3 (polling and selection with respect to two different tributary terminals by the control terminal). The terminal identification for the two different operations is indicated by using upper case alphabetic characters for polling and lower case alphabetic characters for selection (for these examples). The component designations are 1 (for printer) and 6 (for reader) in these examples.

LIMITED CONVERSATIONAL MODE

This mode allows the transmission of heading or text data in reply to a complete message. A conversational reply can be sent in place of an affirmative reply to a block of text that ends with ETX or DLE ETX. Conversational replies are not permitted following a block of heading or blocks of text ending with ETB.

Either SOH or STX can start a conversational reply. The terminal receiving the conversational reply interprets the SOH or STX as the affirmative reply to the last data block it transmitted. Transparent text (beginning with DLE STX) is also a valid conversational reply. A terminal receiving a conversational reply is not permitted to respond by transmitting another conversational reply. An example of the limited conversational mode is given in Figure 3-4.

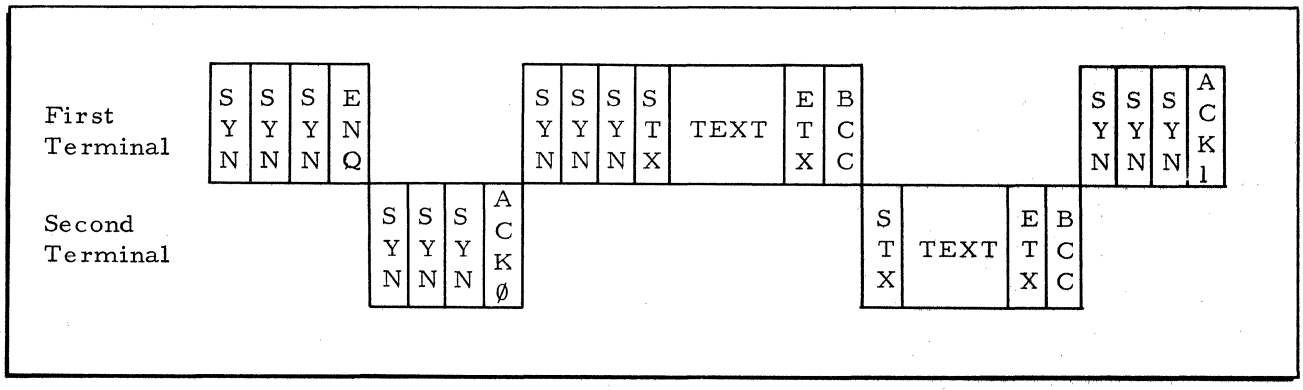


Figure 3-4. Limited Conversational Mode

The limited conversational mode of operation can be used on any network as long as all the terminals in the network realize that this mode of operation is to be used. This mode will most likely be used in networks where a remote terminal and a central processor must swap operating parameters before beginning the communication sequence. When and if this mode is to be used, it will be spelled out in the network specifications; it is mentioned here only to inform the reader that such a mode of operation exists.

SECTION IV
PROGRAMMING INFORMATION

INTERRUPT PROCESSING

The Type 281-2BSC Communication Control (and the Type 285-2BSC Communication Adapter when used with a Type 286-6 or -7 Communication Control) turns on an interrupt signal when it has a message status to report to the central processor. Upon interrupt, the central processor tests the state of the interrupt signal by an appropriate PCB (Peripheral Control and Branch) instruction. Since it is possible for more than one condition to cause an interrupt at any given time, it may be necessary to check all possible interrupt conditions by PCB instructions.

SYSTEM REQUIREMENTS

The central processor must be capable of being interrupted by the communication control. The control operates in the half-duplex mode; therefore, one read/write channel must be assigned to the communication control for the duration of the entire message.

Data Transfer

Each block of data to be transmitted must first be fully assembled in memory and transferred to the communication control character-by-character, as requested by the communication control, using a single PDT (Peripheral Data Transfer) instruction for the entire block. Similarly, in reception the entire block of data must be assembled in memory. When the communication control indicates error-free reception at the end of a block, the entire block should be transferred to the storage device using one PDT instruction.

For maximum line utilization, the memory area containing data to be transmitted should be double-buffered. That is, one buffer contains the last block of data transmitted (to be retransmitted if a NAK is received). The second buffer contains the next block of data to be transmitted if an ACK (positive acknowledgment) signal is received for the previous block. The second block can be prepared from storage while the first block is being transmitted. To prevent loss of information, buffering should also be adapted for reception. A block of data received accurately can be transferred to storage while the communication control is receiving the next block from the line.

SIX-LEVEL CODES

For a 6-level code, such as Six-Bit Transcode, each memory location in a block buffer will contain a 6-bit character which can be transferred to the communication control.

EIGHT-LEVEL CODES

For an 8-level code, such as EBCDIC, a character must be assembled in the communication control from two buffer locations. The two high-order bits of the coded character are contained in the two low-order positions of a buffer location, and are transferred to the communication control first. The six low-order bits of the character are contained in the next buffer location, and are transferred to the communication control next, as requested by the control. The same order is applicable for reception.

SEVEN-LEVEL CODES (PLUS PARITY)

For a 7-level code (plus parity), such as ASCII, the communication control behaves as it does for an 8-level code with the following differences:

1. In transmission the communication control generates parity; therefore, the buffer location that sends the high-order portion of the character to the communication control contains only one high-order bit (in its low-order position).
2. In reception the communication control checks for parity errors, but it transfers both the high-order and parity bits to the buffer as the high-order portion of the coded character.

The above differences apply equally to all characters, whether data or control characters, in a particular code.

PROGRAMMING

Initialization

To prepare the central processor and communication control for communication, the following initialization steps must be taken:

1. Issue an Initialize PCB instruction.
2. Issue a Turn On Allow PCB instruction.
3. Perform miscellaneous housekeeping, such as setting up any constants or fields necessary.
4. Keep a program running in the central processor so that it can be interrupted.

It is absolutely necessary that the communication controls that are to communicate with each other are code-compatible. If communication is attempted between two incompatible communication controls, error conditions will result.

Synchronization

Every transmitted message should begin with four SYN characters supplied by the program. This initial synchronizing sequence is the only one required from the program. All subsequent time-fill and transmit timeout synchronizing sequences are automatically generated by the communication control.

Data Transmission

Data is transmitted by issuing a Transmit PDT instruction to the communication control. The characters to be transmitted are extracted from memory using one memory cycle for a 6-bit code (Six-Bit Transcode) or two memory cycles for either an 8-bit code (EBCDIC) or a 7-bit code (ASCII). If two memory cycles are needed, the first cycle transfers the two high-order bits (or parity and high-order bit) of the character and the second cycle transfers the six low order bits.

NOTE: In normal mode transmissions of 7-bit codes, the parity bit received from memory is discarded and the parity bit calculated by the 281-2BSC is substituted. In the transparent mode, the parity bit position of 7-bit codes is used for data and the control accepts this bit position from memory. When receiving, the 281-2BSC always transfers all eight bits (eight data bits or seven data bits plus parity) to memory regardless of the mode of operation (except when operating with a 6-bit code). The 285-2BSC adapter operates in an identical manner except that it is the 286-6 or -7 communication control that generates the parity bit when necessary.

Regardless of the code set used, the necessary initial synchronization and framing characters (SOH, STX, ETB, ETX, etc.) must be supplied by the program

The transmit RWC (read/write channel) will be released in either of two conditions. The first is the transmission of a character that initiates a line turnaround. When this happens, an End-of-Order signal is transferred to the central processor and an interrupt follows with a status indicator that is dependent upon the status of the message transmitted. The second condition occurs when a record mark is encountered in memory. When this happens, an interrupt occurs with a status of "Frame Demand Out and No Order Stored." The communication control continues to transmit by inserting time-fill (SYN) characters into the void caused by the loss of the RWC until a new Transmit PDT instruction is issued. If a new PDT instruction is not issued within 30 milliseconds of the original interrupt, the "sent abort" status indicator is turned on and the communication control automatically aborts the message.

The transmission of transparent data proceeds in the same manner as the transmission of normal mode data. That is, the data is arranged in memory in 8-bit characters, such as EBCDIC and ASCII (the parity bit in ASCII-equipped communication controls is now used as a

data bit), or 6-bit characters (Six-Bit Transcode) as it would be normally. The beginning characters (DLE STX) are inserted in front of the block of data and the ending sequence (DLE ITB, DLE ETB, or DLE ETX) is inserted at the end of the block of transparent data in the same manner as the SOH or STX characters and the ITB, ETB, or ETX characters would be for normal mode data.

In addition, a 12-bit binary number is inserted into the two line character positions immediately following the DLE STX sequence to define the number of line characters in the transparent text, exclusive of the beginning and ending sequence. This 12-bit binary number is used by the 281-2BSC or 285-2BSC to control some of its internal operations during transparent mode transmissions. It is the programmer's responsibility to see that this number is correct.

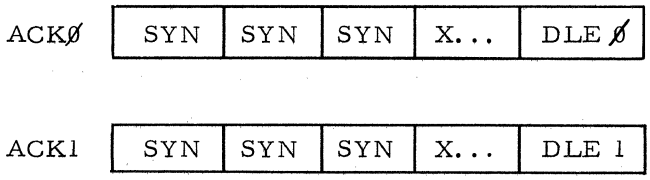
Control Sequences

The entire control sequence to be transmitted should be contained in memory. The sequence should be complete with the necessary synchronization characters, identification characters (if used), and the control characters. The communication control does not generate nor check a block check character for such a sequence but generates and checks parity if it is required for the code set in use. The next memory position, following the last character of the sequence to be transmitted, should contain a record mark to terminate the order. If a record mark is sensed in the middle of a control sequence, the communication control does not draw any additional characters from memory and it causes an interrupt with a "Frame Demand Out and No Order Stored" status indicator. Under this condition, it is possible to transmit illegal sequences; if this occurs, the appropriate status indicator is turned on. The proper placement of a record mark is the programmer's responsibility.

RECEIVING SEQUENCES

In the message transfer mode, a receiving terminal will send one of the following control sequences:

1. ACKØ or ACK1. These sequences signify that a block of data was received with no detectable errors and that the next block may be sent. The transmitted sequences are shown below:



a. "X..." indicates the identification graphics (if used).

b. The characters Ø and 1 are for ASCII. See Table 2-1 for EBCDIC or Six-Bit Transcode.

From the programming point of view, one of these two sequences if the normal response that would be sent when the communication control interrupts.

ACKØ is used to respond to all even-numbered blocks of data. ACK1 is used to respond to the first block in a transmission and all odd-numbered blocks thereafter. This procedure (required by BSC conventions) is designed to allow the transmitting terminal to ascertain with some degree of confidence that the receiving terminal has not missed a block of text.

It should be noted that the single character ACK defined by the code sets (ASCII, etc.) is not recognized by the 281-2BSC, 285-2BSC, or BSC conventions. It cannot be used in lieu of the two-character sequences, ACKØ and ACK1, defined by the BSC procedures.

2. NAK. This character informs the transmitting terminal that a block of data has been received with an error (parity, block check, or illegal sequence). It requests retransmission of the block that was received with errors. When the 281-2BSC control interrupts at the end of the received block, and the status is either Received Message with Error, Received Illegal Sequence, or Received Abort, this response is transmitted. The transmitted sequence is shown below:

SYN	SYN	SYN	X...	NAK

a. "X..." indicates the identification graphics (if used).

3. WACK. This sequence is utilized to indicate that trouble has developed at the receiving terminal, and that the receiving terminal is unable to receive because:

a. The storage medium requires attention, or

b. Some part of the computer system is malfunctioning, etc.

Under such circumstances the receiving terminal proceeds as follows:

a. The receiving terminal waits until the entire block of data has been received.

b. When the communication control has computed and checked the BCC, and indicated to the program whether the received data is good or bad, the program sends a WACK sequence:

SYN	SYN	SYN	DLE ;

NOTE: DLE ; is for the ASCII code set. See Table 2-1 for equivalents in EDCDIC or Six-Bit Transcode.

c. Since there is no way of determining the duration of the interruption, the transmitting terminal sends ENQ's and the receiving terminal responds with WACK's until the condition clears.

d. When the receiving terminal is again ready to receive, its program transmits the ACKØ, ACK1, or RVI, and the communication continues from the point of interruption.

4. RVI. The RVI sequence is a positive acknowledgment sequence used in place of ACKØ or ACK1. The transmitted sequence is:

SYN	SYN	SYN	X...	DLE<

a. "X..." indicates identification graphics (if used).

b. DLE< is for the ASCII code set. See Table 2-1 for EBCDIC and Six-Bit Transcode equivalents.

The RVI sequence, when transmitted by a receiving terminal, acknowledges receipt of data without detectable errors and requests termination of the current transmission because of a high priority message that it must transmit. Successive RVI's cannot be transmitted, except in response to ENQ's.

The transmitting terminal treats RVI as a positive acknowledgment, and responds by transmitting all data that prevents it from becoming a receiving terminal. More than one block transmission may be required to empty the transmitting terminal's buffers.

5. DLE EOT. This sequence is used on switched networks by either terminal to terminate the connection. The transmitted sequence is:

SYN	SYN	SYN	DLE	EOT

Once this message is transmitted, the connection is terminated immediately and can only be re-established by redialing.

All of the preceding sequences pertain to data received. One of these sequences must be sent in response to the block of data received within the 2-second timeout period; otherwise, the transmitting terminal will transmit an ENQ to elicit a response. The next block of data is not transmitted unless a response is transmitted and received correctly.

These responses are transmitted by issuing a Transmit PDT instruction to the communication control at the receiving terminal. The communication control then extracts the characters from memory and transmits them, just as it would any other block of data.

TRANSMITTING SEQUENCES

The transmitting terminal will send one of four control sequences as appropriate. These sequences are:

1. ENQ. This is transmitted to the receiving terminal to signify that a response has not been received for the last block of data transmitted. This may be because the transmitting terminal did not send the response in time or because the response was not recognized (due to noise or some other reason). The transmitted sequence is:

SYN	SYN	SYN	X...	ENQ

NOTE: "X..." indicates the identification graphics (if used).

The ENQ sequence is interpreted by the receiving terminal as a request to "retransmit your last response." The transmitting terminal sends this sequence again if a response is not received before the "response time exceeded" 3-second timer times out.

The program must place a limit on the number of times that the transmitting terminal attempts to acquire a response.

The meaning of ENQ is slightly different for the initial condition when the transmitting terminal is trying to establish contact. Then it signifies a request for the status and attention of the receiving terminal.

A third meaning is applicable to ENQ when it appears in text. In this case it is an abort sequence, and the transmitting terminal tells the receiving terminal to disregard this block and reply with NAK.

2. EOT. This is used to tell the receiving terminal that the transmitting terminal has finished its data transmission. The receiving terminal does not have to respond. The transmitted EOT sequence is:

SYN	SYN	SYN	X...	EOT

NOTE: "X..." indicates the identification graphics (if used).

3. TTD. This sequence is the transmitting terminal's version of the WACK sequence. It is transmitted by a transmitting terminal that wishes to retain control of the line, but that is temporarily unable to retransmit its next block of text. The transmitted TTD sequence is:

SYN	SYN	SYN	X...	STX	ENQ

NOTE: "X..." indicates the identification graphics (if used).

The receiving terminal interprets the TTD as an aborted data block and responds with a NAK as it does for any abort.

4. DLE EOT. DLE EOT sequence is discussed on page 4-6.

These control sequences are transmitted by issuing a Transmit PDT instruction to the communication control at the transmitting terminal. The communication control extracts the characters from memory and transmits them, just as it would any other block of data.

Data Reception

Data is received by issuing a Receive PDT instruction to the communication control. The received characters are transferred to memory using one memory cycle for a 6-bit character or two memory cycles for an 8-bit character (eight data bits or seven data bits plus a parity bit). If two memory cycles are needed, the first cycle will transfer the two high-order bits of the

character and the second cycle will transfer the six low-order bits. The result is that the character is contained in two memory locations. It should be noted that the 281-2BSC does not transfer synchronization characters or block check characters to memory. The 285-2BSC adapter also removes the synchronization and block check characters.

The receive RWC is released by either of two conditions. The first condition occurs with the reception of a character that initiates a line turnaround. When this happens, the character is transferred to memory, an End-of-Order status signal is sent to the central processor, and an interrupt follows with a status signal dependent upon the status of the message received. The second condition occurs when a record mark is encountered in memory. When this happens, the character received is transferred to memory. If the character is not an end character, an interrupt occurs with a status of "Frame Demand In and No Order Stored." The communication control continues receiving and the program must issue another Receive PDT instruction before the control has another character to transfer to memory; otherwise an overwrite condition will occur.

Transparent data is received in the same manner as normal mode data; that is, it is received on a character-by-character basis so that character synchronization is maintained. The transfer of transparent data to memory is also the same as for normal mode transfer.

When receiving transparent text, the receiving communication control will remove the beginning DLE from all control sequences except DLE STX, which begins a transparent block. When intermediate blocking is used in the transparent mode, the receiving control will store the contents of the current location counter in the starting location counter each time the end of an intermediate transparent block is received. The memory address thus stored in the starting location counter is the first data location following the ITB. Therefore, when receiving transparent text, if the communication control interrupts with the single status "received end of transparent," the communications program should store the present contents of the starting location counter in some predetermined location in memory so that it can later find the end of each of the intermediate blocks in the transparent text.

If the "received end of transparent" status occurs at the same time as some other status, the end of a transmission has been received and a reply must be sent. The reply will depend upon the status of the received message as reported by the second status signal. Under this condition, the current location counter will contain the address of the next memory location after the ETX or ETB that ends the block of transparent data.

Receiving Control Sequences

Control sequences are received as though they were normal data transmissions, and the non-sync characters are transferred to memory where they can be interrogated and interpreted by the program. The normal interrupt occurs at the end of reception of a control sequence, and the status signals are turned on according to the format, parity, etc. of the received sequence. There is no block check of a control sequence.

Processing Interrupts

Whenever an interrupt occurs from the communication control, the appropriate status signals are turned on to indicate the cause of the interrupt. For the central processor to determine the status, it must issue PCB instructions to the communication control to test the state of the status signals. Normally, only one status signal is turned on at each interrupt. The exceptions normally indicate a program error, except for the "received end of transparent" signal, which can be valid with or without another status signal being turned on at the same time.

RECORD MARKS

Transmission

Normally, the Type 281-2BSC Communication Control does not need any record marks in memory during transmission because all transmissions are terminated by the recognition of a character that initiates a line turnaround. However, record marks are looked for and do cause control action to take place.

When transmitting 8-bit line characters, the communication control uses two memory cycles to extract a character, as explained earlier. If a record mark is sensed on the first cycle, the RWC is released. The bits extracted will be discarded and an interrupt will occur that can be tested by the Frame Demand Out and No Order Stored PCB instruction.

When transmitting 6-bit line characters, the communication control uses only one memory cycle to extract a character. If a record mark is sensed the RWC will be released and the character that has been extracted will be disregarded. An interrupt will occur as above.

When transmitting 8-bit line codes, it is possible for the programmer to mistakenly place the record mark in the memory location that contains the second portion of a character instead of the next location (where it should be placed). If this happens, the communication control discards the character that it has just extracted and turns on the Frame Demand Out and No Order Stored signal and the Received Illegal Sequence signal.

Reception

Normally, the communication control does not require a record mark in memory since all receptions are terminated by the reception of a character that initiates a line turnaround. However, record marks are looked for and do cause control action to take place.

When receiving 8-bit line characters, the communication control uses two memory cycles to transfer a character to memory, as explained earlier. On the second cycle, used to transfer the low-order bits of a character, the communication control checks for a record mark. If one is found, the transfer is completed and the RWC is released. If the character just transferred is not the one that initiated the line turnaround, an interrupt occurs and the Frame Demand In and No Order Stored signal is turned on. If the character was the one that initiated the line turnaround, the interrupt still occurs, but only those signals that pertain to the status of the message received are turned on.

When receiving 6-bit line characters, only one memory cycle is used to transfer the character to memory. Under these conditions the communication control checks for a record mark in every memory position.

When receiving 8-bit line codes it is possible for the programmer to mistakenly place the record mark in the memory position that accepts the first portion of a character instead of the position that accepts the second portion (where the record mark should be located). If this happens, the communication control interrupts the processor and turns on the Frame Demand In and No Order Stored signal and the Sent Illegal Message signal. The last six bits of the character will be lost.

TERMINATING COMMUNICATION

When a transmitting terminal has finished sending all of its data, it sends an EOT sequence that does not need to be acknowledged. On a switched network, DLE EOT (mandatory disconnect) can be substituted for EOT when all transmissions are completed and either terminal wishes to disconnect. Both the 281-2BSC communication control and 285-2BSC communication adapter recognize the DLE EOT sequence and disconnect from the line.

PERIPHERAL CONTROL AND BRANCH (PCB) INSTRUCTIONS

Format

The format for PCB instructions is shown in Figure 4-1. The significance of the control characters is outlined below the figure.

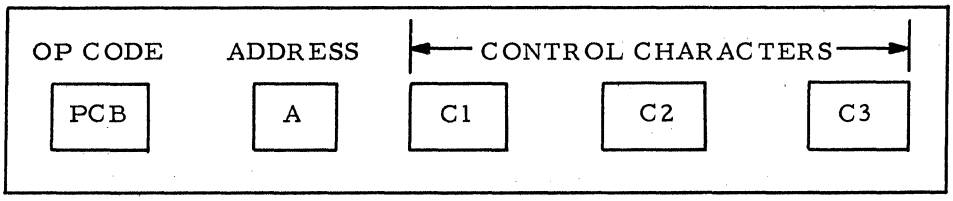


Figure 4-1. PCB Instruction Format

C1

X X X X X X

RWC Assignment:[1]

Same as in PDT instruction. If "busy" status of RWC is not to be tested, all bits of C1 must be zero. Refer to the appropriate programmers' reference manual for coding an RWC busy test.

C2

X X X X X X

Peripheral Address

Sector Bits:[2]

00 = Sector 1
10 = Sector 2
11 = Sector 3

I/O Bit:[3]

Irrelevant

C3

X X X X X X

Control Parameters:

This character provides for questions and control. Multiple queries or actions can be implemented by the use of additional control characters.

[1] RWC assignment depends upon the processor in use and the I/O sector to which the control is connected.

[2] Sector bits apply only to Model 1015 and larger Series 200 processors; these bits specify the sector in which the peripheral control is connected. The designated sector must include the read/write channel assigned in control character C1. Sector bits must be zeros on processors smaller than the Model 1015.

[3] This bit must be zero for programs to be compatible with the Model 8200.

Instructions for Status and Control

All PCB instructions are issued either to obtain communication control status or to control the activity of the communication control. When the communication control interrupts the central processor, the program must determine the source of the interrupt (that is, the peripheral device responsible for the interrupt) and the reason for the interrupt (status). It, therefore, issues a series of PCB "test" instructions to determine the above. When the response to a PCB test is negative, the program normally continues to the next sequential instruction, which is usually another test PCB. When the answer is positive the program normally branches to the A address of the PCB instruction to take the required action in response to the cause of interrupt. The PCB instructions that test the status of the communication control in transmission and reception are listed in Tables 4-1 and 4-2, respectively. The PCB instructions that are used to control the activity of the communication control are listed in Table 4-3.

Table 4-1. PCB Instructions for Transmit Status

Instruction	Function	C3 Octal Code
Busy	This instruction tests the communication control for a busy condition. If the control is busy, the program branches to the A address of the PCB instruction. If it is not busy, the program continues in normal sequence.	Any octal code from 10 to 17.
Sent Message	A positive response to this instruction indicates that a message was sent, and the program branches to the A address. The program must now wait either for the status reply from the remote terminal, or for the next block of data if the message sent was a status reply. The message transmitted should be kept intact in memory since it may not be received in the same condition it was transmitted, necessitating a retransmission.	27
Response Time Exceeded	A positive response to this instruction indicates that the remote terminal has failed to respond to either the last text transmission, a polling/selection sequence, or a line bid. This interrupt occurs three seconds after the completion of a transmission of the type described above if no response is received before the end of the timeout. The program should send an ENQ sequence to acquire the required response from the remote terminal. The program must also have provisions to limit the number of times that it will attempt to elicit a response to a transmission so that the communication lines will not be tied up indefinately. A positive response to the Response Time Exceeded PCB instruction causes the program to branch to the A address; otherwise the program continues in normal sequence.	41

Table 4-1 (cont). PCB Instructions for Transmit Status

Instruction	Function	C3 Octal Code
Sent Abort	A positive response to this instruction indicates that the 281-2BSC was commanded to abort the previous transmission, or an ENQ character was detected in the text and the transmission was aborted because of it, or a record mark was sensed in memory and a new PDT was not issued within 30 milliseconds of the interrupt to complete the transmission. The program must now wait for the reply from the receiving terminal. A positive response to the Sent Abort PCB instruction causes the program to branch to the A address; otherwise the program continues in normal sequence.	45
Sent Illegal Sequence	This instruction can test for several conditions when used in conjunction with other PCB test instructions. A positive response to this instruction coupled with a positive response to the FDI and No Order Stored instruction indicates an improperly placed record mark during reception. It may also indicate incorrect transparency count or an attempt to transmit in transparent mode with an LRC check in ASCII. A positive response to this instruction coupled with a positive response to the FDI and No Order Stored instruction, Sent Message, or Sent Abort instruction indicates that the program-supplied character count of the transparent block was too small. In the first case, a record mark has halted the communication control that is now transmitting time-fill. In the second case, the communication control has recognized an end sequence somewhere in the data after the character count ran out and has ended normally. In the third case, the program has not serviced the interrupt that resulted from a first case condition and the communication control has aborted the transmission. A positive response to this instruction, when not coupled with a positive response to one of the above instructions, signifies that the communication control has detected a format error during transmission of the last data block and that the program should expect to receive a NAK as the response from the receiving terminal. If possible, the program should scan the message sent to correct the format error in order that the data may be retransmitted properly after the response is received.	47

Table 4-1 (cont). PCB Instructions for Transmit Status

Instruction	Function	C3 Octal Code
Interlock Off	A positive response to this instruction signifies that the interlock between the data set and the communication control has turned off. This usually occurs if: 1. The data set has lost power. 2. The control at the other end of the line has disconnected. 3. The control at this end has disconnected because a disconnect PCB has been issued, the 30-second inactivity timer has timed-out, the optional text timeout function has been exceeded, or a mandatory disconnect sequence has been originated at either end.	55
Frame Demand Out and No Order Stored	A positive response to this instruction indicates that the communication control has extracted a character with a record mark from memory. The record mark has caused the communication control to release the RWC even though the present message has not been completed; the character is not transmitted. The program must now issue another Transmit PDT instruction to complete the message transmission. The communication control fills in the void by sending SYN characters (time-fill) until it receives a new PDT instruction. A positive response to this instruction coupled with a positive response to the Received Illegal Sequence instruction indicates an improperly placed record mark during transmission.	60
Interrupt	This instruction tests the communication control for an interrupt-initiated condition. If the communication control has caused the interrupt, the program branches to the A address of the PCB instruction. If the communication control did not cause the interrupt, the program continues in normal sequence. (Usually the next instruction in the sequence will be an Interrupt PCB instruction addressed to another peripheral device.)	75

Table 4-2. PCB Instructions for Receive Status

Instruction	Function	C3 Octal Code
Initial Sync Received	A positive response to this instruction indicates that the communication control has recognized two successive synchronization characters and has, thereby established character synchronization. It is the first indication to the program that the control unit is receiving. The program should respond by issuing a Receive PDT instruction to bring in the message that is being received.	21
Received Good Message	A positive response to this instruction indicates that the communication control has received the last message without detectable errors. The program should respond by transmitting an affirmative reply.	23
Received Message with Error	A positive response to this instruction indicates that either a parity error was detected or that the block check failed in the last received message. The program must respond by transmitting a NAK and by disregarding the entire block of data just received.	33
Sync/Underflow/ Overwrite Fault	A positive response to this instruction indicates one of the following: 1. The communication control has not recognized two successive SYN characters in the message for a period exceeding three seconds and therefore assumes that it has lost character synchronization. The communication control thus sends an End-of-Order status signal to the central processor to release the RWC. The program must discard that portion of the message block already received and any portions subsequently received (as the communication control may find character sync again in the middle of the transmission) and reply with a NAK to this transmission. The next message received may be an ENQ because the communication control may not recognize the End-of-Block character(s), thereby missing the line turnaround following the end of the message. The program must remember this condition so that it may make the proper reply after the line has been turned around to allow a reply to be sent. 2. The data set has had the "carrier on" function turned off for a period exceeding 20 milliseconds, indicating the loss of the received signal. The program must discard the data thus far received in the present block. The program must return a NAK reply to any block of data whose reception is thus interrupted.	35

Table 4-2 (cont). PCB Instructions for Receive Status

Instruction	Function	C3 Octal Code
Sync/Underflow/ Overwrite Fault (cont)	3. The communication control has received and shifted a character into its buffer before a previous character was taken into memory. This would most likely result from program failure to issue a Receive PDT instruction when one was requested. The program must now discard the present message and reply with NAK.	
Receiver Response Time	This instruction tests for an interrupt condition that occurs two seconds after a reception has been completed if the communication control has not started to transmit. A positive response informs the program at a receiving terminal that it should hold the reply to the last data block until the transmitting terminal requests it with an ENQ. If the interrupt occurs at a transmitting terminal that has just previously received a reply, the program at that terminal should immediately send either the next text block, a TTD sequence to retain control of the line, or an EOT to release control of the line.	42
Received Illegal Sequence	This instruction tests for an interrupt that occurs at the end of a message block, intermediate or full. A positive response informs the program that a sequence of control characters not defined by BSC procedures was received. The program should discard the message and reply with a NAK after line turnaround. A positive response to this instruction coupled with a positive response to the Frame Demand Out and No Order Stored instruction indicates an improperly placed record mark during reception.	43
Received Abort	A positive response to this instruction signifies that the current message was terminated prematurely by the transmitting terminal with an abort sequence. The program must discard the data received and reply with a NAK.	45
Received End Of Transparent Block	A positive response to this instruction indicates that the communication control has left the transparent mode and that any data received subsequently will be in the normal mode. A positive response to this instruction that is not coupled with a positive response to any other status test instruction signifies the end of an intermediate block of transparent data. The communication control RWC's current location counter has been stored in the starting location counter. The program must retrieve the location stored in the starting location counter in order that the location of the ITB can be found. The location stored will be the first data position after the ITB.	53

Table 4-2 (cont). PCB Instructions for Receive Status

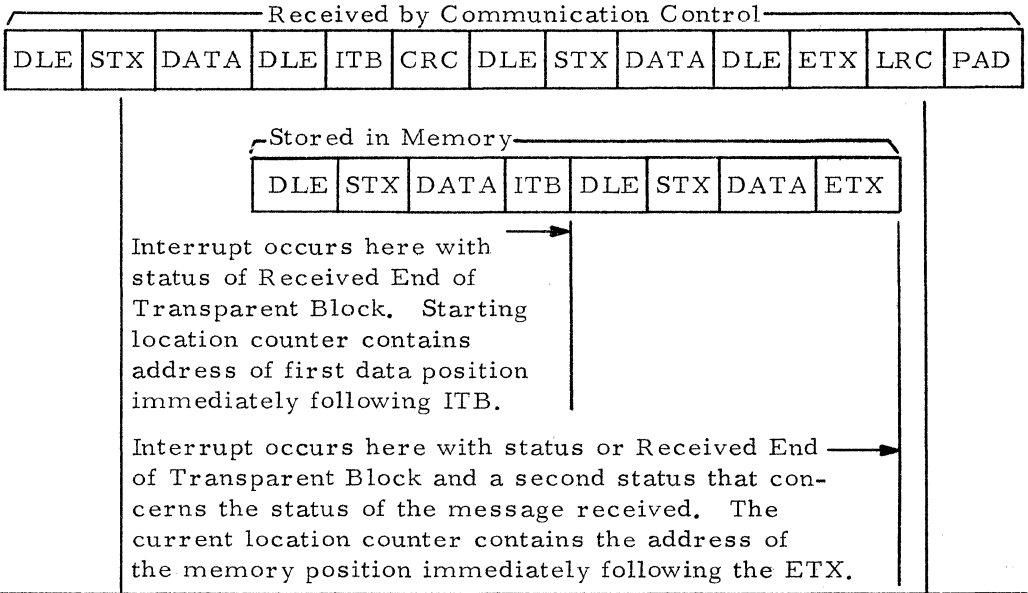
Instruction	Function	C3 Octal Code
		
Frame Demand In and No Order Stored	A positive response to this instruction informs the central processor that the communication control has a character in its buffer and there is no RWC assigned to take it into memory. To correct this situation, the program must issue a Receive PDT instruction to bring in the data being received. A positive response to this instruction coupled with a positive response to the Sent Illegal Sequence PCB instruction indicates an improperly placed record mark during reception.	61
Interrupt	This instruction tests the communication control for an interrupt-initiated condition. If the communication control has caused the interrupt, the program branches to the A address of the PCB instruction. If the communication control did not cause the interrupt, the program continues in normal sequence. (Usually the next instruction will be an Interrupt PCB addressed to another peripheral device.)	75

Table 4-3. Control PCB Instructions

Instruction	Function	C3 Octal Code
Disconnect	This instruction is issued to disconnect the connection between the local and remote communication controls when communication has been completed.	22
Send Abort	This instruction aborts the message being transmitted. It also causes an interrupt that can be tested by the Sent Abort status PCB instruction.	46
Initialize	This instruction simulates the Initialize switch on the control console for the unit addressed by the PCB instruction. It must be coded such that it branches to the next instruction to be executed after initialization, and not to the Initialize PCB instruction itself.	56
Set ASCII Error Control for VRC/CRC	These two PCB instructions select the error control procedures used in communication controls made to work with the ASCII code set. They are not recognized by communication controls that are made to work with either EBCDIC or Six-Bit Transcode.	62
Set ASCII Error Control for VRC/LRC		64
Test Receive	This PCB instruction is used in conjunction with test mode procedures described on the following pages.	63
Test Transmit	This PCB instruction is used in conjunction with test mode procedures described on the following pages.	66
Turn On Allow	This instruction allows the communication control to interrupt the central processor when the control requires service.	71
Turn Off Allow	This instruction prevents the communication control from interrupting the central processor.	70
Turn Off Interrupt	This instruction turns off all interrupts and should not be issued until the cause of interrupt has been determined.	74

PERIPHERAL DATA TRANSFER (PDT) INSTRUCTION

Format

The general format for PDT instructions is illustrated in Figure 4-2. The significance of the control characters is outlined below the figure.

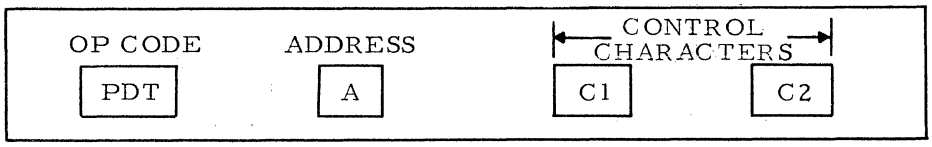
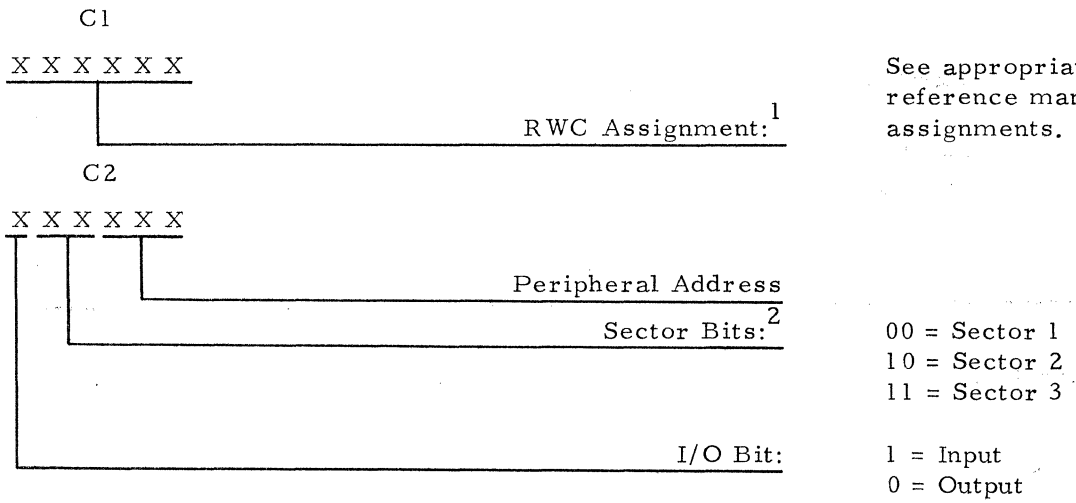


Figure 4-2. PDT Instruction Format



TESTING THE 281-2BSC

The 281-2BSC is provided with a built-in test unit that allows the program to transmit or receive through the 281-2BSC in a local-loop mode as shown in Figure 4-3. The program can transmit through the 281-2BSC (using one read/write channel) and receive the transmitted data through the test unit (using a second read/write channel). The direction of transmission can be reversed to test the reception of the 281-2BSC.

[1] RWC assignment depends upon the processor in use and the I/O sector to which the control is connected.

[2] Sector bits apply only to Model 1015 and larger Series 200 processors; these bits specify the sector in which the peripheral control is connected. The designated sector must include the read/write channel assigned in control character C1. Sector bits must be zeros on processors smaller than the Model 1015.

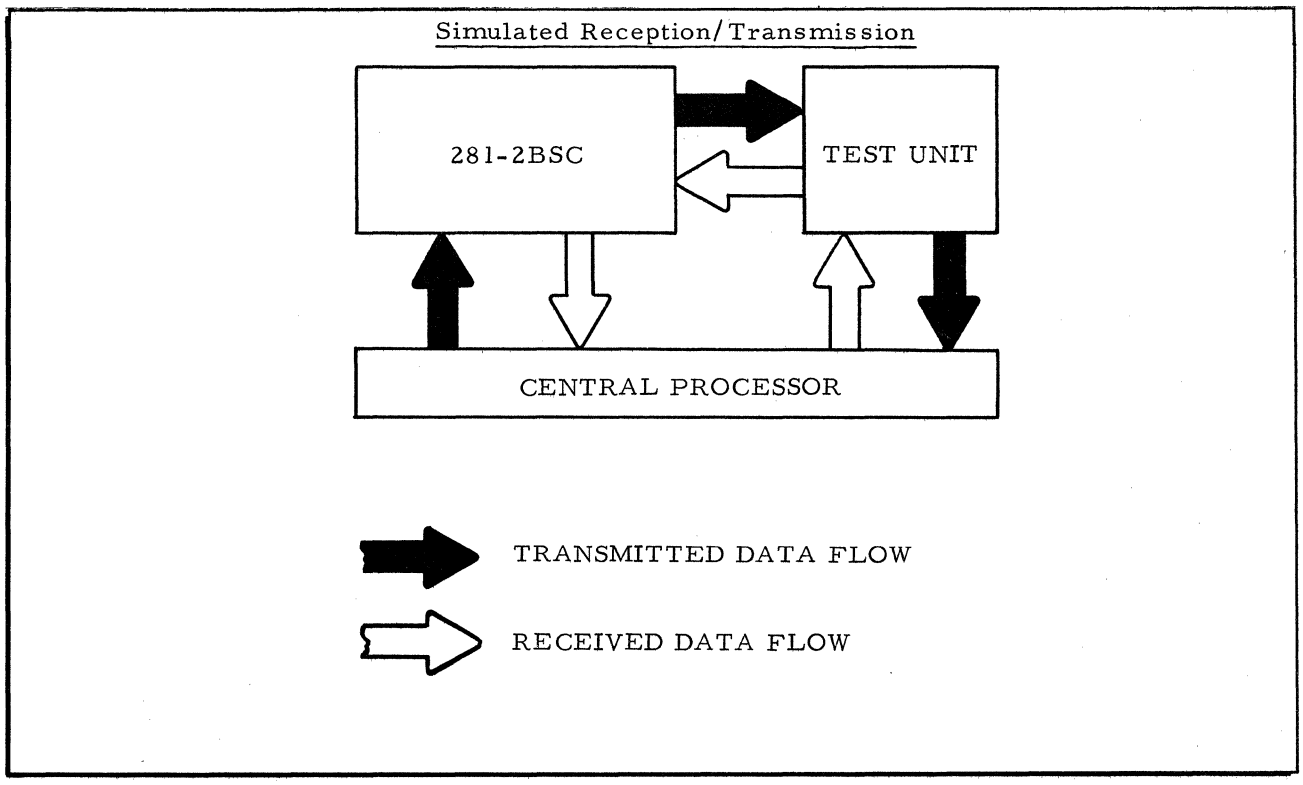


Figure 4-3. Test Loop for Type 281-2BSC Communication Control

Simulated Transmission

The transmit routine is started in the normal manner by issuing a Transmit PDT instruction to the 281-2BSC. This is immediately followed by a Test Transmit PCB instruction. The PCB is followed by a second Transmit PDT instruction issued to the 281-2BSC with a different read/write channel assignment of the same speed as the read/write channel assigned on the first Transmit PDT instruction. This places the 281-2BSC in the test mode, in which it extracts from memory the message at the A address of the first Transmit PDT instruction, using the read/write channel assigned by that instruction, and transmits the message to the test unit. The test unit receives the message and places it into the buffer area at the A address of the second Transmit PDT instruction, using the read/write channel assigned by the second Transmit PDT instruction. It is very important to bear in mind that these two read/write channels are separate. They are not the same read/write channel.

The 281-2BSC operates normally, as though the test unit were the data set. The test unit sends all the characters transmitted by the 281-2BSC to memory without any alteration.

Simulated Reception

The receive routine is started by issuing a Receive PDT instruction to the 281-2BSC, followed immediately by a Test Receive PCB instruction. The A address of the Receive PDT instruction must designate the beginning of the message that the programmer wishes to be transmitted to the 281-2BSC. The test unit starts transmitting the message to the 281-2BSC.

Meanwhile, the 281-2BSC is in its normal receive mode, scanning the incoming data. When it recognizes two consecutive SYN characters in the data transmitted by the test unit, it interrupts the central processor with an Initial Sync Received signal. The central processor recognizes this interrupt in the normal manner. Upon recognition of this condition, the program must issue a Receive PDT instruction having a second read/write channel assignment. It cannot be overemphasized that two different read/write channel assignments are required: one for the first Receive PDT instruction and another for the second Receive PDT instruction.

The 281-2BSC operates normally, as though the test unit were the data set. It performs all its normal character recognition, synchronization checks, BCC accumulations, etc. The test unit must receive a message from memory that appears to have originated at another BSC terminal; that is, the message must contain all the control functions that would have been inserted by a 281-2BSC communication control (SYN characters at 1-second intervals, special doubled DLE characters in the transparent mode, block check characters, etc.).

TESTING THE 285-2BSC

The Type 285-2BSC Communication Adapter is equipped with a built-in test unit that allows both simulated transmission and reception in a local-loop mode as shown in Figure 4-4. The test unit functions in full-duplex transmission and has its own address.

Simulated Transmission

When transmitting a message, data from the 286-6 or -7 multi-line communication control is applied to the adapter's buffer, where it enters a shift register. The 285-2BSC performs all its normal operations on the data and then sends it to the test unit.

From the test unit, data is transferred to the central processor where it is stored in a reserved memory area. The new stored data contains the block check character, SYN characters, and any other characters added by the communication adapter.

Simulated Reception

When receiving a message, data from the 286-6 or -7 communication control is applied to the test unit. The test unit transmits the data to the communication adapter. The adapter

then transfers the data to the 286-6 or -7 as it would any normally received data. During this simulated operation, the communication adapter performs all of its normal operations for data reception and sends the usual status reports to the 286-6 or -7 communication control.

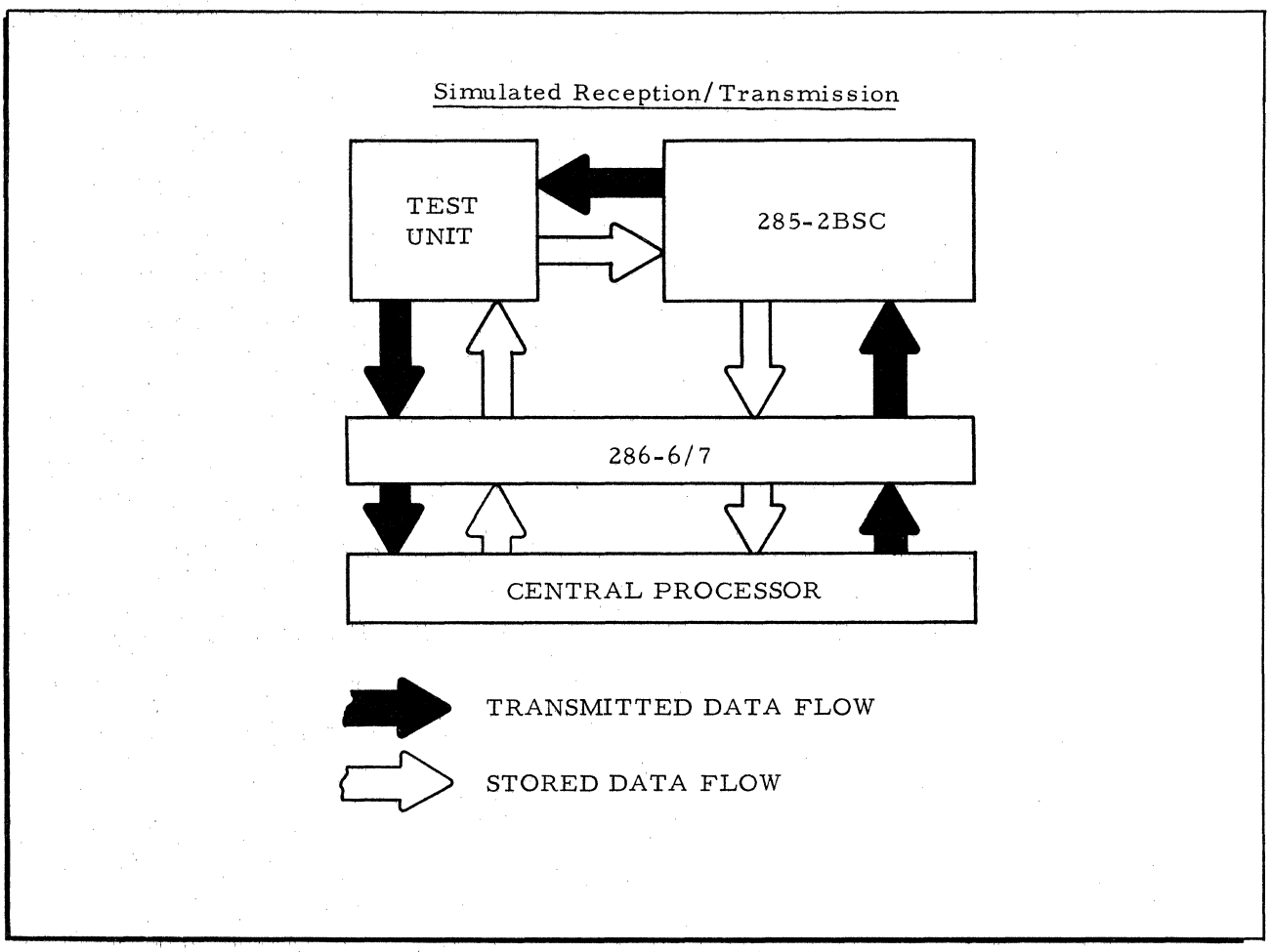


Figure 4-4. Test Loop for Type 285-2BSC Communication Adapter

SECTION V

285-2BSC SPECIAL CONSIDERATIONS

CONTROL SIGNALS

Received Illegal Sequence

The Type 285-2BSC Communication Adapter is provided with hardware format checking. Whenever the adapter receives an illegal sequence, it transmits this information to the 286-6 or -7 multi-line communication control. The program must reply with a NAK to any message received with a Received Illegal Sequence status.

Received Good Message

When the BCC accumulated in the receiving terminal compares with the transmitted BCC, the 285-2BSC recognizes a good message. This information is transmitted to the 286-6 or -7 after the BCC is received.

Received Message with Error

A Received Message with Error indication can appear under three conditions:

1. When a transmitted BCC does not compare with an accumulated BCC.
2. When a parity error is detected.
3. When a "continuous garbage condition" exists for 20 milliseconds.

The program must reply with a NAK to any message received with a Received Message with Error status.

Received Abort

This signal appears when an ENQ or DLE ENQ sequence is detected in the middle of the text. Upon receipt of this signal, the central processor must disregard the last block of data.

Sent Illegal Sequence

This signal is the transmission equivalent of Received Illegal Sequence. Upon receipt of this signal, the central processor must retransmit the last block of data.

Sent Message

This signal is the transmission equivalent of Received Good Message.

Delayed Answer

This signal is given when a 3-second timeout is not reset before 30 seconds expire. The program must respond with an ENQ sequence. After sending this signal the transmitting terminal waits in the Receiving mode.

Underflow

In the transmission mode, an underflow is detected when time-fills are sent for more than 20 milliseconds. When the central processor senses this signal, it may send an order to the 285-2BSC requesting it to send an abort sequence.

Overwrite

In the Reception mode, an overwrite condition exists when two characters overlap in the communication adapter buffer. As a response to this signal, the program must send a NAK sequence.

OPERATION

The Type 285-2BSC Communication Adapter operates under the same BSC conventions as the Type 281-2BSC Communication Control. All initial synchronization and framing characters must be supplied by the communications program in the central processor. If addressing is used, all addresses must be generated and recognized by the program.

APPENDIX A
MISCELLANEOUS INFORMATION

MESSAGE FORMATS

Proper formatting of BSC messages requires the use of the specifically-defined data-link control characters and specific formatting rules provided for heading and text data.

Headings

The heading is a block of data starting with an SOH and containing one or more characters that may be used for message control (for example, message identification, routing, priority, etc.). SOH initiates the block check character accumulation. (An initial SOH is not included in the accumulation.) A block of heading may be of fixed or variable length. The heading may be terminated by an STX.

Text

The text data is the most significant portion of the transmission. It is transmitted in complete units called messages, which are initiated by STX and concluded with ETX. Each message is a complete unit that can stand alone and is not necessarily directly related to other messages being transmitted. A message can be subdivided into smaller blocks for ease in processing and more efficient error control. Each block starts with STX and ends with ETB (except for the last block of a message, which ends with ETX). A single transmission can contain any number of blocks (ending with an ETB) or messages (ending with ETX). An EOT following the last ETX block indicates a normal end of transmission. Message blocking without line turnaround can be accomplished by using ITB.

Control characters or sequences within a block of text are not allowed. Any station receiving a control character within a text block treats the control character or sequence as a format error and may or may not continue receiving, depending upon which control character was received. Figures A-1 through A-7 show the proper formatting of BSC messages.

NOTE: In all figures Ø represents the SYN Characters.

TRANSPARENT TEXT OPERATION

This mode permits greater versatility in the range of coded data that can be transmitted. This is because all data, including the formally restricted data-link line-control characters, are treated only as specific bit patterns when transmitted in the transparent mode. Thus, unrestricted coding of data is permitted for transparent mode operation. All data-link control characters can be transmitted as transparent data without taking on control meaning.

Any data-link control characters must be preceded by a DLE to be recognized as a control function (see Figure A-8). All replies, inquiries, and headings are transmitted in normal mode.

SPECIAL REPLIES AND SEQUENCES

There are two special replies, WACK and RVI, that can be used in place of ACK or ACK1. WACK is used to indicate a temporary "not able to receive" condition; an example is shown in Figure A-9. RVI is used to inform the transmitting terminal that the receiving terminal has a high-priority message that it wishes to transmit; an example is given in Figure A-10.

There is also a special sequence used by a transmitting terminal when it wishes to retain control of the line but is temporarily unable to transmit the next block of data. An example of the use of the Temporary Text Delay (TTD) sequence (STX ENQ) is given in Figure A-11.

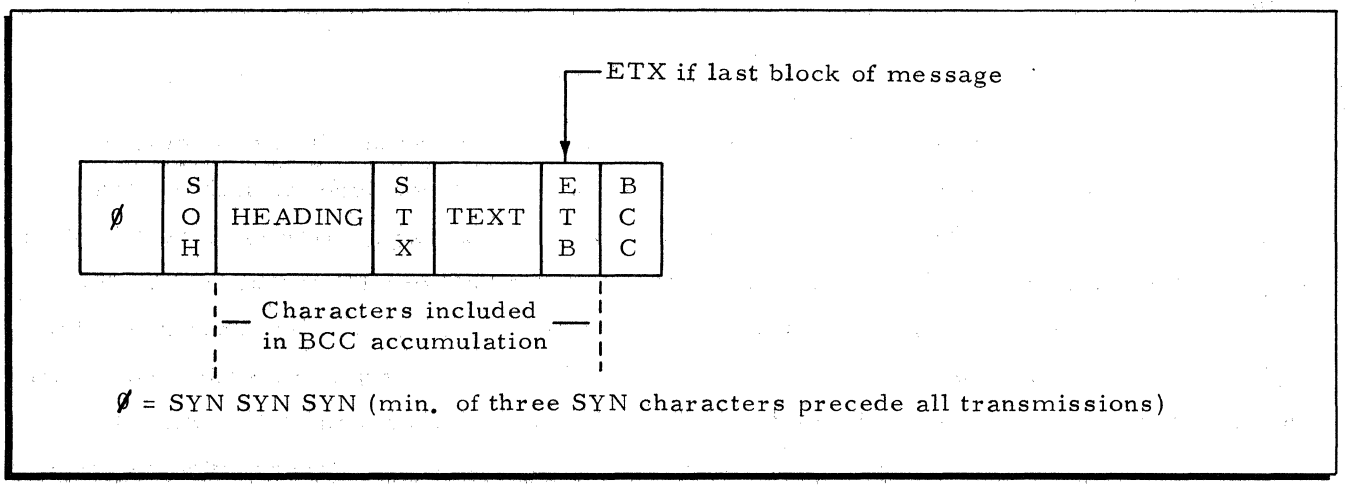


Figure A-1. Block Check Character Accumulation — Entire Transmission

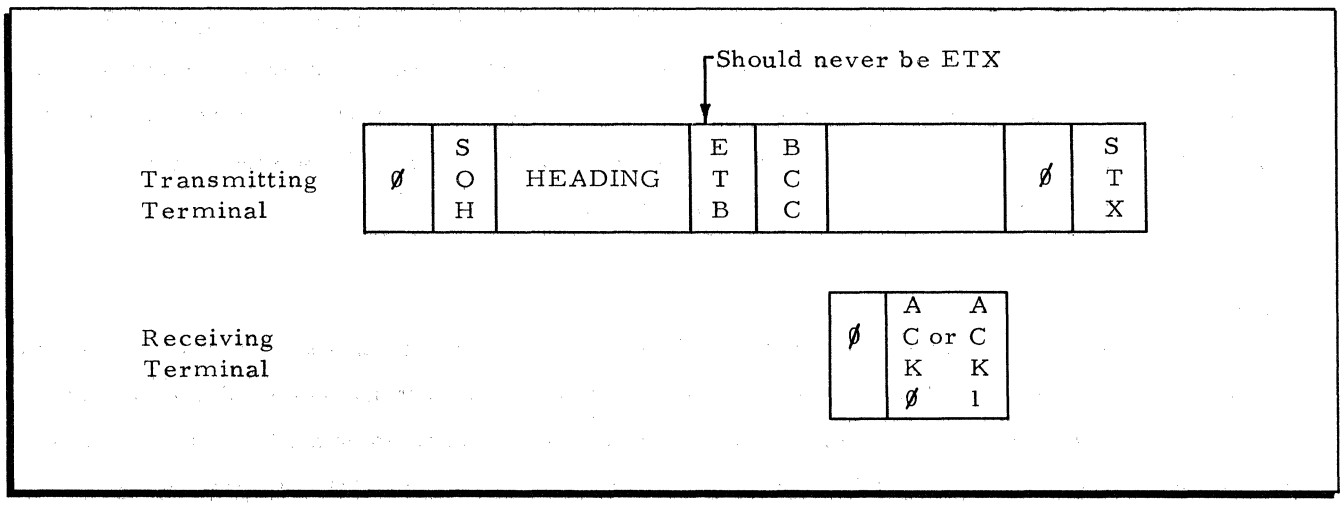


Figure A-2. Block Check Character Accumulation — Heading Only

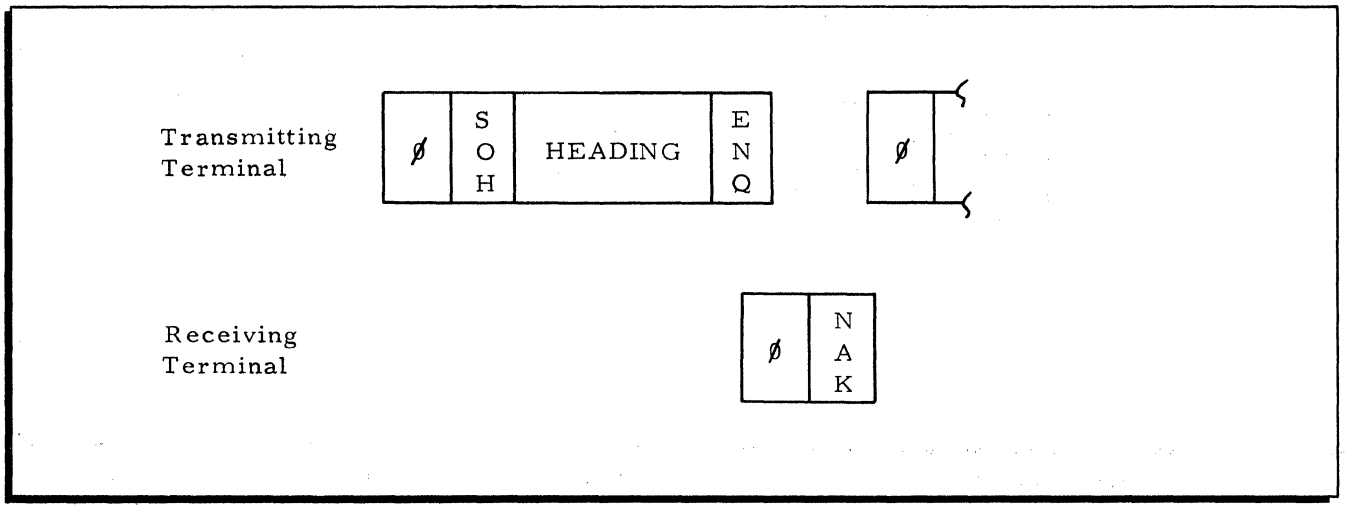


Figure A-3. Use of ENQ to Terminate Heading

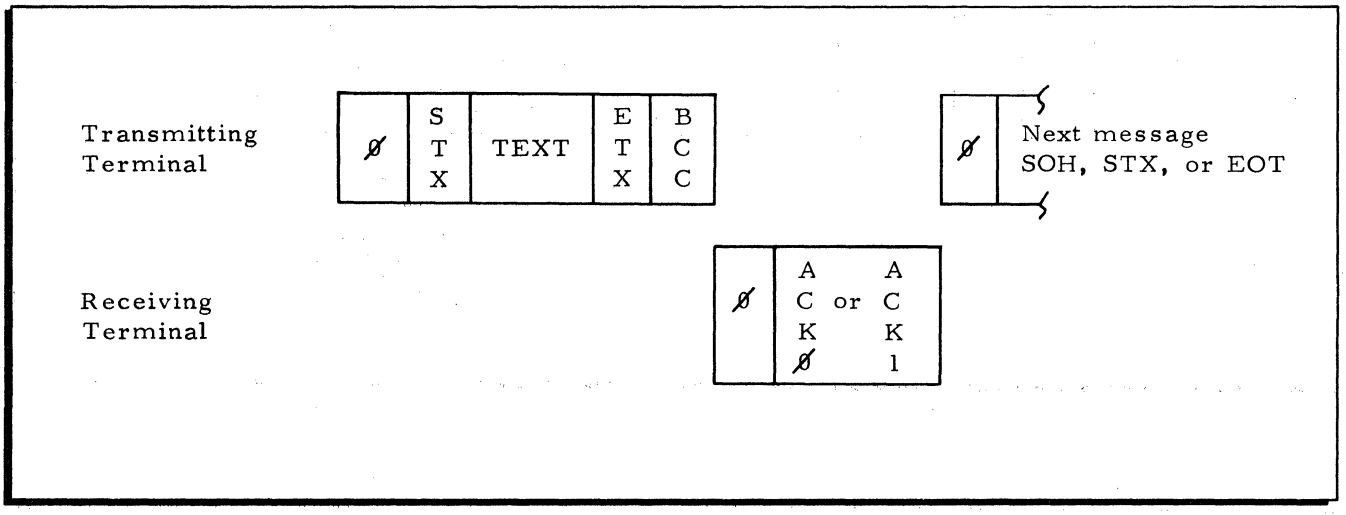


Figure A-4. Format of Last Block

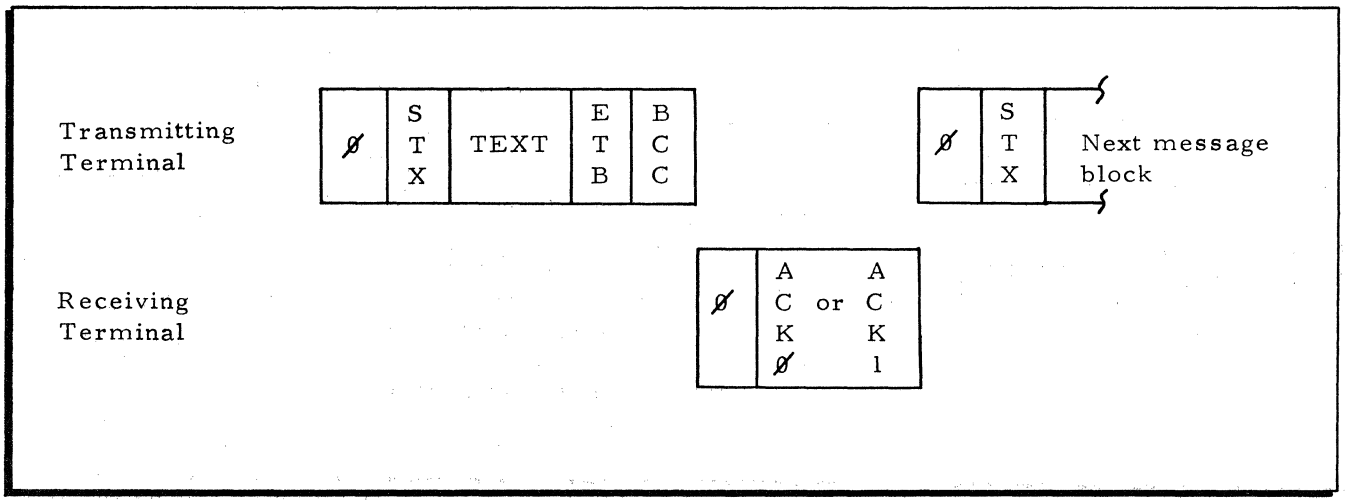


Figure A-5. Format of Normal Text Block

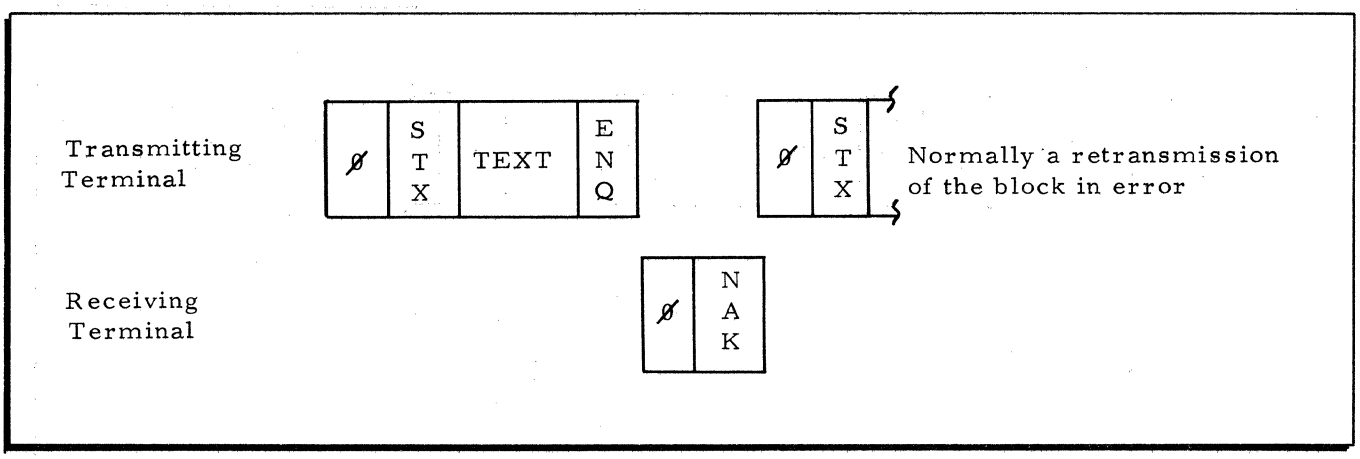


Figure A-6. Format of Block Ended with Forced Error Condition

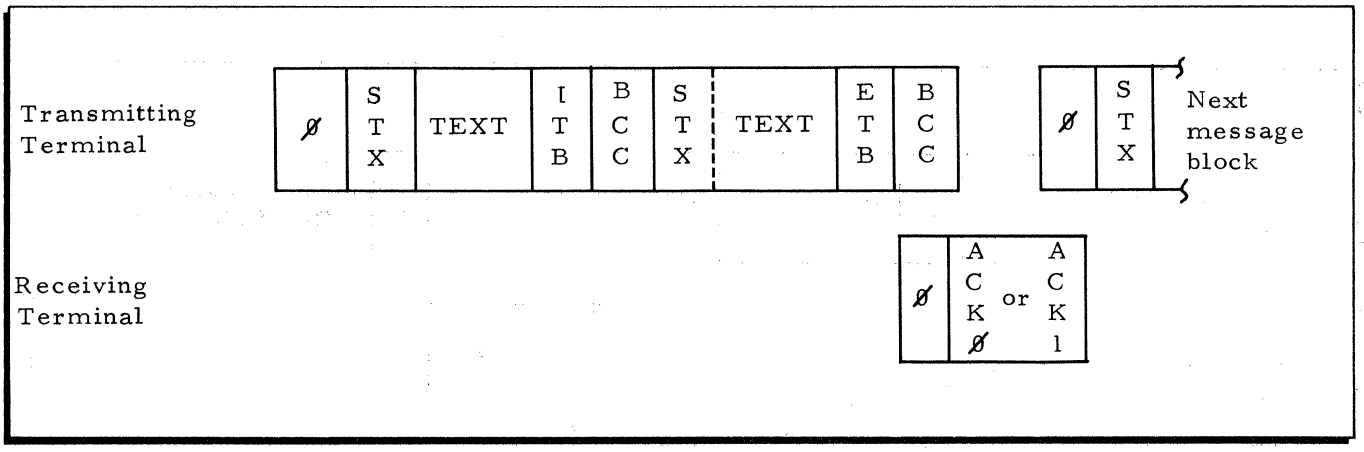


Figure A-7. Format of Block with Intermediate Blocking

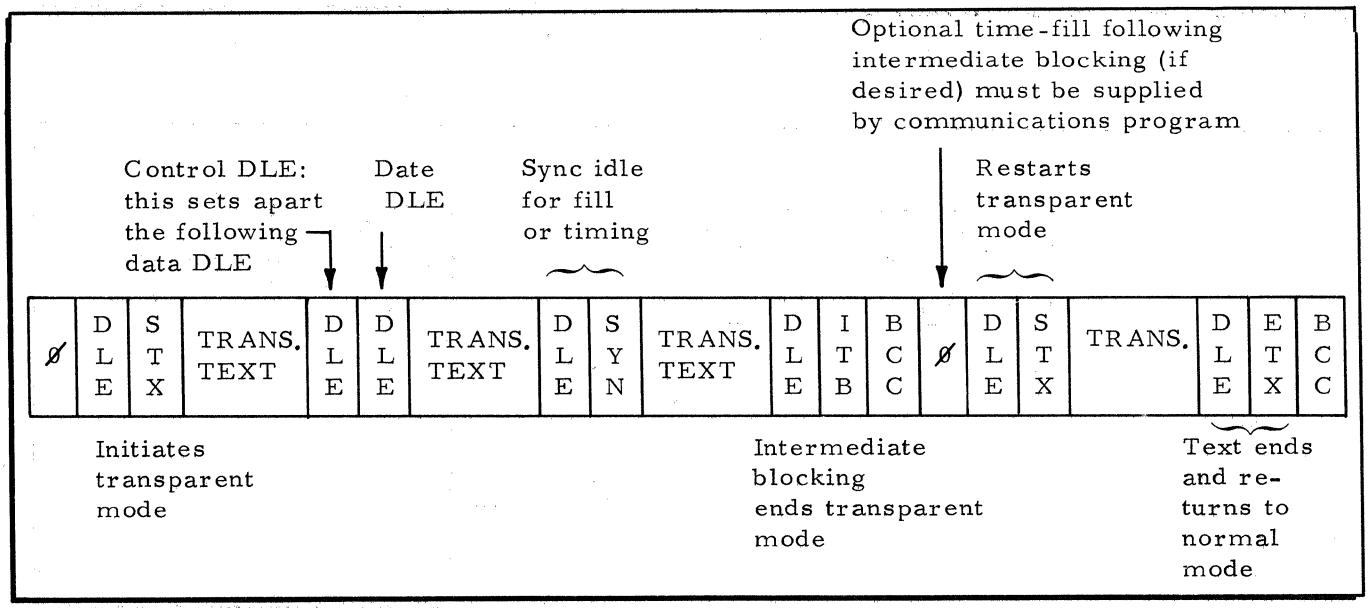


Figure A-8. Transparent Data Block

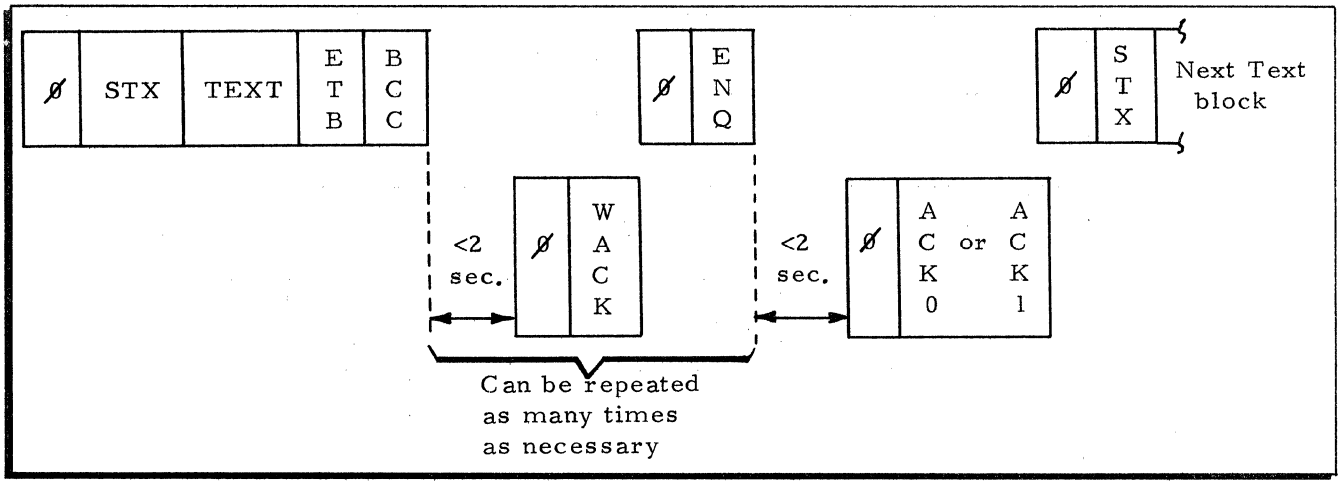


Figure A-9. Example of Use of WACK Reply

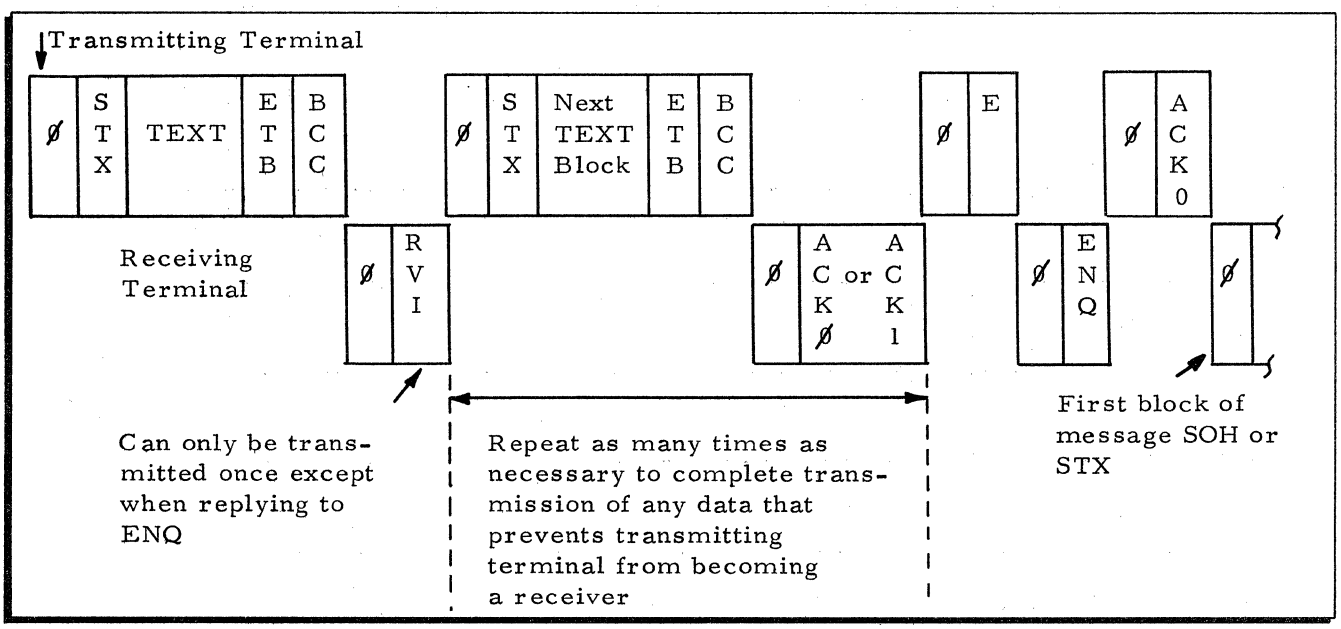


Figure A-10. Example of Use of RVI Reply

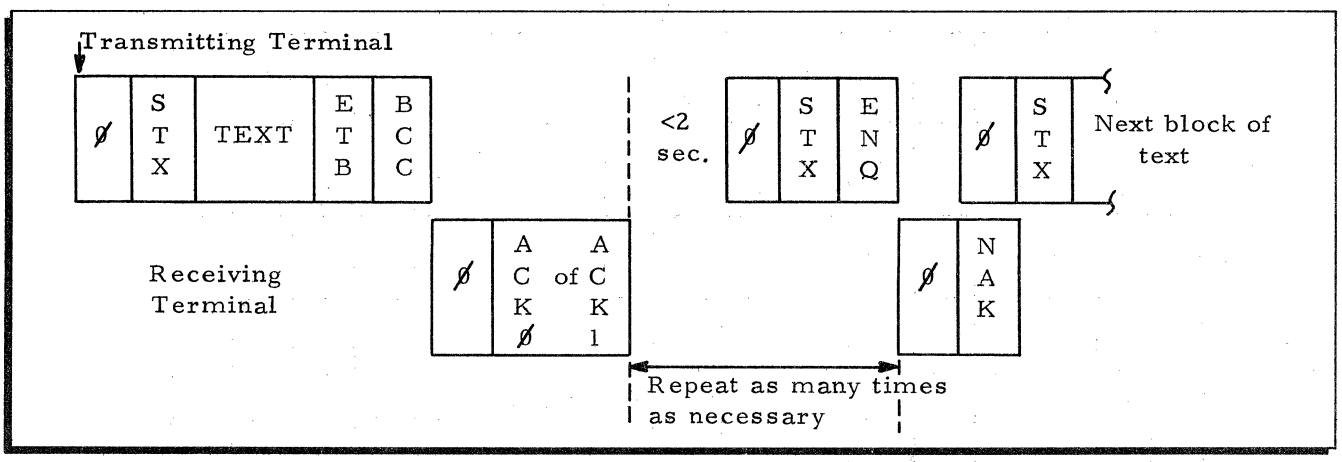


Figure A-11. Use of TTD Sequence

Table A-1. EBCDIC Character Assignments

Bit positions 4,3,2,1	Hex	Bit Positions 8, 7, 6, 5															
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	DLE	DS		SP	&	/						{	}	\	0
0001	1	SOH	DC1	SOS						a	j	~		A	J		1
0010	2	STX	DC2	FS	SYN					b	k	s		B	K	S	2
0011	3	ETX	DC3							c	l	t		C	L	T	3
0100	4	PF	RES	BYP	PN					d	m	u		D	M	U	4
0101	5	HT	NL	LF	RS					e	n	v		E	N	V	5
0110	6	LC	BS	EOB/ETB	UC					f	o	w		F	O	W	6
0111	7	DEL	IL	PRE/ESC	EOT					g	p	x		G	P	X	7
1000	8		CAN							h	q	y		H	Q	Y	8
1001	9	RLF	EM						\	i	r	z		I	R	Z	9
1010	A	SMM	CC	SM		¢	|	¦	:								
1011	B	VT				.	$	,	#								
1100	C	FF	IFS		DC4	<	*	%	@								
1101	D	CR	IGS	ENQ	NAK	(	)	_	'								
1110	E	SO	IRS	ACK		+	;	>	=								
1111	F	SI	IUS	BEL	SUB	|	¬	?	"								

NOTES: 1. ⊘ = Duplicate Assignment
2. ACK = In BSC procedures, ACKØ and ACK1 are used in place of ACK.
   a. ACKØ is used in the 2-character sequence DLE '70' where '70' is a hexadecimal number
   b. ACK1 is used in the 2-character sequence DLE/.

#F11

Table A-2. ASCII Character Assignments

		Bit Positions 8, 7, 6, 5															
		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
Bit Positions 4,3,2,1	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	DLE	SP	0	@	P		p								
0001	1	SOH	DC1	!	1	A	Q	a	q								
0010	2	STX	DC2	"	2	B	R	b	r								
0011	3	ETX	DC3	#	3	C	S	c	s								
0100	4	EOT	DC4	$	4	D	T	d	t								
0101	5	ENQ	NAK	%	5	E	U	e	u								
0110	6	ACK	SYN	&	6	F	V	f	v								
0111	7	BEL	ETB	'	7	G	W	g	w								
1000	8	BS	CAN	(	8	H	X	h	x								
1001	9	HT	EM	)	9	I	Y	i	y								
1010	A	LF	SUB	*	:	J	Z	j	z								
1011	B	VT	ESC	+	;	K	[	k	{								
1100	C	FF	FS	,	<	L	\	l	¦								
1101	D	CR	GS	-	=	M	]	m	}								
1110	E	SO	RS	.	>	N	¬	n	~								
1111	F	SI	US	/	?	O	_	c	DEL								

NOTES: 1. ASCII = American Standard Code for Information Interchange. ASCII code is transmitted with bit 8 assigned odd parity; the control generates the parity bit. When receiving, bit 8 (parity) is transferred to memory.

2. ACK = In BSC, ACK is replaced by the alternating responses, ACKØ (DLE Ø) and ACK1 (DLE 1).

Table A-3. Six-Bit Transcode Character Assignments

		Bit Positions 6,5			
		00	01	10	11
Bit Positions 4,3,2,1	Hex	0	1	2	3
0000	0	SOH	&	-	0
0001	1	A	J	/	1
0010	2	B	K	S	2
0011	3	C	L	T	3
0100	4	D	M	U	4
0101	5	E	N	V	5
0110	6	F	O	W	6
0111	7	G	P	X	7
1000	8	H	Q	Y	8
1001	9	I	R	Z	9
1010	A	STX	SPACE	ESC	SYN
1011	B	.	$	,	#
1100	C	<	*	%	@
1101	D	BEL	US	ENQ	NAK
1110	E	SUB	EOT	ETX	EM
1111	F	ETB	DLE	HT	DEL

Table A-4. Control Character Conversion Chart

Data-Link Character	Code Chart Sequence		
	EBCDIC	USASCII	Six-Bit Transcode
SYN	nc	nc	nc
SOH	nc	nc	nc
STX	nc	nc	nc
ETB	EOB (ETB)	nc	nc
ETX	nc	nc	nc
EOT	nc	nc	nc
ENQ	nc	nc	nc
ACKØ	DLE'70'	DLE Ø	DLE -
ACK1	DLE /	DLE 1	DLE T
NAK	nc	nc	nc
DLE	nc	nc	nc
ITB	IUS	US	US
WACK	DLE ,	DLE ;	DLE W
RVI	DLE @	DLE <	DLE 2
TTD	STX ENQ	STX ENQ	STX ENQ

NOTES: 1. nc - no change.

2. ' ' - Indicates the hexadecimal representation (no graphic assignment).

#F11

CUT ALONG LINE

HONEYWELL INFORMATION SYSTEMS
Technical Publications Remarks Form*

TITLE: SERIES 200 TYPE 281-2BSC COMMUNICATION CONTROL AND TYPE 285-2BSC COMMUNICATION ADAPTER

ORDER NO: 0-F11

DATED: MARCH, 1971

ERRORS IN PUBLICATION:

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION:

(Please Print)

FROM: NAME _____ DATE _____

COMPANY _____

TITLE _____

ADDRESS _____

_____

*Your comments will be promptly investigated by appropriate technical personnel, action will be taken as required, and you will receive a written reply. If you do not require a written reply, please check here. ☐

CUT ALONG LINE

FOLD ALONG LINE

POSTAGE STAMP NOT NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL: FIRST CLASS PERMIT NO. 39531, NEWTON HIGHLANDS, MASS.

POSTAGE WILL BE PAID BY:

HONEYWELL INFORMATION SYSTEMS

60 WALNUT STREET
WELLESLEY HILLS, MASS. 02181

ATTN: U.S. GROUP PUBLICATIONS, MS 050

FOLD ALONG LINE

Honeywell

The Other Computer Company:

Honeywell

HONEYWELL INFORMATION SYSTEMS

1399
2371
Printed in U.S.A.