GA18-2081-1
File No. S360/S370/S3/4300/8100-09

Systems

IBM 3270 Information Display System

3276 Control Unit Display Station Description and Programmer's Guide

IBM

GA18-2081-1
File No. S360/S370/S3/4300/8100-09

IBM 3270 Information Display System

3276 Control Unit Display Station Description and Programmer's Guide

Systems

IBM

Warning: This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions manual, may cause interference to radio communications. It has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment. Operation of this equipment in a residential area is likely to cause interference, in which case, the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

Second Edition (May 1983)

This major revision obsoletes GA18-2081-0. Changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

Changes are made periodically to the information herein, before using this publication in connection with the operation of IBM systems, consult the latest *IBM System/370 Bibliography*, GC20-0001, for the editions that are applicable and current.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address given below. Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Department 812J, 1133 Westchester Avenue, White Plains, New York, 10604, U.S.A. IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Preface

This publication provides management, programmers, and system analysts with detailed reference material relating to the IBM 3270 Information Display System 3276 Control Unit Display Station.

Organization of This Publication

This manual is organized into the following chapters:

Chapter 1, 3276 Control Unit and Data Streams.

Chapter 2, Terminals. This chapter, divided into two main sections, provides general information about displays and printers. The "Display" section presents detailed information about display fields, keyboards, selector-light-pen operations, the security keylock, and the magnetic card reading device. The "Printer" section discusses printer capabilities and control, including formatting, orders, buffered operation, SNA character string, and copy functions. (See also *IBM 3230 Printer Product Description*, GA24-3759, *IBM 3262 Printer Component Description*, GA24-3741, *IBM 3268 Printer Component Description*, GA27-3268, *IBM 3178 Display Station Description*, GA18-2127, *IBM3270 Information Display System: Color and Programmed Symbols*, GA33-3056, *IBM 3287 Printer Component Description*, GA27-3135, and *IBM 3289 Line Printer Component Description*, GA27-3176.)

Chapter 3, Remote Operations–BSC, treats the 3276 Models 1, 2, 3, and 4.

Chapter 4, 3276 SNA/SDLC Communication, describes SNA and SDLC protocols for the 3276. It also presents SNA reference data applicable to the machine.

Chapter 5, Screen Design, introduces important 3270 concepts. Shows an example of what a 3270 display message might look like, what coding elements are required to write this message in your program, and how terminal operator input might be handled.

Chapter 6, Screen Management, suggests macro definitions and programming routines that might be written to encode and decode messages to and from the display.

Reference material is arranged in the following appendixes:

Appendix A. Indicators and Controls

Appendix B. Buffer Address I/O Interface Codes

Appendix C. Status Indicator Codes

Appendix D. APL/Text Feature

Appendix E. Katakana Feature

Appendix F. Encrypt/Decrypt Feature

Appendix G. Record Formatted Maintenance Statistics (RECFMS) Formats

Related Publications

Information concerning the Multiuse Communications Loop, used to attach 3270 devices to 8100 Information Systems, is contained in:

- *IBM 8100 Information System: Communications, Loop, and Display/Printer Attachment Description*, GA27-2883
- *IBM Multiuse Communications Loop Planning Guide*, GA23-0033
- *IBM Multiuse Communications Loop Installation Guide*, GA23-0039

The two Multiuse Communications Loop publications referred to above and the following IBM 4300 Processor publications provide information concerning attachment of the 3276 to the 4331 Processor via the 4331 loop adapter:

- *IBM 4300 Processors Summary and Input/Output & Data Communications Configurator*, GA33-1523
- *IBM 4331 Processor Functional Characteristics and Processor Complex Configurator*, GA33-1526

Publications describing the printers and displays attaching to the 3276, the 3270 data stream, the use of color, and programming information, are listed in the publication *IBM 3270 Information Display System, Library User's Guide*, GA23-0058.

Contents

Figures

Chapter 1. 3276 Control Unit and Data Streams

Introduction

The IBM 3276 Control Unit Display Station is one of the basic units of the 3270 Information Display System Family. The 3276 offers the user a wide selection of components and configurations. Also available are a variety of features which improve performance, provide additional operational capability, and permit expansion of the display system. (The features are described in the publication *IBM 3270 Information Display System: Configurator*, GA27-2849.)

All models of the 3276 can be selected to form 3270 system configurations attachable to System/360, System/370, 303X Processor, 308X Processor, System/3, 4300 Processor, 8100 Information System, and 3790 Communication System configurations as host systems. (See *An Introduction to the IBM 3270 Information Display System*, GA27-2739, for possible system combinations.)

The 3276 is a table-top CRT display station and control unit used for displaying alphameric data up to a maximum of 3440 characters, and for entering data into, and retrieving data from, a host system. The 3276 can be ordered to control up to seven display stations and printers. The 3276 includes one self-contained display which allows a maximum 3276 cluster size of eight terminals.

The base 3276 provides one additional port for attachment of either display stations or printers. Up to three additional terminal adapters can be ordered. Each adapter has two ports which can attach display stations or printers in any combination. A keyboard is needed on every 3276.

The display station provides image display of data transmitted from the host system. A display station with an attached keyboard enables the user to enter, modify, or delete data on the display and to cause the revised data to be returned to the host system for storage or additional processing.

The 3276 can display up to 3440 characters per screen as follows:

- Models 1 and 11 display 960 characters.
- Models 2 and 12 display 1920 characters.
- Models 3 and 13 display 2560 characters.
- Models 4 and 14 display 3440 characters.

When operating in 3277-compatible format, the 3276 Model 1 will display 480 characters (40 characters per line), and Models 2, 3, and 4 will display 1920 characters (80 characters per line).

For EBCDIC and ASCII, the 3276 has a 94-character set (plus space and null).

The 3276 can be attached remotely to a host system (see Figure 1-1). Remote attachment employs common carrier (or equivalent customer) facilities of unlimited length to communicate between the host and the 3276. (The 3276 cannot be attached locally to a host system.) All models of the 3276 can also communicate with a 3704/3705 Communications Controller or a Communications Adapter feature installed in the 4331 without need for communication facilities or a modem (direct connection).

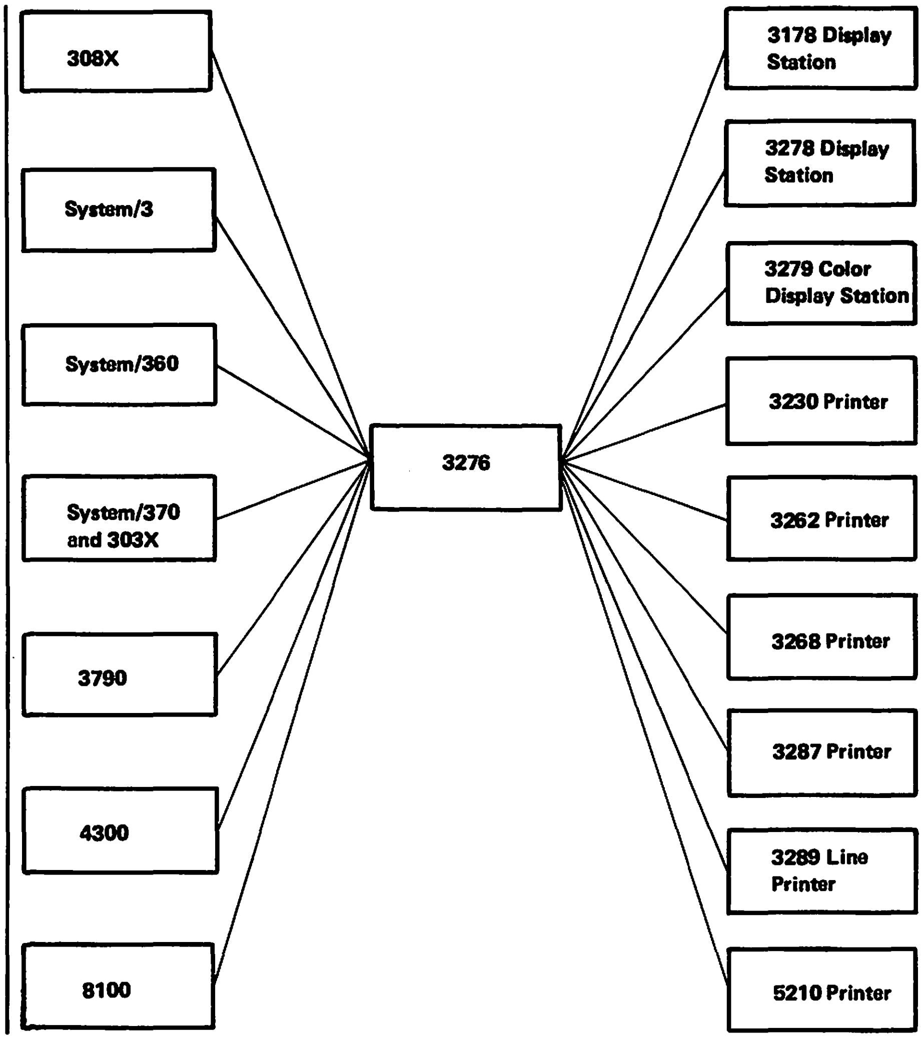


Figure 1-1. Host Control Unit and Device Combinations

The 3276 Models 1, 2, 3, and 4 attach via modems and operate via BSC line protocol at 1200, 2000, 2400, 4800, and 7200 bps. When the models are directly connected to a 3704/3705 Communications Controller, communication speed is limited to 1200 bps.

Models 11, 12, 13, and 14 attach via modems and operate via SDLC line protocol at 1200, 2000, 2400, 4800, 7200, and 9600 bps. When the models are directly connected to the 3704/3705 Communications Controller, communication speed is limited to 1200 bps.

Note: *The 3276 Models 1, 2, 3, and 4 with the SDLC/BSC Switch feature installed can also operate via SDLC protocol at the same communication line speeds as the Models 11, 12, 13, and 14.*

Models 11, 12, 13, and 14 communicate with the 8100 Information System or the 4300 System via modems and an SDLC data link, a directly attached loop, or a data-link-attached loop. Models 1, 2, 3, and 4 (with the SDLC/BSC switch set to SDLC) can communicate with the 8100 system or the 4300 System via modems and an SDLC data link.

The printer provides a printed copy of data displayed at a display station or transmitted from the host system. (In this document, the 3230, 3262, 3268, 3287, 3289, and 5210 Printers are referred to as "terminal printers.")

The 3276 Models 1, 2, 3, and 4 process the host-initiated BSC Copy command. The host-initiated Copy command is used to transfer buffer data from one device to another device via the 3276 to which both devices are attached. After accepting a Copy command addressed to the "to" device, the 3276 initiates the data transfer from the "from" device. Upon transfer of the data to the 3276, the 3276 processes the data and transfers it to the "to" device.

In addition to processing the host-initiated Copy command, the 3276 (all models) also provides an operator-initiated local copy function, which permits direct data transfer from a display station to a printer attached to the same 3276. The local copy function is initiated when the display station operator presses the print key on the display station keyboard. The printer selection is determined by a print-control matrix (called a printer default matrix) in the 3276. The printer default matrix is determined by the physical attachment of the printers to the 3276 at power-on time. In this matrix, each display station is associated with the printer that has the next higher terminal address. Printer assignment can be changed by use of the IDENT key on the 3178, 3276, 3278, or 3279 keyboard.

If the 3276 Models 11, 12, 13, and 14, or the 3276 Models 1, 2, 3, and 4 are equipped with the BSC/SDLC Switch feature and the switch is in the SDLC position, the host-initiated copy function is executed when the host issues a write-type command with the WCC print bit set to 1. Printer selection and servicing of the local copy request proceed in much the same way as in the operator-initiated local copy function.

Interface Codes

Data, commands, and orders transmitted between the control unit and the host system are in the form of interface codes. Two different codes are used in the United States: extended binary-coded decimal interchange code (EBCDIC) and American National Standard Code for Information Interchange (ASCII). The EBCDIC codes are also used in the World Trade countries (ASCII is available only in the U.S.); refer to *IBM 3270 Information System: Character Set Reference*, GA27-2837, for details.

Figure 1-2 shows the United States EBCDIC interface codes for several control unit/ device combinations. Figure 1-3 shows the United States ASCII codes. Figure 1-4 shows the control character codes. Refer to Appendix E for the Katakana codes.

Device Addressing

The port addresses on the 3276 control unit are 00-07 (ports 0-7) in non-SNA and 02-09 (ports 0-7) in SNA.

Bits 0,1		00				01				10				11			
Bits 2,3		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
Bits 4567	Hex 1 / Hex 0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL				SP	&	-						{	}	\	0
0001	1		SBA					/		a	j	~		A	J		1
0010	2		EUA							b	k	s		B	K	S	2
0011	3		IC							c	l	t		C	L	T	3
0100	4									d	m	u		D	M	U	4
0101	5	PT	NL							e	n	v		E	N	V	5
0110	6									f	o	w		F	O	W	6
0111	7									g	p	x		G	P	X	7
1000	8	GE		SA						h	q	y		H	Q	Y	8
1001	9		EM	SFE					`	i	r	z		I	R	Z	9
1010	A					¢	!	¦	:								
1011	B					.	$	,	#								
1100	C	FF	DUP	MF	RA	<	*	%	@								
1101	D	CR	SF			(	)	—	'								
1110	E		FM			+	;	>	=								
1111	F				SUB	|	¬	?	"								EO

Notes:

1. *Character code assignments other than those shown within all outlined areas of this chart are undefined. If an undefined character code is programmed, the character that will be displayed or printed is a hyphen (-); hex code 60 will be returned on a subsequent read operation. For control units with Configuration Support C installed, undefined control codes from X'00' to X'3F' cause a negative response (SNA) or an Op Chk (BSC). IBM reserves the right to change at any time the character displayed or printed and the I/O interface code returned for an undefined character code.*
2. *CR, NL, EM, and FF control characters are displayed and printed as blank characters. The DUP and FM control characters are displayed as $\overline{*}$ and $\overline{;}$ respectively, and are displayed and printed as * and ; when operating in mono-case mode.*
3. *Bits 0 and 1 are assigned for the following characters: AID, attribute, write control (WCC), copy control (CCC), CU and device address, buffer address, sense, and status. Bits 0 and 1 are assigned so that each character can be represented by a graphic character within the solid outlined areas of the chart. See Figure 1-4.*
4. *For BSC data-link control characters, see Chapter 3. For the SCS control codes associated with the SNA Character String feature on terminal printers, see Chapter 2.*
5. *When operating in mono-case mode, the lowercase alphabetic characters are displayed or printed as uppercase characters.*

Figure 1-2. United States EBCDIC I/O Interface Codes for 3276 Unit and Attached Display Stations and Terminal Printers

Bits 4321	Hex 1	000	001	010	011	100	101	110	111
	Hex 0	0	1	2	3	4	5	6	7
0000	0	NUL		SP	0	@	P	`	p
0001	1		SBA	!	1	A	Q	a	q
0010	2		EUA	"	2	B	R	b	r
0011	3		IC	#	3	C	S	c	s
0100	4		RA	$	4	D	T	d	t
0101	5			%	5	E	U	e	u
0110	6			&	6	F	V	f	v
0111	7			'	7	G	W	g	w
1000	8			(	8	H	X	h	x
1001	9	PT	EM	)	9	I	Y	i	y
1010	A	NL		*	:	J	Z	j	z
1011	B			+	;	K	[	k	{
1100	C	FF	DUP	,	<	L	\	l	¦
1101	D	CR	SF	-	=	M	]	m	}
1110	E		FM	.	>	N	^	n	~
1111	F			/	?	O	_	o	

Notes:

1. *Character code assignments other than those shown within all outlined areas of this chart are undefined. If an undefined character code is programmed, the character that will be displayed or printed is a hyphen (-); code 2D will be returned on a subsequent read operation.*
   IBM reserves the right to change at any time the character displayed or printed and the I/O interface code returned for an undefined character code.
2. *CR, NL, EM, and FF control characters are displayed and printed as blank characters.*
   *The DUP and FM control characters are displayed as * and ; respectively, and are displayed and printed as * and ; when operating in mono-case mode.*
3. *AID, attribute, write control (WCC), copy control (CCC), CU and device address, buffer address, sense, and status characters are assigned as specified in Figure 1-4 so that each character can be represented by a graphic character within the solid outlined portion of this chart.*
4. *For BSC data-link control characters, see Chapter 3.*
5. *When operating in mono-case mode, the lowercase alphabetic characters are displayed or printed as uppercase characters.*

Figure 1-3. United States ASCII I/O Interface Codes for 3276 Unit and Attached Display Stations and Terminal Printers

Bits 2-7	Graphic	EBCDIC	ASCII
00 0000	SP	40	20
00 0001	A	C1	41
00 0010	B	C2	42
00 0011	C	C3	43
00 0100	D	C4	44
00 0101	E	C5	45
00 0110	F	C6	46
00 0111	G	C7	47
00 1000	H	C8	48
00 1001	I	C9	49
00 1010 {	¢	4A	–
	[	–	5B
00 1011		4B	2E
00 1100	<	4C	3C
00 1101	(	4D	28
00 1110	+	4E	2B
00 1111 {	|	4F	–
	!	–	21
01 0000	&	50	26
01 0001	J	D1	4A
01 0010	K	D2	4B
01 0011	L	D3	4C
01 0100	M	D4	4D
01 0101	N	D5	4E
01 0110	O	D6	4F
01 0111	P	D7	50
01 1000	Q	D8	51
01 1001	R	D9	52
01 1010 {	!	5A	–
	]	–	5D
01 1011	$	5B	24
01 1100	*	5C	2A
01 1101	)	5D	29
01 1110	;	5E	3B
01 1111 {	¬	5F	
	∧	–	5E

Bits 2-7	Graphic	EBCDIC	ASCII
10 0000	-	60	2D
10 0001	/	61	2F
10 0010	S	E2	53
10 0011	T	E3	54
10 0100	U	E4	55
10 0101	V	E5	56
10 0110	W	E6	57
10 0111	X	E7	58
10 1000	Y	E8	59
10 1001	Z	E9	5A
10 1010	¦ (EBCDIC)	6A	7C
10 1011	,	6B	2C
10 1100	%	6C	25
10 1101	—	6D	5F
10 1110	>	6E	3E
10 1111	?	6F	3F
11 0000	0	F0	30
11 0001	1	F1	31
11 0010	2	F2	32
11 0011	3	F3	33
11 0100	4	F4	34
11 0101	5	F5	35
11 0110	6	F6	36
11 0111	7	F7	37
11 1000	8	F8	38
11 1001	9	F9	39
11 1010	:	7A	3A
11 1011	#	7B	23
11 1100	@	7C	40
11 1101	'	7D	27
11 1110	=	7E	3D
11 1111	"	7F	22

Note: *The characters above are used as attribute, AID, write control (WCC), copy control (CCC), CU and device address, and buffer address. They are also used as status and sense, except by the 3274 and 3276 when operating in BSC. When any of these characters is transmitted to the program, the CU assigns the appropriate EBCDIC code. If transmission is in ASCII, the CU translates the EBCDIC code to ASCII code prior to transmission.*

To use this table to determine the hex code transmitted for an address or control character, first determine the values of bits 2-7. Select this bit configuration from the "Bits 2-7" column. The hex code that will be transmitted (either in EBCDIC or in ASCII) is to the right of the bit configuration.

Use this table also to determine equivalent EBCDIC and ASCII hex codes and their associated graphic characters. Graphic characters for the United States I/O interface codes are shown. Graphic characters might differ for particular World Trade I/O interface codes. Refer to IBM 3270 Information Display System: Character Set Reference, GA27-2837, for possible graphic differences when these codes are used.

Figure 1-4. Control Character I/O Codes

Data Stream

The 3276 data stream consists of user-provided data, commands, and orders which are transmitted between the control unit and the host system (Figure 1-5). Control information, which governs the movement of the data stream, is also transmitted. The control units can differ as to the type of commands and/or transmission protocols employed.

Commands are issued to initiate such operations as the total or partial writing, reading, and erasing of data in a selected device buffer. Control commands initiate control unit and/or terminal operations not involved with data transfer (except for status information). Orders can be included in write data streams, either alone or intermixed with display or print data.

Two types of orders are available – buffer control orders and printer format orders. Buffer control orders are interpreted and executed as they are received by the control unit. They are used to position, define, modify, assign attributes on a field and character basis, and to format data being written to a display character buffer; to erase selected unprotected data in the buffer; and to reposition the cursor. Printer format orders are initially stored in the printer character buffer as data and are interpreted and executed by the printer logic when encountered in the print operation.

The balance of this chapter describes the 3270 data stream. In-depth definition and discussion of the 3270 data stream is provided in the publication *IBM 3270 Information Display System, 3270 Data Stream Programmer's Reference,* GA23-0059.

The 3276 can operate under SNA protocol using SDLC line discipline. In the SNA/SDLC environment, attached 3178s, 3278s, or 3279s function as LU type 2. The data stream RU for a write-type command, for example, consists of the command code, buffer orders, and display data.

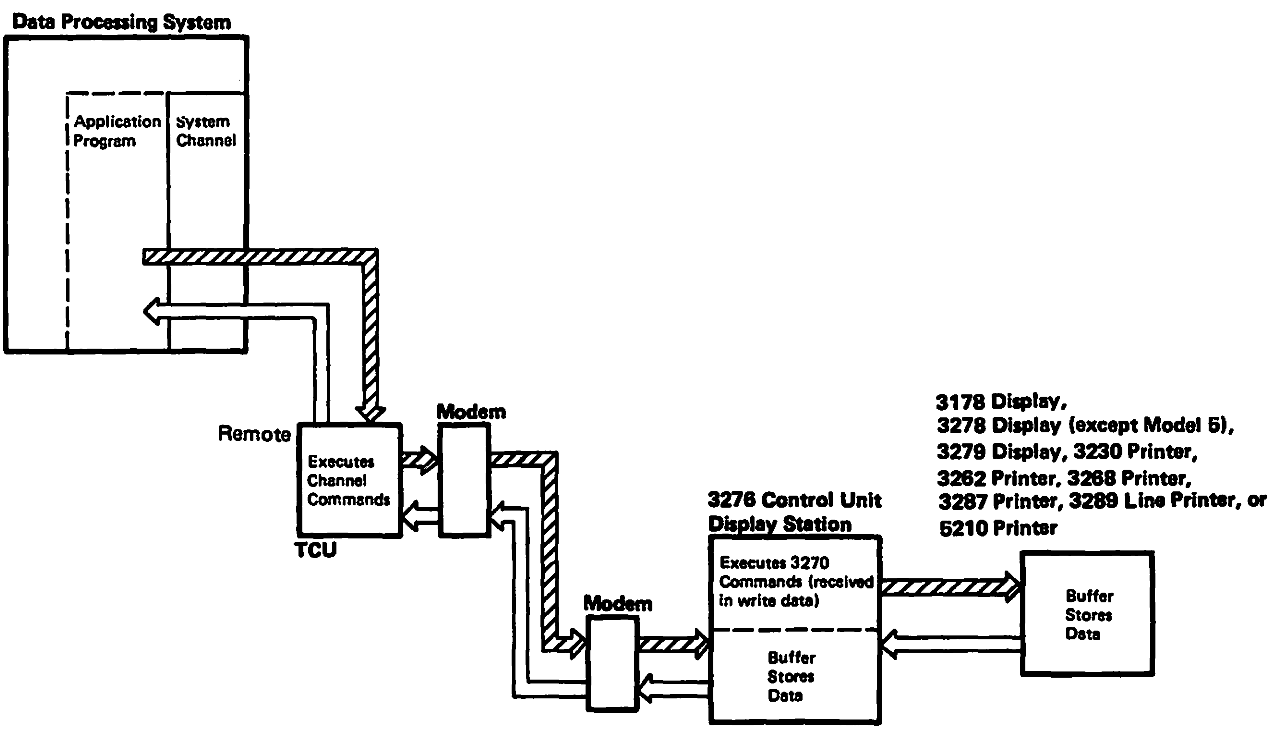


Figure 1-5. Data Flow between Data Processing System and the 3276

The terminal printers attached to a 3276 can also function in BSC or SNA/SDLC protocol. When operating in SNA/SDLC protocol, the terminal printers function as LU type 3. When SCS is installed, the printer functions as an LU type 1. The terminal printers can also operate as local copy devices; that is, data may be sent to a printer(s) from a display station attached to the same 3276, which functions in either BSC or SNA/SDLC discipline.

Commands

Three basic types of commands are used by the 3276:

1. Write commands, which are used to transfer data and orders from main storage to the 3276.

2. Read commands, which transfer 3276 buffer data, keyboard key data, and, for remote configurations, status information to main storage.

3. Control commands, which cause certain printer or display station operations.

Figure 1-6 lists the commands and associated codes that can be executed by the 3276.

Command	3276		
	EBCDIC Hex	ASCII Hex	Graphic
Copy[1]	F7	37	7
Erase All Unprotected	6F	3F	?
Erase/Write	F5	35	5
Erase/Write Alternate	7E	3D	=
Read Buffer	F2	32	2
Read Modified	F6	36	6
Read Modified All[2]	6E	3E	:
Write	F1	31	1

[1]Applicable to 3276-1 through -4 only.
[2]Applicable to 3276-11 through -14 only.

Figure 1-6. Command Codes

When a remotely attached 3276 is in operation, the rate at which data is transferred between the data processing system's main storage and the control unit depends on the type of transmission control unit and on the modems and communication facilities used. The 3276 accepts data from, and provides it to, the transmission control unit/communication facility at the byte rate established by the transmission control unit/communication facility.

All command operations that direct movement of data to and from the 3276 result in transfer of data between the control unit and a device buffer. When commands are not being performed, the control unit and the device buffer interact asynchronously, and the last image displayed by a previous command is continuously regenerated at a visible rate.

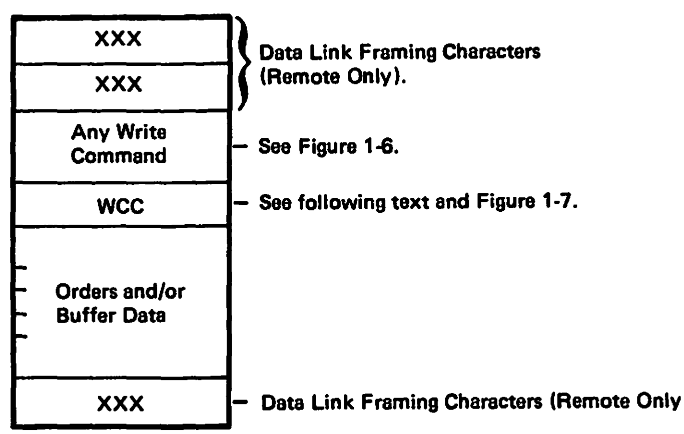
Write Commands

Two write-type commands, Write and Erase/Write, are used to load, format, and selectively erase device buffer data. These commands can also initiate certain device operations such as starting the printer, resetting the keyboard, and sounding the audible alarm. Write and erase/write operations are identical except that Erase/Write causes complete erasure of the device buffer before the write operation is started. Thus, Erase/Write is used to load the buffer with completely new data, whereas Write can be used to modify existing buffer data. Because of this, the 3276 initiates a device-to-control unit buffer transfer before Write command operations, but not before Erase/Write command operations.

A third write-type command, Erase/Write Alternate, performs the erase/write function for the display stations and the terminal printers. It is also used to switch the display or printer into large screen or expanded print capacity mode.
The Erase/Write Alternate command is valid when sent to the 3276.

Write Command

The bytes received by the 3276 for Write command operation consist of a command code, a write control character (WCC), and any orders and/or new buffer data needed to modify the existing buffer contents. Remotely attached 3276s also receive appropriate data link control framing. The sequence of bytes is as follows:



The minimum data stream following a Write command is a 1-byte WCC. This is ensured because the byte count field of the write channel control word (CCW) must be set to a minimum of 1 in BSC operations, or else the command code is not sent. The minimum Write command data stream to a remote 3276 consists of framing characters (e.g., in BSC, STX, ESC, and ETX) and the command code. To be meaningful, a WCC byte should follow the command code; if the BSC data link control character ETX follows the command code, an all-zero default WCC byte is generated by the control unit, and command execution is ended normally. An order or display/print data byte that immediately follows the command code is interpreted as a WCC by the control unit. The WCC byte format is as follows:

*	Reset Bit	Printout Format		Start Print	Sound Alarm	Key-board Restore	Reset MDT Bits
0	1	2	3	4	5	6	7

*Determined by the configuration of bits 2 through 7. See Figure 1-4.

Figure 1-7 describes the function of each WCC bit. When the WCC specifies an operation that does not apply to the selected device (for example, if the Sound Alarm bit is set and the selected device does not have the Audible Alarm feature), the specified operation is ignored. When the WCC byte is followed by order or display/print data bytes, only the Reset MDT Bits function, if specified, is performed before the write operation; any other WCC function is deferred until all data is written and all orders are performed.

Bit	Explanation
0	Determined by the contents of bits 1 through 7 as shown in Figure 1-4.
1	WCC reset bit.
2,3	Define the printout format, as follows: = 00 – The NL, EM, and CR[1] orders in the data stream determine print line length. Provides a 132-print position line when the orders are not present. = 01 – Specifies 40-character print line. = 10 – Specifies 64-character print line. = 11 – Specifies 80-character print line.
4	Start Printer bit. When set to 1, initiates a printout operation at completion of the write operation.
5	The Sound Alarm bit. When set to 1, sounds the audible alarm at the selected device at the end of the operation if that device has an audible alarm.
6	The Keyboard Restore bit. When set to 1, restores operation of the keyboard by resetting System Lock or Wait symbol on 3178, 3276, 3278, or 3279 displays. It also resets the AID byte at the termination of the I/O command.
7	Reset MDT bits. When set to 1, all MDT bits in the selected devices' existing buffer data are reset before any data is written or orders are executed.

[1] The CR order is applicable to the terminal printers only.

Figure 1-7. Write Control Character (WCC)

Orders and buffer data can follow the WCC character. (Orders are described later in this chapter, following the "Commands" description.) Buffer data can be written into any specified location of the buffer without erasing or modifying data in the other buffer locations. Data characters are stored in successive buffer locations until an order is encountered in the data stream which alters the buffer address, or until all the data has been entered. During the write operation, the buffer address is advanced one location as each character is stored.

The buffer location where data entry starts depends upon the following considerations:

1. The starting location may be specified by a Set Buffer Address order that follows the WCC. (This order is described later in this chapter under "Orders.")

2. The starting location will be the buffer address containing the cursor if the Write command is not chained or if it is chained from a Copy or Erase All Unprotected command.

3. The starting location will be the current buffer address if the Write command is chained from a Read or another Write command.

The formatting and placement of write data and the modification of existing buffer data are described under "Orders."

Programming Notes:

1. If the commands are being chained, the Write or Erase/Write command with the Start Print WCC bit set must be the last command in the chain. If not, remote control units perform the print operation and abort the next command.

2. The Printout Format bits are honored only if the Start Print bit is set in the same WCC.

3. In remote operations, if a Write command that includes data is chained from a previous Write command, a Set Buffer Address (SBA) order should immediately follow the WCC to define the starting location at which data entry is to start; this permits recovery in case of an error condition that requires retransmission of that data.

Programming Restriction: A Write command should not be chained from an Erase All Unprotected command. If it is, the operation is undefined.

Erase/Write Command

Execution of the Erase/Write command performs two operations: an erase operation and a write operation. The erase operation clears the entire device buffer to nulls, positions the cursor to character location 0, and resets the buffer address to 0.

Erase/Write then performs the write and WCC operations in the same manner as a Write command. If no WCC is sent, the Erase/Write command will not erase the buffer.

An Erase/Write command can also return a display or printer to the default screen size or character print capacity (as described under "Erase/Write Alternate Command").

Erase/Write Alternate Command

The display stations and the terminal printers with a capacity of 960 characters can function as 480-character devices; 1920-, 2560-, and 3440-character displays and printers can function as 1920-character devices.

For the 3276 BSC, a unique instruction is required from the application program to enable a display or printer to function at greater than 480- or 1920-default-character capacity. The Erase/Write Alternate command is used to switch a display station's screen size or a terminal printer's print capacity to the alternate size indicated by the display model number or specified for the printer as follows:

3178 Model	3230 and 3268 Model	3262 Model	3276 Model	3278 Model	3279 Model	3287 and 3289 Model	5210 Model	Default Character Capacity	Alternate Character Capacity
–	2	13	1,11	1	–	1,2	G01,G02	480	960
C1,C2	2	13	2,12	2	2A,2B	1,2	G01,G02	1920	1920
–	2	13	3,13	3	3A,3B	1,2	G01,G02	1920	2560
–	2	13	4,14	4	–	1,2	G01,G02	1920	3440

Note: *For SDLC machines, the default and the alternate character capacity are defined by the BIND parameter. Thus, the default and the alternate can be exchanged.*

The Erase/Write Alternate command also operates as an Erase/Write command. Once the display or printer is placed in alternate mode, operation continues in alternate mode until: the operator presses the CLEAR, SYS REQ (SNA only), or TEST key; or until an Erase/Write command is received, the SNA session is unbound, power fails at the control unit, display, or printer; or a system reset sequence occurs. Only these conditions return the display or printer to the default-value screen-size or character print capacity. For the 3276 SNA, the Erase/Write Alternate and Erase/Write commands are used to switch a display screen size, or a print capacity to alternate size, or vice versa, according to Bind parameter definition.

When in emulation mode, and with the display not in an LU-LU session, the operator may set the display to its maximum size by pressing the CLEAR key.

A 3178, 3276, 3278, or 3279 display operating as an LU type 2 requires the format shown in Figure 1-8 as part of the bind operation.

If an Erase/Write Alternate command is received while bound, it is processed as a normal Erase/Write command. No state change occurs within the display. Default screen sizes are as follows:

3178 Model	3278 Model	3279 Model	Default Screen Size Assumed with Byte 24 = b'0000000'
–	1	–	480 (12 x 40)
C1,C2	2	2A,2B	1920 (24 x 80)
–	3	3A,3B	1920 (24 x 80)
–	4	–	1920 (24 x 80)

Byte	Bit	Model	Content	Description
20	0–7			Default number of rows
		1	X'01' – X'0C'	1–12
		2	X'01' – X'18'	1–24
		3	X'01' – X'20'	1–32
		4	X'01' – X'2B'	1–43
		5	X'01' – X'1B'	1–27
21	0–7			Default number of columns
		1	X'28'	40
		1–5	X'50'	80
		5	X'84'	132
22	0–7			Alternate number of rows
		1	X'01' – X'0C'	1–12
		2	X'01' – X'18'	1–24
		3	X'01' – X'20'	1–32
		4	X'01' – X'2B'	1–43
		5	X'01' – X'1B'	1–27
23	0–7			Alternate number of columns
		1	X'28'	40
		1–5	X'50'	80
		5	X'84'	132
24	0–7			Session screen size
	0	All	Reserved	reserved
	1–7	1–5	b'000 0000'	Base default (12 x 40 or 24 x 80)
		1	b'000 0001'	Base Model 1 default (12 x 40)
		2–5	b'000 0010'	Base Model 2 default (24 x 80)
		1–5	b'111 1110'	Extended default (size specified in bytes 20 and 21)
		1–5	b'111 1111'	Extended alternate (size specified in bytes 22 and 23)

Note: *Row values outside these ranges and column values other than those listed cause the Bind to be rejected with X'0821'.*

Figure 1-8. LU Type 2 Screen Size Bind Format

Only a Model 1 display can be bound as b'0000001', a base LU type 2 with a 12 x 40 character screen. This coding of the Bind image is rejected with X'0821' on Models 2, 3, 4, and 5.

A Model 2, 3, 4, and 5 display can be bound as b'0000010', a Base LU type 2 with a 24 x 80 character screen. This Bind format, if used for a Model 1 display, causes the Bind to be rejected with X'0821'.

When operating with a screen size of 480 characters, sequential buffer addresses map to the 12 x 40 screen format in row major order. When operating in other screen sizes, sequential buffer addresses map to the defined screen format in row major order (Appendix B).

Byte 24 must be coded X'7E' or X'7F' to use displays in large-screen mode (2560 and 3440 characters) during the LU-LU session.

When bits 1 through 7 of byte 24 are coded X'7E', the screen size of the device is defined in bytes 20 and 21 of the Bind image, and bytes 22 and 23 are ignored. The device operates with the defined screen size during the entire session. An Erase/Write Alternate command is accepted by the device but is interpreted as an Erase/Write command. No state change occurs, and the screen size remains as defined in bytes 20 and 21 of the Bind image. Valid codings of bytes 20 and 21 are as follows:

		Model 1		Model 2	Model 3	Model 4
Byte	Hex	≤X'0C'		≤X'18'	≤X'20'	≤X'2B'
20	Row	≤12		≤24	≤32	≤43
Byte	Hex	X'28'	X'50'	X'50'	X'50'	X'50'
21	Col	40	80	80	80	80

If the Bind specifies an invalid number of columns, or if the number of rows is greater than the maximum row specified (above) for each model, the Bind will be rejected. Buffer wrap will occur at the end of the row specified in byte 20.

When bits 1 through 7 of byte 24 are coded b'0111111', a dynamic switch can be made during the session between a default screen size and an alternate screen size. When byte 24 is coded in this way, bytes 20 through 23 define the default and alternate screen sizes.

Valid codings of these bytes are as follows:

		Model 1		Model 2	Model 3	Model 4
Bytes	Hex	≤X'0C'		≤X'18'	≤X'20'	≤X'2B'
20 and 22	Row	≤12		≤24	≤32	≤43
Byte	Hex	X'28'	X'50'	X'50'	X'50'	X'50'
21 and 23	Col	40	80	80	80	80

The Bind is rejected if an invalid number of columns is coded in the Bind image or if the number of rows is greater than the maximum row value shown for each model (above). When in alternate-size mode, the display will wrap at the end of the row specified in byte 22 of the Bind image. When in default-size mode, the screen will wrap at the end of the row specified in byte 20 of the Bind image.

Once the Bind has taken place, the display is cleared and set to the default screen size and format. Request/Response Units (RUs) that contain SBA, RA, or EUA orders with addresses out of the range of the default screen size are rejected with -RSP (1005) (address out of range) response. Data will wrap at the default screen boundary whether input by the operator or from the outbound data stream, and wrapping will occur at the default screen boundary as defined for all other 3270 operations (for example, Erase All Unprotected, Read Buffer).

The Erase/Write Alternate command dynamically switches the display to the specified alternate screen size. Note that, on a Model 2 display, the Erase/Write Alternate command performs no meaningful function.

If bound to dynamically switch, the device assumes the characteristics of a display with the alternate screen size, upon receipt of an Erase/Write Alternate command. RUs that contain SBA, RA, or EUA orders that have addresses out of the range of the valid alternate screen size are rejected with -RSP (1005) (address out of range).

Write, Erase/Write, and Erase/Write Alternate Commands (LU Type 3)

The terminal printers can operate as LU type 3 and extended LU type 3. Commands and orders used by LU type 2 are applicable to LU type 3 and extended LU type 3 except for the read-type commands: Read Buffer, Read Modified, and Read Modified All. Read-type commands are rejected with -RSP (1003) (invalid command code.)

LU type 3 operations are directed by write-type commands. As specified in the Bind, printers that function as base LU type 3 operate as 480- or 1,920-character devices, and printers that function as extended LU type 3 operate with alternate buffer sizes of 960, 1920, 2560, 3440, or 3564 characters, or the full physical buffer. The alternate size is established by an Erase/Write Alternate command, and the default size is established by an Erase/Write command. Loss of power at the printer or the control unit or unbinding the session returns the printer to the default buffer size.

The WCC for LU type 3 and extended LU type 3 is shown in Figure 1-7. The function of bits 2, 3 (Printout Format), 5 (Sound Alarm), and 7 (Reset MDT bits) is the same as for LU type 2. When bit 4 (Start Print) is set to 1, the printer buffer content is printed after completion of the data transfer. Otherwise, printing does not occur after completion of the data transfer.

Buffered printers that operate as LU type 3 employ the format shown in Figure 1-9 as part of the Bind operation.

Byte 24 establishes the buffer size for both base and extended LU type 3 operations. The base LU type 3 operation supports a 480- or 1,920-character buffer only, using the Erase/Write command. To use larger printer buffer sizes, the Bind must specify Extended LU type 3 operation.

Byte	Bit	Content	Description
19	0–7	Reserved	
20	0–7		Default number of rows
		X'0C'	12
		X'18'	24
		X'1B'	27
		X'20'	32
		X'2B'	43
21	0–7		Default number of columns
		X'28'	40
		X'50'	80
		X'84'	132
22	0–7		Alternate number of rows
		X'0C'	12
		X'18'	24
		X'1B'	27
		X'20'	32
		X'2B'	43
23	0–7		Alternate number of columns
		X'50'	80
		X'84'	132
24	0	Reserved	
	1–7	Session Buffer Size	
		b'0000000'	Extended LU3 uses all available buffer space. No size is specified.
		b'0000001'	Base LU3, 12 x 40
		b'0000010'	Base LU3, 24 x 80
		b'1111110'	Extended LU3 static buffer size is defined in bytes 20, 21.
		b'1111111'	Extended LU, alternate sizes are indicated in bytes 22, 23.
		All other values are reserved and cause the Bind to be rejected with X'0821'.	

Figure 1-9. LU Type 3 Buffer Size Bind Format

The Erase/Write Alternate command is accepted in base LU type 3, but it is processed as an Erase/Write command. No state change occurs. All terminal printers can be bound with b'0000001' or b'0000010'.

When bits 1 through 7 of byte 24 are coded b'0000000', the entire print buffer can be used, regardless of size. Buffer wrap occurs at the end of the physical buffer. An Erase/Write Alternate command is processed as a normal Erase/Write command. No state change occurs.

When coded b'1111110', byte 24 indicates extended LU type 3 operation with the buffer size coded in bytes 20 and 21. Buffer size switching is not allowed. Bytes 22 and 23 are ignored. When an Erase/Write Alternate command is encountered in the data stream, it is interpreted as a normal Erase/Write command.

When byte 24 is coded b'1111111', bytes 22 and 23 are inspected to determine the maximum alternate buffer size to be used during the session; for example, a Bind for 32 rows of 80 characters each permits the use of programs written for 960-, 1920-, and 2560- character buffer sizes. (If programs written for 132-character columns are used, byte 22 must be interpreted differently.) This assumes that programs do not depend upon buffer address wrap during write operations.

If the printer cannot support the required buffer size, the Bind is rejected with a -RSP (0821) response parameter error. A 3287 with a basic 2K buffer cannot, for example, accept an LU3 Bind specifying a 2560-character buffer. Valid Bind parameter values for the 3276 are column counts of 40 or 80, and the product of the row and column counts that are less than or equal to the physical buffer size minus 80. The row/column product determines the print buffer wrap point. Print control is managed by the WCC and not by the Bind parameter values.

For the 3276, other values coded into bytes 20-23 may cause unpredictable results, but the Bind will not be rejected.

Read Commands

Three read-type commands are executed by the 3276: Read Buffer, Read Modified, and Read Modified All. Read Buffer causes the entire buffer contents of the addressed terminal to be read into main storage. The operation initiated by Read Modified is determined by display station operator actions.

The information read during execution of Read Modified or Read Modified All could consist of fields of data modified by keyboard operations, data entered by magnetic reading devices, buffer addresses, or data of selector light-pen or CURSR SEL fields, or the code of a Program Function or Program Access key.

In remote BSC configurations, reading is normally accomplished by a General or Specific Poll sequence (described under "Remote Operations" in Chapter 3). In remote, the 3276 cannot generate attention interruption. Instead, the host program should issue poll sequences periodically. Upon receipt of a poll sequence, the 3276 BSC control unit initiates one of three operations:

1. If status and sense information is pending, this information is sent to the TCU.

2. If an operator action has occurred that requires reading by the program, and status and sense information is not pending, a control-unit-generated Read Modified command operation is performed.

3. If no operator action has occurred and status and sense information is not pending, the control unit sends End of Transmission (EOT) to the TCU, terminating the operation.

Programming Note: Unsolicited read commands are not recommended, because the information read by these commands may be incomplete.

During a read-buffer or read-modified operation, when BSC line discipline is used, a SUB character (3F in EBCDIC, 1A in ASCII) is sent in place of any byte that has bad parity. Also, a Data Check sense condition is recorded. Normal transmission of the read data then continues until the usual ending point. At that time, the operation is terminated with ENQ in place of ETX or ETB.

Read Buffer Command

Execution of the Read Buffer command causes all data in the addressed device buffer, from the buffer location at which reading starts through the last buffer location, to be transferred to main storage. This command is provided primarily for diagnostic purposes. The transfer of data begins:

1. From buffer address 0 if the Read Buffer command is unchained. Certain 3276 emulators also begin data transfer from buffer address 0 if the Read Buffer command is chained from a Copy command.

2. From the current buffer address if the Read Buffer command is chained. Certain 3276 emulators only begin data transfer from the current buffer address if the Read Buffer command is chained from a Write, Erase/Write, Read Modified, or another Read Buffer command. Regardless of where the transfer of data begins, data transfer from the buffer will terminate when the last character location in the buffer has been transferred, or before the last character location has been transferred, when the last character of a text block has been transferred (described under "Remote Operations" in Chapters 3 and 4).

The transferred data stream begins with a 3-character read heading consisting of the AID character followed by a 2-character cursor address. The contents of all buffer locations are transferred, including nulls. Start Field (SF) orders are inserted by the 3276 to identify the beginning of each field.

The possible cursor address byte configurations are shown in Appendix B. The possible Attention Identification (AID) byte configurations are shown in Figure 1-10. An AID configuration other than 60 or E8 is set when the operator at the selected display station has performed an operation that requires program intervention. These operations are (1) pressing a Program Function or Program Access key, (2) reading a magnetic stripe, or (3) detecting an attention field with the selector light pen or CURSR SEL key. The attribute character is shown in Figure 2-4.

Read Modified Command

Read Modified initiates one of three operations, as determined by operator actions at the display station: (1) Read Modified, (2) Short Read, or (3) Test or System Request Read: Figure 1-10 lists the operator actions and the resulting Read Modified command operation initiated by each action. Read Modified commands normally are not used for remote configurations since polling initiates a control-unit-generated read-modified operation if AID is generated and if status is not pending.

A major feature of Read Modified command operations is null suppression. The device buffer is cleared to all nulls when the operator turns power on or presses the CLEAR key, or when the erase portion of an Erase/Write command is executed at the selected device. Also, selected portions of a buffer can be cleared to nulls by the Erase All Unprotected command and certain orders. During Read Modified command operations, null codes are not sent.

Read Modified Operation. During a Read Modified command, if an AID other than selector-light-pen attention, the CURSR SEL key, a PA key, or the CLEAR key is generated, all fields that have been modified by a keyboard, the selector light pen, the CURSR SEL key, or the reading of a magnetic stripe are transferred to the program. All nulls are suppressed during data transfer and thus are not included in the read data stream. As a field is modified by the operator, the modified data tag (MDT) bit is set in the attribute byte for that field. Then, when a read-modified operation is performed, successive attribute bytes are examined for a set MDT bit. When the bit is found, the data in the associated field is read (with nulls suppressed) before the next attribute byte is examined.

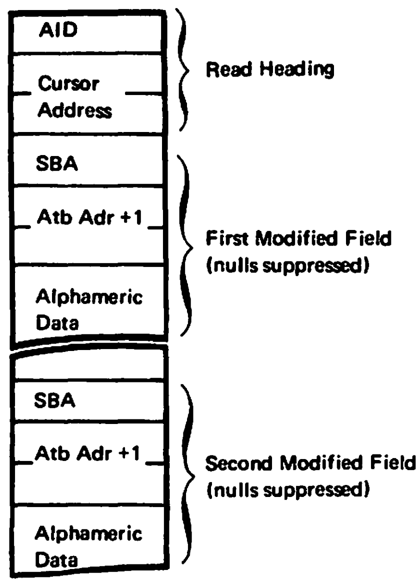
The first 3 bytes of the read data stream are always the AID code (Figure 1-10) and the 2-byte cursor address; these bytes are called the "read heading."

AID	Hex Character (EBCDIC)	Hex Character (ASCII)	Graphic Character	Read Modified Command Operation	Resultant Transfer to CPU
No AID generated (Display or Display Station)	60	2D	–	Rd Mod (Unsolicited Read or Read Modified from Host)	If performing a remote polling operation, no read operation occurs; otherwise, field addresses and text in the modified fields are transferred.
No AID generated (Printer)	E8	59	Y	Rd Mod	
ENTER key and & (Selector-Light-Pen Attention)	7D	27	'	Rd Mod	AID code and cursor address, followed by an SBA order, attribute address +1, and text for each modified field. Nulls are suppressed.
PF 1 key	F1	31	1	Rd Mod	
PF 2 key	F2	32	2	Rd Mod	
PF 3 key	F3	33	3	Rd Mod	
PF 4 key	F4	34	4	Rd Mod	
PF 5 key	F5	35	5	Rd Mod	
PF 6 key	F6	36	6	Rd Mod	
PF 7 key	F7	37	7	Rd Mod	
PF 8 key	F8	38	8	Rd Mod	
PF 9 key	F9	39	9	Rd Mod	
PF 10 key	7A	3A	:	Rd Mod	
PF 11 key	7B (See Note.)	23	#	Rd Mod	
PF 12 key	7C (See Note.)	40	@	Rd Mod	
PF 13 key	C1	41	A	Rd Mod	
PF 14 key	C2	42	B	Rd Mod	
PF 15 key	C3	43	C	Rd Mod	
PF 16 key	C4	44	D	Rd Mod	
PF 17 key	C5	45	E	Rd Mod	
PF 18 key	C6	46	F	Rd Mod	
PF 19 key	C7	47	G	Rd Mod	
PF 20 key	C8	48	H	Rd Mod	
PF 21 key	C9	49	I	Rd Mod	
PF 22 key	4A	5B	¢	Rd Mod	
PF 23 key	4B	2E	.	Rd Mod	
PF 24 key	4C	3C	<	Rd Mod	
Operator Identification Card Reader	E6	57	W	Rd Mod	
Magnetic Slot Reader and Magnetic Hand Scanner	E7	58	X	Rd Mod	
Selector-Light-Pen Attention space null	7E	3D	=	Rd Mod	AID code, cursor address, and field addresses only; no data.
PA 1 key	6C	25	%	Short Rd	AID code only.
PA 2 (CNCL) key	6E	3E	>	Short Rd	
PA 3 key	6B	2C	,	Short Rd	
CLEAR key	6D	5F	–	Short Rd	
TEST REQ and SYS REQ keys	F0	30	0	Tst Req Rd	A test request message. AID transferred on Read Buffer only.

Note: *Graphic characters for the United States I/O interface codes are shown. If a World Trade country I/O interface code is used, refer to* IBM 3270 Information Display System: Character Set Reference, *GA27-2837, for possible graphic character differences.*

Figure 1-10. Attention ID (AID) Configurations

Following the read heading is the alphameric data of each modified field. The data for each field is preceded in the data stream by a Set Buffer Address (SBA) order code followed by the 2-byte buffer address of the first character position in that field (the attribute address + 1). Thus, the read data stream when data has been modified is as follows:



If a space or null selector-light-pen-attention AID is generated, fields are not transferred to main storage during the read-modified operation. Instead, when a set MDT bit is found (indicating selector-light-pen and/or keyboard activity), only the read heading, the SBA order code, and the attribute address +1 are transferred.

Note that if fields are modified by the keyboard but completion of the modification is signaled by a selector-light-pen-attention operation on other than ampersand character-designator fields, a resulting read-modified operation will read only the address of the modified fields; not the modified data. A Read Modified command can be used to obtain both the address of, and the data in, each field that has the MDT bit set to 1.

The buffer location at which the search begins for attribute bytes that define modified fields is a function of command chaining. This location is:

1. Buffer address 0 if Read Modified command is unchained or is chained from a Copy command.

2. The current address if the Read Modified command is chained from a Write, Erase/Write, Read Modified, Read Modified All, or Read buffer command.

The search for modified-field attribute bytes ends when the last buffer location is checked.

The transfer of read data is terminated as follows:

1. If the last modified field is wrapped from the last buffer location (for example, 479 or 1919) to the first location, the operation is terminated after all data in the field is transferred (nulls are suppressed). The buffer address at the end of the operation is the address of the next attribute byte in the buffer. For example, if a modified field extends from address 1900 (the attribute byte) to address 79 (wrapped field), the data from address 1901 through 79 is transferred (nulls are suppressed); in this case, the read operation is terminated with the buffer address set to 80 (the attribute byte of the next field).

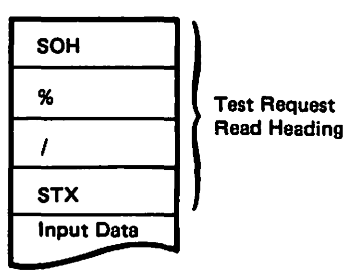
2. If the buffer does not contain a wrapped modified field, and if the channel byte count has not reached zero (local operation only), the modified data stream is terminated when the last modified field is transferred; at the end of the operation, the buffer address is set to 0.

If the buffer is formatted (contains fields) but none of the fields have been modified, the read data stream consists of the 3-byte read heading only.

If the buffer is unformatted (contains no fields), the read data stream consists of the 3-byte read heading followed by all alphameric data in the buffer (nulls are suppressed), even when part or all of the data has not been modified. Since an unformatted buffer contains no attribute bytes, no SBA codes with associated addresses or address characters are included in the data stream, and the modification of data cannot be determined. Data transfer starts at address 0, regardless of command chaining, and continues to the end of the buffer. At the end of the operation, the buffer address is set to 0. This read operation can also be terminated by the channel byte count reaching zero before all data is read; in this case, the buffer address after termination is undefined.

Short Read. The Read Modified command causes a short read operation if the CLEAR, CNCL, or a PA key has been pressed at the selected device. During the Short Read operation, only an AID byte is transferred to main storage. This AID byte identifies the key that was pressed.

Test Request Read. This description applies only to units not using SNA protocol. The Read Modified command causes a Test Request Read operation if the SYS REQ key has been pressed at the selected device. The Test Request Read data stream sent to main storage is as follows:



The Test Request Read heading is generated by the control unit. The remainder of the data stream is the same as described previously for read-modified operations, excluding the 3-byte read heading (AID and cursor address). If the buffer is unformatted, all alphameric data in the buffer is included in the data stream (nulls are suppressed), starting at address 0. If the buffer is formatted, each attribute byte is examined for a set MDT bit. Each time a set MDT bit is found, the alphameric data in the field associated with that bit is sent to main storage (nulls are suppressed); if no MDT bits are set, the read data stream consists of the Test Request Read heading only. The buffer location at which the search for MDT bits begins and the transfer of data ends is the same as described for read-modified operations.

Test Request Read function usage is determined by the access method. Normally, the operator would (1) clear the display, (2) enter test request data in a predefined format, and then (3) press the SYS REQ key.

Read Modified All Command

The Read Modified All command is used with the 3276 unit operating in SNA/SDLC protocol. This command operates like a Read Modified command except that both addresses and data from all modified fields are sent to the host, regardless of the AID byte generated. The Read Modified All command is not generated by the control unit in response to a poll sequence. It must be sent by the host.

Control Commands

Control commands initiate certain control unit and/or device operations not involved with the transfer of data (other than status). Two control-type commands are executed by the 3276: Copy and Erase All Unprotected. The applicable control units are identified within the description of each control command.

Copy Command

The Copy command is used to transfer buffer data from one device to another device attached to the same control unit. The selected device is the "to" device, the one to which buffer data will be transferred. The "from" device, the source of the buffer data to be copied, is identified in the second of 2 bytes that follow the Copy command code; the first byte, called the copy control character (CCC), identifies the type of data to be copied. The CCC can also, at the "to" device, start print operations, specify the printout format for those operations, and, when the device is a display station, sound the audible alarm.

The Copy data stream is as follows:

Byte		
0	STX	
1	ESC	
2	Copy Command Code	← See Figure 1-6.
3	CCC	← See below and Figure 1-12.
4	"From" Device Address	← See Figure 1-13.
5	ETX	

The CCC-byte format is as follows:

0	1	2	3	4	5	6	7
*	1	Printout Format		Start Print	Sound Alarm	Type of Data to Be Copied	

*Determined by the configuration of bits 2 through 7. See Figure 1-4.

Figure 1-12 describes the function of each CCC bit. A CCC and address byte must always follow the command code; if they do not, the control unit aborts the command and generates error status.

The 3276, when operating with SNA/SDLC protocol, does not support the Copy command. A Copy function is provided, however, which is discussed under "Local Copy Function" in Chapter 2.

Bit	Explanation
0,1	Determined by the contents of bits 2 through 7 as shown in Figure 1-4.
2, 3	Define the printout format as follows: = 00 – The NL, EM, and CR[1] orders in the data stream determine print line length. Provides a 132-print position line when the orders are not present. = 01 – Specifies a 40-character print line. = 10 – Specifies a 64-character print line. = 11 – Specifies an 80-character print line.
4	The Start Print bit. When set to 1, initiates a printout operation at the "to" device after buffer transfers are completed.
5	The Sound Alarm bit. When set to 1, sounds the audible alarm at the "to" device after buffer transfers are completed if that device has an audible alarm.
6, 7	Define the type of data to be copied as follows: = 00 – Only attribute characters are copied. = 01 – Attribute characters and unprotected alphameric fields (including nulls) are copied. Nulls are transferred for the alphameric characters not copied from the protected fields. = 10 – All attribute characters and protected alphameric fields (including nulls) are copied. Nulls are transferred for the alphameric characters not copied from the unprotected fields. = 11 – The entire contents of the storage buffer (including nulls) are copied.

[1] The CR order is applicable to the terminal printers only.

Figure 1-12. Copy Control Character (CCC)

Copy command operations are similar to Write command operations. After the 3276, for example, accepts the Copy data stream, it initiates the transfer of all 1920 bytes from the "from" device buffer to the 3276 buffer. Upon completion of this transfer, the 3276 inserts nulls in all character locations that do not contain the type of data specified by CCC bits 6 and 7. The updated control unit buffer contents are then transferred to the selected ("to") device. At the completion of Copy command operations, the cursor is in the same character location at the "to" device as it was at the "from" device at the start of operations.

The "from" device buffer can be "locked" (made incapable of being copied) by writing a protected/alphameric attribute byte (bit 2=1 and 3=0) in address 0 (with BSC only).

The Copy command can specify as the "from" device the same device that is selected (the "to" device). This procedure provides a means of programming selective device buffer "erase" operations as specified by CCC bits 6 and 7. In this case, the device buffer contents are transferred to the control unit, nulls are inserted as determined by the CCC, and the resulting buffer contents are transferred back to the same device buffer.

When the "from" and "to" devices are attached to a 3276 Model 1, 2, 3, or 4, and when the buffer size of the "from" device is smaller than, or equal in size to, the buffer size of the "to" device, screen size switching occurs as listed in Figure 1-13. Invalid transfers are also indicated. The buffer of the "to" device is, in effect, cleared before the copy is performed. The same rules apply for copy-operation transfers to printer buffers.

To	3276/8-1 960	3276/8-1 480	3276 3278-2 3279 1920	3276 3278-2 3279 2560	3276/8-3 1920	3276/8-4 3440	3276/8-4 1920	3178 1920
3276/8-1 960	O	A	●	●	●	●	●	●
3276/8-1 480	V	O	●	●	●	●	●	●
3178/3276 /8-2/9 1920	–	–	O	V	O	V	O	O
3276/8-2/9 2560	–	–	–	O	A	●	A[1]	–
3276/8-3 1920	–	–	O	V	O	V	O	O
3276/8-4 3440	–	–	–	–	–	O	A	–
3276/8-4 1920	–	–	O	V	O	V	O	O

Legend:

O Transfer allowed, no change in screen state required.
– Transfer not allowed, Operation Check returned to host.
● Transfer allowed, no change in screen state (appearance on "from" and "to" device may differ).
A Transfer allowed, screen state changes to alternate size.
V Transfer allowed, screen state changes to default size.

[1]The 3440 screen does not have a 2560 mode; therefore, the screen size is set to 3440.

Figure 1-13. Buffer Transfers for 3276 Models 1 through 4 Copy Command Operation

Programming Notes:

1. Copy should not be chained from a Write, Erase/Write Alternate, Erase/Write Unprotected, or Erase All Unprotected command, since it will copy the data as modified by the Write or Erase command.

2. If the CCC Start Print bit is set and commands are being chained, Copy should be the last command of the chain. If not, the control unit aborts the subsequent command.

3. Copy can be executed from a smaller buffer size to a larger buffer size, but an attempt to copy from a larger to a smaller buffer size will cause an Operation Check.

4. An Operation Check will occur if copying from an APL device in APL mode to a device that does not have the APL feature installed.

Erase All Unprotected Command

This command performs five functions at the addressed device:

1. Clears all unprotected buffer character locations to nulls.
2. Resets to 0 the MDT bit for each unprotected field.
3. Unlocks the keyboard when either the System Lock or the Wait symbol is displayed on the 3178, 3276, 3278, or 3279.
4. Resets the AID byte.
5. Repositions the cursor to the first character location in the first unprotected field of the buffer. If no unprotected fields exist, the cursor is positioned to buffer location 0.

Programming Restriction: Erase All Unprotected should not be chained to a Write, Erase/Write, Erase/Write Alternate, Copy, or another Erase All Unprotected command. If it is, the resulting operation is not defined.

Orders

Orders can be included in Write, Erase/Write, or Erase/Write Alternate command data streams, either alone or intermixed with display or print data. Two types of orders are available: printout format orders and buffer control orders. Printout format orders are initially stored in the buffer as data and are subsequently executed only during a print operation.

The following paragraphs describe buffer control orders, which are executed as they are received in the write data stream by the 3270; these orders are not stored in the buffer. Six buffer control orders (see Figure 1-14) are provided to position, define, and format data being written into the buffer, to erase selected unprotected data in the buffer, and to reposition the cursor.

Notes:

1. Figure 2-4 shows attribute byte, and Figure 1-4 shows coding of this byte.

2. Figures 1-2 through 1-4 show coding of this byte.

3. Appendix B lists the 2-byte code for each possible address. To be a valid address:

a. If the default size is used in BSC mode, the maximum buffer addresses are:

3276-1, 3278-1: 479
3178-C1, -C2; 3276-2, -3, -4; 3278-2, -3, -4; 3279-2, -3: 1919

b. If the alternate size is used in BSC mode, the maximum buffer addresses are specified by the device model number:

Model 1: 959
Model 2: 1919
Model 3: 2559
Model 4: 3439

c. If the SNA/SDLC mode is used, the maximum default size and alternate size are the display size minus 1. The display size is defined in the Bind parameter.

Start Field (SF) Order

This order notifies the control unit that the next byte in the write data stream is an attribute character. The control unit then stores the next byte (the attribute character) at the current buffer address. As the attribute character is stored, the control unit sets a control bit at that address; this bit identifies the byte as an attribute character during subsequent program or device operations with the buffer data.

When received by control units and terminals supporting the extended field attributes, the SF order causes the default value (X'00') for the Color, Extended Highlighting, and Programmed Symbols attribute types to be set in the extended field attribute buffer.

Note: *The byte immediately following the SF order in the data stream is always stored as an attribute character, even when the byte is intended as an order or an alphameric data character.*

During execution of a Read Buffer command, the control unit automatically inserts SF order codes in the read data stream immediately before each attribute character. This permits identification of the attribute characters by the program and also permits correct storage of attribute characters in the buffer if the read data is used for subsequent write operations.

Order \ Order Sequence	Byte 1 (Order Code)		Byte 2	Byte 3	Byte 4
	EBCDIC (Hex)	ASCII (Hex)			
Start Field (SF)	1D	1D	Attribute Character[1]		
Set Buffer Address (SBA)	11	11	1st Address Byte[3]	2nd Address Byte[3]	
Insert Cursor (IC)	13	13			
Program Tab (PT)	05	09			
Repeat to Address (RA)	3C	14	1st Address Byte[3]	2nd Address Byte[3]	Character to Be Repeated[2]
Erase Unprotected to Address (EUA)	12	12	1st Address Byte[3]	2nd Address Byte[3]	

[1] Figure 2-4 shows attribute byte, and Figure 1-4 shows coding of this byte.
[2] Figures 1-2 through 1-4 show coding of this byte.
[3] Appendix B lists the 2-byte code for each possible address. To be a valid address:

a. If the default size is used in BSC mode, the maximum buffer addresses are:
   3276-1, 3278-1: 479
   3178-C1, -C2; 3276-2, -3, -4; 3278-2, -3, -4; 3279-2, -3; 1919
b. If the alternate size is used in BSC mode, the maximum buffer addresses are specified by the device model number:
   Model 1: 959
   Model 2: 1919
   Model 3: 2559
   Model 4: 3439
c. If the SNA/SDLC mode is used, the maximum default size and alternate size are the display size minus 1. The display size is defined in the Bind parameter.

Figure 1-14. Buffer Control Orders and Order Codes.

Set Buffer Address (SBA) Order

This 3-byte order specifies a new buffer address from which write operations are to start or continue. Set Buffer Address orders can be used to write data into various areas of the buffer. An SBA order can also precede another order in the data stream to specify the starting address for a PT, RA, or EUA order; to specify the address at which an attribute byte is to be stored by an SF order or modified by an MF order; or to specify the address at which the cursor is to be repositioned by an IC order.

If the SBA order specifies an invalid address, the write operation is terminated at this point.

When a Read Modified command is executed and an attribute character (initially sent to the device by writing an SF order) is detected with the MDT bit set, the CU inserts, in place of the attribute, an SBA code followed by the 2-byte buffer address of the first character in the modified field (attribute address +1). This permits identification by the control unit of fields that are modified. When a Read Modified command is executed in the 3276, this three-byte sequence is always sent in the same text block. The 3276 does not split this sequence between two successive blocks.

Insert Cursor (IC) Order

This order repositions the cursor to the location specified by the current buffer address. Execution of this order does not change the current buffer address. For example, if IC is issued when the current buffer address is 160 and the cursor is at location 80, the cursor is moved from location 80 and inserted at location 160. The current buffer address at the end of this operation would remain 160.

Program Tab (PT) Order

The PT order advances the current buffer address to the address of the first character position of the next unprotected field. If the PT is issued when the current buffer address is the location of an attribute byte of an unprotected field, the buffer address advances to the next location of that field (one location). In addition, if the PT order in the write data stream does not follow a control command, order, or order sequence such as WCC, IC, or RA (3-character sequence), nulls are inserted in the buffer from the current buffer address to the end of the field, regardless of the value of bit 2 (protected/unprotected) of the attribute character for the field. Whenever a character position is set to null by the PT order, the default value (X'00') for the Color, Extended Highlighting, and Programmed Symbols attribute types is set in the character attribute buffer. When the PT order follows a control command, order, or order sequence, the buffer content is not modified for that field.

The PT order stops its search at the last location in the buffer. If an attribute character for an unprotected field is not found by this point, the buffer address is set to location 0. (If the PT order finds an attribute character for an unprotected field in the last buffer location, the buffer address is also set to zero.)

To continue the search for an unprotected field, a second PT order must be issued immediately following the first one. Since the current buffer address was reset to 0 by the first PT order, the second PT order begins its search at buffer location 0. If the previous PT order was still inserting nulls in each character location when it terminated at the last buffer location, the new PT order will continue to insert nulls from buffer location 0 to the end of the current field.

Repeat to Address (RA) Order

The RA order stores a specified alphameric or null character in all buffer locations, starting at the current buffer address and ending at (but not including) the specified stop address. This stop address and the character to be repeated are identified by the 3 bytes immediately following the RA order in the write data stream, as follows:

Byte		
0	RA Order	
1–2	Stop Address	See Appendix B for address.
3	Character to Be Repeated	See Figures 1-2 through 1-4 for codes

The third character following the RA order is always interpreted as the character that will be repeated. If an invalid stop address is specified, the write operation is terminated at this point without storing the character, and error status is generated. When Color, Extended Highlighting, or Programmed Symbols attributes are specified for the character, the attribute values are entered into the character attribute buffer as each repeated character is written in the data buffer.

When the stop address is lower than the current buffer address, the RA operation wraps from the bottom row of the buffer to the top row. When the stop address equals the current address, the specified character is stored in all buffer locations.

Attribute characters will be overwritten by the RA order if they occur before the RA order stop address.

Programming Note: If the RA order specifies X'ID' to indicate a 2-byte character code (for the Data Analysis/APL), only X'ID' will be repeated. See Figure D-2, Part 2.

Erase Unprotected to Address (EUA) Order

The EUA order inserts nulls in all unprotected buffer character locations, starting at the current buffer address and ending at, but not including, the specified stop address. This stop address is specified by 2 address bytes which immediately follow the EUA order in the write data stream. If an invalid address is specified, the write operation is terminated at this point and error status is generated. Whenever a character position is set to null by the PT order, the default value (X'00') for the Color, Extended Highlighting, and Programmed Symbols attribute types is set in the character attribute buffer.

When the stop address is lower than the current buffer address, the EUA operation wraps from the bottom row of the buffer to the top row. When the stop address equals the current address, all unprotected character locations in the buffer are erased.

Attribute characters are not affected by the EUA order.

Chapter 2. Terminals

This chapter describes the function of the display stations (including keyboard operation, selector light pen, and magnet stripe reader) and printers that can be attached to the 3276 Control Unit Display Station.

Displays

Display Images

Display stations for the 3276 Control Unit Display Station are buffered displays. Data displayed on the screen is stored in coded form in a display buffer; the buffer contains as many locations as there are character positions on the screen. The data may be loaded from the host system by the application program or from a keyboard attached to the display station. Figure 2-1 illustrates the concept of a buffered display.

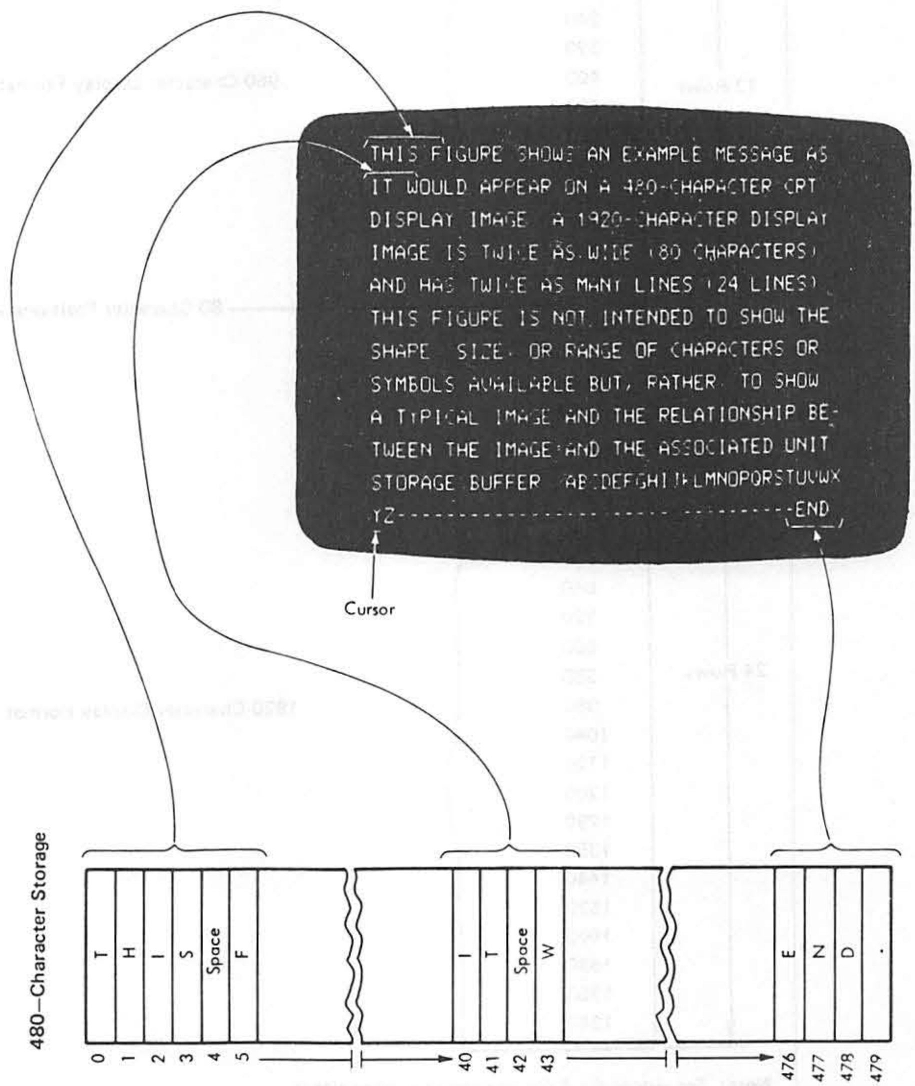
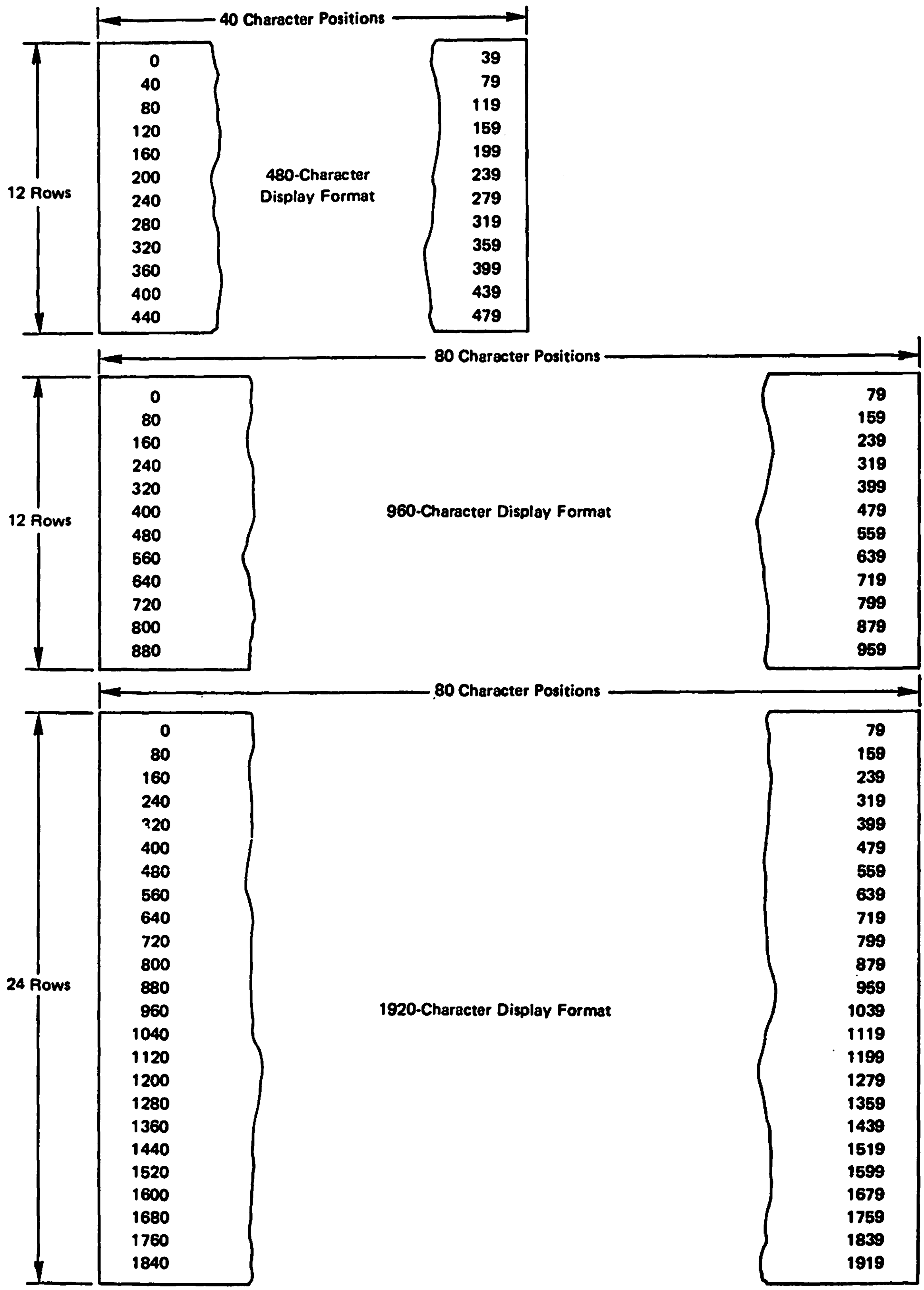


Figure 2-1. Buffer Location and Display Screen Character Position Relationships



Note: *See Appendix B for hexadecimal equivalents.*

Figure 2-2 (Part 1 of 2). Buffer Addressing Layouts for 480–, 960–, 1920–, 2560–, and 3440– Character Terminals

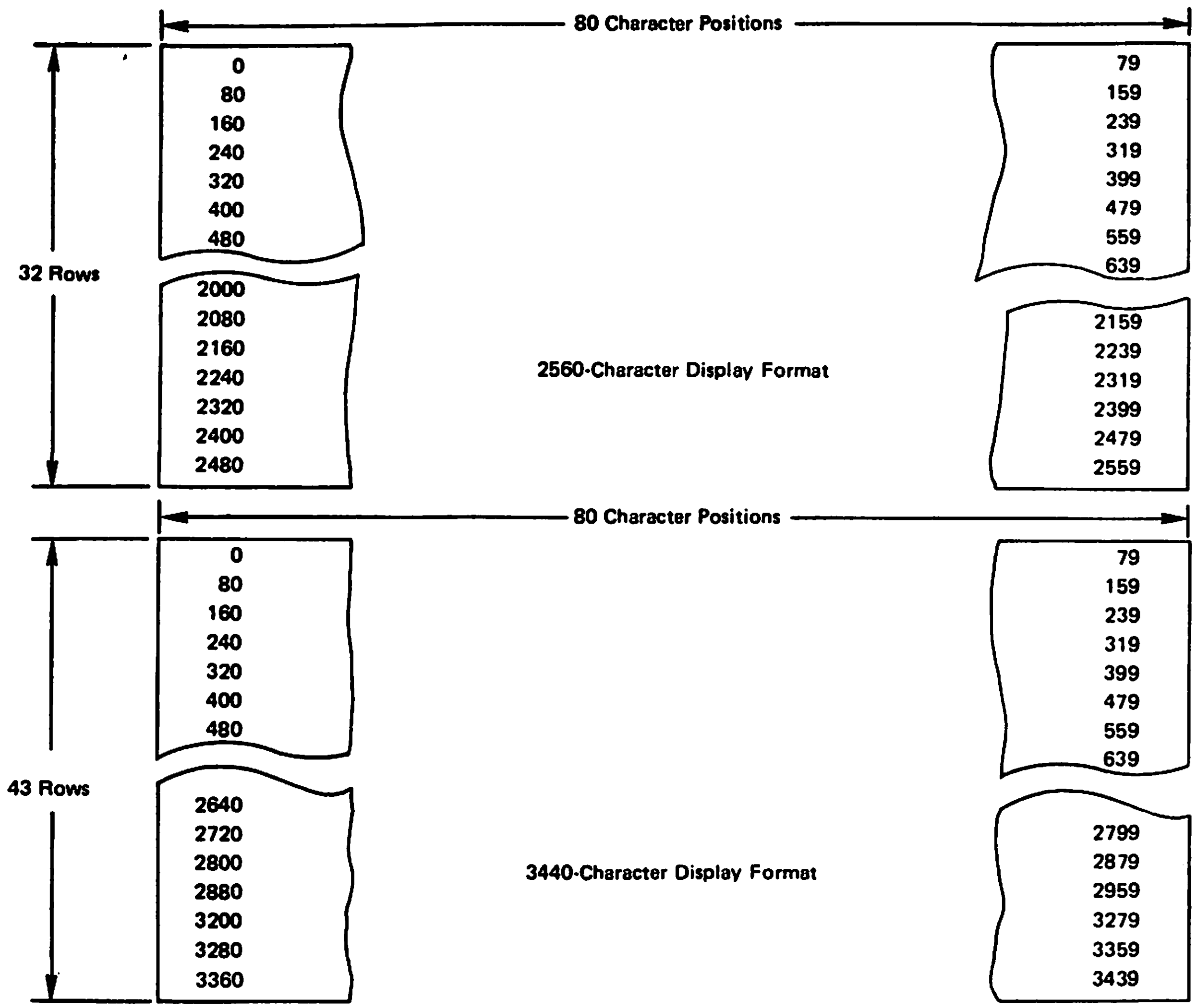


Figure 2-2 (Part 2 of 2). Buffer Addressing Layouts for 480–, 960–, 1920–, 2560–, and 3440– Character Terminals

The display image contains a fixed number of horizontal rows, with a fixed number of character positions in each row. The display station of the 3276 can support the following screen capacities:

Models 1 and 11	960 -	character display	12 rows of 80 characters
Models 2 and 12	1920 -	character display	24 rows of 80 characters
Models 3 and 13	2560 -	character display	32 rows of 80 characters
Models 4 and 14	3440 -	character display	43 rows of 80 characters

There is a fixed relationship between each location in the display buffer and each character position on the display screen. Buffer addresses start from 0, for the character position at the left of the top row, and proceed sequentially along the rows and down the screen to the character position at the right of the bottom row (for example, an image with 960 character positions has buffer addresses from 0 to 959). Figure 2-2 shows the addresses of the first and last character positions in each row, depending upon the available screen capacity.

Each location in the buffer contains 1 byte of storage; codes loaded into the buffer are 2-digit hexadecimal codes. Write commands are used to load the display buffer locations with the code needed to display the required data on the display screen (see Chapter 1). Defined codes that are displayed as alphameric characters are shown in Figures 1-2 and 1-3.

Display images may be formatted or unformatted:

- Formatted Display: A formatted display is one that has separate fields defined by the program. The first character position in each field contains a control character that defines the characteristics of the field. See "Field Attributes," later in this chapter, for a description of the control character.
- Unformatted Display: An unformatted display is one that has no defined fields. An operator may input data into any position on the screen; to access the data, the program must issue a read command for the entire display buffer.

Display Fields

A formatted display contains display fields defined by the program. These fields consist of blocks of character positions bounded by control characters. The control character at the start of a field is set by the program to determine the characteristics of the field; this character contains the field attributes. (For details, see "Attributes," later in this chapter.) Fields containing character positions on more than one row "wrap" from the last character position on one row to the first character position on the next row. A field may wrap the screen; if the first character position on the screen does not contain a control character, the last field on the screen wraps from the last character position to the first. (Some field-oriented operations are terminated early if the field wraps the screen; this effect is noted in the descriptions of the specific operations.)

Display fields simplify operations both for the operator and for the programmer. Headings can be displayed to prompt the operator as to the data that should be entered, and the program can identify fields that contain entered data without reading the entire display buffer. When data is being entered into a formatted display, the presence of a control character acts as a tab stop; pressing the tab key advances the cursor from its current position to the first character position in the next unprotected field. (An unprotected field is one that accepts data input from the keyboard.)

The example in Figure 2-3 illustrates the versatility of formatted displays. In this example, the solid characters represent the displayed form of characters stored in the buffer. The dotted squares represent the character positions corresponding to control characters at the start of each field. The dotted characters represent fields of data that are stored in the buffer, but that have been defined by the program as non-displayable; that is, not to be displayed to the operator.

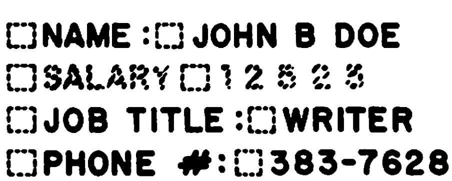


Figure 2-3. Example of Formatted Display

To define the start of a field, the program may issue a Write command transferring a Set Buffer Address (SBA) order and a Start Field (SF) order to the display; the specified buffer address is selected, and the control character specified by the SF order is loaded into the addressed location. Only the start of a field is defined; starting a field ends the previous field at the character position prior to the new control character.

Attributes

Display stations may be programmed with formatted fields. The control character at the start of each field contains the field attributes. Attributes contained in this character apply to all the data contained in the field; for example, the attribute character for the field containing PHONE # in Figure 2-3 might define the field as protected to ensure that the operator does not enter data into that field, and the field containing 383-7628 might be defined as unprotected to allow the data to be changed.

Field Attributes

The field-attribute character occupies the first character position of each display field in a formatted display; the corresponding character position on the display screen is always blank. This 8-bit attribute character is loaded by a Start Field order to (1) define the start of a field and (2) assign characteristics to the field. Bit positions in the character are significant to the display; the value assigned to each bit or group of bits controls whether a specific attribute is applied.

Field Attribute Character

Figure 2-4 shows the significance of bits in the field-attribute character. Characteristics set by the field-attribute character are:

- Protected/Unprotected: An operator cannot enter data into or modify the content of a protected field. Input fields that require data from the operator must be unprotected.
- Alphameric/Numeric: In an unprotected input field, alphameric/numeric defines the type of data that an operator can enter into the field. This attribute has special meaning for protected fields, data entry keyboards, and the Numeric Lock feature.
- Nondisplay/Display/Intensified: Data contained in the field is either not displayed, displayed at normal intensity, or displayed at high intensity. The 3279 does not support two levels of intensity; if no extended attribute is defined, nonintensified fields and intensified fields are displayed in different colors. (The actual colors are determined by the position of the Base Color switch and the value of the Protected/Unprotected attribute.)

  Programming Note: Refer to "Selector-Light-Pen Operations," later in this chapter, for the use of intensified field attributes when formatting selector-light-pen-detectable fields.
- Detectable/Nondetectable: Displayed data in a detectable field can be detected by the selector light pen. (The detectable field must contain a designator character as described under "Selector-Light-Pen Operations" in this chapter.)

Field attributes are protected against input from the keyboard; however, bit 7 (Modified Data Tag) is set to 1 when the operator enters data into the field defined by the attributes. Attribute characters are not protected against operation of the CLEAR key; pressing the CLEAR key erases all locations in the display buffer.

Attribute character bit assignments are summarized as follows:

X	X	U/P	A/N	D/SPD		Reserved	MDT
0	1	2	3	4	5	6	7

EBCDIC Bit	Field Description
0, 1	- Value determined by contents of bits 2–7. See Figure 1-8 for hexadecimal values.
2	- 0 = Unprotected 1 = Protected
3	- 0 = Alphameric 1 = Numeric (causes automatic upshift of data entry keyboard) **Note:** *Bits 2 and 3 equal to 11 causes an automatic skip. See text.*
4, 5	- 00 = Display/not selector-light-pen detectable. 01 = Display/selector-light-pen detectable. 10 = Intensified display/selector-light-pen-detectable. 11 = Nondisplay, nonprint, nondetectable.
6	- Reserved.
7	- Modified Data Tag (MDT); identifies modified fields during Read Modified command operations. 0 = Field has not been modified. 1 = Field has been modified by the operator. Can also be set by program in data stream.

Figure 2-4. Field Attribute Character Bit Assignment

Base Color Mode

The 3279 uses the field attributes for the additional purpose of controlling color.

Models 2A and 3A of the 3279 always decode the field attributes to assign a color to each display field. If the operator sets the Base Color switch to base color (oooo), then the fields are colored in one of four colors - red, blue, green, or white - depending upon the protect and intensify bits. If the operator sets the Base Color switch to monochrome (oo), all data is displayed in green except for intensified fields; intensified data is displayed in white. The particular attributes examined are the protect and intensify attributes. Figure 2-5 shows how the value of these attributes determines the color of characters displayed in a field.

Note: *The integrity of the unprotected/protected attribute is preserved; the operator can enter data only into an unprotected field.*

Field Attribute	Attribute Bit 2	3	4	5	Base Color Switch oo	oooo
Unprotected, normal intensity	0	X	0	X	Green	Green
Unprotected, intensified	0	X	1	0	White	Red
Protected, normal intensity	1	X	0	X	Green	Blue
Protected, intensified	1	X	1	0	White	White

Figure 2-5. Colors Derived from Field Attributes

Keyboard Operations

Keyboards, which may be attached to a display station, enable the operator to change, edit, or create character displays except within fields defined by attribute characters as protected from keyboard operations by the program. As messages are being composed or modified by keyboard operations, the changes are inserted in the buffer and then displayed. When the operator completes an operation and presses the ENTER or an AID generating key, an I/O pending interruption occurs.

Cursor

A special symbol, called a *cursor*, is displayed on the display screen to indicate where the next character entered from the keyboard will be stored. On 3178, 3276, 3278, and 3279 displays, the cursor may appear as an underscore, as a blinking underscore, or as a rectangular or blinking rectangular symbol imposed over a character. The character within the rectangular cursor remains visible. The operator may change the cursor from an underscore to a rectangular symbol, or vice versa, by pressing the Alternate Cursor (ALT CURSR) key. The same operator may cause either type cursor to blink by using the Cursor Blink (CURSR BLINK) key. When the cursor is displayed under one character in a line of characters (Figure 2-1), that character can be changed or deleted by keyboard action. Also, if the cursor is displayed under (or within) a position without a display character, a character can be entered in that position by keyboard action.

One, and only one, cursor must always be in the display buffer. A cursor check occurs when the display station circuitry detects no cursor or more than one cursor in the buffer. When the display is turned on, the cursor is automatically generated and displayed in the first location on the screen. The cursor can be repositioned by the keyboard operator and also by the program. The cursor is not affected by field attributes or by the Security Keylock special feature; it is displayed even when positioned in a nondisplayed/nonprint field and when the Security Keylock special feature (if installed) is turned off.

Keyboards

Six types of keyboards are available: typewriter, data entry, data entry keypunch layout, operator console, APL, and text keyboards. All keyboards have special symbol keys and control keys for entering data. The type of keyboard determines the characters and symbols that can be transmitted from the system for the display image.

Variations between keyboards include 75-key and 87-key versions for the 3178, 3276, 3278, and 3279. The 75-key keyboard provides all the basic operator keys. The 87-key keyboard provides expanded operator-to-program message flexibility with 12 additional keys that may be defined to fit the requirements of the application program. Refer to *3270 Information Display System: Character Set Reference,* GA27-2837, for key layouts and nomenclature.

Typewriter and APL 87-key and 88-key keyboards are available with extended function for the 12 program function keys on the right-hand side of the keyboards.

Key Functions

Alphabetic characters on typewriter keyboards attached to 3276 displays can be entered into the display buffer in either uppercase or lowercase code, depending upon the position of the Shift key. However, only uppercase alphabetic codes can be entered from data entry keyboards. On 3178, 3276, 3278, and 3279 displays, alphabetic characters in the buffer (uppercase or lowercase codes) are displayed as all uppercase or uppercase and lowercase characters, as determined by the setting of the Dual Case/ Mono Case switch. The shift keys on the Katakana keyboards operate differently from the keys described here; refer to Appendix E for details.

Keyboard entry of an alphameric character into the display buffer occurs at the cursor location, provided the cursor is located in an alphameric character location within an unprotected data field. (An attempt to enter an alphameric character into a protected data field or into an attribute character location is blocked.)

Successful keyboard entry of the alphameric character causes the cursor to advance to the next character location within the unprotected data field.

Note: *The following descriptions of key functions are applicable to all keyboards, except where noted. In some cases, descriptions of key functions contain SNA protocol terms, references to local copy operations, or Operator Information Area symbols. For a detailed description of these topics, refer to "Local Copy Function" later in this chapter, to "SNA/SDLC Communication" in Chapter 4, or to Figure A-3 in Appendix A. Operator Information Area symbols in this chapter are designated as "Do Not Enter" symbols in Appendix A.*

The ALT key must be held to activate functions shown on the front of keys on the 3178-, 3276-, 3278-, and 3279- attached keyboards. These functions are SYS REQ, CLEAR, ERASE INPUT, IDENT, TEST, DEV CNCL, PF1-PF12, PA1, PA2, ALT CURSR, and HOME. The ALT key is also used with the ≫ (Right) and ≪ (Left) key to move the cursor two locations at a time instead of one. Using the ALT key with a key that has no associated function produces no effect.

Automatic Skip

Upon entry of a character into the last character location of an unprotected data field, the cursor is repositioned according to the attribute character describing the next field.

If the field attribute character defines the next field as (1) alphameric and either unprotected or protected, or (2) numeric and unprotected, the cursor skips the attribute character and is positioned to the first character location in that field.

If the field attribute character defines the field as numeric and protected, the cursor automatically skips that field and is positioned to the first character location of the next unprotected field.

Character-Oriented Keys

A cluster of four keys (located to the right of the main keyboard) moves the cursor one location at a time into any character location. These are ↑ (Up), ↓ (Down), → (Right), and ← (Left). A fifth key, the Backspace key, occupies its normal position on the keyboard. It performs the same functions as the move-cursor-left key. The cursor may be moved into any character location, including unprotected and protected alphameric character and field attribute character locations, through the use of these keys. Operation of these keys does not affect the MDT bit. The ↑ (Up), ↓ (Down), → (Right), and ← (Left) keys move the cursor one location at a time. When the ALT (Alternate) key is pressed and held, the ≫ (Right) and ≪ (Left) keys will move the cursor two locations at a time.

These keys are all capable of causing the cursor to wrap. Horizontal wrap always involves a vertical movement; the cursor repositions to the next or preceding row of characters. Vertical wrap due to operation of the Up or Down keys involves no horizontal movement; the cursor stays in the same character column.

These keys all have typamatic operation at a repeat rate of approximately 10 operations per second. (When a typamatic key is fully pressed, its function is repeated as long as the key is held pressed.)

Field-Oriented Keys

Any of four keys moves the cursor to the first position in a field on a formatted screen. All four key operations can cause the cursor to wrap from the end of the last line on the display and to continue at the beginning of the top line. Operation of these keys does not affect the MDT bit.

→| (Tab) Key: Moves the cursor to the first character location of the next unprotected data field. In a display with no unprotected fields, the cursor is repositioned to character location 0. The Tab key has typamatic capability at a repeat rate of approximately 10 operations per second.

|← (Back-tab) Key: When the cursor is located in the field attribute character position or the first alphameric character location of an unprotected data field or in any character location of a protected data field, this key moves the cursor to the first alphameric character location of the first preceding unprotected data field. When the cursor is located in any alphameric character location of an unprotected data field other than the first location, this key moves the cursor to the first alphameric character location of that field. In a display with no unprotected fields, the cursor is repositioned to character location 0. The Back-tab key on keyboards attached only to 3178, 3276, 3278, and 3279 units has typamatic capability.

↵ (New Line) Key: Moves the cursor to the first unprotected character location of the next line. If the display has no unprotected data fields, the cursor is repositioned to character location 0. If the display contains no fields, the cursor is repositioned to the first character position of the next line. The New Line key has typamatic capability at a rate of approximately 10 operations per second.

⌂ (Home) Key: Moves the cursor to the first unprotected character position on a 3178, 3276, 3278, or 3279 display screen.

Erase EOF (Erase to End of Field) Key

If the cursor is located in an alphameric character location in an unprotected data field, this key clears the character location occupied by the cursor and all remaining character locations to the right in that field to nulls. The character attributes for all the erased characters are set to X'00'. The operation can wrap from the end of the last line on the display to the end of the field. The cursor does not move as a result of operating this key, and the MDT bit is set to 1.

Operation of this key when the cursor is located in an attribute character location or is within a protected data field causes an input-inhibit condition and disables the keyboard; no character locations are cleared, the cursor is not moved, and the MDT bit is not set.

ERASE INPUT Key

This key clears all unprotected character locations to nulls, resets the MDT bit to 0 in unprotected fields, and repositions the cursor to the first unprotected character location on the screen. The character attributes for all the erased characters are set to X'00'.

On 3178, 3276, 3278, and 3279 displays, the Alternate (ALT) key must be pressed and held first.

In a buffer with only protected data fields, no character locations are cleared and the cursor is repositioned to character location 0.

If the display contains no field, the entire buffer is cleared to nulls, all character attributes are set to X'00', and the cursor is repositioned to location 0.

â INS (Insert) Mode Key (3178, 3276, 3278, or 3279)

The INS MODE key on 3178-, 3276-, 3278-, or 3279-attached keyboards places the keyboard in an insert mode of operation. The Insert symbol is displayed in the Operator Information Area on the 3178, 3276, 3278, or 3279 display screen.

If the cursor is located in an unprotected data field having a null character either in the character location identified by the cursor or in any character location in the field beyond the cursor, operation of an alphameric key causes that alphameric character to be entered at the cursor and the MDT bit to be set to 1. The character formerly occupying the cursor location and all remaining characters within the field (except for null characters or characters to the right of null characters) will be shifted one character location to the right. If the location identified by the cursor location at the time of the insert operation is a null, no character shifting occurs.

After all null characters at or beyond the cursor location in the field have been overwritten, or if there were no null characters, operation of an alphameric key causes the keyboard to become disabled. Field-attribute characters and extended field attributes are not shifted as part of the insert operation. On displays that support extended attributes, the character attributes are shifted with the characters. The character attributes for inserted characters are set to X'00', except where the application program allows attribute-selection and the operator has selected specific attributes.

If more than one row of characters is contained within the field, a character occupying the last character location in the row is shifted into the first character location of the next row.

Operation of an alphameric key while in insert mode when the cursor is located in a field-attribute character location or is within a protected data field disables the keyboard; no character locations are cleared, the cursor is not moved, and the MDT bit is not set.

On 3178, 3276, 3278, and 3279 displays, operation of the RESET key, ENTER key, or any other key that causes host communication returns the keyboard to normal mode. (Operation of the selector light pen or the CURSR SEL (Cursor Select) key also returns the keyboard to normal mode.)

Delete Key (3178, 3276, 3278, or 3279)

If the cursor is located in an alphameric character location in an unprotected field, operation of the Delete key (3178, 3276, 3278, or 3279) deletes the character from the character location identified by the cursor and sets the MDT bit to 1 (if not previously set). The cursor does not move. All remaining characters in the unprotected field, to the right of the cursor and on the same row, shift one character location to the left. If the display supports extended attributes, the character attributes for the deleted character are deleted and the other character attributes are shifted left; the character attributes of vacated character positions are set to X'00'. Vacated character locations at the end of the row are filled with nulls. If the unprotected field encompasses more than one row, characters in rows other than the row identified by the cursor are not affected.

Operation of this key when the cursor is located in a field attribute character location or is within a protected data field disables the keyboard; no character locations are cleared, the cursor is not moved, and the MDT bit is not set.

RESET Key

The RESET key is used to recover from an inhibited keyboard operation that has resulted in a disabled keyboard. When a keyboard is disabled, no other keyboard operations are honored. The RESET key will not reset a disabled keyboard when a command is being executed for the device to which the keyboard is attached, or when a parity error or cursor check is detected in the device buffer.

On 3178, 3276, 3278, and 3279 displays, when a keyboard is disabled, symbols are displayed on the bottom row of the screen. Pressing RESET restores the keyboard or other input devices, except for Printer Busy, Printer Very Busy, Printer Not Working, Time, or Security Key input-inhibited conditions. Pressing RESET once resets multiple input-inhibited conditions.

When operating in BSC after an AID generating key is pressed, the RESET key will be ignored during the period from poll to the end of a transmission to the host. Prior to the poll, a RESET action will cancel both the AID code and I/O pending. After transmission to the host is ended, RESET will reset the AID code.

RESET causes print ID mode to terminate. The cursor then reappears, and the old printer ID is displayed in the indicator row.

DUP (Duplicate) Key

Operation of this key causes a unique character code to be entered into the display buffer, a Tab key operation to be performed, and the MDT bit to be set to 1. The DUP key is provided on all keyboard types except operator console. The DUP character provides a means of informing the application program that a "duplicate" operation is indicated for the rest of the field in which it is located. The DUP character is transferred as a DUP code (Figures 1-2 and 1-3) when the data is read from the display to the program. No duplicate operation is performed at the 3276. The DUP character, when stored in a device buffer, is displayed as an asterisk (*) on 3178, 3276, 3278, and 3279 displays using mono-case mode and is printed as an asterisk (*) on a printer. On 3178, 3276, 3278, and 3279 displays using dual-case mode, DUP is displayed as an asterisk with an overscore ($\overline{*}$).

Pressing the DUP key does not affect the current status of extended attributes; however, the PS selection has no effect on a DUP character.

Operation of this key when the cursor is located in a field-attribute character location or is within a protected data field disables the keyboard; no character locations are cleared, the cursor is not moved, and the MDT bit is not set.

FM (Field Mark) Key

Operation of this key causes a unique character code to be entered into the display buffer and the MDT bit to be set to 1. The field mark character provides a means of informing the application program of the end of a field in an unformatted buffer or subfield in a formatted buffer. The field mark character is transferred as an FM code when the data is read from the display to the program. The field mark character, when stored in a device buffer, is displayed as a semicolon (;) on the 3178, 3276, 3278, and 3279 displays using mono-case mode, and is printed as an asterisk (*) on a printer. On the 3178, 3276, 3278, and 3279 displays using dual-case mode, FM is displayed as a semicolon with an overscore ($\overline{;}$). The Field Mark key is not provided on operator console type keyboards.

Pressing the FM key does not affect the current status of extended attributes; however, the PS selection has no effect on an FM character.

Operating this key when the cursor is located in a field-attribute character location or within a protected data field disables the keyboard; no character locations are cleared, the cursor is not moved, and the MDT bit is not set.

Program Attention Keys

The program attention keys for the 3178, 3276, 3278, and 3279 displays are CLEAR, ENTER, the Program Function (PF) keys, and the Program Access (PA) keys. The use of a PA or PF key during a System Services Control Point (SSCP) session results in an input-inhibited condition. Refer to "Keyboard Disabled (INPUT INHIBITED Indicator Is On)." On 3178, 3276, 3278, and 3279 displays, the operation of the CLEAR key also clears the display screen of all data to nulls (except the indicator row), sets all extended attributes to X'00', and positions the cursor at location 0,0 on the display.

Operation of the CLEAR key does not change shift status except that it will remove the NUM symbol, if displayed. It does not perform a reset function. If an alternate screen size has been selected, the CLEAR key will reset the screen to the default size. When SNA/SDLC is used, the action of the CLEAR key depends upon the type of session. In 3276 BSC, the CLEAR key AID code is sent to the host. When SNA/SDLC is used, the CLEAR key AID code is sent to the host when CLEAR is pressed while in the LU-LU session. While in test mode, the CLEAR key does not cause an AID to be sent to the host.

Note: *Not all program attention keys are available on each type of 3276 keyboard.*

SYS (System) REQ Key

When the 3276 operates in remote SNA/SDLC, the operator can use the SYS REQ key for SSCP-SLU and PLU-SLU session switch procedures. SYS REQ also simultaneously initiates keyboard reset and clear functions. SYS REQ performs these functions despite the presence of input-inhibited conditions except (1) when inbound processing is queued for the display station, in which case the Input Inhibited What symbol appears, and (2) when Printer Busy, Printer Very Busy, or Printer Not Working is displayed, which results in no response when SYS REQ is pressed. (Inbound processing queue is the time from when an AID generating key is pressed until regeneration to the line buffer transfer has been completed.)

In BSC operation, the SYS REQ key performs the test-request function. The automatic reset function is not available. Refer to "Test Request Read" under "Read Modified Command" in Chapter 1.

The ALT key must be pressed and held while the SYS REQ key is pressed.

DEV CNCL (Device Cancel) Key

The operator may use DEV CNCL to cancel a current outstanding print request to a terminal printer if input is inhibited because of a Printer Busy or Printer Very Busy condition. A request initiated by the Print key is dequeued, and the keyboard is restored. A host print request is dequeued, and a negative response is sent to the host. The printer Busy symbol is replaced by the Time symbol.

DEV CNCL is also used to remove Device Not Functional conditions. Any coexisting malfunction-while-printing symbol is also removed.

Following use of the Print key, the keyboard is restored. After a host-initiated print, the Printer Not Working symbol is replaced by the Time symbol.

During other input-inhibited conditions, DEV CNCL causes no response, except that it is queued or detected (with subsequent indication) during certain Time conditions in other situations. Use of DEV CNCL in other situations results in no indication.

The ALT key must be pressed and held while the DEV CNCL key is pressed, to cancel a request and restore the keyboard.

If DEV CNCL is used during a print ID operation at the 3276, the 3276 remains in print ID mode.

⇧ (SHIFT Key) - 3178, 3276, 3278, or 3279

Shift keys perform the upshift function. When the typewriter keyboard becomes ready initially, only characters located on the bottom position of the key tops can be entered from the keyboard. By pressing and holding the Shift key, characters shown on the top position of the key tops can be entered. On 3178, 3276, 3278, and 3279 displays, the shift "up" state is indicated to the operator in the Operator Information Area on the display screen. Pressing the Shift key will reset the Lock key.

(LOCK Key) - 3178, 3276, 3278, or 3279

The Lock key fixes upshift character selection. It is deactivated by pressing the Shift key. When the Shift key on a 3178, 3276, 3278, and 3279 typewriter keyboard is used, the shift state is indicated to the operator in the Operator Information Area on the display screen.

⇧ (NUM Key) - 3178, 3276, 3278, or 3279

The Numeric key on the equivalent 3178, 3276, 3278, and 3279 keyboards is used to perform the upshift function, equivalent to the Shift keys on the typewriter keyboards. The "up" shift state is indicated to the operator in the Operator Information Area on the display screen.

(NUM LOCK Key) - 3178, 3276, 3278, or 3279

The Numeric Lock key on the 3178 data entry, and 3276, 3278, 3279 data-entry/data-entry keypunch layout keyboards fixes the upshifted character selection, but will not disable the Numeric Lock feature. It is released by pressing the Numeric Lock key again. The keyboard then reverts to shift or to programmed control shift. The shift "up" state is indicated to the operator in the Operator Information Area on the display screen whenever the Numeric Lock key is pressed.

⇩ (Alpha Key) - 3178, 3276, 3278, or 3279

When the data entry 3178, 3276, 3278, 3279 or data entry keypunch layout 3268, 3276, 3278 keyboards have been programmed for non-alpha shift, characters shown on the bottom of the key tops can be selected by holding the Alpha key and entering the desired characters. When power is applied, the keyboard is in lowercase alpha mode.

CURSR SEL (Cursor Select) Key

The CURSR SEL key on 3178, 3276, 3278, and 3279 keyboards allows the selector-light-pen-detection function to be performed from the keyboard. The CURSR SEL key may be used on any field defined as a selector-light-pen-detectable field (as described under the heading "Selector-Light-Pen Operations"). However, a cursor-select field does not require the space or null character padding constraints associated with the selector-light-pen-detectable field and cursor-select can occur within the field on a line different from that of the attribute that describes the field.

Cursor-select operations may be immediate or deferred (as defined for selector-light-pen fields). The field used for a cursor-select operation may also be defined in the following format:

- Basic attribute character as defined for selector light pen.
- Designator character as defined for selector light pen.
- Data character(s) Optional
- Basic attribute character Next field.

This format is not applicable when using the selector light pen. When defining a cursor-select field, the attribute character may not be located in the last line of the display with the designator character in the first line.

ATTN (Attention) Key

The ATTN key on the 3178, 3276, 3278, and 3279 keyboards is operable in SNA/SDLC in an SNA LU-LU session, with the following exceptions:

1. When inbound processing is queued for the display.
2. When in Shutdown condition.
3. When in Data Traffic Reset state.
4. When a second or successive ATTN that occurs prior to completion of processing for the first ATTN is ignored (with no indication).

Use of ATTN in any session except LU-LU causes an Input Inhibit Minus Function.

The ATTN key is inoperative in BSC and will cause an Input Inhibit Minus Function when pressed.

CURSR (Cursor) BLINK Key

Pressing the CURSR BLINK key causes the cursor (either the bar or the rectangular cursor) to blink. Activating the key again causes the blinking to stop. This key function is available on keyboards attached to the 3178, 3276, 3278, or 3279.

ALT CURSR (Alternate Cursor) Key

Pressing the ALT CURSR key while holding the ALT key changes the cursor display. The underlined type of cursor is changed to a rectangular cursor. Conversely, the rectangular cursor is changed to the underlined type of cursor by activating the ALT CURSR key. This key function is available on keyboards attached to the 3178, 3276, 3278, or 3279.

TEST Key

The TEST key on the 3178, 3276, 3278, or 3279 keyboard is used to invoke test functions resident in the 3276. Pressing the TEST key (while holding ALT key) clears and resets the display screen, and the test mode indication turns on, despite any input-inhibited conditions, with the following exceptions: If Printer Busy, Printer Very Busy, or Printer Not Working is displayed, or if the security key is locked, use of TEST results in no response. The control unit places the device to be tested in test mode, and the operator identifies the test function desired. The operator terminates test mode by pressing the TEST key again.

When the 3276 uses SNA/SDLC, the control unit enters test ownership state. When the 3276 operates in remote BSC mode, Intervention Required is generated if a command is received for the display when in test mode. When test mode terminates normally, status with Device End is generated.

(Click Key)

A clicking sound may be produced as keys are pressed on keyboards attached to 3178, 3276, 3278, and 3279 displays. The clicking sound is controlled by operating conditions such as input inhibit. For example, if the clicking sound is enabled and an input-inhibited condition occurs, the key clock is then disabled, and vice versa. By pressing the Click key, the operator can activate the clicking sound if it has been turned off or prevent clicking if it has been activated.

(Print Key)

The Print key is used to initiate a local copy function from a keyboard attached to a 3178, 3276, 3278, or 3279 display.

IDENT Key

The IDENT key is used to assign a printer or printer class, while performing a local copy function. (The ALT key must be pressed to activate the IDENT key.) When the IDENT key is pressed, the cursor disappears from the screen, and the Printer Assignment symbol appears with two underlined characters in the "nn" position. The operator may then enter the ID in the "nn" position. (Display stations with one of the PS features always select the base character set for the printer ID; if a symbol set is active when the IDENT key is pressed, it is suppressed and then made active again at the end of the printer ID sequence.)

If the specified printer is not authorized (that is, the matrix does not permit the display to copy to the selected device or class of devices), the keyboard is locked and the Input Inhibited Operator Unauthorized symbol is displayed. If the print ID is not in the matrix, the keyboard is locked and the Input Inhibited What Number symbol is displayed. The contents of the printer status field are displayed for the input-inhibited condition, the cursor appears, and the keyboard is locked. The operator must reset and then retry the print ID sequence.

If the selected print class or printer is valid and authorized for this display, the connection indicator will change to indicate the new connection, and print ID mode is terminated. The cursor reappears, and the keyboard remains unlocked.

When in print ID mode, the following rules apply:

1. Numeric information is displayed at the "nn" position in the indicator row. Each character is then checked for validity.

2. The RESET key and other keys or functions that cause a reset operate normally and cause print ID mode to be terminated. The cursor reappears, and the contents of the printer status field are displayed.

3. The ATTN and DEV CNCL keys, the security key, and unsolicited host read and write operations operate normally in the 3276, except that the 3276 print ID mode is terminated when the Start Print bit in the WCC of the host write command is on. The cursor reappears, and the contents of the printer status field are displayed in the indicator row.

4. Other keys that function during a keyboard inhibit condition also function while in print ID mode without causing termination.

5. All other keys that are not honored during keyboard inhibit conditions cause the Input Inhibit-What symbol to be displayed and terminate print ID mode. In this case, the cursor reappears and the contents of the printer status field are displayed in the indicator row.

Dead Keys, Canadian-French Keyboards

When pressed, the accent keys which show individual accents on the Canadian-French keyboards appear on the display, but the cursor does not move. These accent keys are referred to as dead keys. A subsequent character which receives the accent must be keyed next. If the subsequent character is valid, a unique composite character is formed. Refer to the *3270 Character Set Reference* manual, GA27-2837, for keyboard layouts, I/O codes, and identification of valid accent characters.

Pressing an accent key places the keyboard in dead key mode until a valid second key is pressed. When the second character of a dead key sequence is invalid, only the Shift, DEV CNCL, ALT, Click, ALT CURSR keys, and the Dual Case/Mono Case switch and security key are operational. Use of ATTN in this case causes the Input Inhibited Minus symbol to appear. Use of any other key terminates the operation and causes an Input Inhibited Accent Plus What symbol to appear on the screen.

The selector light pen and the magnetic slot reader (MSR) do not function while in a dead key sequence. If used, they cause the dead key sequence to be aborted, and the keyboard is inhibited, with the What symbol displayed.

All other nonkeyboard-related functions that occur during a dead key sequence are performed normally. If performance of the function causes the dead key sequence to be aborted, the keyboard is inhibited and the What symbol is displayed after the function has been performed.

In all of these conditions, the dead key sequence is aborted, and an accent only is displayed at the cursor position. The operator must reset and rekey both the accent and the valid character.

Numeric Lock Feature Operation

When the Numeric Lock feature is installed, the character (0-9), decimal sign, minus sign (-), and DUP may be entered by the operator in a field identified in the field-attribute byte as numeric and unprotected. MSR input is also accepted. Operating any other key that can enter a displayable character causes an input-inhibited condition. In addition, the NUM symbol lights on the 3178, 3276, 3278, and 3279 displays. Operating the RESET key enables the keyboard (if disabled), and the NUM symbol (3178, 3276, 3278, 3279) goes out. The nondisplay/nonprint attribute bits 4 and 5 and MDT bit 7 operate normally.

For 3178, Numeric Lock is a basic function.

The Numeric Lock feature can be overridden as follows:

1. On a data-entry keyboard, any character can be entered by pressing (and holding) the Numeric Shift key or the Alpha Key, depending upon the character to be keyed, and then pressing the desired key(s).

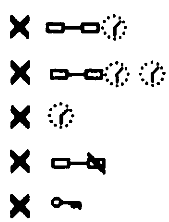
2. On a typewriter keyboard, any uppercase character or symbol can be entered by pressing (and holding) the Shift key and then pressing the desired key(s).

3. On an APL or a text keyboard, any non-APL or non-Text uppercase character or symbol can be entered by pressing (and holding) the shift key and then pressing the desired key(s); also, any APL or Text uppercase or ALT-Shift character can be entered by placing the keyboard in APL mode or text mode (pressing APL ON/OFF with ALT or TEST ON/OFF with ALT), pressing (and holding) the Shift key or the ALT key (depending upon the character to be keyed), and then pressing the desired key(s).

Note: *If any devices with attribute-select or overlay keyboards are attached to a control unit, numeric lock for those keyboards is set by an option taken during customizing. The option taken applies to all devices with attribute-select and overlay keyboards; if numeric lock is set off, all these devices have numeric lock off.*

Keyboard Disabled (Do Not Enter Condition)

When the Do Not Enter (X symbol) condition is indicated in the Operator Information Area, the keyboard and other input devices are disabled. In cases caused by operator key action, the input-inhibited condition can be cleared by using the RESET key unless one of the following Do Not Enter indications is present:



Do Not Enter is turned on by:

1. Operation of a Program Attention key.

2. A selector-light-pen attention that caused an I/O interruption or that resulted in an operator error.

3. A magnetic slot reader (MSR) that caused an I/O interruption.

4. Turning the security key to the off position when the Security Keylock feature is installed, when power is applied initially.

5. A system-initiated I/O operation addressed to that unit.

6. Operation of any alphameric key or of the DUP, FIELD MARK, ERASE EOF, or DEL key, when the cursor is in a protected field.

7. Operation of any alphameric key not included in the numeric key grouping when the cursor is in a numeric field, without simultaneously operating either the Alpha or Numeric shift key on a data entry keyboard or the Shift key on a typewriter keyboard, when the Numeric Lock feature is installed on a keyboard.

8. Copying of data in the refresh buffer to another terminal.
9. The occurrence of a Machine Check, Program Check, or Communications Check.
10. The terminal's being in receive state under SNA protocol.

Do Not Enter is turned off by:

1. On 3178, 3276, 3278, and 3279 displays: Receipt and execution of a WCC with the Keyboard Restore bit on when the System Lock or Time symbol is displayed.
2. On 3178, 3276, 3278, and 3279 displays: Receipt and execution of an Erase All Unprotected command when the System Lock or Time symbol is displayed.
3. Turning of the security key to the on position (if the Do Not Enter indicator was turned on because the security key was in the off position).
4. Operation of the RESET (except as noted under "Reset Key"), TEST, or SYS REQ key in BSC operation.
5. Depression of the DEV CNCL key after receipt of a Printer Not Working symbol.
6. Termination of a Time condition.

An I/O operation that leaves the 3276 in a send state but does not unlock the keyboard can be cleared by using the RESET key on the 3178-, 3276-, 3278-, or 3279- attached keyboards. When the Do Not Enter condition, INPUT INHIBITED, exists on a 3276, 3278, or 3279 display, manual input to the unit from the keyboard or selector light pen is inhibited, except for use of the Shift, ALT CURSR, CURSR BLINK, and Click keys.

Do Not Enter is cleared by a reset action from the control unit or the operator. During a buffer transfer when the 3276 is executing a Copy command in BSC, keystrokes are accepted for processing. The 3276 will queue at least two keystrokes and will process the input, if the queue is not exceeded, after the poll sequence to the keyboard is restored.

If the queue capacity is exceeded, all queued keystrokes are discarded, and the What symbol is displayed. The What symbol is also indicated if input is attempted during Time Symbol conditions or during Print Busy or Printer Not Working input-inhibited conditions.

If the input-inhibited condition is caused by a Machine Check, only an operator reset action can reset the device (if it can be reset). Only an operator reset action will reset a device that shows a Communication or Program Check condition. The Communications Check inhibit symbol does not reappear unless it is reencountered by pressing a host communication key on the display keyboard.

Selector-Light-Pen Operations

The selector light pen, shown in Figure 2-6, is a light-sensitive pen that can detect the light emitted from characters displayed on the 3276, 3278, or 3279 displays. With the selector light pen, the operator can select from a list or table of displayed items and can then cause those selections to be identified to the application program.

The selector light pen is operated by pressing the tip of the pen against the screen on fields programmed for selector-light-pen operations.

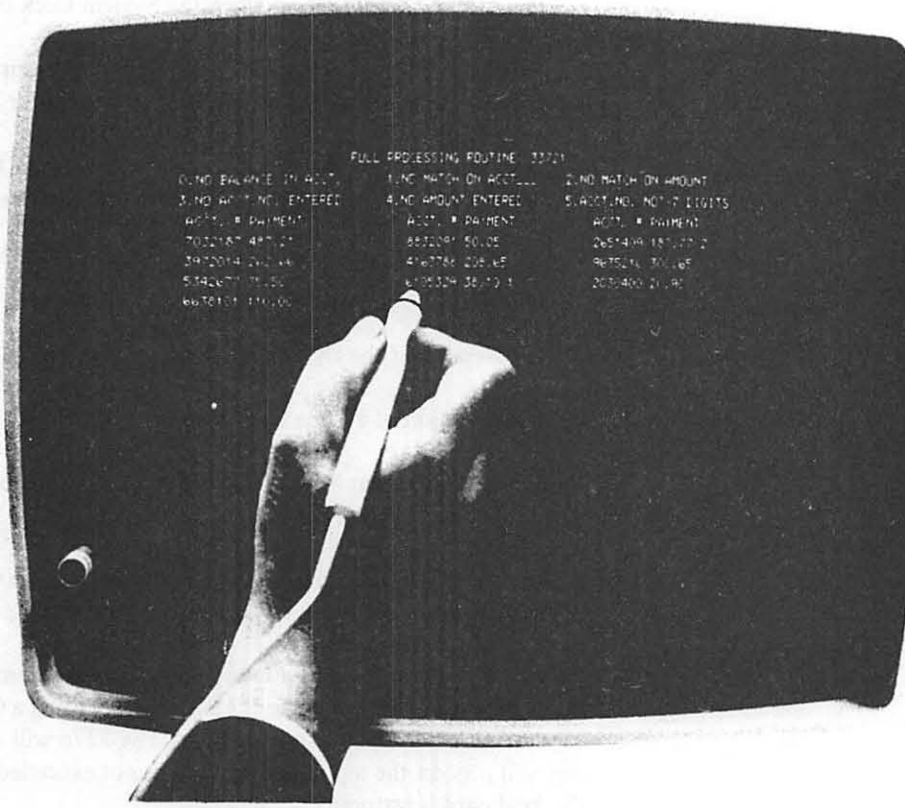


Figure 2-6. Selector Light Pen

Selector-Light-Pen Field Format

A field that is to be used for selector-light-pen operations must be defined in the following format:

	Data Character	Preceding field (on the same line as the SPD field).
	3 Space or Null Characters	Three space or null characters must precede the field-attribute character defining the SPD field unless the attribute character is the first character on the line.
	Field Attribute Character	The field-attribute character defines the field as displayed and selector-light-pen-detectable. (An SPD field may be protected or unprotected, alphameric or numeric.)
SPD (Selector Pen Detectable) Field	Designator Character	The designator character which defines the type of operation that will be performed by detection on this field.
	Displayed Data	One or more displayed alphameric characters for sensing by the selector light pen.
	3 Space or Null Characters	Three space or null characters are required when a new field follows on the same line as the SPD field.
	Field Attribute Character	Succeeding field (on the same line as the SPD field).
	Data Character	

The field-attribute character, the designator character, and displayed alphameric characters must be on the same line. If the field extends beyond one line, only those characters on the same line as the attribute character can be detected by the selector-light-pen. A maximum of 12 detectable fields in the 3276 may precede the last detectable field on any given line.

Designator Characters

Designator characters are used to define two types of selector light pen fields: selection fields and attention fields. Each type of field performs a different selector-light-pen operation.

The selection field is defined by a question mark (?) designator character. When the selector light pen detects on a selection field, the MDT bit in the field-attribute character for that field is set (1) in the display buffer. Also, the designator character is automatically changed on the screen to a greater than (>) sign to provide a visible indication to the operator that the detection was successful. If a mistake was made and the operator again detects on that same field, the > changes to a ? and the MDT bit for that field is reset (0).

The attention field is defined by a space or null designator character. A detection on an attention field causes an I/O pending (attention) at the display. This I/O pending indicates to the program that the selector-light-pen operation has been completed. The program may then issue a Read Modified command to obtain the address of each field that was selected or modified by the operator.

A second type of attention field for the 3276 is defined by an ampersand (&) designator character. A selector-light-pen detection on a field containing an ampersand designator sets the MDT bit and causes an ENTER key I/O pending condition at the 3276. The display responds to a poll or Read Modified command, and both the address and the data in each field that was modified by the operator are returned to the application program.

Programming Notes:

1. The application programmer should be aware that both normal intensity and high-intensity unprotected fields can be modified by the display station operator to become selector-light-pen-detectable fields.

2. Use of the Selector Light Pen feature without the ampersand (&) designator character is anticipated to be such that the program will correlate the address of each SPD field with the data associated with it. Therefore, to minimize TP line loading, channel loading, and buffer size requirements, only the addresses of selector-light-pen-detected fields are required to be sent to the application program; the field data is not included.

3. Users who wish to combine selector-light-pen-detect input with keyboard input must use the keyboard or the ampersand designator character to generate the I/O pending. Use of the selector light pen on a space or null designator field or on an attention field to generate the I/O pending will result in transmission of only the addresses of the fields in which the MDT bit was set.

Figure 2-7 shows a sample display with fields defined for selector-light-pen operation. In this sample, "FULL", "50MG", and "4 TIMES" are all preceded by > designator characters to indicate that they were selected by the operator. When the operator detects on the word "EXIT", which has no displayed designator character, an I/O pending occurs and the program obtains the addresses of the three selected fields.

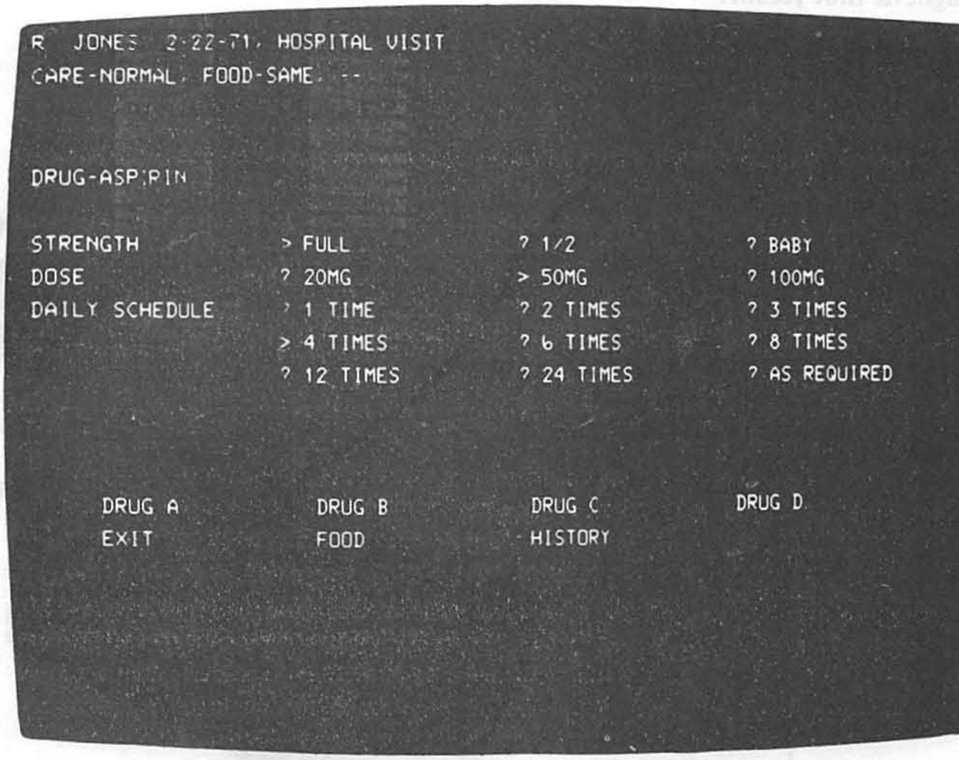


Figure 2-7. Sample Display Screen for Selector-Light-Pen Operations

Security Keylock

The Security Keylock is a security-enhancement special feature that provides a key-controlled lock for 3276 display. When the key is in the "off" position or is removed from the display station, the message buffer is "locked," which prevents entry, modification, and display of data. The display station is unavailable to programmed read or write operations and to operator inputs such as keyboard entry, card reader entry, and selector-light-pen operations.

Programmed attempts to access display stations that have the key turned off or removed from the lock result in responses being returned to the CPU by the 3276. 3276 responses are device- and operation-dependent. They are summarized in the following table:

Device Attachment	Operation	Response
3276-1, -2, -3, -4 (Note 2)	Specific Poll	IR Status and Sense
	General Poll	EOT
	Selection Addressing Sequence	RVI
3276-11, -12, -13, -14	Normal Flow Requests	IR (Negative Response 0802)

Notes:

1. *Each operation in the Operation column applies to each corresponding unit in the Device Attachment Column.*
2. *When the SDLC/BSC Switch feature is installed and the switch is in the SDLC position, the response from the terminal with the key off to normal flow requests is IR (neg resp 0802).*

Magnetic Slot Reader

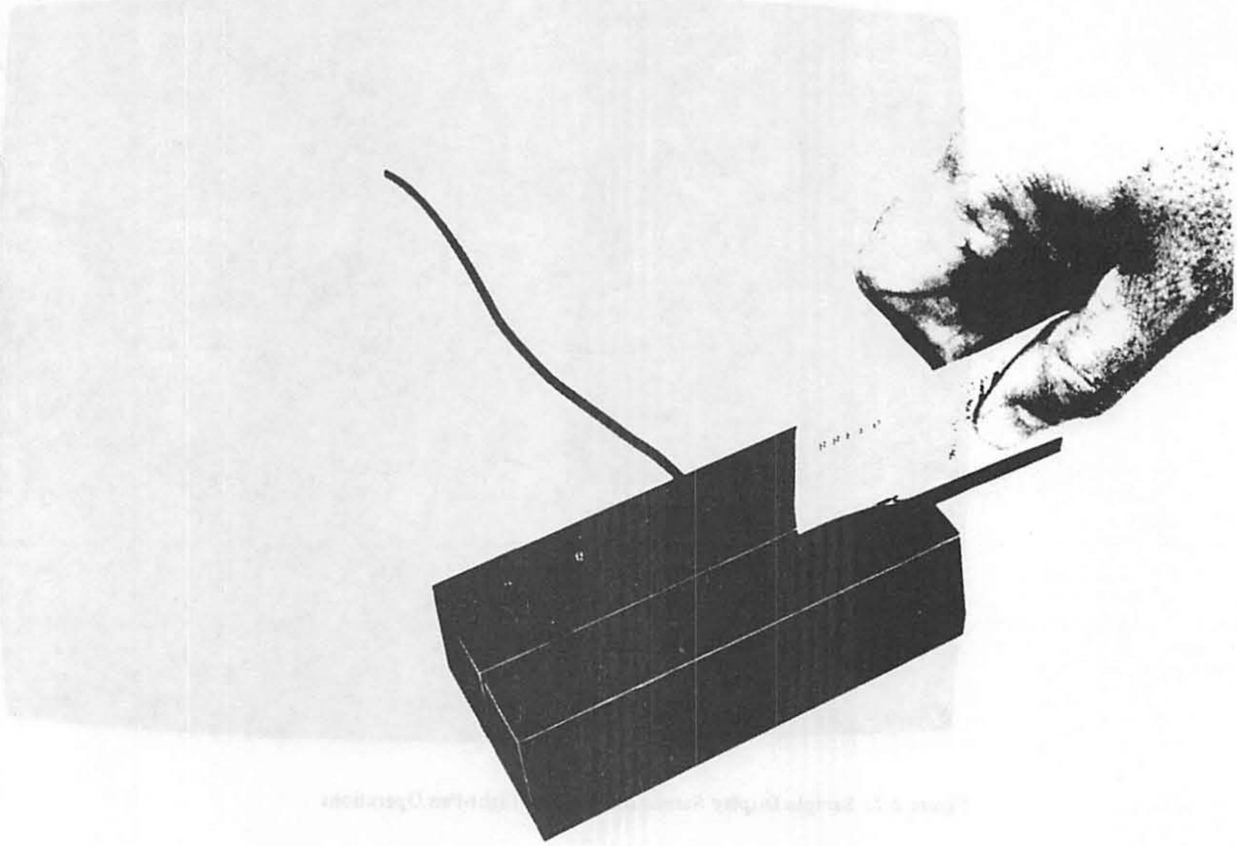


Figure 2-8. Magnetic Slot Reader (3276, 3278, and 3279 Attachments)

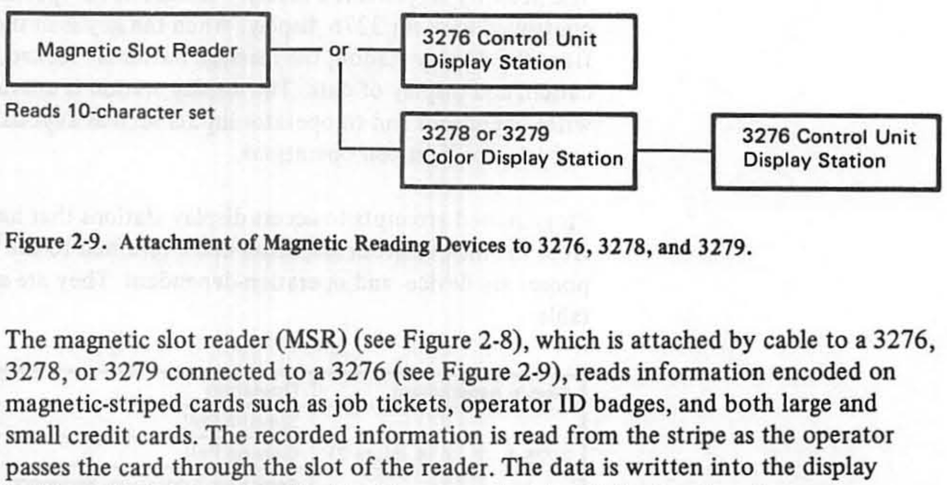


Figure 2-9. Attachment of Magnetic Reading Devices to 3276, 3278, and 3279.

The magnetic slot reader (MSR) (see Figure 2-8), which is attached by cable to a 3276, 3278, or 3279 connected to a 3276 (see Figure 2-9), reads information encoded on magnetic-striped cards such as job tickets, operator ID badges, and both large and small credit cards. The recorded information is read from the stripe as the operator passes the card through the slot of the reader. The data is written into the display buffer at the location specified by the cursor, but is not displayed on the screen. If the device supports the Structured Field and Attribute Processing option, the Extended Attribute Buffer (EAB) is updated. After the information is read, an I/O pending is generated at the display to inform the program that the data can be retrieved and transferred to main storage.

Character		Bit Pattern 2^0	2^1	Direction of Recording 2^2	2^3	P	Hex Code	I/O Interface Code (Note 4) EBCDIC	ASCII
Data	0	0	0	0	0	1	0	F0	30
	1	1	0	0	0	0	1	F1	31
	2	0	1	0	0	0	2	F2	32
	3	1	1	0	0	1	3	F3	33
	4	0	0	1	0	0	4	F4	34
	5	1	0	1	0	1	5	F5	35
	6	0	1	1	0	1	6	F6	36
	7	1	1	1	0	0	7	F7	37
	8	0	0	0	1	0	8	F8	38
	9	1	0	0	1	1	9	F9	39
Control	(Special - See Note 1)	0	1	0	1	1	A	7A	3A
	SS, or RSS	1	1	0	1	0	B	7B	23
	EOI (Note 2)	0	0	1	1	1	C	7C	40
	Field Separator	1	0	1	1	0	D	7D	27
	(Unassigned)	0	1	1	1	0	E	7E	3D
	ES (Note 3)	1	1	1	1	1	F	7F	22

Note:

1. *This character is reserved for operator identification only and must be located in the first data character position.*
2. *MSR: SS (Start Sentinel); RSS (Reverse Start Sentinel).*
3. *EOI (End of Inquiry) is treated as an error by the MSR (3276, 3278, and 3279 displays). The card is rejected, and the MSR red light is turned on.*
4. *MSR: ES (End Sentinel).*
5. *Programmers use only the four least-significant bits of the I/O interface code.*

Figure 2-10. 10-Character Set Used with Magnetic Slot Reader

With the 10-character set shown in Figure 2-10, the maximum number of characters that can be read is:

- 40 characters at 3 bits per millimeter (75 bits per inch) and at 8.3 bits per millimeter (210 bits per inch)
- 100 characters at 5 bits per millimeter (128 bits per inch)

Note: *A minimum of seven characters must be encoded between the Start Sentinel and End Sentinel Characters.*

The 10-character set may be used to log on and log off in SNA mode (LU-LU session only; not SSCP-LU session) or in a non-SNA mode.

10-Character Set

The 10-character set shown in Figure 2-10 comprises 10 numeric characters plus a Field Separator and control characters. Each character is composed of a 4-bit pattern plus an odd-parity bit. This bit pattern is recorded with the low-order bit recorded first. A longitudinal redundancy check (LRC) character is placed at the end and is protected by an odd-parity bit of its own.

Characters are recorded, low-order bit first, beginning at the left-hand side of the magnetic stripe when the stripe is at the bottom of the card or badge as you face the magnetic material. The characters are read in one direction only.

Magnetic-Stripe Format

The format used on the magnetic stripe is in the sequence shown in Figure 2-11.

When the SS character is read from the magnetic stripe, a field-attribute character is entered automatically into the cursor-identified location of the buffer (provided the cursor is at an unprotected character location). This attribute character defines the following data field as protected, alphameric, and nondisplay or nonprint. As the data characters are read into the buffer, they are stored starting at the first character location after the field-attribute character. As each data character is stored in the buffer, the cursor advances one buffer location. The cursor advancement is all the operator sees on the display screen when using the operator identification card reader. When the operator uses the magnetic slot reader, the cursor does not move as the card is passed through the slot, but is repositioned after the card has been read.

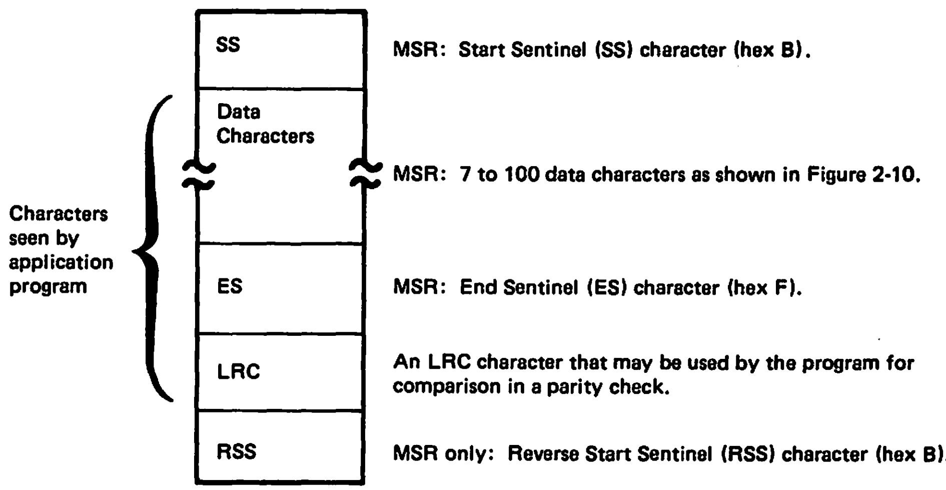


Figure 2-11. Magnetic-Stripe Format (MSR Using 10-Character Set)

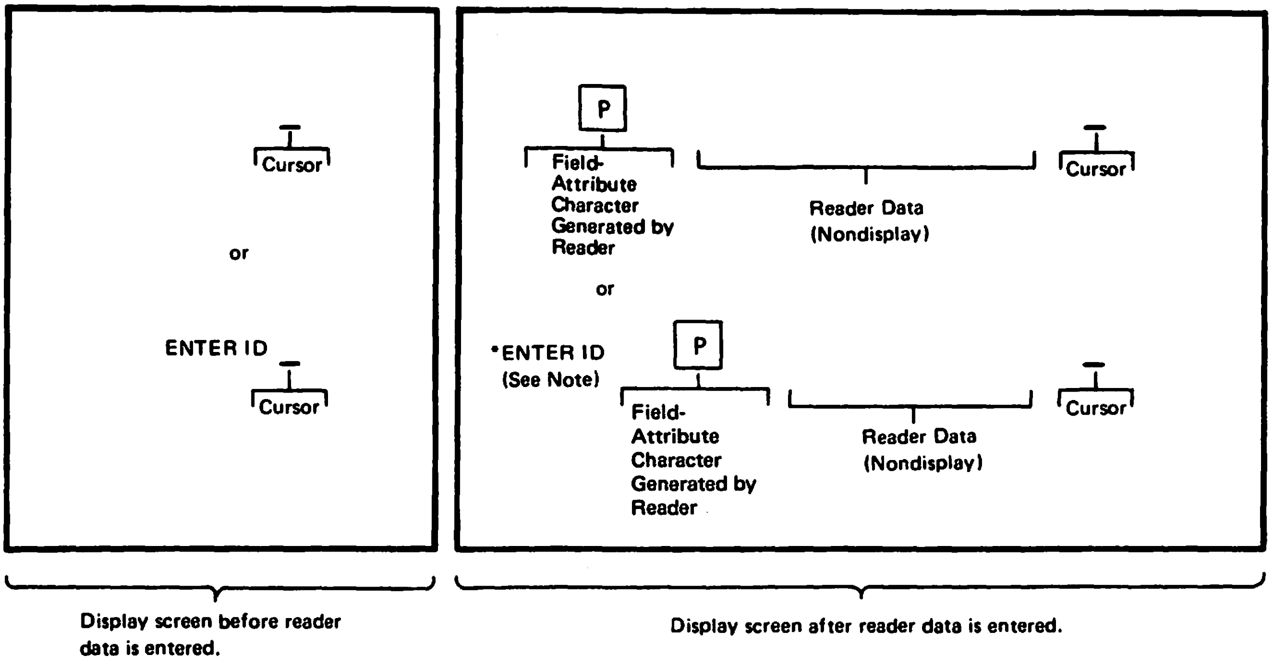
Operational Differences because of Screen Format

When the 10-character set of Figure 2-10 is being used with the magnetic slot reader (MSR), differences exist in the content of the data stream sent to the application program, depending upon whether the display screen is unformatted or formatted.

When an unformatted screen (that is, a screen without attribute characters of fields) is being used, the operation of the display results in an inbound data stream as shown in Figure 2-12.

The reader operation formats the screen by the automatic generation of the field-attribute character at the cursor position by the reader.

A formatted screen has at least one field-attribute character defined at initial presentation. This may be the only field-attribute character, as in the instruction sequence ENTER ID; or one or more attributes may be required, as, for example, in the instruction sequence NAME, TITLE, ID CARD READER.



Note: *The ENTER ID is not displayed because it is within a nondisplay field, defined by the reader attribute character.*

[P] = Protected field-attribute character

Inbound Data Stream

Inbound Data Stream	
AID	Set to indicate input from a magnetic-stripe reading device.
Cursor Address	Address of the cursor upon completion of the reader operation.
SBA	Set Buffer Address.
Start of Data Address	Address of the first data character following the field-attribute character.
Data	The reader data followed by any additional information present in the display buffer. The additional information can be initiated by the application program as ENTER ID (as shown in the example) or entered by the operator before the reader operation is started.

Note that with an unformatted screen the reader data is the first text in the data stream sent to the application program.

Figure 2-12. Operation of the Display with an Unformatted Screen (MSR Using 10-Character Set)

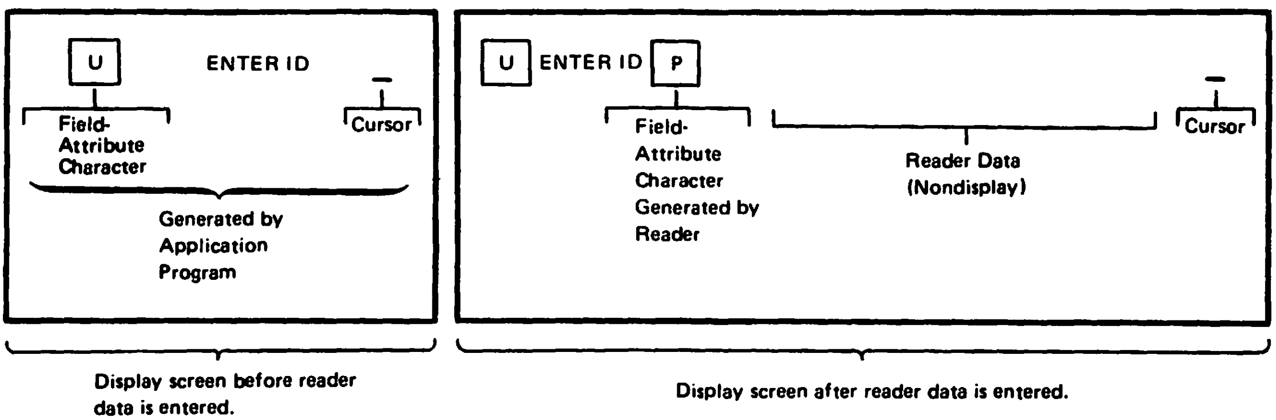
The operations of the 3276, 3278, and 3279 with the MSR are identical when formatted screens are used.

Two fields (new data field and previous data field), with the MDT bits set, are sent to the application program, because the displays treat all information from the reader as data until after the information is written into the display buffer. Also, the MDT bit is set in the reader-generated field-attribute character that was initiated when the data was entered.

The following examples are included to help clarify operation of the reader with a formatted screen.

Example 1:

If the MSR field is set up by the application program as an unprotected field and contains instruction information, the inbound data stream is as shown in Figure 2-13.



Inbound Data Stream

AID	Set to indicate input from a magnetic-stripe reading device.
Cursor Address	Address of the cursor upon completion of the reader operation.
SBA	Set Buffer Address.
Start of Data Address	Address of the unprotected (U) field-attribute character + 1.
Data	ENTER ID, in the example above.
SBA	Set Buffer Address.
Start of Data Address	Address of the protected field-attribute character +1. In this case, the address of the first data character from the reader following the protected field-attribute character.
Data	The reader data (and any data between the cursor and the next field-attribute character).

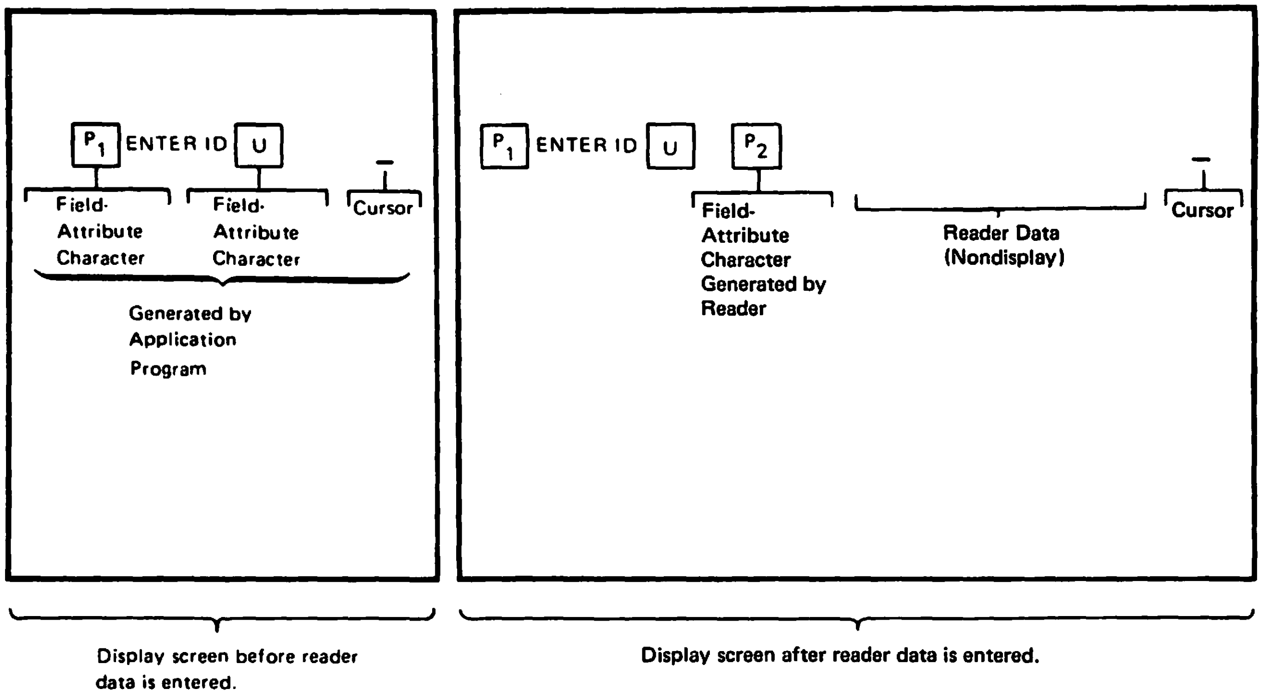
U = Unprotected field-attribute character

P = Protected field-attribute character

Figure 2-13. Operation of the Display with a Formatted Screen (Using 10-Character Set), Example 1

Example 2:

When the MSR field is set up by the application program as an unprotected field, with the cursor directly following an unprotected field-attribute character, the inbound data stream is as shown in Figure 2-14.



U = Unprotected field-attribute character

P = Protected field-attribute character

Note: *Rules for positioning modified data on formatted screens apply. The position of reader data in the inbound data stream depends on the field position in the format.*

Inbound Data Stream	
AID	Set to indicate input from magnetic-stripe reading device.
Cursor Address	Address of cursor upon completion of reader operation.
SBA	Set Buffer Address.
Start of Data Address	Adress of the unprotected (U) field-attribute character +1. In the example above, it will be the address of the P_2 field-attribute character.
SBA	Set Buffer Address.
Start of Data Address	Address of the P_2 field-attribute character +1. In this case, the address of the first data character from the reader following the P_2 field-attribute character.
Data	The reader data (and any data between the cursor and the next field-attribute character).

Figure 2-14. Operation of the Display with a Formatted Screen (Using 10-Character Set), Example 2

Error Conditions

MSR data will not be written into the display buffer if any of the following error conditions exist when the magnetic stripe is read:

- The SS (MSR) character is not successfully connected to a field-attribute in the display buffer.
- The cursor is located in a protected field.
- The cursor is located in a field-attribute character location.
- The display is busy performing another operation.

MSR Validity Test

The proper use of the MSR as an identification and data-entry device requires that the application program perform certain validity tests. The following guidelines are recommended for proper operation:

1. No field should be accepted as reader input unless the reader AID code is set.

2. For preformatted displays, the application program must know the location of the field defined to receive the reader data and the exact location of the entered data, based upon the hardware operation that was previously defined. The use of the cursor address present in the inbound data stream, in combination with the AID byte to ensure reader input, is an additional technique that can be used to ensure the integrity of the data. For unformatted displays, the reader data is always presented as the first data entry in the input record to the application program.

3. For preformatted displays, it is advisable to terminate the reader data field with another attribute byte.

4. Upon completion of the reader operation, the application program should check for the presence of the ES character. Absence of this character means the reader data has not been transferred successfully. This condition can occur under the following error conditions:

   a. The detection of a parity error in any data character in the ES character.

   b. An interruption of normal data flow from the reader.

   c. The cursor has been moved to a field-attribute character location. This means the field defined for reader input is too small or the cursor was not initially positioned at the beginning of a correct-length field.

5. Upon completion of the reader operation and a successful check for the ES character, the LRC character may be used for a parity check to ensure integrity of the data.

   Because of the makeup of the 10-character set codes (4 bits plus parity bit), only the right-hand 4 bits are of concern. The application program should set up a 1-byte field initialized to X'0B'. This is the SS character, which is not included in the inbound data stream but which is used to compute the LRC. As each character is checked for validity, it is exclusively ORed into this field. This operation should include the ES character and the LRC, resulting in the byte containing zero. If the byte is nonzero, it means the result of the check on the data characters, including ES, does not equal the LRC, and a parity error has occurred.

6. If the reader input field is to be reused, the application program must remove the hardware-generated field-attribute character and reader input data. The location of this character can be derived from the inbound data stream by using one less than the start of the data address preceding the input data. Additionally, the cursor is located one position beyond the end of the reader data field.

   The card field may be reused if more than one card input is required or if the original attempt was unsuccessful and the application program desires to retry the operation.

7. Text for all fields having the MDT bit set is transferred to main storage when the reader data is retrieved in response to the reader-generated I/O pending.

8. The cursor must be moved out of the reader-generated field before further keyboard activity is allowed.

9. A test card, PN1742659, is delivered with each 3276/3278/3279 Magnetic Reader Control feature. The test card data placed in the display buffer is as follows:

   01234567899876543210012345 67F4

   Care should be taken that the card is not accidentally auto-entered. The display should be placed in test mode to avoid auto-entering magnetic-stripe information to the host.

MSR Operator Indicators and Alarm

The magnetic MSR contains three operator indicators and a buzzer. The indicators are color-coded green, yellow, and red. When all indicators are off, power has not been applied to the MSR.

Green Indicator On: Indicates that the MSR is ready to read a magnetic stripe. This indicator is turned on when:

1. The 3276, 3278, or 3279 is turned on.

2. The 3276, 3278, or 3279 Test/Normal switch is operated.

3. The MSR data is placed in the 3276, 3278, or 3279 display buffer.

The green indicator is turned off when the yellow or the red indicator is turned on. At this time, the Time symbol is displayed in the Operator Information Area of the screen until turned off by the host.

Yellow Indicator On: Indicates that MSR data is being processed. This indicator is turned on when the magnetic stripe has been read successfully by the MSR hardware. Subsequent read operations are ignored while the yellow indicator is on.

The yellow indicator is turned off when either the red or the green indicator is turned on.

Red Indicator On: Indicates that the MSR data is rejected. The red indicator is turned on during an MSR red operation when:

1. Invalid magnetic-stripe information (for example, invalid character, LRC error, parity error) is detected by the MSR hardware.

2. The keyboard is already locked. The operator should check the symbols in the display's Operator Information Area and take the appropriate action. (See Appendix A.)

3. An unsuccessful read operation is detected. The keyboard is locked.

The red indicator is turned off when the yellow indicator is turned on.

The buzzer on the MSR gives a short tone (one quarter second) when the green indicator turns on and a longer tone (one second) when the red indicator turns on.

Printers

Printers for the 3276 Control Unit Display Station provide a printed copy of information that is displayed at a display station or of information written from the program. Printed data appears in the same alphameric characters and symbols that appear on a display, and printouts can be formatted. Cursor information is ignored by the printer.

The 3230 Model 2, 3262 Model 13, 3268 Models 2 and 2C, 3287 Models 1, 2, 1C, and 2C, the 3289 Models 1 and 2, and 5210 Models G01 and G02 can be attached to the 3276.

Print Line Formatting

Printout operations are specified by a Write command or a Copy command (the 3276 Models 1, 2, 3, and 4, using BSC only), addressed to the printer. The print line format in which the data is to be printed from the buffer can be specified as part of the command in one of three printer formats. These formats define the print line length: 40, 64, or 80 character positions per line. If a format is not specified, the print line length is 132 character positions on all terminal printers. Print line length can be set to values less than 132 character positions by the operator on all terminal printers.

When the 3178, 3276, 3278, or 3279 Print key is used to initiate a printout, or when the 3276 SNA host copy operation described under "Local Copy Functions" is executed, the print line length will be the same as that of the source display. Print line length formats are specified below.

Operation	Command	Addressed Terminal	Format Specification
Host Write (except SCS)	Write	Printer	WCC bits 2 and 3
BSC Host Copy	Copy	Printer	CCC bits 2 and 3
SNA Host Copy	Write	Display	Same as display
Print Key	NA	NA	Same as display

The relationship between the printer buffer and a printout is shown in Figure 2-15.

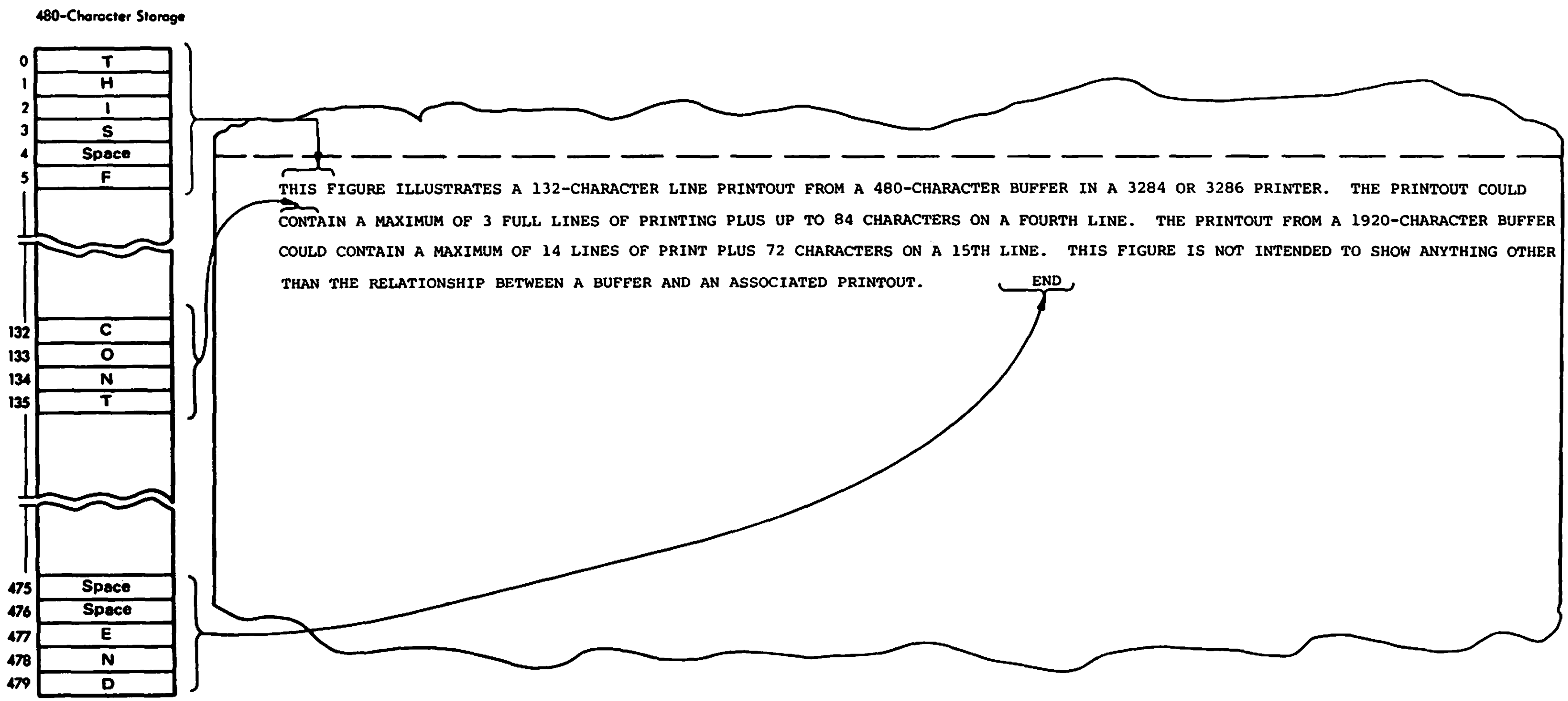


Figure 2-15. Relationship between Buffer Data and Printed Data

Printer Orders

Printer orders are transferred as part of the data stream from the application program. They are stored in the buffer as data.

Programming Note: Devices without the Extended Character Set Adapter (ECSA) feature support 182 characters in the base character set, while devices with the ECSA feature support 191 characters in the base character set. If characters from the 191-character set that are not supported by the 182-character set are directed to a device without the ECSA feature, then that device may interpret certain of these unsupported characters as control codes.

New Line (NL) and End of Message (EM) (All Printers)

The NL order is executed only when encountered during an unformatted printout, that is, a printout that does not have a line-length format specified. When an NL order is encountered in the buffer, the printer performs a new line function. If no NL order is encountered before the printer reaches the end of a line (as determined by the maximum print position), the printer automatically performs a new line function and continues printing. If an NL order is encountered at one character position past the maximum print position, 3230 and 3287 printers will perform two new line functions; the 3262, 3268, 3289, and 5210 printers will perform one new line function.

The NL order is not executed when located in a nondisplay/nonprint field; it is treated as an alphameric character and printed as a space. In addition, the NL order is not executed when encountered during formatted printout. Instead, it is printed by the terminal printers attached to 3276s as a space character.

For buffered printer operation (described under "Buffer Printer Operation") the EM order is executed only when encountered during an unformatted printout. The EM order is not executed when located in a nondisplay/nonprint field; it is treated as an alphameric character and printed as a space. In addition, the EM order is not executed when encountered during a formatted printout. Instead, it is printed by the terminal printers attached to 3276s as a space character.

Forms Feed (FF) (All Printers)

Valid Forms Feed (FF) orders are executed by the terminal printers during either formatted or unformatted printouts. (The FF order is described under "Page Length Control.") When a valid FF order is encountered in the first print position of a line, the print form indexes to a predetermined print line on the next form.

Carriage Return (CR) (All Printers)

When the Carriage Return (CR) order code is found in the data stream, the next print position will be the leftmost character position on the current print line. CR orders are not executed when they occur in nonprint fields and when the printout is formatted (printer format bits in the WCC indicate a line length). In both cases, the CR order is printed as a space character.

Buffered Printer Operations

When a command specifying a printout is received from the system, the contents of the addressed printer are transferred to the 3276 control unit buffer. If the WCC Start Print bit is set to 1, the printout starts after the control-unit-to-printer-buffer transfer is completed.

During a formatted print operation, data characters in the printer buffer are scanned one line at a time before they are printed. A line feed is executed after each line is printed. If a line contains one or more space characters only, a line feed is performed to cause a blank line in the printout. When null characters, attribute characters, or alphameric characters in nonprint field are encountered, they are treated as follows:

- If embedded in a print line, they are printed as spaces.
- If they constitute an entire line, they are ignored and the line feed is not performed; as a result, a blank line does not appear in the printout, and the data is compressed vertically one line.

During an unformatted operation, printout of the buffer data begins at buffer location 0 and continues until the last position of the buffer is printed or until a valid EM character is encountered. Each print line is left-justified. At the end of each printout, a final line feed is executed so that the printer is ready to start the next printout. When the print-terminating EM order appears in the first print position of the print line, a final line feed is not executed because the printer is already positioned at the left margin for the next printout.

Page Length Control Operations

The ability to index forms vertically under program control to a predetermined print line is provided by the Page Length Control function for the terminal printers, except the 3289 printer.

When a valid Forms Feed (FF) order is encountered in the buffer during a printout, the form skips to a predetermined line. Printing begins on the predetermined line; the first print position, the buffer location containing the FF character, is printed as a space character. Printing and skipping continue until the printout is terminated. The printer is "busy" while printing and skipping.

There is no limit on the number of FF orders that can be included in the printer buffer or on the frequency of their occurrence. However, for an FF order to be considered valid and thus initiate skipping, FF characters must be placed in buffer locations corresponding to the first position of a print line in a field designated either print or nonprint. This can be accomplished by placing the FF character (1) in the first character after the WCC in a write, erase/write, or erase/write alternate data stream to the printer (2) after a valid NL or CR order.

When an FF character is placed in the first character position of any print line (for example, in character position 41 in a buffer with a printout format of 40 characters per line specified, or in character position 133 in a buffer for an unformatted printout), the form skips to line 1, position 2.

An FF order in any other position (than the above) in the printer buffer is considered invalid; the skip operation is not executed, and the FF character prints as a space character on all terminal printers, except when the FF order is located in a nonprint field.

During a print operation, if a valid FF order is encountered when the form is located at the predetermined skip stop line (the first print line of each form) of a terminal printer, the terminal printer will not skip a blank form.

Programming Note: Placing the FF order at the end of a print buffer is not recommended. When a valid FF order is placed at the end of a print buffer and is followed by an EM order, the terminal printers will stop printing and skip to line 2 of the next form.

Before beginning Page Length Control operations, forms must be loaded in the printer and aligned to the print line where skipping should stop and printing begin. If the forms are not aligned properly while initially being loaded, all forms will be misaligned.

The two Selector switches must be set to the number corresponding to the total number of print lines from one skip stop line to the next for each Page Length Control application. There can be up to 99 lines between successive skip stop lines. When uniform length forms are used, the setting for the switches is computed by multiplying the forms length in inches by the lines-per-inch setting, 6 or 8 lines per inch for all terminal printers.

Programming Notes:

1. If an NL order and an FF order appear on the last line of a terminal printer's printout, subsequent printing will begin on a new form.

2. The value of the Page Length switch (Selector switch on 3287) when power is turned on or when the Form Feed switch is pressed, is interpreted as the operator-selected MPL value. (For 3289, pressing the Form Feed switch does not initialize the MPL value to the Selector switch value.)

SNA Character String (All Printers)

The SNA Character String (SCS) control codes provide printed page format control. They also can set modes of operation, define data to be used in a unique way, and allow communication between a terminal operator and an application program.

The SCS data stream consists of a sequential string of control and data characters.

Note: *To ensure format integrity, any change in print format control must be followed by the appropriate synchronizing event (CR, NL, FF, etc.).*

SCS Control Codes

SCS control codes are honored by the terminal printers when operating as LU type 1 attached to the 3276. The terminal printers using SCS support can perform a variety of page-editing functions. These are the SCS control codes and their definitions:

Code	EBCDIC (hex)	Name
BS	16	Back Space
BEL	2F	Bell Function
CR	0D	Carriage Return
ENP	14	Enable Presentation
FF	0C	Forms Feed
HT	05	Horizontal Tab
INP	24	Inhibit Presentation
IRS	1E	Interchange-Record Separator
LF	25	Line Feed
NL	15	New Line
SHF	2BC1	Set Horizontal Format
SLD	2BC6	Set Line Density
SVF	2BC2	Set Vertical Format
TRN	35	Transparent
VCS	04XX	Vertical Channel Select
VT	0B	Vertical Tab

Back Space (BS) – a format control that moves the print position horizontally one position to the left. If the print position is at column 1, the function is inoperative. Left margin settings are ignored.

Carriage Return (CR) – a format control that moves the print position horizontally to the left margin on the same line. If the print position is already at the left margin, the function is inoperative.

Enable Presentation (ENP) – a formatting control character used to enable the printing of keyboard input data on the presentation space. This code performs no function on the LU type 1 device, but it is accepted without error response and without affecting format.

Form Feed (FF) – a format control that moves the print position to the top and left margin of the next form. If the maximum presentation line (MPL) value has not been set and there is no default value, the MPL defaults to 1, and the print position moves to the left margin of the next line.

Horizontal Tab (HT) – a format control that moves the print position horizontally to the next tab stop setting. Horizontal tab stop values are set by using the Set Horizontal Format (SHF) function. If there are no horizontal tab stops set to the right of the current print position, the horizontal tab function results in a space.

Programming Note: Horizontal tab placed after the MPP will cause a space in the first print position on the next line.

Inhibit Presentation (INP) – a format control character used to inhibit the printing of keyboard input data. This code performs no function on the LU type 1 device, but it is accepted without error response and without affecting format.

Inter-Record Separator (IRS) – a separator character, normally used on the LU-SSCP session. If received on an LU-LU session, the IRS defaults to a New Line (NL) function.

Line Feed (LF) – a format control that moves the print position vertically down to the next line.

New Line (New Line) – a format control that moves the print position to the left margin and vertically down to the next line. NL is functionally equivalent to CR followed by LF.

Set Horizontal Format (SHF) – a data-defining control used to set the horizontal format controls. These include left and right margins and horizontal tab stops. A 1-byte binary count follows the SHF code that indicates the number of bytes to the end of the SHF string, including the count byte. The first 3 bytes following the count byte define the maximum presentation position (MPP), the left margin (LM), and the right margin (RM), respectively. Tab stop settings follow the right margin position. All values are expressed as 1-byte binary numbers.

The minimum SHF sequence is 1-byte length, which sets the horizontal format controls to their default conditions. The SHF sequence is:

(SHF)(cnt)(MPP)(LM)(RM)(T1)(T2). . . (Tn)

This value is used to define a line length less than, or equal to, the maximum print position. The MPP default value is the maximum print position (132) or the value set up by the printer operator.

Programming Note: If the MPP is set to a value greater than the physical page width, data may be lost (for example, printing on the platen or print head jams at the right margin).

LM specifies the column value of the leftmost print position. The LM also serves as the first horizontal tab stop. Valid LM values are less than, or equal to, the MPP. The LM default values is 1.

RM is not used in printing operations.

T1 . . . (Tn) are horizontal tab stop settings. The tab stops do not have to be in order. Valid tab stop values are equal to or less than MPP.

Set Line Density (SLD) – specifies the distance to be moved for single-line vertical spacing, as in LF or NL. This function changes values that were previously set during printer initialization or by pressing the Change LPI switch (6LPI/8LPI switch for 3289) on the operator panel. The SLD code (X'2BC6') is followed by a one-byte count (CNT) and a one-byte line density parameter (LPI) as follows:

2BC6 CNT LPI

CNT = the number of bytes following the SLD code (01 or 02), including the count itself. If the value is not 01 or 02, an "invalid parameter" response is generated. Printing terminates immediately.

LPI = the line density parameter that specifies the distance (measured in lines per inch) to be moved for single-line vertical spacing (one inch equals 25.4 mm).

Acceptable values are:

2BC60218	=	3 lpi
2BC60212	=	4 lpi
2BC6020C	=	6 lpi
2BC60209	=	8 lpi
2BC60200	=	default to 6 lpi
2BC601	=	default to 6 lpi (3287, 3289), or to operator panel setting of 6 or 8 lpi (3230, 3262, 3268, 5210)

Programming Note: If the SLD is changed without a corresponding change in the MPL (and vice versa), printing may occur on the form fold, or the MPL may be defaulted.

Density values not implemented are rejected with a negative response of X'1005' parameter error.

Set Vertical Format (SVF) – sets vertical format controls, including the maximum presentation line (MPL), top margin (TM), bottom margin (BM), and vertical tab stops. A 1-byte count field follows the SVF character to indicate the number of bytes, including the count byte, in the SVF string.

The first three values following the count in an SVF string are the maximum presentation line, the top margin, and the bottom margin, in that order. A zero for any of these values results in the function assuming the default value. Vertical tab stop values follow the bottom margin. All values are expressed as 1-byte binary numbers.

The SVF sequence is:

(SVF)(cnt)(MPL)(TM)(RM)(T1)(T2). . . (Tn)

MPL defines the page depth. All values between 0 and 102 (3287), 0 and 127 (3230/3262/3268/3289), and 0 and 255 (5210) are valid. A page depth defined by the SVF takes precedence over the device default value. The MPL default value for the 3287 is 1; the MPL default value for the 3230, 3262, 3268, and 3289 is 1 or the contents of the Selector switch. If the Selector switch is set to 00 and power is turned on, the MPL defaults to 1; if the Selector switch is set to 00 and the Reset switch is pressed, the MPL remains unchanged. For the 5210, the MPL defaults to 66.

Programming Note: If the MPL is set to a value greater than the physical page length, printing may occur on the form fold.

TM specifies the line value used as the top representation line on the page. The top margin is also the first vertical tab stop. Valid TMs are equal to, or less than, MPL.

The default TM value is 1.

Programming Note: After the TM is initialized, the TM should not be changed, because a TM change requires operator intervention to align the physical page. The printer cannot detect physical line 1; therefore, it is assumed the operator has aligned physical line 1 to the printer's logical line 1. If a printer must be used in an intermixed SCS/non-SCS environment, the operator should always set the physical page line 1 at the first line to be printed and should always set the TM to a value of 1.

TM may be changed on the 5210. When forms are initially loaded, they are aligned to the form fold. The 5210 automatically indexes to the TM upon receipt of the first non-25 control after having executed a form feed.

BM specifies the line value that, if exceeded, causes an automatic skip to a new page. BM must be greater than, or equal to, TM, and less than, or equal to, the MPL. The default BM value is the MPL value.

Transparent (TRN) – a data-definition character, which provides for the transmission of data in transparent mode. A 1-byte binary value follows the TRN code which specifies the number of bytes of transparent data to follow. The length does not include the length byte. Transparent data is user-defined and is not scanned for SCS control codes. As each data byte is interpreted, the print mechanism moves one character position. Valid graphics are printed. Invalid graphics are printed as hyphens (-).

Vertical Channel Select (VCS) – a device control code that allows selection of one of 12 vertical channels to control vertical format. The first character of the code is the select code, followed by a function value which selects the appropriate channel.

Vertical Tab (VT) – a format control that moves the print position vertically down to the next vertical tab stop setting. Vertical tab stops are set by using the Set Vertical Format (SVF) function. If there are no vertical tab stops below the current print position, the vertical tab function results in an LF function.

Graphic Escape (GE) – a character selection code that immediately precedes a codepoint. It is used to indicate that the character to be displayed or printed is to be selected from the character set stored in ROS 1. (The base character set for the machine is stored in ROS 0.)

Note: The 5210 does not support GE and prints a hyphen in its place. The codepoint interpreted as an SCS control or graphic.

Program Attention (PA) and Cancel Print Switches

The PA1/PA2 and Cancel Print switches are provided when SCS is installed on terminal printers attached to the 3276. (SCS is a specify feature for the 3287, and a standard feature for other terminal printers.) These switches allow the operator to communicate with the host system in SCS mode, and are used with the Hold Print/Enable Print switch. Operator- or host-initiated operations can be performed.

CANCEL PRINT. The Cancel Print switch causes the printer to terminate the current print operation. Portions of a chain which have not been passed to the printer are purged by the control unit.

Cancel print is meaningful when the printer is printing SCS data or waiting for the next data in a chaining operation. If the Cancel Print switch is pressed and the printer is not processing SCS print data, an invalid switch operation is indicated at the printer. The control unit is not made aware of this condition.

PA1/PA2. The PA1/PA2 switch causes an attention to be sent to the control unit. The status indicator on the printer will indicate acceptance of the code, and printing is resumed if it was in progress prior to the PA switch sequence. The two-digit code is then cleared from the status indicator.

The operator may then initiate another PA switch selection if the previous selection is overwritten. PA switch information is not stacked within the subsystem.

The control unit of an SCS printer transmits the PA switch codes to the PLU as FM data, as follows (note that there is a blank between APAK and the PA switch code digits):

PA Switch	Text String Transmitted
1	APAK01
2	APAK02

If the printer is not in SCS mode (for example, performing a local copy operation), an invalid switch operation is indicated, and no PA switch sequence can be initiated.

Print Format Control

The format of the printed data is determined by the following parameters:

Maximum Print Position-MPP
Maximum Page Length-MPL
Lines per Inch-LPI
Single/Double-space
Mono/Dual Case

The terminal printers, except 3287, allow the operator to change the machine default values of these parameters. They can be set by the host or controller in SCS and non-SCS print modes. See the terminal printer's Component Description for details.

When the 3287 is operating in SCS mode, the operatore can change the machine default of only Single/Double-space. The default values are MPP = 132, MPL = 1, LPI = 6, and Mono/Dual Case = Dual.

Local Copy Function

In addition to processing the BSC Copy command in remote configurations, the 3276 Models 1, 2, 3, and 4 provide a local copy function which allows direct data transfer from a display station to a printer(s) attached to the same 3276. The local copy function is directed by a print-control matrix. The print-control matrix for the 3276 is called the *default matrix.*

The local copy function can be operator- or host-initiated. For operator-initiated copy, the Print key on a keyboard attached to a 3178, 3276, 3278, or 3279 may be used by the operator to initiate a local copy request. The local copy request is serviced by a printer selected under control of the print control matrix.

In SNA models, host-initiated local copy requests are initiated by issuing a write-type command with the WCC Print Bit set to 1; that is, systems using 3276 Models 11, 12, 13, and 14. Printer selection and servicing of the local copy request proceed in much the same way as for operator-initiated local copy requests.

Do not attempt to copy graphics dependent on more than one character position for their presentation. If the graphic data is not completely contained within one character position, the printout will be inaccurate because of the differing block matrix sizes and dot densities between display and printer. Also, attempting to copy to a printer not featured for Programmed Symbol operation, or not containing a matching symbol set (with the one in the display station), results in default to the I/O interface character set installed in the printer.

3276 Default Matrix

At the time the control unit is powered on, a reset is issued to each attached terminal. As each terminal responds positively, it is posted in the default matrix. The matrix identifies each terminal in ascending order, by port. For example:

Port	0	1	2	3	4	5	6	7
Terminal	D	D	P	P	D	D	P	P
Assignment	02	02	X	X	06	06	X	X

Note: *X = not applicable*

Displays (D) are assigned the first printer (P) occurring at a higher port number. In this example, display terminals at ports 0 and 1 will be assigned the printer on port 2. Display terminals on ports 4 and 5 are assigned the printer on port 6.

If power is off at a terminal when the control unit is powered on, nothing is posted in the matrix for that terminal. Therefore, the control unit assumes that the device at that port is a display. Power off at ports 1, 4, or 5 does not alter the definition of the matrix in this example. Power off at ports 2 and 3 (prints) results in display terminals at ports 0 and 1 being assigned to the printer at port 6.

If a terminal is powered off after it has been posted in the matrix, the terminal is considered "not ready." The matrix is not altered. Thus, if the printer at port 2 is powered off after being posted in the print matrix, a not-ready condition would be signaled if a local copy operation is attempted by the displays at ports 0 or 1. However, by switching power on the 3276 off and on again, printer 2 is removed from the default matrix, which then appears:

Port	0	1	2	3	4	5	6	7
Terminal	D	D	X	P	D	D	P	P
Assignment	03	03	X	X	06	06	X	X

If a terminal is initially powered off, and then powers on some time after the control unit has been powered on, the control unit is notified, and the matrix is updated. For example, if the printer attached to port 6 was not powered on, the default matrix appears as:

Port	0	1	2	3	4	5	6	7
Terminal	D	D	P	P	D	D	X	P
Assignment	02	02	X	X	07	07	X	X

Applying power to a printer at port 6 at a later time will change the assignments for displays 4 and 5 to printer 6, as in the previous examples.

As configured in the first example, the printers attached to ports 3 and 7 will not be used for local copy from display stations. They are available for uninterrupted use by the host for direct print and BSC Copy command operations. The printers on ports 2 and 6 may also be used by the host for direct print and BSC Copy command operations. In this case, such operations may have to wait or be interrupted by execution of local copy requests.

In 3271-compatible operations, host and local copy print requests are handled on a first-in, first-out basis; however, when using SNA protocol, local copy requests may be executed only when the host printer session is "between brackets."

Printer Selection. The IDENT key on the keyboard of the 3276, or on the keyboard of the attached 3178/3278/3279 display station, may be used to change the printer ID assigned by the default matrix as described under "IDENT Key" in Chapter 2. For example, by using the ALT key, and keying IDENT 03 at the display attached to port 1, the default matrix becomes:

Port	0	1	2	3	4	5	6	7
Terminal	D	D	P	P	D	D	P	P
Assignment	02	03	X	X	06	06	X	X

By switching 3276 power off and on again, the original default matrix is restored.

3276 Local Copy Operation

Operator-Initiated Copy. The operator may initiate a local copy operation by pressing the Print key on the display keyboard. The 3276 will then attempt to execute the local copy function on the printer with ID shown in the "connect" indicator in the Operator Information Area.

The Print key is active in an SNA environment under the following conditions:

1. No session has been established (prior to receipt of ACTLU, or after receipt of DACTLU).

2. Session owner is "Unowned."

3. The terminal is in Test mode, and the keyboard is unlocked.

4. Session owner is the SSCP, and the keyboard is unlocked.

5. Session owner is the PLU, the keyboard is unlocked, and the SLU is not in receive state.

The Print key is active in a BSC environment whenever the Time symbol is not displayed.

If the printer is busy doing local copy operations for other displays, the Input Inhibited Printer Busy (short term) symbol is displayed. In SNA, if the printer is busy because it is "in" brackets with a host application, or in BSC during a host write-type operation, the Printer Very Busy (long term) symbol is displayed. In either case, the request is queued, and the keyboard is locked until the copy can be performed or the operator cancels the print request. The RESET key has no effect while a print request is on the queue; however, the operator can cancel the local copy request by pressing the DEV CNCL key. This turns off the Input Inhibited symbol, unlocks the keyboard, and dequeues the print request. The operator is then free to perform another task.

In BSC, an operator-initiated local copy operation to a printer is not executed if the printer has status pending from a previous host-directed print operation. General or Specific polling will clear the printer status and free the printer for local copy usage.

If the printer is not functional because of an intervention-required or permanent-error condition, then the Input Inhibited Printer Not Working symbol is displayed and the keyboard is locked. The operator must depress the DEV CNCL key to continue. This action turns off the Input Inhibited symbol and unlocks the keyboard. The print request is not queued. The operator may then choose an alternative action. When the Printer Not Working symbol has been turned on as a result of an operator-initiated copy request, this symbol, and an associated Printer Failure symbol, if displayed, will be turned off by receipt of any outbound FM data request.

If the operator attempts to print again, and the selected printer is still not operational, the Input Inhibited Printer Not Working symbol reappears. Some operator action, for example, loading paper in the printer, may be required to clear a not-functional condition. If no connection indicator is displayed and the Print key is depressed, the Input Inhibited Operator Unauthorized symbol is displayed and the keyboard is locked. The symbol remains on until the operator presses the RESET key.

When a valid printer is selected, and the display-to-printer buffer transfer begins, the display keyboard is locked and the Input Inhibited Time symbol is displayed. This symbol remains on and the keyboard remains locked until the buffer transfer is completed successfully.

If the printer stops during a local copy operation (out of paper, paper jam, etc. - a data check on the printer does not fall in this category), the Printer Failure symbol replaces the Printer Printing symbol and the print is terminated. The keyboard locks and the Printer Not Working symbol is also displayed, calling the operator's attention to the failure. The Printer Failure symbol specifies the failing printer. In this state, the DEV CNCL key will remove both symbols from the display.

Operator-Unauthorized Condition. If the display cannot perform the copy operation because the most-available printer does not have a large enough buffer, the operator will be alerted by an inhibit condition with the Operator Unauthorized symbol. This may occur, for example, when the operator attempts to copy to a 1920-character buffer printer from a 3440-character display.

Host Interference with Operator Copy (SNA). Once the display operator has initiated a local copy operation, any outbound FM data request will be rejected with a busy indication, X'082D', during the time that the operator request is queued or the buffer is being transferred, and an outbound FM data request is received for the display. Once the buffer transfer has been completed the display is free to receive outbound FM data requests. If a negative response has been sent because of this condition, an LUSTAT of X'0001D000' will be sent at the completion of the buffer transfer to notify the host that the busy condition no longer exists. FM data may be written into the display buffer as soon as the buffer transfer is complete.

If the host is in session with the printer, the local copy operation will not change the selected size of the printer buffer as set by the host session.

Host-Initiated Local Copy Using SNA/SDLC

The host application program may initiate a local copy function in an SNA environment by sending to the display station a write-type command with the Start Print bit in the WCC turned on. (The copy function under SNA ignores WCC bits 2 and 3.) The control unit performs the local copy function as required, using the print class or printer assigned to the display and displayed in the Operator Information Area. When a write-type command is sent to the display with the Start Print bit on, the display first interprets the orders and data in the write data stream and updates the display buffer. During this time, the Input Inhibited Time symbol is displayed. Once the buffer write is completed, the control unit attempts to use the printer(s) it assigned to the display. The Time symbol remains on while the copy operation takes place. Once the buffer transfer is completed, the Printer Printing symbol replaces the Printer Assignment symbol. The Printer Printing symbol always shows the specific terminal address of the printer actually doing the print operation.

The keyboard remains locked, regardless of keyboard Restore, until the print operation is completed. When the print operation is completed, the keyboard unlocks according to the keyboard Restore in the WCC. The Time symbol is removed, and the Assignment symbol replaces the Printer Printing symbol.

To perform the host-initiated local copy described above, the host program must send a write-type command with the Start Print bit turned on in the WCC as an RQD chain or an RQE, CD, EB chain. Otherwise, the synchronization may be lost or the request rejected with response X'0843'.

Printer Busy Condition. If, after performing the display buffer update operation, the control unit finds that the connected printer or all printers in the selected print class are busy with other local copy operations, the print request will be queued.

If the 3276 LU repeats a previous host-initiated copy request, and is waiting for availability of a printer, further print requests will not be queued but are rejected. On displays attached to a 3276, the keyboard remains locked and the Input Inhibited Time symbol is replaced by the Input Inhibited Printer Busy (short term) symbol. The operator may cancel the request by operating the DEV CNCL key. This will dequeue the print request and replace the Printer Busy symbol with the Time symbol. A negative response X'0807', printer busy, is sent to the host. This allows the host to take an alternative action.

Similarly, on a 3276, if the selected printer or all printers in the selected class are found to be "in" brackets with the PLU, the copy operation is refused. After the write operation is complete, the control unit will respond negatively to the print request with X'0807', printer busy.

Once a print request has been refused with "printer busy," the SLU sends an LUSTAT of '0001B000' to the PLU when a printer becomes available. (Only one LUSTAT is returned per SLU, regardless of the number of times the PLU may have requested a local print operation.)

The PLU may choose not to wait for the LUSTAT but to continue with other display work. Even though the SLU is taken out of the ERP.1 state by the PLU, it is still bound to send in the LUSTAT at the first opportunity when the printer becomes available.

After sending the LUSTAT '0001B000', if obligated, the 3276 holds the printer until:

- It is released because a valid FM data request is received which does not specify start print.
- It is released because of a Clear request; the session is unbound.
- The copy is completed after the PLU sends a write type command with the Start Print bit turned on in the WCC.
- The copy fails and a negative response is returned to the host because of one of the following:
  - A permanent error in the printer is detected during printing.
  - The display operator turns off the security keylock.
  - The display operator turns off display power.
  - Ownership of the display is changed to other than the PLU.
  - A permanent error in the display is detected.
  - A temporary error in a printer or display is detected.
  - Intervention Required condition in a printer was detected.

Printer Not Assigned Condition. If a printer is not assigned to the SLU at the time the printer is selected, the control unit responds to the write-type command with negative response (0801) "printer not assigned."

"Printer not assigned" will also be sent to the PLU when a copy request is made, and the selected printer cannot perform the copy because of a feature mismatch between the display device and the printer.

In all cases mentioned above, once the negative response has been sent to the host, the 3276 enters the receive state.

Printer Not Functional Condition. If the most-available printer is not functional at the time the printer is selected, the Printer Not Working symbol replaces the Time symbol. The Write command is responded to with negative response (083E) intervention required, or negative response (082F) permanent printer error. The display LU goes into the ERP.1 state as defined for printer busy. When intervention-required is returned, recovery may require operator action, e.g., loading forms. When the intervention-required condition has been cleared, the control unit will generate an LUSTAT 0001B000 to the PLU in session with the display. After receiving the LUSTAT, the PLU may reinitiate the copy request by sending a Write command with the Start Print bit in the WCC and with no data.

If the operator operates the DEV CNCL key while the Printer Not Working symbol is being displayed, the Printer Not Working symbol is replaced by the Time symbol.

If the PLU transmits any FM data request to the display and the Printer Not Working symbol has not been cleared, the FM data request will remove the Printer Not Working symbol and an associated Printer Failure symbol, if displayed, and may take the SLU out of the ERP.1 state.

No LUSTAT is required when 082F (permanent error) is sent as a response to the Write command.

If the printer malfunctions during the print operation, both the Printer Not Working and the Printer Failure symbols are displayed. The print operation terminates, and the Write command is responded to with negative response (082E) or negative response (082F). The keyboard remains locked and the system waits for some recovery action as defined above.

Note that any FM data requests from the PLU will clear a Printer Not Working symbol. This requires careful planning by an installation in the use of host- and operator-initiated printing.

Local Copy Performed without SNA Protocol

In a BSC environment, host-initiated local copy is initiated through use of the Copy command (remote only). The description of operator indicators under "Host-Initiated Local Copy Using SNA/SDLC" does not apply to the Copy command. Operator-initiated copy in a non-SNA subsystem is the same as defined under "Operator-Initiated Copy."

When a printer or class of printers is in shared mode, the contention between host and local copy use of the printer is resolved according to the following procedure:

1. If, during processing of an operator-initiated copy operation, the host sends a selection addressing sequence to the printer, the control unit will respond with an RVI and will set Intervention Required. When the local copy queue no longer exists and the printer becomes available, Device End (DE) is sent in response to a poll (remote) or as asynchronous sense/status (local) to signal that the printer is then available.

2. To provide security in systems that operate in a non-SNA environment, the printer buffer is cleared after successful operator-initiated local copy operations are completed. A read-buffer or read-modified operation will not return the contents of a printer buffer just used in a local copy operation by another display operator.

3. A host program may use several messages to load a buffer with data to be printed or for temporary data storage. Once the program initiates loading of the buffer, operator-initiated local copy operations cannot be performed until print operation is completed, or until there is a permanent error. An operator-initiated print request via the Print key during this period is queued, and the Device Very Busy symbol is displayed. The host system should issue an Erase/Write command with the Start Print bit "on" to release the printer for local print operations.

4. The host application program can use the printer when there are no operator-initiated local copy requests outstanding. If it is required that the host have sole ownership of the printer for data integrity or performance considerations, the printer should be designated as a system mode printer in the printer authorization matrix.

5. If a host transmission to the display is received while an operator-initiated copy request is queued, the host transmission will be accepted and written to the display. No change will be made to the status of the operator-initiated copy. If the copy is queued and buffer transfer has not taken place, the new screen will be copied. If buffer transfer has started before arrival of the host transmission to the display, the transfer will be completed before writing to the display. In this case the old screen will be copied.

6. Each time the local copy queue is completed, a Device End will be transmitted to the CPU by the 3276, thereby signaling that the printer is available. The printer buffer is set to the default size after each copy queue is completed.

Mono/Dual Case Control

When power is applied, the 3287 is activated to print mono-case; other terminal printers are automatically activated to print the dual-case character set.

In dual-case operation, the alphabetic character codes sent by the host determine whether uppercase or lowercase characters are printed, provided that the print belt has the dual-case character set. In mono-case operation, the lowercase alphabetic character codes print equivalent uppercase characters.

The Change Case switch can be pressed to change the print case on the terminal printers. However, when operating with LU1 printers in SNA, the data character codes and the print belt character set determine whether mono- or dual-case characters are printed, regardless of the Change Case switch setting.

In a BSC environment, when using the Copy command to transfer data from a display to a printer, the setting of the Change Case switch on the "from" display determines mono- or dual-case in the "to" printer. When the Copy command transfers data from a display or a printer to a display, the Change Case switch on the "to" display determines whether mono- or dual-case is displayed.

Format Control during Shared Printer Operations

When shared printers respond to uncoordinated print requests, control of the horizontal and vertical print position format is governed by the operating mode(s) and the format selected.

In BSC printer operations, sharing occurs on a buffer load basis, between local copy requests and host-initiated printer output, by means of write-type or Copy commands. When using SNA protocol, local copy requests for display buffer data originating from an LU2 session may share a printer with either LU3 or LU1 host output. Sharing of LU2 and LU3 devices is comparable to BSC operation.

In BSC, and in SNA LU2 printer operations when performing local copy, the entire buffer content, including nulls, attribute, and buffer control characters of a "from" display or a "from" printer (non-SNA only), can be transferred to a printer buffer.

During formatted print operations, the data is scanned a line at a time. If a line contains one or more data characters (including Space, NL, EM, and CR) in a display/print field, the line is printed and a line feed is performed. To produce a blank line, at least one Space character must be present.

A valid FF character is executed regardless of the attribute of the field, except for the 3289-1 and -2. The 3289-1 and -2 do not execute or print any characters in a nonprint field, including the FF character. If the FF character is invalid, it is not executed and prints as a blank in a field that is not defined as nondisplay/nonprint.

If a line contains only nulls, attribute characters, or alphameric characters (including Space, NL, EM, FF, or CR) in a nonprint/nondisplay field, no line is printed and no line feed is performed. A screen facsimile can only be obtained by inserting at least one space character in the blank lines.

In BSC, and in SNA LU3 printer operations when directly printing from the host, the identical procedure is followed as described above once data has been loaded in the buffer and the print operation is started. Thus, when a print operation is completed, a line feed will have been automatically performed after printing of the last line (blank or not). Therefore, the next buffer load of data, regardless of the source, starts printing on the next line, ignores the previous horizontal position, and is contiguous with the previous output except for blank lines as provided in either or both buffer data.

A valid FF control character in the data at either the beginning or end of a form (one or more buffer loads) ensures synchronization of the forms with the data. Interleaving of a local copy operation within a host output print operation using VFC will usually cause local copy to be printed on part of a completed form or cause at least one form to be misprinted. This may best be avoided by configuring the printer in system mode, thus excluding its use for local copy.

In BSC unformatted print operations, the completed print operation terminates at a new line position. Thus, the next print operation is also contiguous with the previous output except for possible blank lines as specified in the data. (SNA LU type 1 devices do not perform unformatted printouts.)

When operating as an SNA LU type 1 device, an automatic LF, NL, etc., is not sent at the end of a bracket or a session. Therefore, the print position may be one position to the right of the last printed character. The first printed line resulting from a local copy operation performed with an LU2 device is printed on the line that is currently available. Overprinting may occur if the first line is not specified as a blank line. When the local copy operation is completed, the LU1 session resumes with a new bracket at the horizontal print established by the preceding LU1 bracket.

Error Conditions

Four error conditions may be encountered at the printers. In each of the following cases, when an error is detected, the program is notified.

Not Ready. A printer is defined as not ready when it is out of paper, its cover is open, or it is mechanically disabled (unable to advance to its proper position). For 15 seconds, the mechanism will automatically attempt to recover. If the recovery attempt is successful, the printer will return to the ready condition. If the recovery attempt is not successful after 15 seconds, the printer will become not ready, as indicated by Intervention Required (IR) status.

A 3262 displays an error code in the status indicator. The operator may be able to clear the error condition and continue printing.

If a printer (not the 3289) is not ready at the start of a printout, or if it becomes not ready during a printout operation, the print operation terminates. Error status is sent to the channel once when the condition occurs during a printout and, then, again each time a printout is initiated.

Parity Error. If a parity error is detected on a character about to be printed, an error graphic (prx 10T,L) (3287) is printed in place of the character with incorrect parity. The buffer continues printing until all printable characters have been printed. The printer prints a graphic X. The isolated X character (specify feature on the 3287) serves to indicate the detection of the parity error.

An x (an X overprinted with an O) prints in place of an incorrect character on all other terminal printers. An x also prints in the left margin of the next line.

Command-Chaining. In local operations, if any command is chained to a command that initiates a print operation, an error condition occurs: no printout is performed, the command is aborted, and the system channel is notified of the error. In remote operations, if command chaining is attempted, error status is sent to the system channel but the printout is completed.

Chapter 3. Remote Operations - BSC

Introduction

The 3276 Control Unit Display Station Models 1, 2, 3, and 4 use Binary Synchronous Communication (BSC) mode of operation and can communicate with the program via an IBM 2701, 2703, 3704, 3705, or an equivalent Integrated Communications Adapter (hereafter called *TCU*) and appropriate data sets as specified for the control unit.

The 3276 uses BSC procedures over duplex or half-duplex facilities (nonswitched or privately owned); these communications use the Multipoint Data Link mode of operation only.

Code Structures

The 3276 can operate with one or two code structures: EBCDIC (Extended Binary-Coded Decimal Interchange Code) or ASCII (American National Standard Code for Information Interchange). The choice of code depends on the application. However, for system compatibility, the same code must be chosen for all units on a particular communications line.

Channel Program Concepts

In remote configurations, the TCU becomes the intermediary between the 3276 and the channel program. As such, the TCU, not the 3276, executes channel commands and initiates I/O interrupts. At the start of each I/O operation involving the TCU, the Start I/O instruction addresses the TCU and a communications line attached to that TCU; it does not address an individual remote control unit on that line. Subsequent CCWs in the channel program initiate TCU operations; they specify TCU commands, not 3276 commands.

Selection of control unit and all subsequent command operations are specified by character sequences in TCU Write CCW data streams. Write CCW data to the TCU communications line selected by Start I/O can contain (1) address bytes to select a control unit on that line, (2) the code of a command (such as Erase/Write or Write) to initiate a control unit operation, or (3) orders and/or display/print data for the control unit buffer. In addition, this write data will contain the appropriate data-link control characters. Thus, all characters sent by the TCU to a 3276, with the exception of SYN, pad, and BCC characters, originate from the data stream of a Write CCW addressed to the TCU.

Programming Note: All Write commands should be set for CCW chaining to a Read command when a response is expected. This prevents a loss of data received by the TCU in response to Write command operations. An exception to this requirement is when the Write command is used to issue EOT to the 3276.

Text Blocking

The 3276 performs inbound text blocking. Each block of data can contain a maximum of 256 text characters. Of that total, each block contains the STX and ETB (or ETX) data link control characters. Two address bytes (CU poll address and device address) precede the read heading in the first block only and are included in the 256 character total. The last block of a message is terminated with ETX, which is also included in the 256 character total.

Programming Note: If the automatic polling facility (Auto Poll) is used by the TCU, the Auto Poll index byte will add 1 byte to the text block created by the 3276.

Block check characters (BCC) are transmitted as the last characters of a data stream. (See "Redundancy Checking.") BCC is not counted as text because it follows the ETX and ETB data link characters. Upon successful comparison of the received BCC with the accumulated BCC, the program should respond with ACK to read the next block of text; each subsequent block is preceded by STX to initiate BCC accumulation by the TCU.

Text blocking does not disjoin the 3-byte SBA order sequence (SBA code and 2-byte field address) generated during execution of a Read Modified command. Therefore, the last characters of a block ending with an SBA sequence would be . . . SBA, Address, Address, ETB (or ETX).

Related Publications

Readers who are unfamiliar with the binary synchronous method of communications should review the following publications, as applicable:

- *General Information - Binary Synchronous Communications,* GA27-3004
- *IBM 2701 Data Adapter Unit Component Description,* GA22-6864 (especially the section that describes the Synchronous Data Adapter - Type II)
- *IBM 2703 Transmission Control Component Description,* A27-2703 (especially the section on BSC capabilities)
- *Introduction to the IBM 3704 and 3705 Communications Controller,* GA27-3051

Multipoint (Nonswitched Line) Data Link Control

The 3276 can operate on a nonswitched communications line with multiple stations. Time-sharing of the line is accomplished by interleaving transmissions between the TCU and all units on the line. A 3276 operates multidropped on the same line with other properly featured units.

The TCU is the control station of the multipoint, centralized network. All units attached by communications lines to the TCU are called tributary stations. The control station is the focal point of the network and maintains, under program control, an orderly flow of network traffic by initiating all data transfers. The control station is either the transmitter or receiver of every communication.

3276 Modes of Operation

In the multipoint environment, the 3276 is always in one of three modes of operation: control mode, text mode, transparent-monitor mode. The 3276 does not provide transparent-mode.

Control Mode

The 3276 enters control mode whenever it transmits or receives a valid EOT sequence. While in control mode, the unselected 3270 CU monitors the communications line for the following:

1. A valid selection or poll addressing sequence, by which the 3276 will become selected for entry into text mode.

2. A DLE-STX sequence, placing the 3276 in transparent-monitor mode.

Text Mode

Once a 3276 is successfully selected, it enters text mode. In text mode, the 3276 is either a master station or a slave station, as is the TCU. This status depends on the operation being performed. The station that is transmitting a message is called the MASTER STATION, whereas the station that is receiving and acknowledging the message is called the SLAVE STATION.

The 3276 becomes the master station (and the TCU the slave station) once it sends STX to the TCU while executing a Read command or a poll operation. As the master station, it can (1) transmit text messages and (2) transmit ENQ to request a reply or retransmission from the TCU. After transmission of the message is completed, the 3276 returns to control mode.

The 3276 becomes the slave station (and the TCU the master station) when executing a write-type command. As a slave station, it responds appropriately to master-station (TCU) transmissions.

Transparent Monitor Mode

The 3276 does not operate in transparent mode, but can operate on a communications line with other types of terminals that can operate in transparent mode.

Transparent-monitor mode is provided with an EBCDIC 3276. It permits the transmission of data in any of the 256 possible EBCDIC bit patterns between the TCU and another unit on the same communications line with the 3276. This data may be independent of the selected transmission code (EBCDIC). Examples of such format-independent data are packed-decimal data, programs (both source and object), core images, and other binary data. Thus, link control characters within this data will not inadvertently initiate a 3276 operation.

When an EBCDIC 3276 decodes a DLE STX sequence while in control mode, it enters transparent-monitor mode. While in this mode, the 3276 disregards all data configurations that may appear on the communications line except for (1) a transparent text sync sequence (DLE SYN) or (2) a transparent text-terminating sequence (DLE ITB, DLE ETX, DLE ETB, or DLE ENQ). The 3276 leaves transparent-monitor mode and returns to control mode (1) if a transparent text sync sequence is not received within any 3-second period or (2) if a transparent text-terminating sequence is decoded.

Redundancy Checking

A redundancy check is performed on the following communications line data:

1. 3276 command-sequence characters (including the write data of a Write, Erase/Write, or Erase/Write Alternate command).

2. Data transmitted to the TCU in response to a read-type command or to a polling sequence.

A block check character (BCC) is accumulated for each block of data at both the TCU and the 3276. If EBCDIC code is used, a 2-byte BCC is generated (cyclic redundancy check accumulation); if ASCII code is used, a 1-byte BCC is generated (longitudinal redundancy check accumulation).

BCC accumulation is initiated by, but does not include, the first STX or SOH framing character. All characters following this STX or SOH, up to and including the end-of-block character (ETB or ETX), are part of the accumulation. Following the ETB or ETX character, the transmitting unit transmits its BCC character(s). The receiving unit then compares this character(s) with the BCC it has accumulated. If the redundancy accumulations are different, a transmission error has occurred.

When the 3276 is the receiving unit and detects a BCC error, it responds to the transmission by sending NAK to the TCU. When the TCU is the receiving unit, it will set Unit Check in the ending status for the TCU command being executed when the BCC error was detected; also, it will set Data Check in the sense byte.

Note: *BCC characters are removed from the data stream when received for comparison by the TCU or by the 3276; they are not stored in main storage or in the 3276 buffer.*

In both EBCDIC and ASCII, transmission formats (data link controls) are rigidly screened so that communication is orderly and accurate. Improper transmissions are ignored or rejected to avoid the acceptance of faulty messages. Received or transmitted data blocks are counted odd-even-odd-even, etc., by both the transmitter and receiver (by means of ACK 0's and ACK 1's), and their counts must agree at each block-check point.

Data-Link Control Characters

Two types of characters are transmitted between the TCU and the 3276: CU data-link control characters and 3276 message data. Data-link control characters are used for such purposes as message framing, acknowledgment that received message data was valid or invalid, and identification of the start- or end-of-text transmission. Data link control characters are used (singly or in sequences) by the TCU (under program control) and by the 3276 to establish and control all data link operations in an orderly fashion. The 3276 message data consists of all address, command, order, and display/print characters sent to the 3276 and of all buffer data, AID bytes, and status/sense bytes read from the 3276. Data-link control characters are described individually in the following paragraphs and are described with 3276 message data later in this section (under "Operational Sequences").

The data-link control characters, with their EBCDIC or ASCII codes, are as follows:

Data-Link Control Character	EBCDIC (hex)	ASCII (hex)
ACK 0 (two bytes)	1070	1030
ACK 1 (two bytes)	1061	1031
DLE	10	10
ENQ	2D	05
EOT	37	04
ESC	27	1B
ETB	26	17
ETX	03	03
ITB	1F	1F
NAK	3D	15
RVI (two bytes)	107C	103C
SOH	01	01
STX	02	02
SYN	32	16
TTD	022D	0205
WACK	106B	103B

All control characters transmitted by the TCU (except pad and SYN) are issued by the channel program as part of a TCU Write CCW data stream. All control characters transmitted by the 3276 to the TCU are generated by the control unit; a Read command to the TCU is used to store these characters (except pad and SYN) into main storage for subsequent analysis by the access method.

Pad

Pad characters, leading and trailing, are generated by TCU or 3276 hardware to ensure complete transmission or reception of the first and last significant character of each transmission.

SYN (Synchronous Idle)

Two consecutive SYN characters are generated by TCU or 3276 hardware to establish character synchronization. The TCU can also embed SYN characters in text for time-fill to maintain synchronization; the 3276 discards these SYN characters (does not store them in the buffer). Once this synchronization is lost, a Poll or a Selection sequence to a 3276 must be preceded by an EOT.

DLE (Data Link Escape)

DLE is always the first byte in the following 2-byte control characters: ACK 0, ACK 1, WACK, and RVI. DLE is also used as the first character in several 2-character sequences that are used in transparent-monitor mode (described earlier in this chapter under "Transparent Monitor Mode").

ACK 0 (Even Acknowledge)

ACK 0 is a 2-byte character, as follows:

- EBCDIC: 1070 (hex)
- ASCII: 1030 (hex)

ACK 0 is transmitted by the 3276 after a successful selection addressing (not poll) sequence to indicate to the TCU that the 3276 is ready to accept transmission. ACK 0 is also transmitted by the 3276 or by the TCU upon receipt and validation of an even-numbered (second, fourth, etc.) text block.

ACK 1 (Odd Acknowledge)

ACK 1 is a 2-byte character, as follows:

- EBCDIC: 1061 (hex)
- ASCII: 1031 (hex)

ACK 1 is transmitted by the 3276 or TCU upon receipt and validation of an odd-numbered (first, third, etc.) text block.

NAK (Negative Acknowledgment)

NAK is transmitted by the 3276 in response to a TCU text transmission that (1) terminates with ENQ, (2) has ENQ embedded in text, (3) has invalid BCC, (4) contains a TTD sequence (STX ENQ), or (5) has ETX missing.

When NAK is received by the 3276 in response to a text transmission, the 3276 retransmits the last block of text.

Programming Note: The TCU should be programmed to respond with NAK to an ENQ (that ends a text block) from the 3276; this NAK causes the 3276 to send EOT and retain the status for error recovery.

ENQ (Enquiry)

The 3276 transmits ENQ (1) to request a reply from the TCU following a 3-second timeout, (2) to request retransmission of the previous reply from the TCU, or (3) as the last character of a text message in which a data check was detected by the 3276. (See Programming Note above.)

When the 3276 receives ENQ in response to a transmission, the last 3276 transmission to the TCU is repeated. The 3276 responds with NAK when ENQ is received (1) as the last character of a TCU-aborted text transmission, (2) embedded in text, or (3) as part of a TTD sequence (STX ENQ).

To be addressed successfully, the 3276 must receive ENQ as the last character of a polling or selection addressing sequence.

WACK (Wait before Transmit)

WACK is a 2-byte character, as follows:

- EBCDIC: 106B (hex)
- ASCII: 103B (hex)

WACK is generated by the 3276 (1) in response to a selection addressing (not poll) sequence when a printer is busy, and (2) in response to a Write or Copy command text transmission when the Start Printer bit is set in the WCC or CCC. The 3276 responds with ENQ to a WACK from the TCU.

RVI (Reverse Interrupt)

RVI is a 2-byte character, as follows:

- EBCDIC: 107C (hex)
- ASCII: 103C (hex)

RVI is generated by the 3276 in response to an attempted selection (not poll) by the TCU when the 3276 has a status and sense message to be transmitted. Whenever the 3276 accepts RVI from the TCU, the CU responds with EOT and resets all pending status and sense information. The 3276 accepts RVI in place of ACK 0 or ACK 1 and then only when they would have been valid. If RVI is received at the 3276 in response to RVI, a timeout occurs at the 3276 unit.

STX (Start of Text)

The 3276 receives STX as the first character of a command or TTD sequence. The STX causes the 3276 to clear its BCC and start accumulating a new BCC (STX is not included in the accumulation). Subsequent STX (and SOH) characters are included in the BCC accumulation. STX is transmitted by the 3276 to the TCU as the first character of a read-data text block except in a status or test-request message; this STX causes the TCU to start accumulating a new BCC (STX is not included in the accumulation).

The first character in status and test-request messages is SOH, with STX following two header characters. With a message of this type, the TCU starts BCC accumulation upon receipt of the first SOH; the subsequent STX character is included in the BCC accumulation.

SOH (Start of Heading)

The 3276 generates SOH in a 3-character heading sequence that identifies the accompanying data as a status message (SOH, %, R, STX, - - -) or as a test-request message (SOH, %, /, STX, data - - -). The TCU starts BCC accumulation upon receipt of SOH (SOH is not included in the accumulation).

ETB (End of Transmission Block)

During a message transfer operation, ETB informs the receiving unit that BCC follows. The 3276 treats ETB as though it were ETX by checking BCC and then generating the appropriate response; the 3276 does not accept conventionally blocked outbound text.

ETX (End of Text)

During a message transfer operation, ETX informs the receiving unit that BCC follows. The 3276 transmits ETX at the end of the last (or only) block of a text message. Then, upon successful comparison of the received BCC with the accumulated BCC, the program should respond with ACK to the 3276. If the BCC comparison is unsuccessful, the TCU interrupts the program (Channel End, Device End, and Unit Check status, with Data Check set in the sense byte); the program should respond with NAK to the 3276. Receipt of ETX by the 3276 initiates a BCC comparison, causes a line turnaround, and causes generation of an appropriate response to the TCU.

EOT (End of Transmission)

EOT is transmitted by the 3276 (1) when the 3276 is a slave station and is unable to perform an operation requested by the TCU; (2) when the 3276 is a master station, as normal termination of a read operation; (3) when the 3276 has completed General Poll operations with each attached device; (4) as an answer to RVI sent by the TCU. Line synchronization is dropped, and the 3276 is returned to control mode. Note that the program can also issue EOT to the 3276 in order to drop line synchronization and return the 3276 to control mode. EOT does not reset status and sense in the 3276; therefore, it should not be sent as a response to a status message.

Following receipt of a valid selection addressing sequence, if an error occurs during buffer transfer, the 3276 will provide a positive response to the selection sequence and internally set DC and US status. EOT is sent in response to the following 3276 command or poll.

ITB (End of Intermediate Transmission Block)

The 3276 does not accept conventionally blocked text. However, to coexist on a BSC multipoint line on which ITB may be used, the 3276 includes the ITB and associated BCC in its own BCC accumulation but then removes them from the data stream so that they are not stored in the buffer. The 3276 does not perform a BCC comparison at that time, but continues the receive operations until ETB or ETX is decoded.

ESC (Escape)

ESC must precede the command code in each command-sequence data stream transmitted to the 3276, as follows: STX, ESC, CMD, - - -. The 3276 does not generate ESC.

TTD (Temporary Text Delay)

TTD is a 2-character sequence: STX ENQ. The 3276 responds to TTD by transmitting NAK to the TCU. The 3276 does not generate TTD. TTD may also be used by the master station to terminate an operation (that is, initiate a forward abort). The 3276 (slave station) will always respond with a NAK, expecting the master station to transmit EOT. In this case, the slave station interprets this sequence as a controlled forward abort rather than an end of transmission.

Operational Sequences (Nonswitched Line)

The following paragraphs describe the various data and control sequences that can be performed with the 3276 operating on a nonswitched line. These sequences are divided into four categories:

1. Specific and General Poll.

2. Selection addressing.

3. Write and control type commands.

4. Read-type commands.

The description of each category is associated with a Sequence/Response Diagram, which shows (1) 3276 responses to program-generated transmissions by the TCU and (2) normal program-handling of 3276 transmissions. These diagrams show the I/O supervisor/access method as examining 3276 response to determine which operation to initiate next; however, for specific applications, additional usage of command chaining in the channel programs may be desirable.

A selection addressing sequence selects a 3276 and an attached device for subsequent command operations. Polling sequences are selection sequences used specifically to obtain pending status at a device. Either a Specific Poll sequence requesting status from a particular device or a General Poll sequence sent to all devices may be executed.

Remote Chaining of 3276 Commands

For remote operations, 3276 command codes are included in the data stream of a Write CCW to the TCU. Remote chaining of 3276 commands is defined as the transmission of more than one command sequence to a 3276 following a single selection addressing or poll sequence. This chaining normally is accomplished with separate Write CCWs in the channel program. For example, the channel program could (1) write a selection addressing sequence and read the response for evaluation by the I/O supervisor/access method, (2) write a 3276 Write command and text block and read the 3276 response for evaluation, and then (3) write a 3276 Write command followed by a second text block and read the 3276 response for evaluation.

The program may chain 3276 commands following a selection addressing sequence, provided that the BSC rules governing limited conversational mode are observed. (Refer to *General Information - Binary Synchronous Communications,* GA27-3004.) The 3276 permits any valid command to be chained following a poll sequence; however, Read Buffer or Read Modified should not be chained because the BSC rules for limited conversational mode (a maximum of two consecutive data transfers without an intervening ACK) will be violated.

Any 3276 command (except Erase All Unprotected) may be chained from a Write, Erase/Write, Erase/Write Alternate, or Copy command. However, if one of the commands has started a print operation, the 3276 will abort the subsequent chained command (the print is completed normally).

General and Specific Poll Sequences

When a General or Specific Poll sequence is issued (Figure 3-2), one of three possible results occurs:

1. If status and sense information is pending with or without an AID present, a status and sense message is generated.

2. If status and sense information is not pending and an AID is present, a Read Modified command is executed.

3. If there is no status or sense information or no AID pending, an EOT response is generated.

Figure 3-9 lists the conditions under which status and sense messages are transmitted.

Control unit and device address bytes transmitted for the General and Specific Poll sequences are as follows:

1. General Poll address byte sequence:

```
3276 CU Poll Address     }  (See Figure 3-1.)
3276 CU Poll Address     }
7F (EBCDIC) or 22 (ASCII) } (Used in place of the two
7F (EBCDIC) or 22 (ASCII) }  device-address bytes.)
```

2. Specific Poll address byte sequence:

```
3276 CU Poll Address  }
3276 CU Poll Address  }
Device Address        }  (See Figure 3-1.)
Device Address        }
```

The selected 3276 remains selected at the completion of a poll operation so that the program can issue a Write, Erase/Write, Erase/Write Alternate, Copy, or EAU command without reselecting the 3276 and the device; command operations will be with (1) the device that was selected by Specific Poll or (2) the device from which a response was last received during the General Poll operation. Selection is dropped when the 3276 transmits EOT; the 3276 transmits EOT when it has no pending status or messages or after it receives NAK from the TCU in response to a message that ends with ENQ.

Specific Poll addresses the 3276 and one device to determine if status and sense information or a manually entered message is awaiting transfer to the TCU. The pending status and sense information or message is transferred automatically by the 3276 upon receipt of the Specific Poll addressing sequence.

General Poll addresses the 3276 and examines each attached device in sequence (starting at a random device address) to determine if a status and sense or a manually entered message is awaiting transfer to the TCU. If a message is present, it is transferred to the TCU. Each message is accompanied by the address of the device from which it originated.

Upon completion of this transfer, an ACK response from the program causes the 3276 to continue the General Poll operation, either by transferring another block of a text message or by examining other attached devices for pending messages. The program could issue a command rather than ACK to the device from which the message was just received, only after inbound blocks that end with ETX. The 3276 will ignore any commands that are sent in response to a block of data that ends with ETB. Once the 3276 has examined all attached devices and has successfully transferred all pending messages, it generates EOT and returns to control mode. If the program wishes to terminate the General Poll, an RVI may be issued to the 3276, forcing an EOT response. A command issued rather than the ACK (after blocks that end with ETX) will also terminate the General Poll.

Column 1
Use this column for:
- Device Selection,
- Specific Poll,
- General Poll, and
- Fixed Return Addresses

CU or Device Number	EBCDIC I/O Char.	EBCDIC Hex (Note 3)	ASCII I/O Char.	ASCII Hex
0	SP	40	SP	20
1	A	C1	A	41
2	B	C2	B	42
3	C	C3	C	43
4	D	C4	D	44
5	E	C5	E	45
6	F	C6	F	46
7	G	C7	G	47
8	H	C8	H	48
9	I	C9	I	49
10	¢	4A	[	5B
11	.	4B	.	2E
12	<	4C	<	3C
13	(	4D	(	28
14	+	4E	+	2B
15	I or !	4F	!	21
16	&	50	&	26
17	J	D1	J	4A
18	K	D2	K	4B
19	L	D3	L	4C
20	M	D4	M	4D
21	N	D5	N	4E
22	O	D6	O	4F
23	P	D7	P	50
24	Q	D8	Q	51
25	R	D9	R	52
26	!	5A	]	5D
27	$	5B	$	24
28	*	5C	*	2A
29	)	5D	)	29
30	;	5E	;	3B
31	¬ or ∧	5F	∧	5E

Column 2
Use this column for:
- 3270 CU Selection Addresses
- Test Requests

CU Number	EBCDIC I/O Char.	EBCDIC Hex (Note 3)	ASCII I/O Char.	ASCII Hex
0	-	60	-	2D
1	/	61	/	2F
2	S	E2	S	53
3	T	E3	T	54
4	U	E4	U	55
5	V	E5	V	56
6	W	E6	W	57
7	X	E7	X	58
8	Y	E8	Y	59
9	Z	E9	Z	5A
10	¦	6A	|	7C
11	,	6B	,	2C
12	%	6C	%	25
13	_	6D	_	5F
14	>	6E	>	3E
15	?	6F	?	3F
16	0	F0	0	30
17	1	F1	1	31
18	2	F2	2	32
19	3	F3	3	33
20	4	F4	4	34
21	5	F5	5	35
22	6	F6	6	36
23	7	F7	7	37
24	8	F8	8	38
25	9	F9	9	39
26	:	7A	:	3A
27	#	7B	#	23
28	@	7C	@	40
29	'	7D	'	27
30	=	7E	=	3D
31	" (Note 1)	7F	"	22

Examples:

3276 Addressing			
		EBCDIC	ASCII
General Poll CU5	CU Address	C5 C5	45 45
	Device Address	7F 7F	22 22
Specific Poll Device 4 on CU5	CU Address	C5 C5	45 45
	Device Address	C4 C4	44 44
Select Device 4 on CU5	CU Address	E5 E5	56 56
	Device Address	C4 C4	44 44

Notes:

1. *I/O character address (") is used as the device address to specify a General Poll operation.*
2. *Graphic character for the United States I/O interface codes are shown. Graphic characters for EBCDIC 4A, 5A, 5B, 7B, 7C, and 7F might differ for particular World Trade I/O interface codes. Refer to* IBM 3270 Information Display System: Character Set Reference, *GA27-2837, for possible graphic differences when these codes are used.*

Figure 3-1. Remote Control Unit and Device Addressing

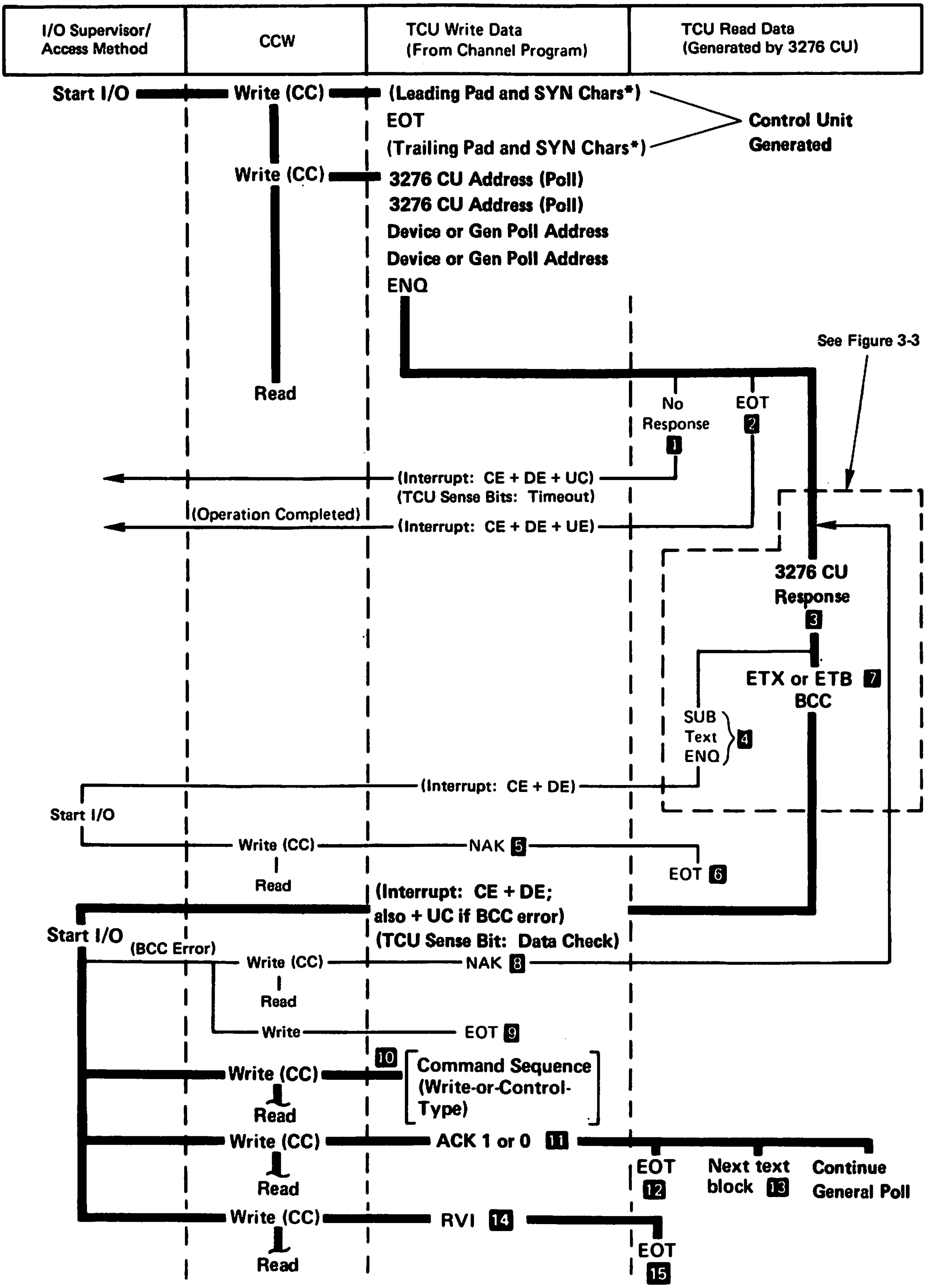


Figure 3-2 (Part 1 of 2). General Poll and Specific Poll, Sequence/Response Diagram

Notes:

1 *The 3270 CU will fail to respond to the addressing or polling sequence, causing a TCU timeout, for any of the following reasons:*

- *The 3270 CU is "unavailable" (has power off, is "offline", or is not attached).*
- *Any character in the polling sequence is invalid.*
- *The characters in the polling sequence are out of order.*
- *The polling sequence is incomplete (less than seven characters).*
- *The 3276 CU address is incorrect in the write data stream.*
- *The addressed 3276 CU was left selected from the previous transmission.*

2 *There is no I/O pending nor pending status. For General Poll, the CU sends EOT only after polling all devices.*

3 *The device response is a function of the kind of device and its status. Types of responses include: Text, Status, and Test Request messages. (Refer to Figure 3-3.)*

For General Poll, the search for a response starts at some random device address and continues sequentially (as long as ACKs are received in response to text transmissions) until all devices are given the opportunity to respond.

4 *Upon detection of an internal parity check or a cursor check, the 3276 CU (1) substitutes the SUB character for the character in error, (2) records Data Check status, and (3) transmits an ENQ in place of ETX (or ETB) and BCC at the end of the text block. The General Poll process is stopped.*

5 *Mandatory program response to a text block terminated in ENQ.*

6 *Terminates the operation. The nature of the error (parity or cursor check) does not warrant a retry. This response indicates that status and sense information is stored and that internal 3271/device polling is stopped. The status retrieval information included in Figure 3-6, Note 2, applies.*

7 *ETB is used to frame each block of a blocked text message, except the last block. ETX is used to frame the last block of a blocked text message.*

8 *BCC error has been detected. The program issues NAK to cause the 3276 CU to repeat its last transmission.*

9 *Response issued by the program to terminate the operation if the TCU is unsuccessful in receiving a valid BCC following "n" attempts by the 3276 CU to transmit the message. This response does not cause the 3270 CU to reset its sense/status information. Therefore, the same status message will be transmitted if a Specific Poll is immediately issued to the same device.*

10 *This transmission must be a write or control-type command sequence (described in Figure 3-5). A read-type command would violate BSC standards on limited conversational mode.*

For General Poll, this transmission stops the polling operation. The General Poll must be reinitiated to ensure receipt of all pending device messages.

11 *Positive acknowledgement. The text block has been successfully received by the TCU. The program issues ACK 1 in response to the first and all odd-numbered text blocks and issues ACK 0 in response to the second and all even-numbered text blocks.*

12 *Normal termination of a Specific Poll.*
Normal termination of a General Poll.

13 *The second and all suceeding text blocks are framed as the first except they do not include the 3270 CU/device address sequence.*

14 *RVI to terminate polling sequence.*

15 *Termination of polling sequence on receipt of RVI.*

Legend:

(CC) = Chain Command (CC) Flag in CCW is set to 1.

(Interrupt) = TCU-generated interrupt (CE = Channel End, DE = Device End, UE = Unit Exception, UC = Unit Check).

1 Reversed numbers refer to notes.

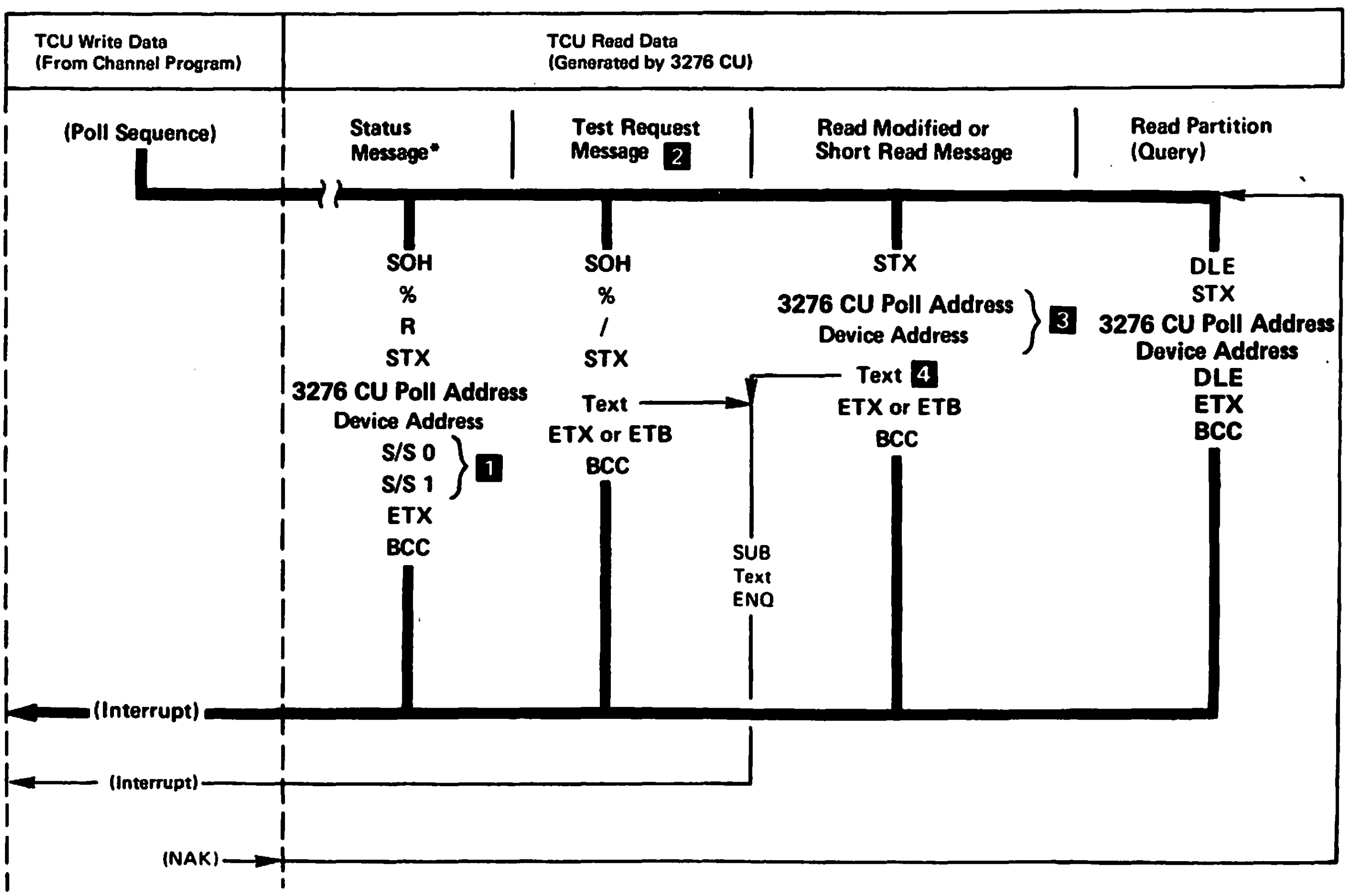
*Only the critical framing characters (sync pattern and pad) are shown. All other framing characters are also hardware-generated as required. See SL *General Information - Binary Synchronous Communications,* GA27-3004, for a complete description.

Figure 3-2 (Part 2 of 2). General Poll and Specific Poll, Sequence/Response Diagram

Figure 3-3 shows the message formats. The Test Request, Read Modified, and Short Read operations and the resulting data are described under "Read Modified Command" in Chapter 1. Note that a device address is not provided in the heading of a Test Request message. An address must be manually entered by the operator as part of the text; this is because the operator may specify the address of another device for test operations with the program.

The status and sense bits are described later in this chapter under "Status and Sense (S/S) Bytes."

(Note: This figure is referenced in Figures 4-2 and 4-6.)



*Response to General Poll or Specific Poll only (not program-generated Read Modified command)

Notes:

1 *A status message response is issued to a General or Specific Poll if (1) the 3276 CU has pending status (General Poll ignores Device Busy and device "unavailable" and, if the 3276 continues polling of next device), or (2) if error status develops during execution of the poll. Status and sense bit assignments are described in Figure 3-7.*

2 *A Test Request Message response is issued to a General or Specific Poll if a SYS REQ key is pressed at a 3278 or 3279 attached to a 3276.*

3 *This address is included only in the first block of a blocked text message.*

4 *The text portion of this message is the result of either a Read Modified or Short Read operation by the 3270 CU. Figure 3-5 lists each operator action and the resulting read operation that will be performed.*

Legend:

(Interrupt) = TCU-generated interrupt.

1 Reversed numbers refer to notes.

Figure 3-3. 3276 CU Message Response to Polling or Read Modified Command

Selection Addressing Sequence

The selection addressing sequence (Figure 3-4) specifies a 3276 and an attached device in preparation for write-, control-, or read-type command sequences. It is similar in format to a Specific Poll sequence in that a CU address is sent, followed by a device address, but different I/O characters and hex codes are used to represent the CU address bytes.

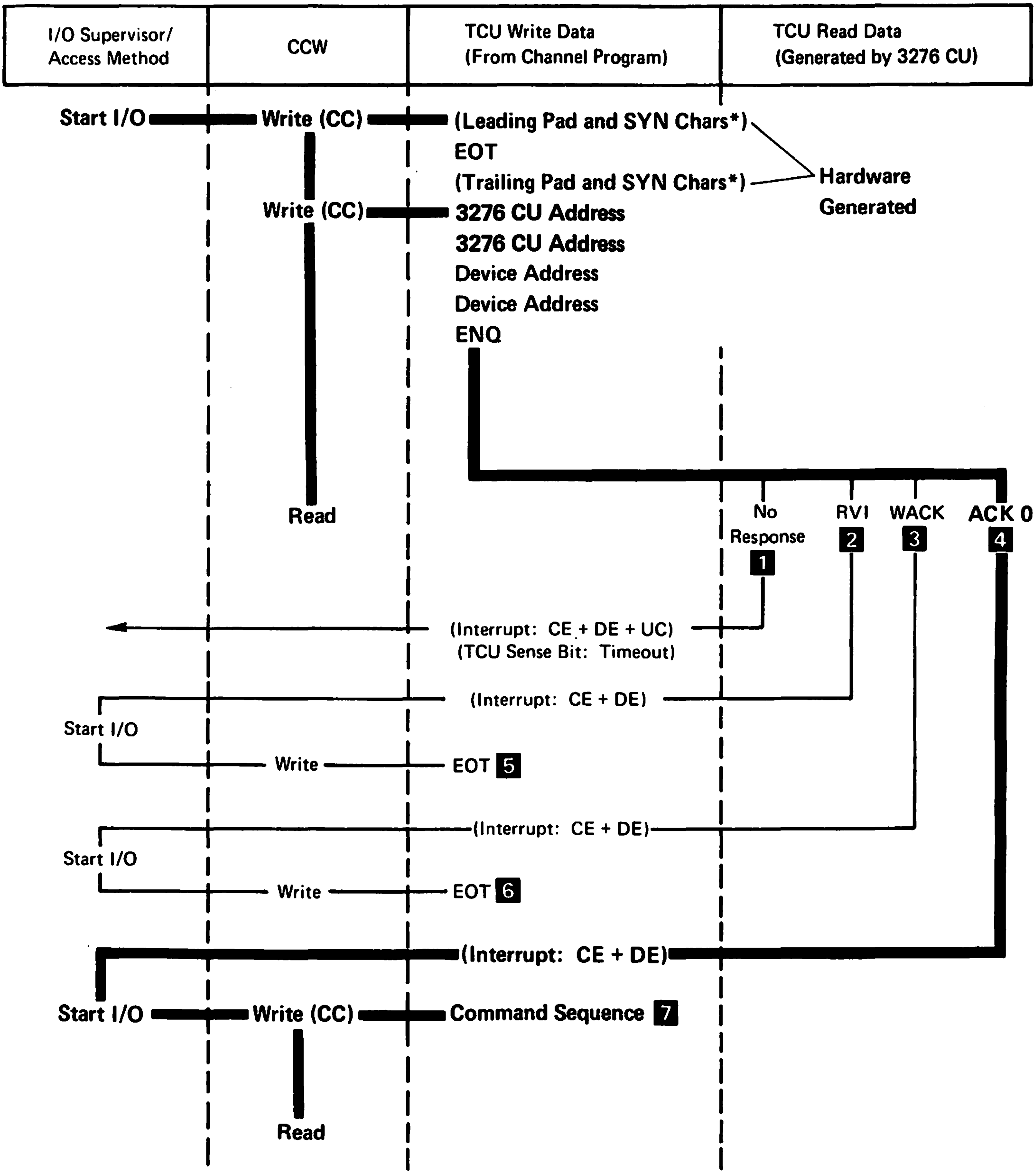
Column 1 in Figure 3-1 lists the characters and hex codes used to complete the selection addressing sequence. Comparative examples showing CU and device address codes for General Poll, Specific Poll, and selection addressing sequences appear at the bottom of Figure 3-1.

For the 3276, the selection addressing sequence performs a function similar to a local Select command in that it causes a device-to-control unit buffer transfer. The 3276 provides a positive response to a selection sequence before transfer of a device buffer to the 3276. If an error occurs during buffer transfer, following receipt of a valid selection addressing sequence, a positive response to the selection sequence is provided by the 3276, and DC and US status are internally set. EOT is sent in response to the following 3276 command.

Write-Type and Control-Type Command Sequences

The program initiates a Write, Erase/Write, Erase/Write Alternate, Copy, or EAU operation (Figure 3-5) by first writing a command and, except for EAU, a data sequence to the selected 3276 and, then, reading the response. All write-type commands and Copy commands must be followed by a minimum of one data byte (the WCC or CCC byte). If the program reads a positive response (ACK) from the 3276, it can terminate the operation or continue with another command. The program can write blocks of text to the 3276 by initiating, after receipt of each ACK, a Write command sequence for each block to be written.

Write data is blocked to devices attached to a 3276 as follows: Each time the 3276 receives a selection addressing sequence, it begins to transfer the device buffer contents to the control unit buffer. As the Write command data is received by the control unit, updating occurs, and the result is asynchronously transferred to the buffer of the addressed device. The device buffer contents not affected by the write data stream remain unaltered in the device buffer. If the transmission of a block of data to the control unit is successful (ACK reply), a device-to-control-unit buffer transfer is begun. If the transmission of a block of write data to the control unit is unsuccessful (e.g., NAK reply), the buffer contents previously stored in the control unit buffer are immediately transferred to the device buffer before another Write command is received. These contents include any previous text blocks that were written successfully. Thus, the 3276 can receive retransmission of the block that was unsuccessfully received.



*Only the critical framing characters (sync pattern and pad) are shown. All other framing characters are also hardware-generated as required. See SL *General Information – Binary Synchronous Communications,* GA27-3004, for a complete description.

Figure 3-4 (Part 1 of 2). Selection Addressing, Sequence/Response Diagram

Notes:

1 *The 3276 CU will fail to respond to the addressing or polling sequence causing a TCU timeout, for any of the following reasons:*

- *The 3276 CU is "unavailable" (has power off, is "offline", or is not attached).*
- *Any character in the polling sequence is invalid.*
- *The characters in the polling sequence are out of order.*
- *The polling sequence is incomplete (less than seven characters).*
- *The 3276 CU address is incorrect in the write data stream.*
- *The addressed 3276 CU was left selected from the previous transmission.*

2 *The addressed device has pending status (excluding Device Busy or Device End).*

3 *The addressed 3276 device is busy. No S/S information is stored. An RVI response takes precedence over a WACK response.*

4 *The address has been successfully received, no status is pending.*

5 *Termination of attempted addressing sequence:*
Availability of valid status and sense information cannot be ensured unless a Specific Poll is issued to the responding device as the next addressing sequence issued to this 3276 CU.

6 *Termination of attempted addressing sequence.*

7 *Refer to Figure 3-5 or 3-6 for the desired command sequence.*

Legend:

(CC) = Chain Command (CC) Flag in CCW is set to 1.

(Interrupt) = TCU-generated interrupt (CE = Channel End, DE = Device End, and UC = Unit Check)
1 Reversed numbers refer to notes.

Figure 3-4 (Part 2 of 2). Selection Addressing, Sequence/Response Diagram

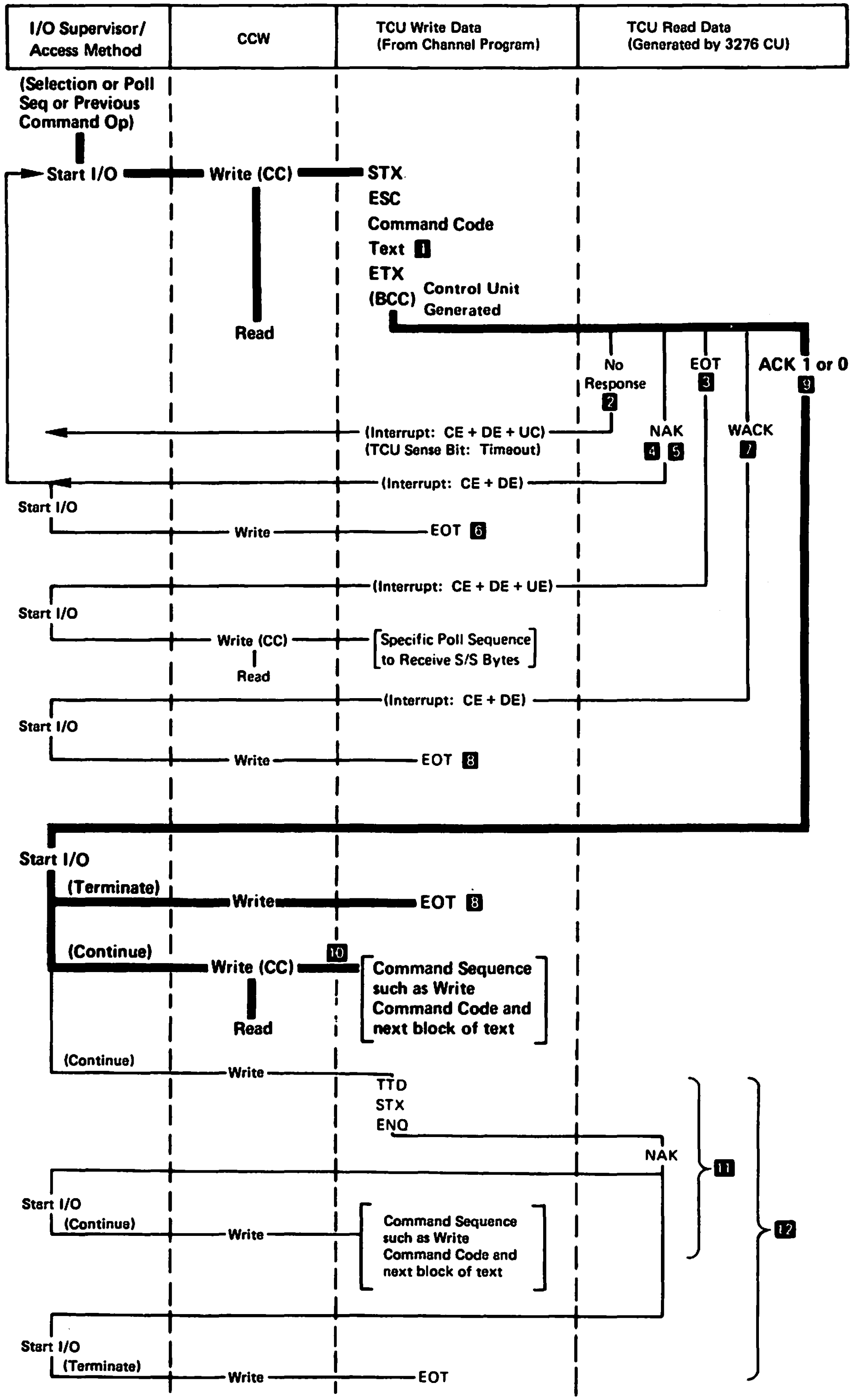


Figure 3-5 (Part 1 of 2). Write-Type and Control-Type Commands, Sequence/Response Diagram

Notes:

1 *No text is transmitted on an EAU command transmission.*

2 *Command transmission was not successfully received because of invalid framing (STX missing). Causes a timeout at TCU.*

3 *The control unit is unable to perform the operation indicated in the command transmission because of a busy/unavailable/not ready device.*

4 *If a transmission problem causes both a 3276 CU detected check condition and a BCC error, the BCC error takes precedence over all other check conditions, and a NAK is transmitted to the TCU.*

5 *BCC error or missing ETX has been detected. The NAK response requests the program to repeat its last transmission.*

6 *Response issued by the program to terminate the operation if the 3276 CU is unsuccessful in receiving a valid BCC following "n" attempts by the program to transmit the message.*

7 *If the Start Printer bit is set in the WCC or CCC, a WACK response indicates that the text transmission was successfully received but that the printer is now busy and an additional chained command cannot be accepted.*

If any of the conditions cited in Note **3** *prevail, the EOT response takes precedence over the WACK response.*

8 *Normal termination of the operation by the program.*

9 *Command execution has been successfully completed.*

10 *Repeat the operation shown in this figure or in Figure 3-6 for the next command sequence.*

11 *Example of a Temporary Text Delay (TTD) sequence.*

12 *Example of terminating an operation using TTD (a forward abort sequence).*

Legend:

(CC) = Chain Command (CC) Flag in CCW is set to 1.

(Interrupt) = TCU-generated interruption (CE = Channel End, DE = Device End, UE = Unit Exception, UC = Unit Check).

1 = Number in parentheses refers to note.

Figure 3-5 (Part 2 of 2). Write-Type and Control-Type Commands, Sequence/Response Diagram

Read-Type Command Sequences

Programming Note: Read Buffer is used primarily for diagnostic purposes, and Poll (General and Specific) is normally used in place of Read Modified for remote read operations.

The program initiates a read operation (Figure 3-6) by first writing a command sequence to the selected 3276, and then reading the response. If the 3276 responds with text followed by ETB, and if BCC comparison at the TCU is successful, the program should write ACK to retrieve the next text block. This should continue until an error is detected or until a text block is followed by ETX. After ETX is received, the program should write ACK to the 3276 and then read the EOT reply. The three types of Read Modified message responses are shown in Figure 3-3.

The 3276 will retransmit text up to 15 times when NAK or an incorrect ACK is received or when ENQ is received in response to a conversational text reply to a Read command. The 3276 supports limited-conversational-text mode. If the host transmits a text block following receipt of a text transmission which ends in ETB, a timeout occurs at the 3276 unit and ENQ is sent to the host.

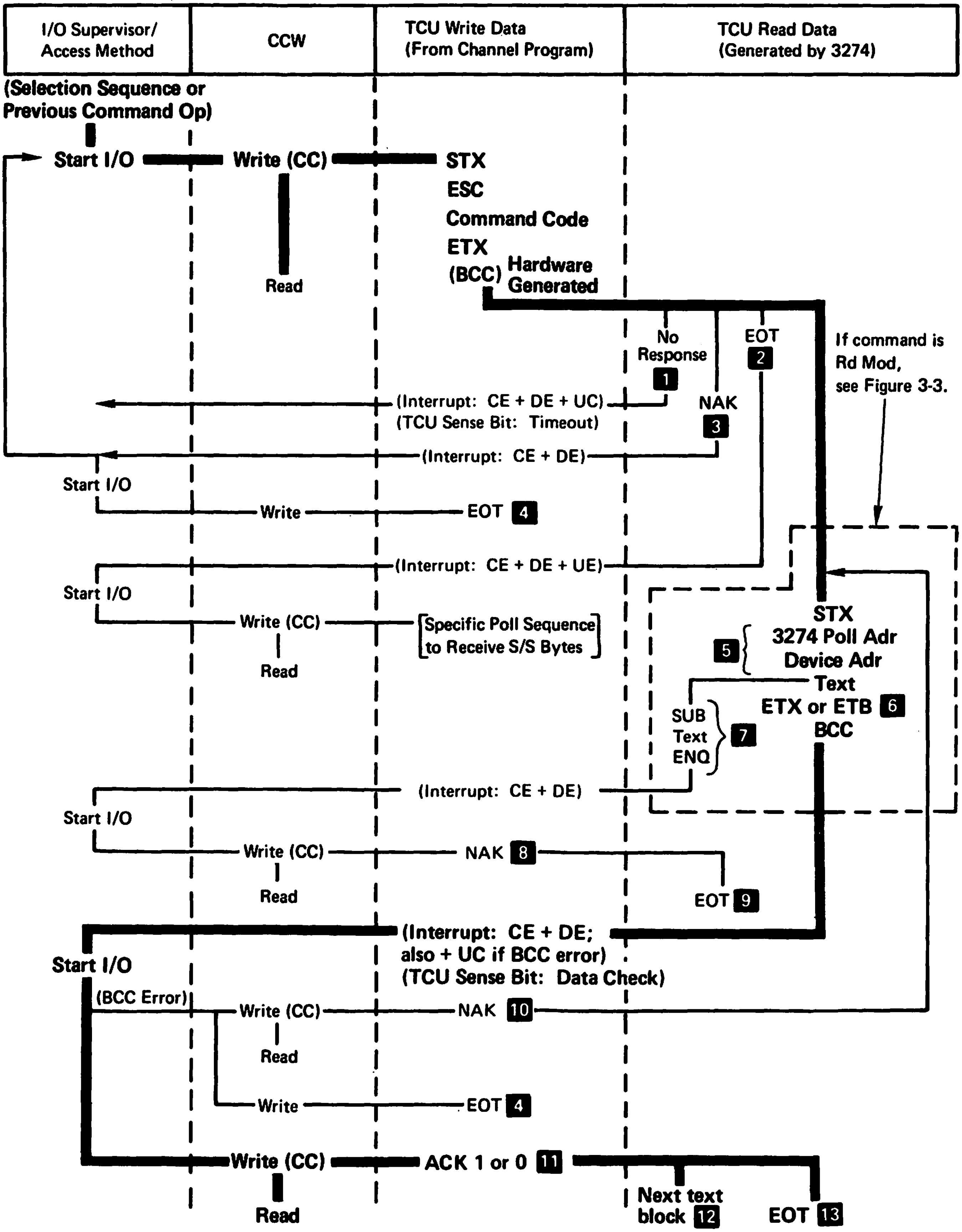


Figure 3-6 (Part 1 of 2). Read-Type Command, Sequence/Response Diagram

Notes:

1 *Command transmission was not successfully received because of invalid framing (STX missing). Causes timeout at TCU.*

2 *The 3276 CU is unable to perform the operation indicated in the command transmission because of a busy/unavailable/not ready device or a 3276 CU-detected check condition (receipt of an illegal command/order sequence, failure to decode a valid command, or an I/O interface "overrun"). The EOT response to a command transmission indicates that status information is stored in the 3276 CU.*
To ensure retrieval of valid status, a Specific Poll must be issued to the device-responding EOT as the next addressing sequence issued to this 3276 CU.

3 *If a transmission problem causes both a 3276 CU-detected check condition and a BCC error, the BCC error takes precedence over all other check conditions, and a NAK is transmitted to the TCU.*

4 *Response issued by the program to terminate the operation if the 3276 CU is unsuccessful in receiving a valid BCC following "n" attempts by the program to transmit the message.*

5 *This address sequence is included only in the first block of a blocked text message.*

6 *ETB is used to frame each block of a blocked text message, except for the last block. ETX is used to frame the last block of a blocked text message.*

7 *Upon detection of an internal parity check, the 3276 CU automatically substitutes the SUB character for the character in error.*
If a parity or cursor check is detected, ENQ 1, transmitted in place of ETX (or ETB) and BCC at the end of the text block and appropriate status and sense information is stored. This is also used by the 3276 if, after transmitting the first block, the transmission cannot be completed due to power being off at the terminal.

8 *Mandatory program response to a text block terminated in ENQ.*

9 *Response to terminate the operation. The nature of the error (parity or cursor check) does not warrant a retry. This response indicates that appropriate status and sense information is stored. The status retrieval information included in Note* **2** *applies.*

10 *BCC error has been detected. The program issues NAK to cause the 3276 CU to repeat its last transmission.*

11 *Positive acknowledgement. The text block has been successfully received by the TCU. The program issued ACK 1 in response to the first and all odd-numbered text blocks and issued ACK 0 in response to the second and all even-numbered text blocks. This response to a text block terminated in ETX turns on the device SYSTEM AVAILABLE indicator.*

12 *The second and all succeeding text blocks are framed as the first except that they do not include the 3276 CU/device address sequence.*

13 *Normal termination of the operation following transmission of the last text block.*

Legend:

(CC) = Chain Command (CC) Flag in CCW is set to 1.

(Interrupt) = TCU-generated interrupt (CE = Channel End, DE = Device End, UE = Unit Exception, UC = Unit Check)

1 Reversed numbers refer to notes.

Figure 3-6 (Part 2 of 2). Read-Type Commands, Sequence/Response Diagram

Status and Sense (S/S) Bytes

All remote status and sense conditions are combined into 2 bytes. These 2 bytes are always sent in a status message. In EBCDIC code, the bits are transmitted as indicated in Figure 3-7. If the sense bytes are transmitted in ASCII code, the EBCDIC code defined below is translated to ASCII before transmission.

Status and sense conditions are recorded by the 3276 for each device. These conditions may include busy or ready status or detected errors. Figure 3-8 shows how these status and sense conditions are interpreted for each error response transmitted by the 3276 in response to a poll sequence from the TCU.

Error-Recovery Procedures

Errors detected at the 3276 are indicated to the system processor by the following responses: RVI, NAK, EOT, or sense/status information. The meaning of the responses depends upon their sequences, as defined in Figures 3-2 through 3-6.

When errors occur in the 3178, 3278 and 3279, the error condition is reported once to a General Poll. The 3276 allows parts of messages to be transmitted to the host before all data is transferred from the 3178, 3278, and 3279 to the 3276. If a terminating condition prevents completion of data transfer from the 3178, 3278, and 3279 to the 3276 after inbound link transmission has started, the 3276 sends STX . . . . . SUB ENQ. The 3276 responds to a Specific Poll with DC status. Following a selection addressing sequence, a write-type command is accepted but a read-type command is rejected, and DC status is returned by the 3276.

When the host selects the 3276 and issues a Read Modified command, the 3276 transmits a single block of text followed by ETX. If the host makes an error by starting a new command sequence with STX, the 3276 responds with ENQ. If more than one text block is transmitted to the host, with ACK received from the host after each ETB, the host may respond to ETX on the last block, with a new command sequence beginning with STX, ESC.

Figure 3-9 lists the various error combinations of sense/status bits (with the exception of Device Busy (DB), which is not an error) and the recommended error-recovery procedure for each combination. Although there are 256 possible combinations of status and sense bits, only a portion of this total is normally used. Combinations other than those listed may occur. For example, an unpredictable catastrophic hardware failure could induce an undefined combination of status and sense bits. Errors that occur at the "from" device during a Copy command are identified by an Operation Check (OC) sense bit in addition to the sense bit representing the detected error.

Bit No.	Bit Definition
	S/S Byte 0:
0	Dependent upon setting of bits 2-7.
1	Always a 1.
2	Reserved.
3	Reserved.
4	*Device Busy (DB)* — This bit indicates that the addressed device (except the 3178, 3278, or 3279) is busy executing an operation or that a busy detection was previously made by a command or Specific Poll. The device is busy when it is executing an Erase All Unprotected command or a print operation, accepting data from the operator identification card reader, or performing various keyboard operations (ERASE INPUT, Backtab, and CLEAR). This bit is set with Operation Check when a Copy command is received which specifies a "busy" device with its "from" address. This bit is set with Unit Specify when a command is addressed to a busy device. This can occur by chaining a command to a Write, Erase/Write, Erase/Write Alternate, or Copy command which started a printer or by chaining a command to a Specific Poll addressed to a busy device. **Note:** DB is not returned for the 3178, 3278, or 3279 when executing an Erase All Unprotected command, accepting data from the MSR, or performing ERASE INPUT, Backtab, or CLEAR keyboard operations.
5	*Unit Specify (US)* — This bit is set if any S/S bit is set as a result of a device-detected error or if a command is addressed to a busy device.
6	*Device End (DE)* — This bit indicates that the addressed device has changed from unavailable to available and not ready to ready, or busy to not busy. This bit is included during a Specific or General Poll but is not considered pending status by a selection-addressing sequence. If a selection-addressing sequence detects that the addressed device has pending status and also detects one of the above status changes that warrants a Device End, then the Device End bit is set and preserved along with the other pending status, and an RVI response is made.
7	Reserved.
	S/S Byte 1:
0	Dependent upon setting of bits 2-7.
1	Always a 1.
2	*Command Reject (CR)* — This bit is set upon receipt of an invalid 3270 command.
3	*Intervention Required (IR)* — This bit is set if: • A Copy command contains a "from" address in its data stream which specifies an unavailable device. • A command attempted to start a printer but found it not ready. The printout is suppressed. • The 3276 receives a selection-addressing sequence or a Specific Poll sequence for a device which is unavailable or which became not ready during a printout. A General Poll sequence does not respond to the unavailable/not ready indication and proceeds to determine the state of the next device. • The 3276 receives a command for a device which has been logged as unavailable or not ready.
4	*Equipment Check (EC)* — This bit indicates a printer character generator or sync check error occurred, the printer became mechanically disabled, or a 3276 detected bad parity from the device.
5	*Data Check (DC)* — This bit indicates the detection of bad parity from the device, or 3276 operation to a device was unsuccessful (i.e., the device was disabled with DC returned to the host; IR will be returned on subsequent retry by the host).
6	*Control Check (CC)* — For the 3276, this bit indicates a timeout check. A timeout check occurs when a device fails to respond to 3276 communications within a specified time period or when a device fails to complete an operation within a specified time period.
7	*Operation Check (OC)* — This bit, when set alone, indicates one of the following: • Receipt of an illegal buffer address or of an incomplete order sequence on a Write, Erase/Write, or Erase/Write Alternate command. • The device did not receive a CCC or a "from" address on a Copy command. • Receipt of an invalid command sequence. (ECS is not received in the second data character position of the sequence.) • The internal buffering capability is exceeded on a 3276. This bit is set with Unit Specify to indicate that the "from" address on a Copy command specified a device with a "locked" buffer (the device data is secure).

Figure 3-7. Remote Status and Sense Byte Definitions –BSC

Device Response	Command	S/S Explanation
RVI	Selection	Outstanding Status – Pending information from a previous operation with the same device. (If the addressed device is busy, WACK is sent to the TCU instead of RVI, and no S/S bit is set.) Note: *A selection-addressing sequence does not recognize a Device End as pending status. If there is no other pending status, it resets this bit and proceeds with the selection. If the addressed device has other pending status, Device End remains set with it, and the RVI response is made as usual.* CC – A timeout check is caused by the addressed device. The operation is tried twice before this bit is set. CC, IR – Power on reset occures during selection. IR – The addressed device is unavailable. DE, IR – The addressed printer is out of paper, its power has been turned off, or its cover is open. DE, IR, EC, US – The addressed printer is mechanically disabled and cannot recover. DE, DC, US – A parity error is detected at the printer. DC, US – A parity check or cursor check is detected by the addressed device on the data it is sending to the control unit. For a 3276, an operation to a terminal was unsuccessful. The terminal was disabled and DC US returned to the host. On subsequent retry by the host, IR will be returned to the host.
EOT	Read Commands	CR – Invalid 3270 command is received. OC – Invalid command sequence (ESC is not in the second data character position), or data follows the command in the data stream received at the device. DB, US – The addressed device is busy. The command was chained to a Write, Erase/Write, Erase/Write Alternate, or Copy command which started a print, or it was chained to a Specific Poll. IR – A command is addressed to an unavailable device. DC – The 3276 is unable to complete a Read command operation after the first block has been sent to the host, because either there was an error in the terminal or the terminal was powered off after the first block was sent. A SUB character and an ENQ character are placed in the buffer. DC, US – A parity check or cursor check is detected by the addressed device on the data it is sending to the control unit. For a 3276, an operation to a terminal was unsuccessful. The terminal was disabled and DC US returned to the host. On subsequent retry by the host, IR will be returned to the host.
EOT	Write Commands	CR – An invalid or illegal 3270 command is received. OC – An invalid command sequence (ESC is not in the second data position), an illegal buffer address or an incomplete order sequence is received, or a data byte was sent to the device during the Write command before the operation required by the previous data byte was completed. DC, US – The device detects a parity or cursor check on its buffer during the command operation. For a 3276, an operation to a terminal was unsuccessful. The terminal was disabled and DC US returned to the host. On subsequent retry by the host, IR will be returned to the host. CC – The device fails to complete an operation. DB, US – The addressed device is busy. The message is accepted but not stored in the 3276 buffer. The command is aborted.

Figure 3-8 (Part 1 of 2). Remote Error Status and Sense Responses – BSC

Device Response	Command	S/S Explanation
EOT	Copy Command	CC, OC – The "from" device fails to complete an operation or respond to the 3271 in a certain time (timeout check). (Not used for the 3274 or 3276.) DB, OC – The "from" device is busy. (The device is busy executing an operation, a printout, reading data from the operator identification card reader, or performing a keyboard operation.) The Copy command is aborted. IR, OC – The "from" device is not available. OC, US – The "from" device has a locked buffer. CC – The data stream contains other than two bytes (the CCC and the "from" address). The command is aborted. OC – The "from" device buffer is larger than the "to" device buffer. OC – The buffer of the "from" device (as APL/Text feature) contains APL/Text characters (entered since an Erase/Write or Erase/Write Alternate command or a CLEAR key operation) and the "to" device does not have the APL/Text feature. DC, OC, US – Set when "from" device detects an internal parity or cursor check. For 3274 or 3276, an operation to a terminal was unsuccessful. The terminal was disabled and DC US returned to the host. On subsequent retry by the host, IR will be returned to the host. DB, US – The addressed "to" device is busy.
EOT	Write, Erase/Write, Erase/Write Alternate, Copy Commands	IR – Addressed device is not available, or addressed printer is not ready.
EOT	Erase All Unprotected Command	OC – One or more data bytes followed the command (buffer overrun).
	Specific and General Poll	DE, IR, EC, US – An unrecoverable mechanical failure is detected at the printer. DC, US – A parity check or cur: ɿr check is detected by the addressed device on the data it is sending to the control unit. For a 3276, an operation to a terminal was unsuccessful. The terminal was disabled and DC US returned to the host. On subsequent retry by the host, IR will be returned to the host. DC – The 3276 is unable to complete a Read Command operation after the first block has been sent to the host, because either there was an error in the terminal or the terminal was powered off after the first block was sent. A SUB character and an ENQ character are placed in the buffer. DE – The poll finds a device (1), previously recorded as busy, now not busy or, (2), previously recorded as unavailable *or* not ready, now available *and* ready. IR, DE – The poll finds a device, previously recorded as ready, available, and busy, now not ready and not busy, or the printer went not ready during a printout. DC, US, DE – A parity error is detected at printer. CC (Specific Poll only) – The poll finds a device, previously recorded as unavailable, still unavailable (timeout check).
	Specific Poll	DB – The addressed device is busy.
NAK	Read and Write Commands	NAK is transmitted by the 3276 when it detects a block check character (BCC) error on the TCU transmission. A BCC error has priority over all other detectable error conditions. If, for example, a BCC error and parity error are detected during the same command transmission, the parity error condition is reset, and a NAK response is set by the 3276.

Figure 3-8 (Part 2 of 2). Remote Error Status and Sense Responses – BSC

Sense/ Status Bits	Detected during 3270 Operation						Transmitted in Response to:		Error Recovery Procedure
	Hex		Selection Addressing Sequence	Specific Poll Sequence	General Poll Sequence	A 3270 Command	Specific Poll	General Poll	3276
	EBCDIC	ASCII							
CR	40 60	20 2D				D, P	D, P		6
OC	40 C1	20 41				D, P	D, P		6
OC, US	C4 C1	44 41				D, P	D, P		13
IR	40 50	20 26	D, P	D, P		D, P	D, P		4
IR, OC	40 D1	20 4A				D, P	D, P		5
DC	40 C4	20 44	D, P	D, P	D, P	D, P	D, P	D, P	2
DC, US	C4 C4	44 44	D, P	D, P	D, P	D, P	D, P	D, P	2
DC, OC, US	C4 C5	44 45				D, P	D, P		3
DC, US, DE	C6 C4	46 44		P	P		P	P	8
IR, DE	C2 50	42 26		P	P		P	P	4
IR, EC, US, DE	C6 D8	46 51		P	P		P	P	7
DB	C8 40	48 20	D, P	D, P			D, P		9
DB, US*	4C 40	3C 20				D, P	D, P		10
OC, DB*	C8 C1	48 41				D, P	D, P		11
DE	C2 40	42 20		D, P	D, P		D, P	D, P	None
CC, IR	40 D2	20 4B		D, P	D, P	D, P	D, P		4

Note: *The attached device errors that are detected asynchronously do not cause a sense bit to set until the device is polled for status during a selection-addressing, Specific Poll, or General Poll sequence. Those error S/S bit combinations that contain DE were detected during a printout.*

*The DB, US, and OC S/S bits will be combined if a Copy command is addressed to a busy "to" device and the command also specifies the "from" device the same as the "to" device.

Legend:
NA – Not Applicable
D – Display (3178, 3276, 3278, 3279)
P – Printer

Figure 3-9. Remote 3276 BSC Status and Sense Conditions

The error-recovery procedures recommended in Figure 3-9 are as follows:

1. Execute a new address selection addressing sequence and retransmit the message, starting with the command sequence that was being executed when the error occurred. If, after two retries, the operation is not successful, this should be considered a nonrecoverable error. Follow supplementary procedure B after two retries.

2. Reconstruct the entire device buffer if possible, and retry the failing chain of commands (within the BSC sequence of operations). The sequence of commands used to reconstruct the buffer should start with an Erase/Write or Erase/Write Alternate command. If the information in the screen buffer is such that it cannot, or need not, be reconstructed, the operation may still be retried. If an unrecoverable 3178, 3278, and 3279 buffer error or an error occurring on a transfer between the 3276 and the 3178, 3278, and 3279 is detected, the entire buffer is cleared and the host system is informed of the error by receiving DC, US status but is not informed of the clear operation. If, after three retries, the operation is not successful, this should be considered a nonrecoverable error. Follow supplementary procedure A.

3. The error occurred during execution of a Copy command. Execute procedure 2, except that it is the buffer of the "from" device specified by the Copy command that should be reconstructed. After three retries, follow supplementary procedure B, listed below under "Supplementary Procedures."

4. The error indicates that the printer is out of paper, has its cover open, or has a disabled print mechanism; or it indicates that the device is unavailable. Request (or wait for) either the display or system operator to ready the device. Then, retry the printout by issuing a Write command with the proper WCC and no data stream. (There is no data error, and the data is still intact in the device buffer and can be reused.) OR, follow procedure 2.

5. The error indicates that the "from" device specified by a Copy command is inavailable. Note that the device address associated with the error status and sense information does not indicate the device that actually required "readying." The device that requires the corrective action is the device specified by the "from" address in the Copy command. When the device is determined and made "ready," follow procedure 1.

6. The operation should be tried up to six times. Continued failure implies an application programming problem which can be detected by analyzing the failing write data stream.

7. The error occurred during a printout operation and indicates either a character-generator error or a disabled print mechanism. There is no data error. The proper error-recovery procedure is application-dependent since the user may or may not want a new printout. If a new printout is required, follow procedure 4.

8. A data error occurred in the device buffer during a printout; procedure 2 should be followed.

9. A Specific Poll detected that the addressed device is busy. Periodically issue a Specific Poll to pick up the Device End sense/status bit sent by the device when it becomes not-ready (unless this status change is detected on a selection addressing sequence).

10. Indicates that a command was erroneously addressed to a busy device. Periodically issue a General or Specific Poll to pick up the Device End sense/status bit sent by the device when it becomes not busy. Then follow procedure 1.

11. Indicates that, in attempting to execute a Copy command, the "from" device was found to be busy. Follow procedure 1 when the "from" device becomes not busy. Note that the device address associated with the status and sense message is the address of the "to" device and not that of the busy "from" device. The "from" device will transmit Device End via a Specific or General Poll when it becomes not busy.

12. An attempt was made to execute a Copy command, but access to the "from" device data was not authorized. The device address associated with the error sense/status bits is that of the copy "to" device.

Supplementary Procedures

A. Request maintenance for the device that is giving trouble. After repair, reconstruct the screen buffer image. The sequence of commands used to reconstruct this image should start with an Erase/Write command. Retry the failing chain of commands according to the procedure that referred you to this supplementary procedure.

B. The "from" device specified by the Copy command in the failing chain of commands (CCWs) is malfunctioning. The "from" device should be determined from the data-stream information, and maintenance should be requested for the device. After the repair, reconstruct the buffer image. The sequence of commands used to reconstruct this image should start with an Erase/Write command. Retry the failing chain of commands according to the procedure that referred you to this supplementary procedure.

C. Same as procedure 1, except a new selection addressing sequence is not performed, and this message is transmitted as part of the present device selection.

D. Same as procedure 1, except retransmit the entire failing chain of commands.

NAK to a Text Block

When the 3276 detects a BCC error at the end of a text transmission, it transmits a NAK. The following recovery action should be taken.

If the text is a write command sequence chained from a previous Write, Erase/Write, or Erase/Write Alternate command, and if the failing write command data stream contains more than one byte but does not contain an SBA order sequence immediately following the WCC, then procedure 2 (above) should be executed.

In all other cases, supplementary procedure C (above) should be executed, except the number of retries should be six. If after these six retries the problem is not corrected, the program should issue an EOT and follow supplementary procedure A (above).

Note: *An FF (hex) character in a data field does not cause a BCC error when operating with the 3276 units.*

EOT to a Text Block

The recommended recovery procedure depends upon the type of detected error. A Specific Poll must be issued immediately following the EOT to obtain the error sense/status information. Then the recovery procedures referenced in Figure 3-9 should be executed.

Errors Detected during a Specific or General Poll Sequence

Any errors that result from execution of the poll sequence itself are contained in Figure 3-9, and those recovery procedures apply. The detected error bits are transmitted to the TCU in a Status Message during the poll sequence.

RVI to Selection Addressing Sequence

A Specific Poll must be issued immediately following the RVI to a selection addressing sequence to obtain the error sense/status information. Then the recovery procedures defined in Figure 3-9 should be followed.

Chapter 4. 3276 SNA/SDLC Communications

This chapter provides information to aid the system analyst and the system programmer in establishing the host-to-3276 communications, using System Network Architecture (SNA) protocols. A knowledge of the Network Control Program (NCP), IBM access methods, and/or 3790 concepts, where the 3276 is attached to 3790, is assumed. The IBM access methods supporting SNA are VTAM, TCAM, and EXTM.

Additional information on SNA can be found in *Systems Network Architecture Format and Protocol Reference Manual: Architecture Logic,* SC30-3112. Information to assist the host programmer in planning the use of SNA commands and access method macros can be found in the following publications:

VTAM:

VTAM Concepts and Planning, GA27-6998.
VTAM Macro Language Reference, GC27-6995.
VTAM Macro Language Guide, GC27-6994.

TCAM:

TCAM Concepts and Application OS/VS, GA30-2049.
TCAM Programmer's Guide OS/VS1, GC30-2054.
TCAM Programmer's Guide OS/VS2, GA30-2041.

EXTM Option of CICS/DOS/VS:

EXTM Version 1.0 General Information, GH20-1597.
EXTM Version 2.0/3.0 General Information, GH20-1702.

3790 Communication System:

Introduction to the IBM 3790 Communication System, GA27-2807.

Network Control Program:

IBM 3704 and 3705 Communications Controller Network Control Program/VS Generation and Activities Guide and Reference Manual, GC30-3008.

Transmission Formats

The host program and the 3276 communicate using half-duplex, flip-flop, send-receive protocols. When the host program or the 3276 program is transmitting data, it assumes the role of the sending Logical Unit (LU). The LU to which the transmission is directed is the receiving LU. [An LU is the logical entity that communicates on behalf of an end user (such as a terminal or application program).] The term *outbound* refers to transmissions from the host to the 3276. The term *inbound* refers to transmissions from the 3276 to the host.

The portions of a transmission between the host and the 3276 that are discussed in this chapter are:

- Request/Response Header (RH). This header describes the type of message being transmitted and contains indicators that control SNA protocols.
- Request/Response Unit (RU). This contains the data or commands that flow in the transmission. (Note that occasional reference is made to a Null RU, that is, an RU that contains no data.)
- Transmission Header (TH). This header contains format identification, mapping fields, and an expedited flow indicator.

The 3276 can communicate with the host system by means of a teleprocessing network that uses the synchronous data link control (SDLC) transmission format. A description of SDLC transmission format is found in the *IBM Synchronous Data Link Control General Information* manual, GA27-3093.

Session Components

Within SNA, communication takes place between LUs. For 3276 operation, the host always contains the Primary Logical Unit (PLU), and the 3276 contains the Secondary Logical Unit (SLU). The 3276 can have from 1 to 8 SLUs (addresses 2 through 9).

A set of logical connections, called *sessions,* is required to control the exchange of data and control information between the host program and a 3276 SLU. At the host system, the access method provides the System Services Control Point (SSCP) function for all sessions that are established with the 3276. The SSCP maintains information that allows a PLU to establish and maintain an LU-LU session with a specific 3276 LU.

SNA Sessions

The sessions that must exist between the host system and the 3276, for an access method application program and the 3276 to exchange information, are as follows:

SSCP-PU [access method - 3276 Physical Unit (PU)]

SSCP-PLU (access method - host program)

SSCP-SLU (access method - 3276 SLU)

PLU-SLU (host program - 3276 SLU) (referred to as LU-LU)

The following paragraphs discuss the sessions individually and identify how they are established and terminated. The SNA commands that establish and terminate the sessions are identified. SNA commands are discussed in detail under the heading "SNA Commands."

SSCP-PU Session

Before establishing the SSCP-PU (access method - 3276 control unit) session, the physical transmission or channel connection to the host must be established.

The SSCP-PU session must be established before establishing the SSCP-SLU or LU-LU sessions. When the access method network operator activates a specific 3276, the access method issues the Activate Physical Unit (ACTPU) command to the control unit. A predefined start procedure for the access method may also request the activation of specific 3276 control units. The SSCP-PU session is the first session established between the host system and the 3276.

The SSCP-PU session is terminated when the access method network operator deactivates the 3276. When all SSCP-LU sessions for the control unit have been terminated, the access method issues the Deactivate Physical Unit (DACTPU) command. When the 3276 returns a positive response to the DACTPU command, the SSCP-PU session is terminated.

Figure 4-3 lists commands that are valid for the SSCP-PU session.

SSCP-Secondary LU Session

When the SSCP-PU session is established, an activate command may be issued to the access method to establish the SSCP-SLU session. The access method will issue an Activate Logical Unit (ACTLU) for the appropriate SLU or SLUs in the 3276. The SSCP-SLU session must be established before establishing the LU-LU session.

The SSCP-SLU session is terminated when the access method sends a Deactivate Logical Unit (DACTLU) command to the specified SLU. When the control unit returns a positive response to the DACTLU command, the SSCP-SLU session is terminated.

Figure 4-3 lists commands that are valid for the SSCP-SLU session.

LU-LU Session

Initiating an LU-LU Session

Three types of LU-LU sessions are supported by the 3276. Further description of these sessions is provided later in this section.

The LU-LU session types are:

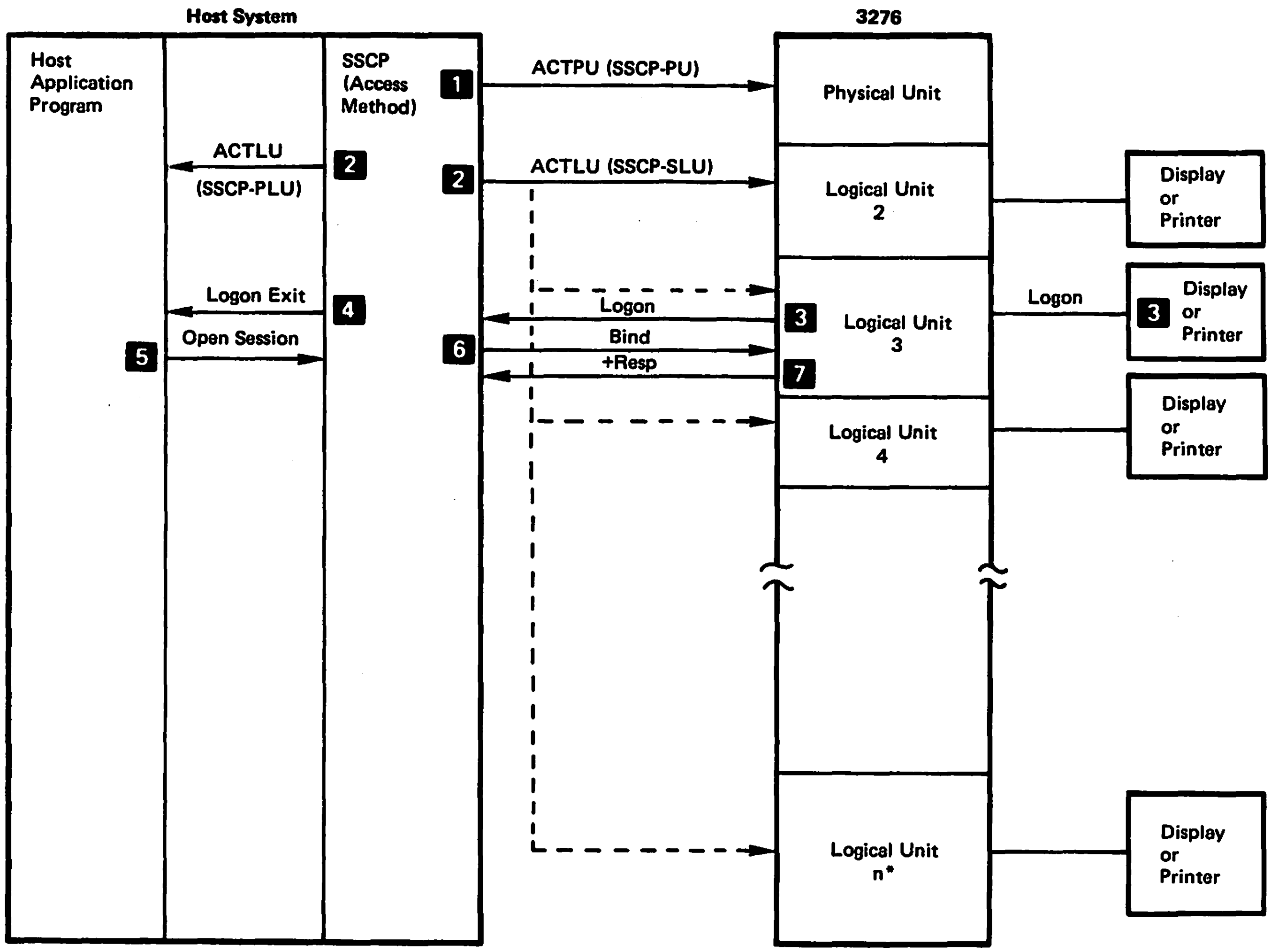
Type 1 – The device attached to the 3276 SLU is a printer, and the data stream is the SNA Character String (SCS).

Type 2 – The device attached to the 3276 SLU is a keyboard/display, and the data stream is in the 3270 data stream compatibility (DSC) mode format.

Type 3 – The device attached to the 3276 SLU is a printer, and the data stream is in the 3270 DSC mode format.

The SNA Bind command is used to differentiate these types of sessions.

The command flow sequence required to establish a session is summarized in Figure 4-1. The command flow nomenclature is generalized, and access method specific macro names are not used. The example assumes that no sessions are active between the host and the 3276. The access method sends the ACTPU command to establish the SSCP-PU session 1. ACTLU commands 2 are then sent to establish SSCP-PLU and SSCP-SLU sessions. The SSCP-PLU session can be established by the host application any time prior to logon. The network is now ready for LU-LU sessions to be established.



*The highest LU number for a 3276 is 9. (Note that LU1 is reserved for the 3276.)

Figure 4-1. Establishing a Session with a 3276

An LU-LU session is started by the host application program when it issues the Bind request. The LU-LU session may be initiated by the host application program (for example, acquiring the terminal or by a simulated logon) or by the display terminal operator 3 (a character-coded logon). If a character-coded logon is received by the access method, the access method translates the logon request and schedules a logon exit 4 for the PLU. After the PLU receives control at the logon exit, or when the PLU acquires a terminal, the PLU passes an open session request to the access method 5 which results in an SNA Bind 6 being passed to the SLU. The 3276 LU examines the session parameters of the Bind and, if they are acceptable, allows the session to be established by sending a positive response 7 to the Bind command. If the session parameters are not acceptable, the 3276 LU rejects the Bind command by returning a negative response, indicating that the session parameters are invalid (sense code X'0821'). Also, if power is not on at the device, a negative sense code X'080A' or X'0845' is returned to the Bind. Figure 4-4 identifies the bind parameters that can be specified for 3276 sessions.

After the Bind command has been accepted with a positive response, the host program can issue the Start Data Traffic command to allow data traffic to flow for the session.

The manner in which an LU-LU session may be initiated depends on the type of session being started. A type 1 or type 3 session must be initiated by the PLU. A type 2 session may be initiated by either the PLU or SLU.

3276 Attachment to a 3790

When the 3276 is attached to a 3790, the 3790 provides the services otherwise provided by the host access method. The logon message from the terminal operator is intercepted by the 3790 and examined to determine whether the session is to be established with the 3790 itself or with an application program in a host that is communicating with the 3790.

Terminating an LU-LU Session

The PLU can terminate an LU-LU session by requesting that the SSCP close the session. The SSCP then sends the Unbind command to the secondary LU and the LU-LU session is terminated.

Type 2 sessions can also be terminated by the display operator in either of two ways. The first method is to notify the PLU (where supported), on the LU-LU session, that termination is desired; the PLU then terminates the session. In the second method, the display operator changes from an LU-LU session to an SSCP-SLU session by use of the System Request (SYS REQ) key and enters a logoff message. The SSCP then passes the logoff request to the PLU, if the logoff message is conditional, or issues the Unbind for the PLU if the logoff message is unconditional. When the 3276 is attached to the 3790, all logoff requests are treated as unconditional.

A PLU may close the session in an orderly fashion by issuing a Shutdown command. When the host program issues the Shutdown command, the 3276 returns the Shutdown Complete command after completing any outstanding operation and entering the Between Bracket state. Note that the PLU must close a bracket with end bracket before the Shutdown command is effective.

The 3276 terminals support FID2 transmission headers (TH). The transmission header consists of 6 bytes:

TH0:	FID (Bits 0-3)	Format Identification
	MPF (Bits 4,5)	Mapping Field
	RES (Bit 6)	Reserved
	EFI (Bit 7)	Expedited Flow Indicator
TH1:	RES (Bits 0-7)	Reserved
TH2:	DAF' (Bits 0-7)	Destination Address Field (See Figure 4-2 and "Device Addressing" in Chapter 1)
TH3:	OAF' (Bits 0-7)	Origin Address Field
TH4,5:		Sequence Number on Normal, ID Number on expedited flow requests and responses

Device Number	Device Address Field Bits: 0	1	2	3	4	5	6	7
PU	0	0	0	0	0	0	0	0
**	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1	0
1	0	0	0	0	0	0	1	1
2	0	0	0	0	0	1	0	0
3	0	0	0	0	0	1	0	1
4	0	0	0	0	0	1	1	0
5	0	0	0	0	0	1	1	1
6	0	0	0	0	1	0	0	0
7	0	0	0	0	1	0	0	1

**Address reserved.

Figure 4-2. Device Addressing for SNA Terminals

The 3276 handles transmission headers received on outbound requests as follows:

1. All reserved parameters are ignored on requests.
2. MFP – The 3276 supports outbound segmenting for FM data.
3. EFI – The expedited flow indicator identifies normal (0) or expedited (1) flow requests.

EFI=1

The 3276 supports the following requests as outbound expedited flow requests:

RU Category	Request
SC	ACTPU, DACTPU, ACTLU, DACTLU, BIND, UNBIND, CLEAR, SDT
NC	Not supported
DFC	SIGNAL, SHUTDOWN
FMD	Not supported

When the 3276 receives any requests listed above with correct categories and EFI=1, they will be passed through for further processing. When the 3276 receives any requests listed above with incorrect categories and EFI=1 or any requests not listed above with EFI=1, it will reject them with the negative response sense code X'1003'. The 3276 handles all expedited flow requests as if they have ONLY-IN-CHAIN, NO BRACKET, NO CD, NO QRI, and NO PACING.

EFI=0

The 3276 supports the following requests as outbound normal flow requests:

RU Category	Normal Request
SC	Not supported
NC	Not supported
DFC	CANCEL, BID, CHASE
FMD on PLU-SLU	Any request
FMD on SSCP-SLU	Any in SCS format
FMD on SSCP-SPU	REQMS

When the 3276 receives any of the requests listed above associated with the correct categories and EFI=0, they will be passed through for further processing. When the 3276 receives any requests listed above with incorrect categories and EFI=0 or any requests not listed above with EFI=0, they will be rejected with the negative response code X'1003'.

SNA Commands

SNA commands define a set of controls to establish and terminate sessions, and to assist in the management of host-to-3276 data flow and sessions. Three types of SNA commands are discussed:

- Session Control (SC) commands – These commands establish and terminate sessions in the network.
- Data Flow Control (DFC) commands – These commands control the flow of data in an LU-LU session.
- Function Management Data (FMD) command – This command is used to transfer data in the LU-LU session.

Commands Supported

The SNA commands supported by the 3276 are listed in Figure 4-3.

SNA Command							
Name	Type	SSCP →	PU ←	SSCP →	SLU ←	PLU →	SLU ←
ACTPU	SC	X					
DACTPU	SC	X					
ACTLU	SC			X			
DACTLU	SC			X			
BIND	SC					X	
UNBIND	SC					X	
SDT	SC					X	
CLEAR	SC					X	
CANCEL	DFC					X	X
CHASE	DFC					X	
LUSTAT	DFC						X
SHUTD	DFC					X	
SHUTC	DFC						X
RTR	DFC						X[1]
BID	DFC					X	
SIGNAL	DFC					X	X[2]
DATA	FMD					X	
REQMS	FMD	X					
RECFMS	FMD		X				

[1] Only SLU types 1 and 3
[2] Only SLU types 1 and 2

Figure 4-3. SNA Commands Supported by the 3276

Command Description

Activate Physical Unit (ACTPU)

The ACTPU command is sent by the access method to establish the SSCP-PU session with a 3276 control unit. The SSCP-PU session is established when the 3276 returns a positive response to the ACTPU command.

The ACTPU command can be transmitted when the SSCP-SLU and LU-LU sessions are active; for example, when an NCP restart procedure occurs. When the 3276 receives the ACTPU command, all active sessions are terminated immediately. The 3276 returns a positive response to the ACTPU command, and the SSCP-PU session is reestablished.

Deactivate Physical Unit (DACTPU)

When the 3276 receives the DACTPU command, all LU-LU and SSCP-SLU sessions and the SSCP-PU session are terminated. If a command other than ACTPU is received after a positive response has been returned for the DACTPU command, the 3276 returns a negative response with sense data indicating PU not active (sense code X'8008').

Activate Logical Unit (ACTLU)

The ACTLU command is sent by the access method to establish the SSCP-SLU session with each 3276 control unit LU. The SSCP-SLU session is established when the 3276 returns a positive response to the ACTLU command. The SSCP-PU session must be established prior to the receipt of ACTLU to allow the 3276 to return a positive response to this command. If the 3276 receives a command other than ACTPU, ACTLU, DACTPU, or DACTLU before the SSCP-LU session is established, a negative response is returned with sense data indicating LU not active (sense code X'8009'). Note that the SLU is in the 3276 and that the session can be activated without a display or printer being powered on or attached.

When an SSCP-SLU session has been previously established and the 3276 receives an ACTLU command for that LU, any active session between that LU and a host program is terminated. The 3276 returns a positive response to the ACTLU command, and the SSCP-LU session is reestablished.

Deactivate Logical Unit (DACTLU)

Receipt of this command terminates the SSCP-SLU session. If an LU-LU session is established when the DACTLU command is received, the session is terminated. When the 3276 receives a command other than DACTPU, ACTPU, or ACTLU after a positive response has been returned for the DACTLU command, a negative response is returned with sense data indicating SLU not active (sense code X'8009').

Bind

This command is sent by the access method to request an LU-LU session between an application program and a 3276 SLU. The 3276 returns a positive response to establish the LU-LU session. When the session cannot be established, the 3276 returns a negative response with sense data that describes the reason the session was rejected.

The 3276 examines session parameters that are received with the Bind command. The values required depend on the type of session established. Figure 4-4 provides a detailed description of the session parameters that are sent with the Bind command.

When the SSCP-SLU session is established and the 3276 receives a command that flows in the LU-LU session, other than Bind, a negative response is returned with sense data indicating no session established (sense code X'8005').

If the device attached does not have power on or is physically detached from the 3276 cable port, a negative response is returned with sense data indicating power off (sense code X'080A').

When an LU-LU session exists, that is, one Bind has been accepted, and the 3276 receives a subsequent Bind command for the LU, a negative response is returned with sense data indicating session already exists (sense code X'0815') if the Bind sender address is the same as the session already found. A negative response indicating function active (sense code X'0805') is returned if the Bind sender address differs from the session already found.

Byte	Hex Value	Bit Setting	Meaning
0	31		Identifies this RU as a Bind command.
1	01		Bind type and format. The only Bind type supported is Hex 01.
2	03		Function management (FM) profile. Specifies that the data flow control commands and the request/response protocols that are to be used for this session conform to FM Profile 3.
3	03		Transmission services (TS) profile. **Specifies that the 3276 conforms to TS Profile 3, that** is, pacing and sequence numbers are used with normal flow transmission and that data traffic is controlled by the Clear and Start Data Traffic commands.
4			Primary LU Protocols.
		X...	Chaining use: 0 The PLU can send only single-element chains. 1 The PLU can send single- or multiple-element chains.
		.X..	Request mode selection: 0 Immediate request mode is used. Only one definite response can be outstanding at a time. That response must be received before the PLU can send another RU.
		..XX	Chaining responses: 01 The PLU can only request exception-only responses. 10 The PLU can only request definite responses. 11 The PLU can request definite or exception-only responses.
		.... 00..	Reserved.
		.... ..X.	Compression indicator: 0 Must be 0.
		X	Send End Bracket Indicator (EB): 1 The PLU can send the EB.
5			Secondary LU Protocols.
		X...	Chaining Use: 0 **The 3276 can send only single-element chains.** 1 **The 3276 can send single-or multiple-element chains.** **Note:** *0 or 1 for LU type 1 or 3.* *1 for LU type 2.*
		.X..	Request mode selection: 0 Immediate request mode is used. **The 3276 can issue a request for a single definite response. No further transmissions are sent until the 3276 receives the requested response.**
		..XX	Chaining responses: 01 **The 3276 can only request exception-only resposes.** 10 **The 3276 can only request definite responses.** 11 **The 3276 can request only exception-only respose.**
		.... 00..	Reserved.

Figure 4-4 (Part 1 of 3). Bind Command Session Parameters

Byte	Hex Value	Bit Setting	Meaning
		.... ..X.	Compression indicator:
			0 The 3276 cannot send compressed data.
		X	Send End Bracket indicator (EB).
			0 The 3276 cannot send the EB.
6			Common Protocols.
		0...	Reserved.
		.X..	Function management (FM) header usage:
			0 The PLU and the 3276 cannot exchange FM headers.
		..X.	Brackets usage:
			1 Bracketed session is used. Both the PLU and the 3276 must use bracket protocols.
		...X	Bracket termination protocol:
			1 Bracket termination rule 1 is used (refer to "Bracket Protocol" for a description of bracket termination rule 1).
		.... X...	Alternate Code selection:
			0 Both the PLU and the 3276 must use EBCDIC.
			1 Both the host program and the 3276 can use an alternate code. An example of an alternate code is ASCII.
		.... .000	Reserved.
7			Common Protocols.
		XX..	Normal Flow Send/Receive mode (selection):
			10 This session uses half-duplex, flip-flop (HDX FF) transmissions. Refer to "Session Processing States."
		..X.	Recovery responsibility:
			0 The PLU is responsible for error recovery.
		...X	Brackets first speaker:
			0 The 3276 is always the first speaker.
		.... 000.	Reserved.
		X	Contention resolution:
			0 Contention (simultaneous transmissions from the host program and the 3276) is resolved in favor of the 3276.
8		00xx xxxx	Secondary-to-primary LU pacing count. If set to zeros, pacing is not used.
9		00xx xxxx	The primary-to-secondary pacing value defines the number of RUs that may be received by the 3276 before a pacing response must be returned to indicate readiness for another block of RUs. If set to zeros, pacing is not used. See "Pacing" for recommendations of pacing values.
10	XX		Maximum RU size sent by the secondary LU. This value represents the largest RU that can be sent by the 3276. It is expressed as a mantissa (8 through F) and an exponent value of 2 by which the mantissa is multiplied. For example, when the mantissa is specified as 8 and the exponent of 2 is 5 (hex 85), the RU size represented is 256 bytes. Examples of mantissa and exponent values used by the 3276 are shown below with the RU size they represent: 85=256 86=512 C6=768 87=1024 A7=1280 C7=1536 E7=1792 88=2048

Figure 4-4 (Part 2 of 3). Bind Command Session Parameters

Byte	Hex Value	Bit Setting	Meaning
			See "RU Lengths Supported" for detailed information about values supported by the 3276.
11	XX		Maximum RU size sent by the primary LU. This value represents the largest RU that can be sent by the PLU and is specified in the same format as for the secondary LU (byte 10). See "RU Lengths Supported" for detailed information about values supported by the 3276.
12, 13	0000		Reserved.
For SLU Type 1:			
14	01		Type 1 print function using SCS data stream.
15-17	00		Reserved.
18	E1		Sent but not checked by the 3276 for LU type 1.
19	00		Reserved.
20-24			Not supported for LU type 1.
For SLU Types 2 and 3:			
14	02		Type 2 3270 data stream compatibility mode.
14	03		Type 3 3270 print function using 3270 data stream.
15-19	00		Reserved.
20-24	XX		Refer to Figure 1-8 for LU type 2. Refer to Figure 1-9 for LU type 3.
For all SLU Types:			
25+			Reserved.

Figure 4-4 (Part 3 of 3). Bind Command Session Parameters

Session parameters included in the Bind command RU define the protocols that govern the session. Figure 4-4 describes contents of a Bind command RU that are supported by the 3276 and explains how the session parameters are used. A generalized setting for the access method logmode table is listed under "Bind Default" later in this chapter. Also listed (under "Bind Check") are the checks that the 3276 makes when the Bind command is received. Specific customer optimization or device features may require changes for each installation.

Also listed in this chapter (under "Logical Unit Status") are the checks made by the 3276 for each logical unit type. Failure to properly specify the required session parameters results in rejection of the Bind command by the control unit because the session parameters are invalid (sense code X'0821').

Unbind

Receipt of this command directs the 3276 to terminate the LU-LU session between a host program and a 3276 SLU. The LU-LU session is terminated when the 3276 returns a positive response to the Unbind command.

Clear

Receipt of the Clear command causes the 3276 to enforce the data-traffic-reset state upon the LU-LU session. Clear also causes the 3276 to initialize all inbound and outbound transmission buffers. When data-traffic-reset state is activated for an LU-LU session, only the following commands are valid for that session: Clear, Unbind, and Start Data Traffic (SDT).

Start Data Traffic (SDT)

This command allows data traffic to flow during an LU-LU session. The SDT command must be issued after a Bind command has established the LU-LU session. It is also sent after Clear to complete a session resynchronization sequence with the 3276. SDT is valid only when the data-traffic-reset state is active for an LU-LU session.

To complete a session resynchronization sequence, the host program must request transmission of the SDT command from the access method.

Cancel

When received, normal SNA usage of this command directs the receiver to discard all elements of the chained transmission being received. However, the 3276 processes data RUs to the display or printer as they are received without waiting until end-of-chain. Therefore, the Cancel command serves the purpose of providing a proper termination for an otherwise incomplete chain. A Cancel command received between chains only affects the 3276 state controlled by the change direction (CD) and end bracket (EB) bit settings carried in the RH with the Cancel command. Processing of a chained transmission is terminated when the Cancel command is received. EB or CD may be sent with the command.

When a chained transmission is in progress, and the 3276 returns a negative response to an element of that chain, the PLU should terminate that chained transmission and issue the Cancel command if the last chain element has not already been sent to the 3276.

When sent by the 3276 type 2 SLU, the Cancel command directs the PLU to stop processing a chained transmission and to discard all elements of the chain that are currently being received. The Cancel command is substituted for the end of the chain if a 3178, 3278, or 3279 failure or operator action prevents transfer of all data from the display to the 3276.

When the PLU returns a negative response for an element of a chain, the following will happen:

- For a 3276 when inbound pacing is not used, the entire chain will be transmitted before the PLU response is examined. Cancel will not be sent.
- For a 3276 when inbound pacing is used, the negative response from the PLU will be examined only if the 3276 must look for a pacing response. If the negative response is examined, the 3276 will send Cancel and will not transmit the remaining elements in the chain. If the negative response is not examined, the entire chain will be transmitted and Cancel will not be sent.

In either case, the PLU should discard all elements of a chained transmission after sending a negative response.

Chase

Chase is used to confirm that all preceding requests have passed through the network and have been processed. When this command is received, the 3276 returns a positive response to the PLU, indicating all previous chains have been processed.

The PLU should complete or cancel the current chained transmission before issuing the Chase command. When a chained transmission is sent with exception-only responses requested, the Chase command can be used to verify that all responses for that chain have been received. The EB or CD indicators can be issued with the Chase command.

Bid

The Bid command is sent by the PLU to a 3276 SLU to request permission to begin a bracket. The use of Bid avoids long chains of data using transmission time and then being discarded because SLU won bracket contention. If the Bid is accepted by the SLU, a positive response is returned and the SLU goes to begin-bracket-pending state and waits for the request containing BB.

A 3276 SLU can reject a Bid command by winning bracket contention for the following reasons:

1. LU Type 2
   - The 3276 is already In Bracket (INB) and a PLU protocol error exists. The sense code returned is X'0813'.
   - The operator has initiated an inbound data stream carrying Begin Bracket (BB). The sense code returned is X'0813'.
   - An operator has started to enter data on the screen but has not initiated an inbound data stream. The sense code returned is X'081B'.
2. LU Type 1 or 3
   - The SLU is already INB and a host program protocol error exists. The sense code is X'0813'.
   - A printer attached to the 3276 is busy doing a local copy operation. The sense code returned is X'0814'. The 3276 will send the Ready to Receive (RTR) command to the host program when the printer becomes not-busy and a BB can be accepted by the secondary LU.

Signal

The PLU can send the Signal command to the 3276 SLU to request the Change Direction (CD) indicator. The SLU will complete any chained transmissions that are in progress and send the CD to the PLU. A request with CD but no data (a Null-RU) will be sent if the SLU is in send state but has not started transmitting. If the SLU is already in receive state, BETB, or ERP1 state (see "Session States"), the Signal is positively responded to but no SLU action is taken.

The 3276 will send the Signal command (X'00010000') when the terminal operator presses the keyboard ATTN key or, for an LU type 1, either of the printer PA switches. The command is expedited and has no effect on SLU states. Once Signal has been sent by an SLU, pressing the ATTN or PA keys will not cause a second Signal until the 3276 has received a response to the first Signal.

LU Status (LUSTAT)

The 3276 SLU sends the LUSTAT command to notify the PLU that a processing error has been detected or that a change in the operational status of a device has occurred. A 4-byte status code is sent by the 3276 SLU to describe the error condition or the device status change.

For LUSTAT codes and conditions that determine which LUSTAT is sent, refer to "Logical Unit Status" later in this chapter.

Ready to Receive (RTR)

A 3276 type 1 or 3 SLU sends this command to indicate when a previously rejected bracket (with sense code X'0814') can be initiated by the host program. The RTR command is allowed only when the session is ready to receive new bracket.

When the RTR command is sent and a positive response is received from the host program, the printer LU enters begin-bracket-pending state and expects the host program to begin a bracket.

REQMS

The Request Maintenance Statistics (REQMS) command is sent by the SSCP to a 3276 when the Network Determination Aid Processor (NDAP) requests PU performance statistics. Four types of requests can be made:

- Type 1 – Link Test Statistics
- Type 2 – Summary Counters
- Type 3 – Communication Adapter Data Error Counts
- Type 5 – 3276 Machine Level Information

The state of the RESET/NO RESET indicator in the REQMS request determines whether or not the log area where the transmitted maintenance statistics are stored is cleared.

An REQMS request that cannot be executed by the 3276 is rejected with a negative response; an accepted REQMS request receives a positive response and the requested statistics (formatted as RECFMS) as an inbound message.

RECFMS

Record Formatted Maintenance Statistics (RECFMS) is sent by the 3276 to the SSCP in response to an REQMS command (the 3276 will not send unsolicited RECFMS requests to the host). The RECFMS maintenance statistics are recorded at the host by the Network Communications Control Facility (NCCF).

When the 3276 accepts an REQMS request, it transmits the maintenance statistics requested. If the REQMS specified "RESET", the error log area referenced by the REQMS is reset by the 3276 after the 3276 receives a positive response to the RECFMS; otherwise, the error log area is not reset.

For descriptions of the RECFMS responses, refer to Appendix G.

Shutdown

The PLU sends the Shutdown command. Receipt of this command directs the 3276 SLU to prepare for a session termination sequence. The 3276 returns a positive response to the PLU, but data-transfer sequences are not inhibited.

The Shutdown command causes the session to enter shutdown-complete-pending state. The pending state is maintained until the SLU completes normal flow processing and goes between bracket (BETB). The SLU then sends the Shutdown Complete command to the PLU.

Shutdown Complete

This command is sent by the 3276 after the Shutdown command has been received from the host program and an End Bracket has caused the SLU to go to BETB state.

When the Shutdown Complete command is sent to the PLU, the session enters shutdown state. When shutdown state is active, no data transmissions can be sent to the PLU; the PLU, however, may continue to send data to the 3276.

The PLU may either terminate the session using Unbind when the Shutdown Complete command is received from the 3276, or use Shutdown as a means of quiescing traffic. Exit from Shutdown Complete requires a Clear and SDT if the command is used as a quiesce function.

FM Data

This command is used to transfer data in the LU-LU session or in SSCP-LU session. It may only be sent in LU-LU session when data traffic is allowed (SDT has been issued and received a positive response).

When communicating with a 3276 SLU, the following FM data protocols are used:

Bracket: Bracket protocol is used to delimit a series of related inbound and outbound FM data request units (RUs); for example, all the RUs required to complete a transaction.

Chaining: Chaining logically connects one or more RUs from a single LU; for example, all RUs required to complete a display image

Change Direction: Change direction informs the receiving LU that the sending LU has completed transmission and expects the next transmission to be from the receiving LU; for example, the PLU has transmitted a complete form image and expects the next transmission to be from the display operator when the blank fields in the form image are filled in.

Bracket Protocol. The 3276 provides a bracket protocol to delimit a series of related inbound and outbound requests. A bracket may consist of one input and one output, many sets of inputs and outputs, or a series of requests flowing in a single direction. The Begin Bracket (BB) and End Bracket (EB) indicators are used to delimit a bracket. References are made to bracket states (BETB and INB); these states are described under "Bracket States."

A bracket is initiated when the Begin Bracket (BB) indicator is accepted by the primary or secondary LU. The bracket is usually ended when the End Bracket indicator (EB) is received by the secondary LU. The specific conditions that end a bracket are defined by SNA bracket termination rule 1 (see below). Two commands, Bid and Ready to Receive (RTR), are implemented to further define the initiation of a bracketed session. These commands are described under "SNA Commands."

The following protocols apply for 3276 bracket processing.

For sessions with type 2 SLUs, the SLU may begin a bracket any time the session is between brackets. The PLU may request permission to begin a bracket using Bid. If the SLU returns a positive response, the PLU may begin a bracket. If the SLU returns a negative response, the PLU must wait for the next BB from the SLU.

For type 1 and 3 sessions, the PLU may begin a bracket any time the session is between brackets (the only time the SLU will begin a bracket is when the operator presses the PA key). The PLU may start a bracket by sending a transmission that contains BB or by sending Bid, waiting for a positive response, and then sending a transmission that contains BB.

The PLU may attempt to initiate a bracket by simply sending a transmission with BB. If a contention situation exists (the SLU begins a bracket before receiving BB from the PLU), the SLU returns a negative response to the PLU's transmission and then discards all portions of the chain from the PLU. The SLU assumes that its transmission will be accepted by the PLU.

If a Bid or BB from the PLU is rejected, the 3276 will do the following:

- For a session with a type 2 SLU, the SLU sends BB when it next has data to send. The PLU may return its data when it receives Change Direction (CD).
- For a type 1 or 3 session with a 3276, the SLU will only reject the PLU's Bid or BB if the printer is performing a local print function or when a protocol error is detected. When the local print is completed, the SLU will send RTR.

The host program can end a bracket. The 3276 cannot end a bracket.

Bracket protocol establishes the following restrictions on beginning and ending brackets:

1. BB and EB cannot be sent with response RUs.
2. The EB cannot be sent with the Bid or RTR command. All other normal flow DFC commands can end the bracket.
3. All outbound chains that begin a bracket but do not carry EB must be sent with definite response requested.

The 3276 supports bracket termination rule 1 as follows:

1. When EB is received and the last element of a chain required definite response, the 3276 will enter between-bracket (BETB) state from in-bracket (INB) state after +RSP to the chain or stay INB after -RSP.

2. When EB is received and the last element of a chain requires exception response, the 3276 will enter BETB from INB immediately.

The 3276 ignores the BB bit on all outbound requests except FM data, and ignores EB on all outbound requests except FM data and DFC commands Cancel and Chase.

Chaining Protocol Definition. A data chain is a complete unit of data that originates at a single LU. Data RU chaining provides a method of logically defining a complete unit of data regardless of whether the data is transmitted as a single RU or as a series of consecutive RUs. Each RU is associated with only one chain. An individual RU may be the beginning, middle, ending, or only (both beginning and ending) RU in the chain; the chaining indicators, Begin Chain (BC) and End Chain (EC), are contained in the request header. The following are definitions of each type of RU in a chain:

First in Chain (FIC) – Identifies an RU that begins a chained transmission (RH=BC,¬EC).

Middle in Chain (MIC) – Is transmitted with all RUs following the BC transmission, with the exception of the last RU in that chain (RH=¬BC¬EC).

Last in Chain (LIC) – Identifies the RU that completes a chained transmission (RH=EC¬BC).

Only in Chain (OIC) – Both the BC and EC indicators are included to indicate a transmission that consists of a single RU. That RU is termed a single-element chain (RH=BCEC).

A chain is correct if the RUs consist of:

1. FIC, LIC
   or
2. FIC, MIC, . . . , LIC
   or
3. OIC.

Any other sequence of chaining indicators will cause a chaining error.

Chaining Operations. When the 3276 receives a chain with chaining indicators in an improper sequence (for example, FIC, MIC, FIC), a negative response, with sense data indicating a chaining error (sense code X'2002'), is returned to the host program. The 3276 purges the chain, ignoring subsequent elements of the chain until a data RU with the LIC or a Cancel command is received. Receipt of an OIC data RU terminates the purging of a chain; the OIC message is also purged. Sending RUs having chaining indicators in the sequence FIC, MIC, OIC is a violation of chaining protocol. In this case, when the 3276 receives the OIC transmission, the chaining error is detected, the OIC transmission is purged, purging of chain elements is stopped, and a negative response is sent for the OIC transmission. The 3276 is now ready to normally process the next chain.

Change Direction. The 3276 uses a half-duplex, flip-flop (HDX-FF) mode to transfer normal flow data. Only one of the two LUs in the session may send at a given time. The flip-flop protocol demands that, when one LU is sending, the other must be prepared to receive. Therefore, the two states of send and receive (RCV) exist on each end of the session.

A bit in the request header, called the Change Direction (CD) indicator, is used to keep the two end-point LUs in synchronization. Each time an LU accepts this CD in a request, it means it is that LU's turn to send. Each time an LU sends the CD in a request, that LU must then be prepared to receive. The 3276 always sends CD with EC or OC in an FMD RU. Exceptions may occur following negative responses. See "ERP1" state.

Pacing

Inbound and outbound pacing is supported by the 3276. Pacing is used as a tuning parameter for the system. Usage comments are included here; however, control is under the user's discretion at NCP or equivalent definition time.

The pacing count (N) determines the number of normal flow request RUs that can flow before a pacing response is required to allow the next group of N RUs to continue. A special response designated as Isolated Pacing Response (IPR) is used to return the pacing response if a response to the outbound request is not required at the time the pacing response is required. The 3276 will indicate readiness with a pacing response as soon as printer buffers become available after receiving the pacing request. Thus, the number of normal flow RUs allowed in the network due to pacing is up to 2N-1. RUs may vary in length as specified in the Bind parameter.

LU Type 1

For the 3276, device dependencies exist because the printer is slower than the displays. Care must be exercised in the use of pacing and/or definite response protocol so that waiting RUs and/or chains are not stacked in the 3276 link buffers.

Within a chain, the 3276 transfers RUs from the link buffer pool to the printer buffer as they are received. The pacing parameter is then used to ensure that there is adequate printer buffer space so that the link buffer pool does not fill and restrict data flow to the keyboard displays or other printers.

During the transmission of multiple chains, interaction occurs between pacing and the type of response requested. When a definite response is requested, a response for a chain must be received by the PLU before it can send the next chain. When exception response is requested, the PLU may send any number of consecutive chains without waiting for a response. Therefore, a definite response enforces a type of pacing.

When OIC RUs are used that are less than, or equal to, 256 bytes, it is redundant to specify both pacing and definite response; unnecessary network traffic will occur if both are specified. When chains with multiple RUs are used, pacing is necessary even though definite response is requested.

During the transmission of multiple chains, the 3276 waits for each chain to be processed by the terminal before removing the next chain from the link buffer pool. Therefore, while OIC RUs of 256 bytes or less may be acceptable (based on the available buffer capacity), the link buffer pool may be depleted and data flow to the keyboard displays restricted if the pacing count is greater than one and exception response is used. The pacing count should not be greater than two or three; one is recommended.

If 3276 SLU type 1 receives more normal flow requests than it is guaranteed by using the outbound pacing mechanism, and the printer buffer does not have enough space left to store the outbound data, a -RSP using sense code X'0801' will be returned. The 3276 will respond to the RU causing the overrun.

LU Type 2 and 3

For LU type 2, the 3276 will generally operate faster than the link, and pacing is not required for the controllers.

For LU type 3, the definite response required when the WCC Start Print bit is set is an effective alternative to pacing.

In telecommunication networks where RUs are processed through more than one communication controller (for example, a 3704 and a 3790 or two 3705s), outbound pacing may be required for type 2 and 3 LUs to prevent data traffic congestion in these controllers.

Inbound pacing is supported by the 3276. Usage in a tree-structured network may not be required. Usage in large telecommunication networks may require inbound pacing to prevent congestion at communication controllers in the network. If a 3276 is attached to a 3790, refer to 3790 documentation for detailed information about inbound pacing support.

SNA Responses

The RH contains indicators that describe the type of response given: Definite Response 1 (DR1) or Definite Response 2 (DR2). The RH also contains an Exception Response (EX) indication that is used when describing the response protocol. Definite response protocol (DR1¬EX or DR2¬EX) specifies that a response, either positive or negative, must be given. Exception response protocol (DR1 EX or DR2 EX) specifies that only a negative response may, or need be, returned.

The only definite response type requested by the 3276 is Definite Response 1 (DR1). The response protocol requested by the 3276 (definite response and/or exception response) is defined in the Bind.

The 3276 will respond to a message from the host with any requested response type (DR1, DR2, or both). The 3276 supports definite response or exception response protocols.

No distinction is made (within this chapter) between the specific response types. The term "positive response" indicates successful receipt of a command or data RU. The term "negative response" indicates that the receiving LU detected an error, which is reported to the sending LU.

Summary of SNA Commands

Figure 4-5 summarizes the validity of SNA commands received by the 3276 relative to the sessions (SSCP-PU, SSCP-LU, and LU-LU) to two LU-LU session processing states (Data Traffic Reset and In Brackets). Figure 4-6 shows the same for SNA commands sent by the 3276.

				LU-LU Session Processing States			
				Data Traffic Reset		In Bracket	
SNA Command Received	SSCP-PU Session Active	SSCP-LU Session Active	LU-LU Session Active	On	Off	On	Off
ACTLU	R	E	T				
ACTPU	E	T	T				
DACTLU	R	T	T				
DACTPU	R, T	T	T				
BIND			E, I	X			X
UNBIND			R, T				
CANCEL			R		R		
CHASE			R			R	
CLEAR			R	X			X
SDT			R		X		
SIGNAL			R		R		
SHUTDOWN			R		R		
FM DATA			R		R	R	
REQMS	R						

Legend:
R – Required state for this command to be valid.
I – Command invalid if in this processing state.
E – Command establishes this session.
T – Command terminates this session.
X – Command sets the processing state to the indicated status.

Figure 4-5. Summary of SNA Commands Received

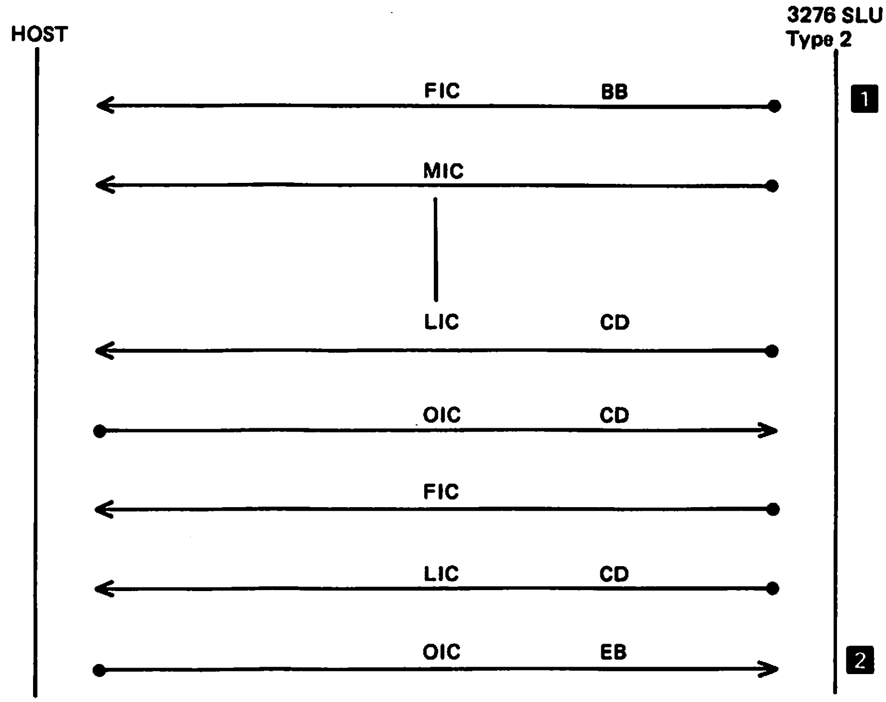
SNA Command Sent	SSCP-PU Session Active	SSCP-LU Session Active	LU-LU Session Active	LU-LU Session Processing States			
				Data Traffic Reset		In Bracket	
				On	Off	On	Off
LUSTAT			R		R		
SIGNAL			R		R		
CANCEL			R		R	R	
READY TO REC.			R		R		R
SHUTDOWN COMPLETE			R		R		R
FM DATA			R		R	R	
RECFMS	R						

Legend:
R — Required state for this command to be valid.

Figure 4-6. Summary of SNA Commands Sent

Sample SNA Command Sequences

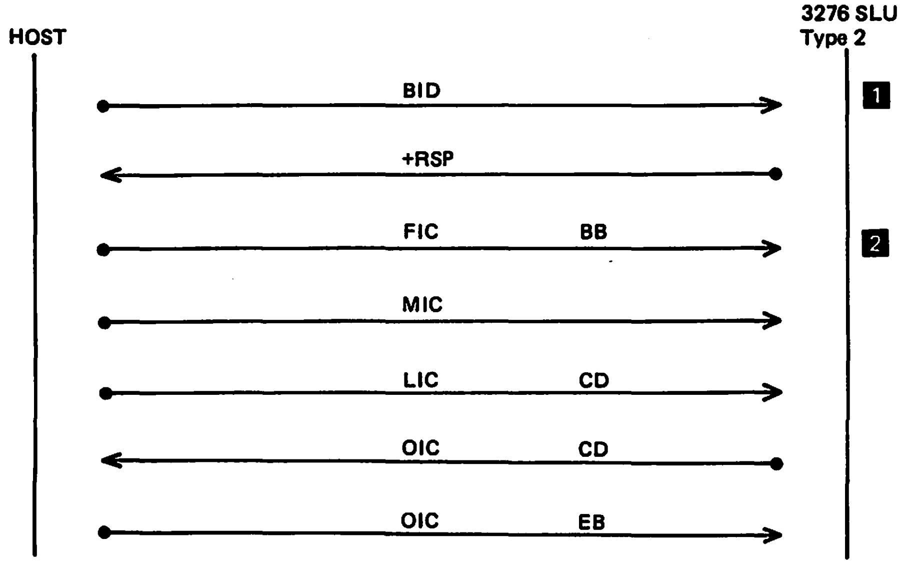
Figures 4-7 through 4-13 illustrate the use of SNA commands. Responses to commands are not shown unless the response is a necessary part of the example.



1 Initial conditions: Session established and both ends in contention-between-bracket state. SLU type 2 initiates a bracket and sends a chain as a result, for example, of Enter key depression.

2 After the required exchange of chains is completed, the host ends the 'unit of work' by sending EB (an LU type 2 cannot send EB). The EB chain may contain data: for example, a write to the screen; or it may be a Null RU chain, that is, only RHs.

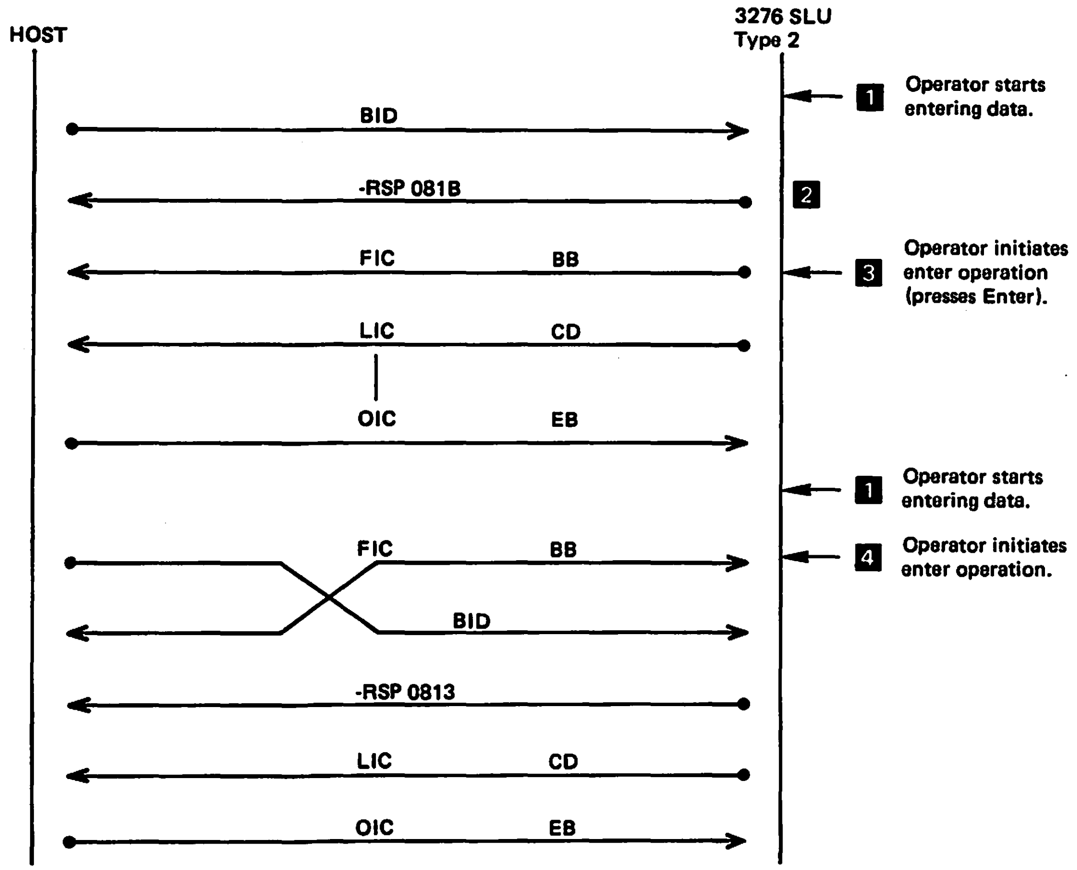
Figure 4-7. Bracket/Chain – LU Type 2 Initiated (without Contention)



1 Initial conditions: Session established and both ends in contention between bracket state. Host sends Bid to indicate intention to begin a bracket.

2 The +RSP was SLU type 2, go ahead to the host. The host initiated the 'unit of work' with BB. **Note:** *the host has the option of going directly to 2, that is, skipping the Bid. However, there is a possibility of Bid rejection, which would result in resending the data associated with 2.*

Figure 4-8. Bracket/Chain – Host Initiated (without Contention)



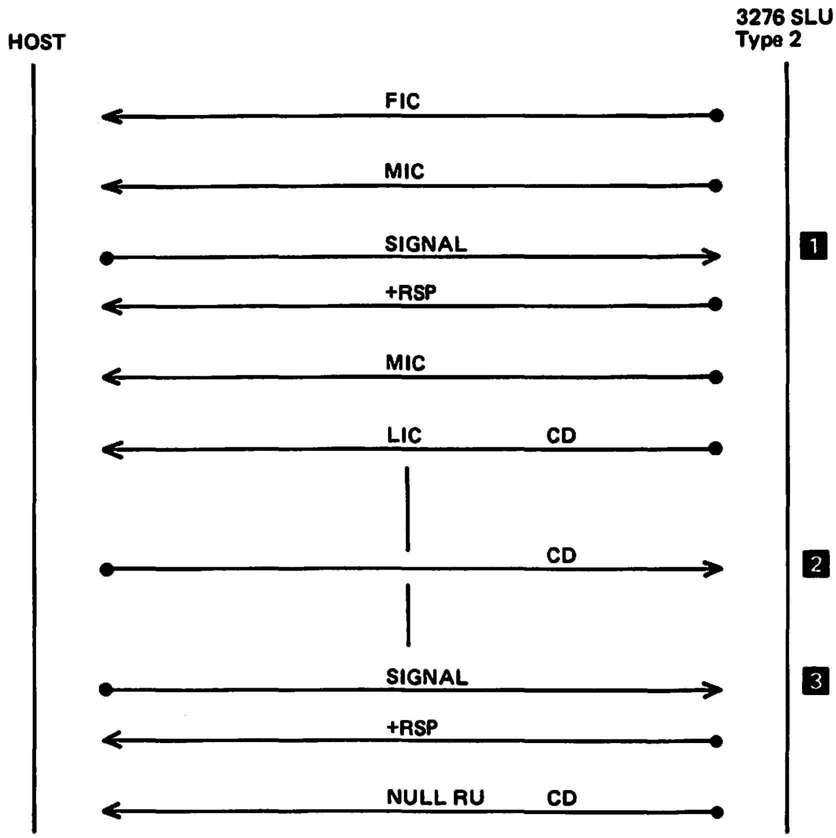
1 Initial conditions: Session established and both ends are in between-bracket state.
The first operator keystroke puts the type 2 SLU in the send (but not transmitting) state.
The type 2 SLU remains in BETB state.

2 The type 2 SLU will reject a Bid (or BB) with 081B. Receiver in transmit mode.

3 The operator initiates an enter operation; for example, he presses the ENTER key.
The type 2 SLU begins a bracket and transmits the operator-entered data.

4 When the operator presses the ENTER key, type 2 SLU goes to in-bracket (INB) state.
Type 2 SLU begins a bracket and starts sending data. The host end has sent a Bid (or BB) before the type 2 SLU first chain element was received. The type 2 SLU rejects the Bid (or BB) with 0813.
The sense code differs from reference 2 because the bracket check is made before the HDX state check.
In reference 2, the bracket check was good.

Figure 4-9. Bracket/Chain – Host/SLU Contention

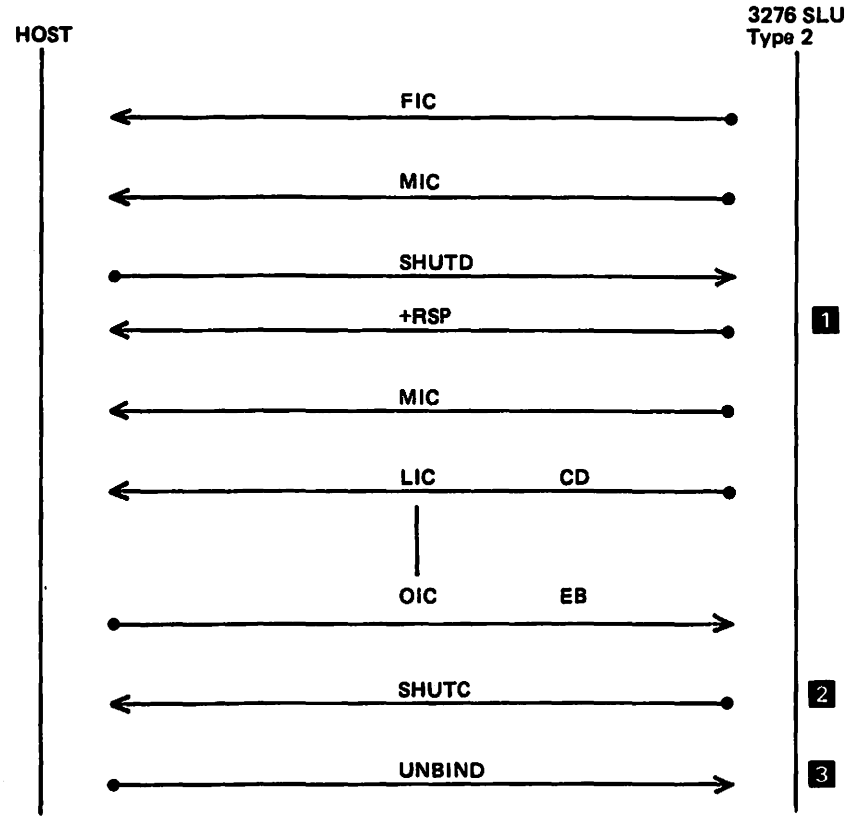


1 The SLU type 2 receives Signal while sending. The +RSP is returned to acknowledge receipt of Signal. The Signal is effectively treated as a NO-OP, and the SLU completes sending of the chain. The SLU type 2 always sends CD with the end of a data chain.

2 CD allows the SLU to send. The operator starts keying in data.

3 Before the operator initiates sending of data, for example, presses the ENTER key, the host sends Signal. The SLU sends +RSP to Signal, locks the keyboard, and sends CD.

Figure 4-10. Signal from Host

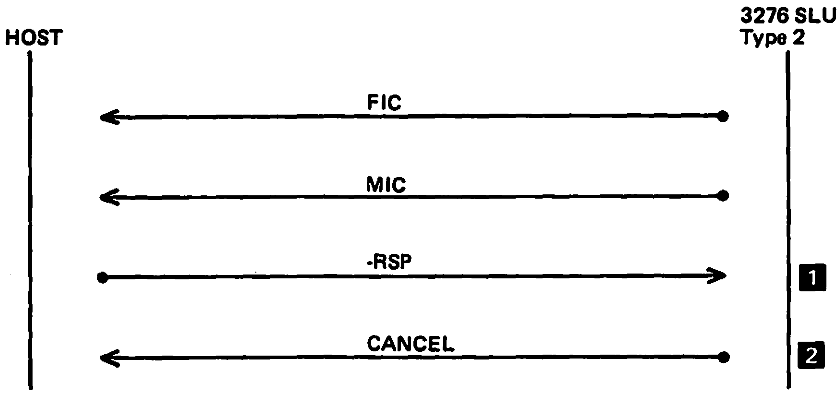


1 The SLU type 2 is alerted that the host wants to shut down. However, a synchronizing EB must be received before effecting shutdown.

2 The SLU goes into shutdown; that is, inbound normal flow (including Signal) is inhibited.

3 The host terminates the session. (**Note:** *The host could clear the condition and continue by sending Clear, SDT instead of terminating the session.)*

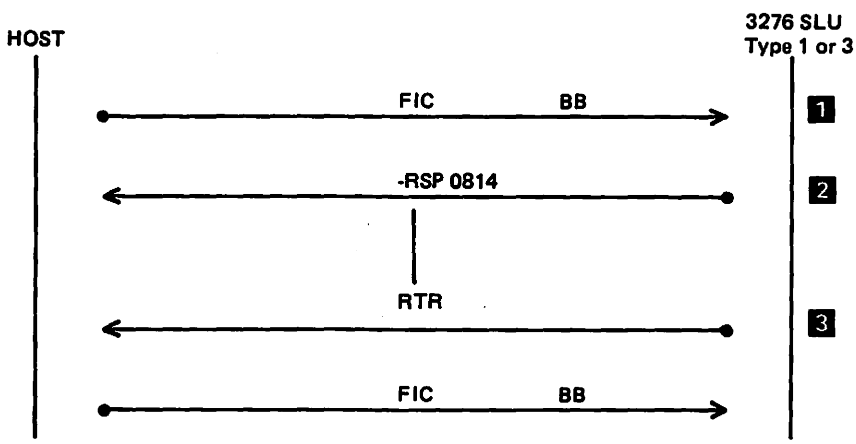
Figure 4-11. Shutdown/Shutdown Complete



1 *The type* 2 SLU receives-RSP to a chain element. **Note:** *Normally, the 3274 or 3276 will not examine any response until the entire chain has been sent and will therefore not send Cancel as the result of receiving a -RSP. However, when inbound pacing is in effect, responses are examined when the SLU must receive a pacing response before continuing transmission. A-RSP will then be detected and cause Cancel to be sent.*

2 The type 2 SLU sends Cancel to direct the host to discard the chain elements already received. The SLU goes to receive state, waiting for host recovery action.

Figure 4-12. CANCEL, SLU Type 2 Sending



1 The printer associated with the SLU type 1 or 3 is not available because a local copy is being done. Consequently, the SLU type 1 or 3 cannot honor the host BB (or Bid).

2 The SLU type 1 or 3 rejects BB (or Bid) with -RSP X'0814' (Bracket Reject, RTR to follow).

3 The printer becomes available, and SLU type 1 or 3 send RTR to indicate to the host that a bracket may be started.

Figure 4-13. RTR – LU Type 1 or LU Type 3 Send

Session Processing States

The 3276 controls the processing of SNA commands, responses, and user data transmissions with a set of session states. Some of these states are defined by SNA and others are unique 3276 definitions that cause SNA state transitions. When the 3276 receives the Clear or Bind command, all 3276 session states are reset.

This section describes the processing states used by the 3276. When several states relate to a common processing function such as bracket or chain processing, they are described under a common heading. The remaining processing states are described individually.

Data Traffic (Reset/Active) State

Reset of all SNA LU-LU states in the 3276 is assured by entering Data Traffic Reset state. This state is entered when a Bind or Clear command is received from the PLU. When Data Traffic Reset state is turned off by SDT, the state is referred to as *Data Traffic Active.*

When in Data Traffic Reset state for any LU-LU session, the 3276 SLU cannot transmit data or commands to the host program. The host can send only session-recovery and session-termination commands when in this state. The 3276 accepts only data RUs for an LU-LU session during Data Traffic Active state.

When in Data Traffic Reset state and a data RU or a command other than SDT or Unbind is received from the host program, the 3276 returns a negative response with system sense data indicating that data traffic is inactive (sense code X'2005'). No other state, except Contention, can exist when the SLU is in Data Traffic Reset state.

Contention (CONT) State

The Contention state on the LU-LU session exists only between brackets. In this state, the LU resources are not allocated. All associated I/O devices are enabled and the SLU can accept data from either the terminal or the host, whichever occurs first. The first arrival triggers a change to Send or Receive state.

For the SSCP-SLU session, Contention state exists between the successful completion of all chains.

Send (SEND) State

The Send state is common to both contention and HDX FF modes of operation.

In Send state, the 3276 LU resources are allocated for inbound (to the primary) operations. Internally, there are two subdivisions of the Send state. These are referred to as Send¬xmit (Send-not-transmit) and Send-xmit (Send-transmit). Send¬xmit exists while the control unit is entering data from a keyboard, MSR, or selector light pen into the device buffers. The state is entered from contention by the first keystrokes capable of changing data on the display, or by initial input from the type 2 SLU MSR or selector light pen or the type 1 SLU PA key. The state is maintained until exited to Send-xmit by an action causing the data to be sent inbound, generally the ENTER key. The transition from Send¬xmit to Send-xmit also causes the transition to In Bracket (INB) state when leaving contention. The transition always causes the keyboard to be locked and the Input Inhibit (3276 keyboard/display, and 3278 and 3279)

| and Wait (3276 keyboard/display or 3178, 3278, and 3279) indicators to be turned on. When In Bracket, Send--.xmit is entered from Receive state or ERP1 state after successfully processing an outbound chain carrying CD but not EB.

The type 2 SLU keyboard does not automatically unlock when the Send state is entered from either Receive state or ERP1 state. The keyboard is unlocked only if:

- A previous WCC specified keyboard restore, or
- The SLU is in Send state and the terminal operator presses the RESET key.

After going from Contention to Send--.xmit state, any normal outbound requests received on that session will be discarded and a negative response "Receiver in Transmit Mode" with sense code X'081B' will be sent. Once INB, any normal outbound requests received on that session (FMD with BB or Bid) while in Send State will be discarded and a negative response "Bracket Bid Reject" with sense code X'0813' will be sent. Neither of these responses causes any state change in the 3276 SLU. If INB and in Send state, a request received that does not carry BB will be rejected by the 3276 with sense code X'081B'.

During Send-xmit state, the data is being transferred from the device buffer to the PLU. Except for a possible LUSTAT, all normal flow chains on the LU-LU session will carry the CD. The transition out of Send-xmit depends upon the response type carried with the inbound request. If a definite response is requested, the transition from Send-xmit to Receive takes place after the response to the inbound request is returned to the 3276. If an exception response is requested, the transition from Send to Receive takes place as soon as the end-of-chain has been successfully transferred to the transmission link.

The SSCP-SLU session operates in definite-response mode only. Therefore, the transition is from Send-xmit to Contention upon the receipt of a positive response, or from Send-xmit to Receive if a negative response is returned.

Receive (RCV) State

The Receive state is common to both contention and HDX-FF modes of operation. In this state, the 3276 LU resources are allocated for outbound (from the PLU) operations.

When RCV state is active, inbound normal flow requests cannot be sent. Responses, as requested, and control commands of the expedited flow can be sent inbound.

Input devices may be activated by a WCC character that specifies Keyboard Restore. However, an attempt to send data to the PLU by an operator, by using the selector light pen or MSR, or by pressing the ENTER, PA, or CURSR SEL key will not be allowed.

Normal flow traffic from the PLU is passed to the device when it is in Receive state. This is allowed to halt local device operations by causing the keyboard to be locked and the Input Inhibited and Wait indicator to be turned on. A request with a WCC containing the Keyboard Restore bit set to zero is treated as a NO-OP for the keyboard states; that is, if the keyboard was unlocked before the write, it will remain unlocked after a successful write. If the keyboard was locked before the write, it will remain locked after the write.

For the LU-LU session, Receive state is entered from Contention state if an outbound normal flow message is accepted for processing. It is entered from Send-xmit after receiving a response from an inbound request carrying CD and definite response, or after successfully transferring the chain to the data link when the request carries CD and exception response. For the SSCP-LU session, Receive state is entered from Contention if an outbound normal flow message is accepted for processing. It is entered from Send-xmit if a negative response is received for an inbound request.

For the LU-LU session, Receive state is changed to Send--.xmit after successfully processing a last-of-chain carrying the CD. Receive state is changed to Contention state after successfully processing and responding to a chain carrying EB, or after receiving a chain carrying EB which carries exception response requested. Receive state is changed to ERP1 state if any negative response except X'0813', X'0814', or X'081B' is returned to the outbound request.

For the SSCP-LU session, Receive state is changed to Contention after returning the response to the outbound request.

ERP1 State

ERP1 is a special state created to allow for error recovery protocols. The PLU is always responsible for error recovery; therefore, the SLU state structure generally is awaiting an outbound request to correct the error condition. However, there are times when the SLU must first recover and notify the PLU of its recovery by use of LUSTAT command before the PLU can take action. Thus, the SLU ERP1 state allows a form of contention mode within brackets. This state has the characteristic of being able to receive any request, but only sending LUSTATs.

When an LUSTAT flows inbound, the SLU remains in ERP1 state. This allows successive LUSTATs to flow without requiring the general exchange of CD between each LUSTAT. LUSTAT does not request change direction when sent while in ERP1 state.

ERP1 state is entered by an SLU after responding with any negative response except X'0813', X'0814', and X'081B'. If the negative response does not change the state to between-brackets (BETB), the transition to ERP1 takes place at end-of-chain.

ERP1 state is changed by accepting an outbound chain carrying CD. Following processing of the CD bit, the transition is made to Send state.

When in ERP1 state, the keyboard is locked, except for the SYS REQ, ATTN, and TEST REQ keys.

Bracket States

The 3276 has three major states associated with bracket protocols. These states are Between Bracket (BETB), In Bracket (INB), and Pending Begin Bracket (PEND.BB). These states are used to ensure synchronization of traffic between the PLU and the SLU. Transitions between these states are controlled by the BB and EB bits and by the Bid command.

Between Bracket (BETB) State

BETB state exists when the PLU and SLU are in contention to begin a bracket. This is the state entered after the SDT command is accepted. When the Bid or BB is accepted from the PLU or sent by the SLU, BETB state ends. If the host program cancels the chain containing the Begin Bracket, or if the SLU sends negative response for the chain containing the Bid or BB, the 3276 returns to BETB state. BETB state is normally assumed when an EB has been processed successfully.

When a chain carrying both BB and EB is being processed, BETB state is not changed.

The 3276 sets BB on the first RU transmitted when the control unit enters INB from BETB.

BETB is terminated and INB is entered when the first (or only) element of a chain with BB bit on is ready to be transmitted; that is, an ENTER, PA, PF, or other attention key is pressed.

Pending Begin Bracket (PEND.BB) State

In the PEND.BB state, the 3276 is waiting for a bracket to be begun by the host system. The 3276 either has returned a positive response to a Bid command or has received a positive response to a Ready to Receive command. When the host program attempts to begin a bracket and the 3276 is in PEND.BB state, the 3276 will not reject the bracket with sense code X'0813' or X'0814'.

In Bracket (INB) State

INB state is entered when the 3276 receives a BB without the EB or when the 3276 begins a bracket. INB state is maintained by the 3276 until the positive definite re sponse to the EB chain is returned to the host or until the 3276 receives the last element of the EB chain when exception response is requested.

3276 Bracket State Errors

Error codes generated for bracket error conditions are shown below; the bracket state conditions remain unchanged after sending the error code.

State \ Command	CHASE & EB	CHASE & ¬ EB	BID	CANCEL & EB	CANCEL & ¬ EB	FMD & BB	FMD & ¬ BB
BETB	2003	–	–	2003	–	–	2003
INB	–	–	0813	–	–	0813	–
PEND.BB	2003	–	–	2003	–	–	2003

RU Lengths

Outbound to the 3276

The maximum RU length that a PLU is permitted to send is defined in byte 11 of Bind. The 3276 accepts a maximum RU size within the following constraints. Note that where multiple constraints apply, the maximum RU size is limited to the smallest size calculated by applying each constraint.

For a type 1 SLU in a 3276: The following formula applies:

$$MRU \leq 256 \times L \left[\frac{1}{PC} \times L \left(\frac{BUFF\text{-}80}{256}\right)\right]$$

where:

MRU is the maximum RU size specified in byte 11 of the Bind.
PC is the pacing count specified in byte 9 of the Bind.
BUFF is the device buffer size.
L is the symbol that means round down to the next integer.

Example: If the printer buffer size is 2048 bytes, and a pacing count of 2 is selected, then the maximum MRU that may be specified in Bind byte 11 is 768 bytes.

A Bind reject with sense code X'0821' will occur if the Bind specifications do not meet these limits.

For type 2 and 3 SLUs in a 3276: There are no 3276 restrictions.

Inbound from the 3276

The 3276 accepts only a 'Multiple Element Chains' Bind for inbound operation. The maximum RU size can be controlled by the PLU through byte 10 of the Bind request. For the 3276, the maximum RU size is 2048. If the value of byte 10 is greater than the 3276 capabilities, the Bind will be accepted, but the actual RU size will be limited to device capabilities.

The minimum value that may be specified by byte 10 of the Bind request is 256 bytes for the 3276. If lesser values are specified, the Bind will be rejected with a negative response, sense code X'0821'. For the 3276, if a mantissa value of byte 10 is less than 8, then minimum value is selected as a default.

Segmenting Description

RUs sent to network terminals are often larger than acceptable for optimum transfer of data by the link connecting the terminal to the network. Therefore, a Basic Information Unit (BIU) consisting of RH and RU may be divided into smaller elements, called *segments,* that are transmitted over the link. The 3276 supports inbound and outbound segmenting on the LU-LU session.

The segment elements are defined as follows. The First in Segment (FIS) element is equated to Begin-BIU, not End-BIU. The Last in Segment (LIS) element equates to End-BIU, not Begin BIU. The Middle in Segment (MIS) equates to not Begin-BIU, not End-BIU. An Only in Segment (OIS) contains the entire BIU.

Sequencing of segments is in the correct order if the sequence consists of:

1. FIS, LIS
2. FIS, MIS, . . ., LIS
3. OIS

Segmenting Outbound

Errors due to improper sequencing of the segment elements will cause the 3276 to enter normal disconnect mode. This action does not permit sending a negative response to the PLU. The 3276 will also deactivate the physical unit and all logical units. The 3276 Program Check indicator will be turned on and show the segmenting error. The 3276 will also turn off the ON LINE indicator. See Appendix C.

The 3276 passes segment elements through for processing and immediate display or printing when the terminal is attached using a Terminal Adapter Type A (for example, a 3278).

The maximum size for segment elements (the NCP MAX DATA SIZE parameter) delivered to the 3276 must not exceed 256 bytes of data plus 6 Transmission Header (TH) bytes and 3 Request/Response Header (RH) bytes for the FIS or OIS. The maximum size for MIS or LIS must not exceed 256 bytes of data plus 6 bytes of TH. (Because the maximum MIS or LIS length is 262 bytes, specify the MAX DATA value equal to 262 bytes for the 3276 control unit.)

Continuous rejection of a segment element that is too long is expected to cause a retry failure in the communication controller, and results in a station inoperative disconnect by the node. The 3276 will return a Command Reject for this condition. The 3276 depends on the sending node to limit the data length in a segment element to 256 or fewer bytes of data, and does not check for the overrun error that could occur in the MIS or LIS. The bytes of data exceeding 256 will be lost.

The Communication Check indicator showing buffer overflow is turned on for all operational 3178, 3278, or 3279 displays connected to the 3276 and the 3276 display, when the control unit detects buffer overflow.

When the 3276 is connected to NCP, the NCP buffer size should be set for one of the following byte sizes:

Optimum:	64, 128, or 256 bytes.
Second choice:	84, 124, 248, or 252 bytes.

Segmenting Inbound

Segmenting inbound is supported by the 3276 on the LU-LU session under the following conditions:

1. When maximum RU size is specified as 256 and accepted at Bind time, no segmenting is used by the 3276.

2. When maximum RU size is specified as greater than 256, the RUs are segmented into segment elements containing 256 data bytes each for FIS or MIS, provided sufficient data is transmitted to cause segmenting.

When the Bind maximum RU size is greater than 256 bytes, considerations other than maximum RU size and amount of data to be transmitted may determine the actual RU length (≤max RU size) that is sent. The 3276 will never send an RU having more than 2048 bytes. The number of segment elements allocated to an RU by the 3276 is variable and depends on the availability of link buffers when the RU is assembled for transmission. For example, if the maximum RU size is set to 2000, a sequence of sending 2500 bytes of data might appear on the line as follows:

FIC	FIS, LIS	≤512 data bytes
MIC	FIS, MIS, MIS, MIS, MIS, LIS	≤1280
MIC	OIS	≤256
LIC	FIS, MIS, LIS	remainder

The 3276 Errors

Data Link

For data link control, action is as discussed in the *IBM Synchronous Data Link Control General Information* manual, GA27-3093. Unique action is that the Set Normal Response Mode command causes the 3276 to reset from an Activated Physical Unit to a Deactivated Physical Unit. All sessions must be restarted by the sequence starting with ACTPU.

A segmenting error will not be reported by an SNA negative response, but will cause the 3276 to go to normal-disconnect mode and do an internal DACTPU.

LU-LU Session Error Reporting

A protocol has been established for the reporting of transmission errors and processing errors during sessions. When the host program or the 3276 SLU is the receiving LU, errors are reported by returning a negative response to the sending LU, with descriptive sense data included.

The format of the 4-byte sense data RU, sent with a negative response, is as follows:

0	1	2 and 3
System Major Code	Sense Modifier	User Sense

Byte 0 of the sense data RU is bit-encoded to reflect one of six transmission error categories, as follows:

Byte 0 in Hex	Major Code B
'80'	Path Error
'40'	RH Error
'20'	State Error
'10'	Request Error
'08'	Request Reject
'00'	User-Defined Error

Byte 1 of the sense data RU is a binary modifier that further defines the error condition. The modifier encoding is unique to each major code.

Bytes 2 and 3 are zeros for all negative responses sent by the 3276. The section "SNA Sense Codes" later in this chapter defines modifier encoding for each major code of system sense data issued by the 3276.

Note that the 3276 will not examine the sense data in a negative response from the host. All negative responses on the LU-LU session cause the 3276 to enter RCV state and await further action by the host.

3276 Session Interaction

Three sessions exist for the 3276 when operating with SNA protocols. These sessions are: SSCP-PU, SSCP-SLU, and LU-LU (PLU-SLU). The protocols and interactions between sessions are next described.

The three sessions can exist simultaneously. The SSCP-SLU and LU-LU sessions may wish to use the display simultaneously.

An interactive protocol is used with the 3276, in which, at any given time, only one of the sessions is defined as the device (display screen, keyboard, and data buffer) owner. During ownership, any attempts by the nonowner session to send FM data is rejected by the 3276.

The state diagram (Figure 4-14) shows the transfer of device ownership between the SSCP-SLU and the LU-LU session. Prior to ACTLU, or following DACTLU, no session can own a device. Local operations initiated by the TEST key are not defined as sessions.

Device ownership is indicated to the operator by symbols in column 3 of the Operator Information Area. (Refer to Figure A-4 for a detailed explanation of Operator Information Area symbols.) Prior to ACTLU or following DACTLU, this column is blank. ACTLU causes the Unowned symbol to appear.

After ACTLU is received, the SYS REQ key may be used by the operator to control which session owns the device. When the LU-LU session is not bound and the Unowned symbol appears in column 3, the SYS REQ key, or an RU from the SSCP, transfers device ownership to the SSCP-SLU session. At this time the System Operator Symbol appears in column 3. The operator can then communicate with the SSCP.

If the attached device is a printer or a display without a keyboard, an FM data request to the SLU from the SSCP while in the unowned state will be rejected with category not supported sense code X'1007'.

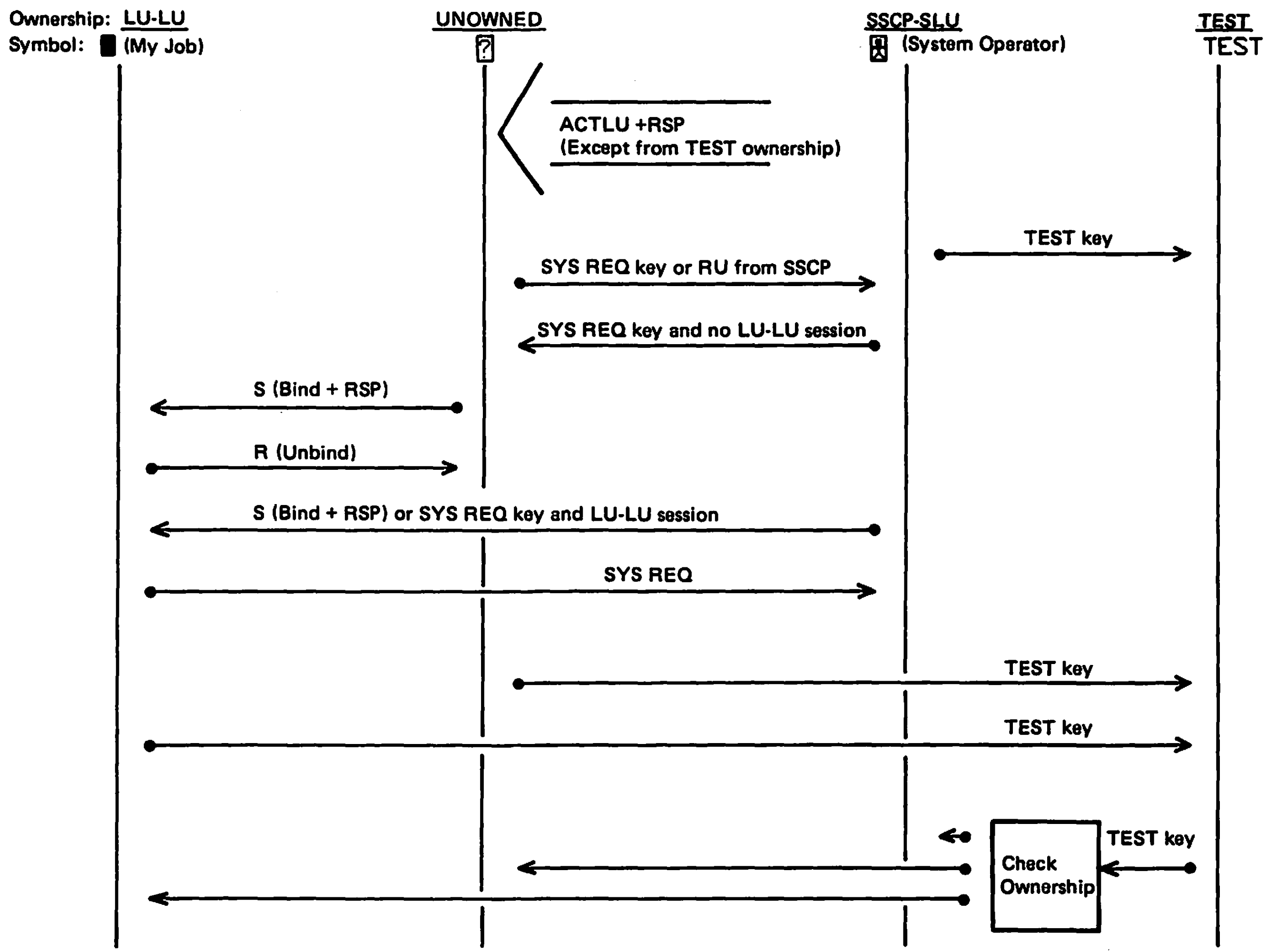


Figure 4-14. State Diagram for Session Ownership of Device

When a Bind command is received and positively responded to, ownership is transferred immediately from the SSCP-SLU session, or the unowned state, to the LU-LU session, and the My Job symbol appears in column 3. Note that Bind commands may be PLU-initiated without operator logon.

The SYS REQ key is also used to transfer ownership from the LU-LU session to the SSCP-SLU session. This transfer of ownership interrupts communications taking place during the LU-LU session without waiting for completion of outbound chains. Inbound chains will complete unless a test is made for a pacing response. As long as the LU-LU session remains bound, another depression of the SYS REQ key will cause ownership transfer back to the LU-LU session. Note that if the LU-LU session is not bound, the SYS REQ key will cause ownership transfer to the unowned state.

Pressing the TEST key causes the device to go into or leave the test ownership state. This state removes the device from the SLU and makes it unavailable to either the SSCP or PLU. If the PLU sends an FM request, the SLU sends -RSP X'082D'. If the SSCP sends an FM request, the SLU sends -RSP X'081B'. These responses assume that all other requirements for an active session have been met. When leaving the test state, a check is made for SSCP or PLU device ownership. Return will be to the session whose ownership is indicated by the check or to the unowned state if neither the PLU nor SSCP is the owner.

Setting the Screen Size

When ownership changes, the screen size may change. When changing from the unowned state to SSCP-SLU ownership, the screen size is set to the maximum physical size. When the screen enters the unowned or test state, the initial screen size is the size set by the previous owner; pressing the CLEAR key will set the screen to the maximum physical size. Operation and control of the screen size when the owner is the LU-LU session is discussed under "Erase/Write Alternate command" in Chapter 1.

Pressing the SYS REQ key causes the screen to be cleared. The screen also is cleared by the transfer of ownership from unowned to SSCP-owned when this state transfer is caused by an outbound RU from the SSCP.

Operation in SSCP-SLU Session

The following paragraphs describe the operational characteristics of the 3276 when exchanging display data on the SSCP-SLU session.

SSCP-SLU Contention Operation

The 3276 supports FM profile 0. Immediate control and immediate response is followed, and all requests are treated as definite requests.

HDX contention is implemented, and a normal flow request must be processed and acknowledged by a response before an opposite-direction normal flow request can be accepted or processed.

The 3276 SLU is in Contention state whenever SSCP-SLU session ownership mode is entered by use of the SYS REQ key.

Nonerror Operation

For nonerror operation, the receipt of a positive response, or transmission of the response, initiates the transition to Contention state. The transition from Contention to Receive state is initiated by the recognition of an outbound request. The transition from Contention to Send-not-xmit is made when the first data key pressed is accepted. Refer to "Send (SEND) State." The transition to Send-xmit is made when the ENTER key is accepted.

The keyboard is controlled by state conditions. It is unlocked when in Contention or Send-not-xmit, and locked when in Receive or Send-xmit. The operative keys that are locked or unlocked are the same as for the LU-LU session.

Error Operation

When a normal flow request has been transmitted inbound and a negative response is received, the SLU goes into Receive state and waits for an outbound request from the SSCP.

When the 3276 SLU detects a temporary or permanent error condition while in Send or Contention state, the SLU goes into Contention state. The SSCP is not notified of the error.

When a normal flow request is received but cannot be accepted because of error or a not-available condition, the SLU goes into Contention state following the negative response.

Outbound Message Handling

The SSCP may send messages to a display when the SSCP-SLU session owns the display. The messages are byte strings consisting of SCS control codes and SSCP-supported graphic codes. There is an outbound limit of 256 bytes of data. The only valid SCS control codes for the 3276 are NL and, when the APL/Text feature is installed, the Graphic Escape character. NULL, IFS, and IRS are treated as graphics and displayed as blank, *, and ; respectively. Any other binary combination in the SCS data stream will be treated as if it is a graphic. The characters appearing on the screen for code-points other than supported graphics are unpredictable.

Each message from the SSCP is displayed at the current cursor address. When the 3276 receives an NL control code in the SSCP message, it will insert nulls in the character positions remaining in the display line being written and position the cursor at the leftmost position of the next line. Characters following the NL code are displayed beginning at the new cursor position. The message wraps to the top of the screen if the last line on the screen is written and additional characters remain in the message.

After displaying the data in the received chain, the 3276 places the cursor in the position next to the last character if NL does not follow. If the message is ended by NL, the remainder of the line is set to nulls and the cursor appears in the first character position of the next line. This cursor position address is called the *initial cursor address* and is stored to identify the starting position of the operator's display input data.

Inbound Message Handling

When the System Operator symbol is displayed, an operator can enter the message bound for the SSCP from the character position occupied by the cursor.

After entering a message, the operator must press the ENTER key to initiate a transmission of the inbound message to the SSCP. Pressing other PA keys has no effect, except for the CLEAR key. Data transmission does not occur. If other PA or PF keys are depressed, Input Inhibited and Minus Function symbols are turned on. Pressing the CLEAR key causes the display screen to be cleared, and the initial cursor address is reset. The ERASE INPUT and ERASE EOF keys operate as defined under "Key Functions" in Chapter 2.

Chains sent on the SSCP-SLU session are OIC, and have a maximum RU length of 256 bytes. The 3276 will search the screen including and following cursor position to end of screen, or until a 256-byte RU has been assembled. Null characters are suppressed and not sent.

System Logon (3178; 3276; 3278; or 3279 Attached to 3276)

By means of the logon sequences, the terminal operator requests that a session be established with a PLU. The logon sequence is as follows:

1. The terminal operator checks the symbol displayed in column 3 of the Operator Information Area (see Appendix A). If the My Job symbol is displayed, the terminal is already connected to a PLU, and system logon is not required.

2a. If the Unowned symbol is displayed, the terminal operator presses the SYS REQ key to enter the SSCP-SLU owned session and then keys in a character-coded logon request in a syntax defined by the installation. The operator presses the ENTER key and the logon message is sent to SSCP.

2b. If the System Operator symbol is displayed, the display station is already owned by the SSCP-SLU session. In this case, the operator performs step 2a, except the SYS REQ key is not pressed.

3. SSCP receives the logon request and sends a positive response (X SYSTEM disappears).

4. SSCP may send a message, such as a prompting or error message, to the 3276 if necessary. When the 3276 receives this message, it sends a +RSP if accepted for display, or -RSP X'081B' if device ownership has been transferred to the LU-LU session.

5. A successful logon causes the My Job symbol to appear. An error message leaves the System Operator symbol displayed; the operator may retry, starting with step 2b.

Note: *An SSCP-SLU message confirming LOGON should not be used since this may arrive after the Bind command and confuse the operator by displaying the Message Received symbol.*

System Logoff (3178; 3276; 3278; or 3279 Attached to 3276)

By performing the logoff sequence, the terminal operator requests the SSCP to terminate a session with the PLU. The logoff sequence is as follows:

1. The terminal operator presses the SYS REQ key to enter the SSCP-SLU owned session and keys in a character-coded logoff request in a syntax defined by the installation. When the operator presses the ENTER key, the logoff message is sent to SSCP.

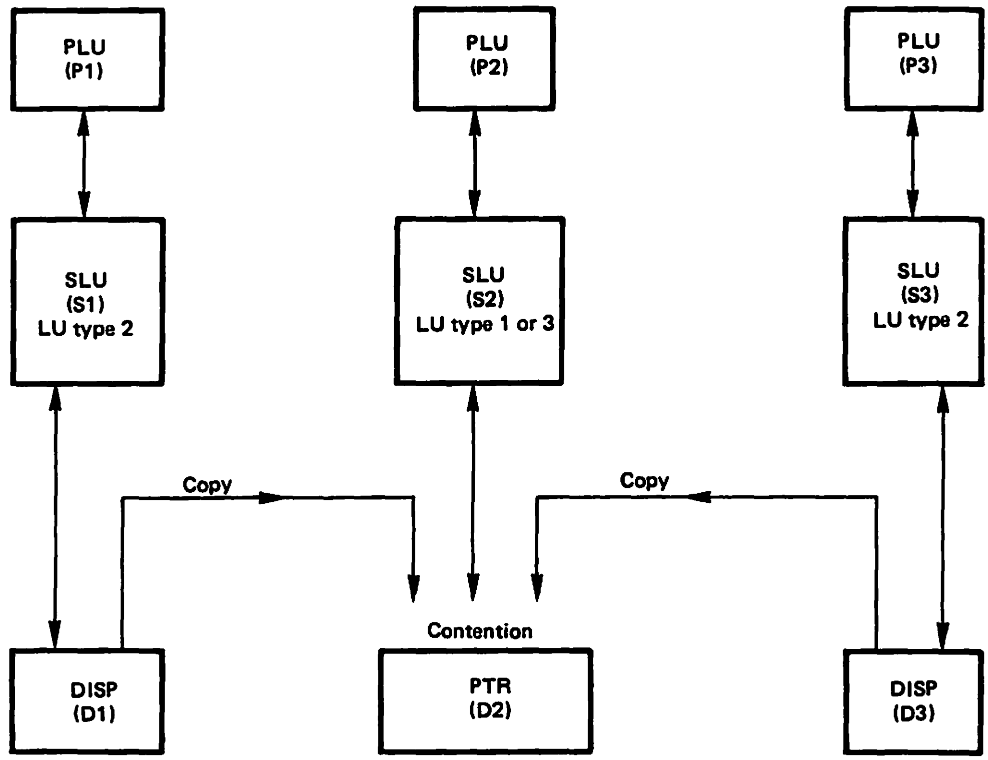
2. SSCP receives the logoff request and sends a DR response.

3. SSCP may send a message. When the 3276 receives the message, it sends a +RSP if accepted for display, or -RSP X'081B' if device ownership has been transferred.

SNA Printer Control Sessions

The following paragraphs describe the structure of the SNA session and the SNA control for printer operations. Details and constraints of subsystem operation are described under "3276 Local Copy Function" in Chapter 2.

Figure 4-15 shows a typical example of a logical subsystem and the point at which contention for the printer occurs.



P1, P2, P3	:	PLUs at the host.
S1, S3	:	SLUs in the 3276 operating as LU type 2.
S2	:	SLU in the 3276 operating as LU type 1 or LU type 3.
D1, D3	:	Display device controlled by S1 and S3, respectively.
D2	:	Printer device controlled by S2 or copied to from D1 or D3

Figure 4-15. Logical Subsystem

Printers attached to the 3276 are always configured in shared mode. In shared mode, both the SLU type 2 and the SLU type 1 or 3 may compete for use of the printer. The printer is used by the SLU type 1 or 3 as a principal device and by the SLU type 2 as a subsidiary device. For the 3276, sharing may be done between brackets.

When in shared mode, printer contention is allowed to occur between brackets. When the printer's SLU enters BETB state (or if a session does not exist), the printer is available for either a local copy from an SLU type 2 or an SLU type 1 or 3 bracket, whichever occurs first. If a local copy function is being performed for either a single SLU type 2 or a queue of SLU type 2 requests, a BB request for the type 1 or 3 SLU will be rejected with sense code X'0814' (Bracket Reject, RTR to Follow). When all local copies are completed, the type 1 or 3 SLU acquires the printer and sends RTR to the PLU. If the type 1 or 3 SLU is in-bracket, the printer is not available for local copy functions. (See the description of the copy function for details.)

SDLC Transmission Frames

SDLC transmission frames are composed of a series of 8-bit binary-coded bytes which contain addressing, data, control, and checking information. Transmission between the controller and the 3276 units takes place according to a predefined frame format which consists of the following sequence of bytes:

Flag (F) Sequence –1 byte
Secondary Station Address (A) – 1 byte
Control (C) Field – 1 byte
Information (I) Field – up to 256 bytes of message data, preceded by header information
Frame Check Sequence (FCS) – 2 bytes
Flag (F) Sequence – 1 byte

Bit synchronization preceding transmission of an initial flag and following a line turnaround is achieved by transmission of 16 zero bits, after the Clear to Send signal is turned on and the NRZI encoder (when used) is enabled.

When sending or receiving over an SDLC link, these units operate in modulo-8 mode– that is, up to seven frames at a time.

For a detailed description of the SDLC frame format, refer to *IBM Synchronous Data Link Control General Information,* GA27-3093. Support of the frame sequence, flag byte, address byte, and frame check sequence bytes conforms to the referenced document.

Response Modes

The 3276 unit functions in two link-operating modes: normal response mode (NRM) and normal disconnect mode (NDM). In NRM, the 3276 can initiate transmission only as a result of receiving a frame from the communication controller which contains the P bit set to 1. Single or multiple frames may be sent by the 3276. The last frame (or a single frame) transmitted by the 3276 in response to a command received with the P bit set to 1 must have the F bit set to 1. When the 3276 has completed a transmission, a new transmission cannot be initiated until a subsequent frame is received from the communication controller which contains the P bit set to 1. A response transmission initiated by the 3276, which requires acknowledgment from the communication controller, is repeated each time the communication controller polls until the acknowledgment is received. There is no limit to the number of transmissions. Responses that require acknowledgment from the communication controller are I frames, FRMR, and RR when transmitted with the F bit set to 0, to report clearing of a busy condition.

When in NDM, the 3276 cannot accept or transmit I or supervisory (S) frames. Nonsequenced responses are not transmitted unless the 3276 is solicited to reply. Invalid or nonimplemented commands received in NDM cause the 3276 to transmit a DM response at the next response opportunity. DM can be retransmitted until an SNRM or DISC command is received. Command reject conditions are not present in NDM.

The following paragraphs describe the 3276 support of the control and information fields.

Control Field

The control field designates the frames as supervisory (S), nonsequenced (NS), or information (I).

Supervisory Commands

The 3276 supports only the supervisory commands Receive Ready (RR) and Receive Not Ready (RNR).

The C-field formats are as follows:

RR	Nr	P/F	00	01
	012	3	45	67

RNR	Nr	P/F	01	01
	012	3	45	67

The 3276 will transmit RNR when the control unit cannot accept further data from the link.

When the reported RNR condition is cleared, the control unit will transmit an I frame or RR with the F bit on after a frame with the P bit on is received.

If the 3276 has received an RNR, an I frame will not be transmitted until an RR or I frame with the poll bit on is received.

The transmission or receipt of an NS frame does not indicate the RNR condition has cleared.

Nonsequenced Commands and Responses

The following nonsequenced commands and responses are supported by the 3276:

Command/Response	C-Field	Hex Code
Set Normal Response Mode (SNRM) Command	1 0 0 P 0 0 1 1 0 1 2 3 4 5 6 7	93
Disconnect (DISC) Command	0 1 0 P 0 0 1 1 0 1 2 3 4 5 6 7	53
Unnumbered Acknowledgment (UA) Response	0 1 1 F 0 0 1 1 0 1 2 3 4 5 6 7	73
Disconnect Mode (DM) Response	0 0 0 F 1 1 1 1 0 1 2 3 4 5 6 7	1F
Frame Reject (FRMR) Response	1 0 0 F 0 1 1 1 0 1 2 3 4 5 6 7	97
Test Command/Response	1 1 1 P/F 0 0 1 1 0 1 2 3 4 5 6 7	F3
Exchange Station ID Command/Response	1 0 1 P/F 1 1 1 1 0 1 2 3 4 5 6 7	BF

The SNRM command sets the 3276 in NRM. Receipt of SNRM causes the 3276 to deactivate the physical unit if it is in active state. The On-Line and Ownership symbols are turned off.

The DISC command sets the 3276 in NDM.

The UA response is sent by the 3276 to acknowledge receipt and acceptance of the SNRM and DISC commands.

The Test command is used to initiate one round-trip transmission of test data both in NRM and NDM. The 3276 station will return the Test response without data if buffering is not available to hold the complete test data, or with data if buffering is available.

The Disconnect Mode (DM) response is sent by the 3276 in normal disconnect mode (NDM) to request on-line status. DM is sent in response to any command except Test and XID. DM is sent in response to the SNRM command when the 3276 cannot enter NRM.

The FRMR response is implemented by the 3276 as described in GA27-3093. The FRMR will be sent in response to any poll until an SNRM or DISC is received to reset the control unit.

The exchange station identification (XID) number must be added to the host programs for a 3276 attached to a switched network, but the XID is not critical when the 3276 is attached to a nonswitched network.

The XID command and response contains additional data beyond the C byte. The 3276 responds to the XID command in NRM or NDM, except when an FRMR condition exists, in which case the FRMR response takes precedence over XID. The request/response unit (RU) of the XID response consists of 48 bits, defined as follows:

Bits	Meaning
0 – 3	ID format B'0000'
4 – 7	PU type B'0010'
8 – 15	Self description X'00'
16 – 27	X'018' (3276)
28 – 47	Terminal ID

Bits 28-47 are a unique terminal ID that can be obtained from the seven digits either shown in the machine history list supplied with the 3276 or engraved on the side frame of the 3276. These seven digits are the machine serial number and should be converted into the proper station ID by one of three methods:

- Method 1 if the first two digits of the machine serial number are 55 or 82, or if the first two digits are 00 or 23 and the remaining five digits of the machine serial number are less than 65536.
- Method 2 if the first two digits of the machine serial number are 00 or 23 and the remaining five digits of the machine serial number are from 65536 to 80535.
- Method 3 if the first two digits of the machine serial number are 00 or 23 and the remaining five digits of the machine serial number are greater than 80535.

Method 1: The first two digits of the machine serial number are converted into the following bits.

First two digits:	Bits 28–31
00 or 23:	B'0000' or X'0'
55:	B'1100' or X'C'
82:	B'1111' or X'F'

The remaining five digits are converted into bits 32–47 by use of the *IBM System Reference Card,* GX20-1850 or GX20-1703.

Example: If the seven digits of the machine serial number are 00–15263, 15263 is converted into hexadecimal. The result is X'3B9F'.

Bits

28–31	X'0' or B'0000'
32–47	X'3B9F' or B'0011101110011111'

The complete terminal ID is X'020001803B9F'.

Fixed Variable

Method 2: The first two digits of the machine serial number are converted into the following bits:

First two digits:	Bits 28–31
00:	B'0000' or X'0'
23:	B'0000' or X'0'

The remaining five digits are converted by use of the *IBM System Reference Card,* GX20-1850 or GX20-1703. The highest bit of the binary is then removed. The remaining 16 bits are used for bits 32–47

Example: If the seven digits of the machine serial number are 00–71234, 71234 is converted into hexadecimal. The result is X'11642'. The highest digit is then removed. The result is X'1642'.

Bits

28–31	X'0' or B'0000'
32–47	X'1642' or B'0001011001000010'

The complete terminal ID is X'020001801642'.

Fixed Variable

Method 3: The first two digits are not converted. The last five digits of the machine serial number are converted into hexadecimal by use of the *IBM System Reference Card,* GX20-1850 or GX20-1703. The converted hexadecimal value is bits 28-47.

Example: If the seven digits of the machine serial number are 00–98765, 98765 is converted into hexadecimal. The result is X'181CD'.

Bits

28–47	X'181CD' or B'00011000000111001101'

The complete terminal ID is X'0200018181CD'.

Fixed Variable

Terminal Identification and Addressing

Terminal ID

Each 3276 control unit operating under SDLC has a permanent, unique, 6-byte identification that it will transmit in response to a request for its ID (XID command). This identification is fixed at the time of manufacture and is not selectable.

SDLC Station Address

The SDLC station address is a 1-byte address that must be selected by the customer at setup time.

For details, refer to *IBM 3270 Information Display System Planning and Setup Guide*, GA27-2827. An SDLC station address of either X'00' or X'FF' should not be assigned.

Information (I) Frame

The information frame is used to transmit message data. When transmitted, the I frame contains a maximum of 256 bytes of RU message data preceded by 6 bytes of transmission header (TH) and, optionally, 3 bytes of request/response header (RH). For further information, refer to "Segmenting Description" in this chapter.

Sequence Error Recovery Procedures

A sequence error occurs when the 3276 receives an I frame with an incorrect Ns sequence count and valid FCS bytes. The 3276 does not accept the I frame that caused the sequence error and rejects all following I frames until an I frame is received which contains the correct Ns value, at which time the sequence error condition is reset.

The 3276 transmits I frames in the sequence indicated by the last Nr count received, which may include retransmission of previously transmitted I frames that have not been acknowledged.

All I frames are transmitted in contiguous sequence according to the Ns value within the constraints of the modulo count.

Abort Function

The abort function is used by the communication controller or by the 3276 when a frame being transmitted is to be discarded. The abort function is performed by transmitting eight contiguous 1 bits without zero insertion at the earliest possible time following recognition of an abort situation. No FCS is transmitted. When, for example, the 3276 receives seven contiguous 1 bits, it discards the aborted frame. The 3276 employs the abort function when an equipment malfunction occurs that causes an erroneous transmission.

Timeout Controls

The 3276 supports automatic disconnection from the line as shown in Figure 4-16.

			3276	
			Models 1, 2, 3, 4	Models 11, 12, 13, 14
Integrated Modem	Switched	Auto Answer	No	Yes
		Manual Answer	No	No
	Non-Switched	Without Switched Network Backup	No	No
		With Switched Network Backup (Auto Answer)	Yes	Yes
		With Switched Network Backup, Manual Answer (Auto Answer)	No	No

Note: *Automatic disconnection is supported for external modems whenever switched network is used. However, the line is not actually disconnected by this function when the CDT coupler, or its equivalent, is used between the modem and the line.*

Figure 4-16. Automatic Disconnection Support by 3276

In SDLC mode, DTR is deactivated if no frames which address the station are received for 48 to 64 seconds. In BSC mode, DTR is deactivated if no link activity is detected for 40 to 50 seconds. DTR will be activated after DSR has been deactivated.

In SDLC mode, the DISC command will cause automatic disconnection from the line in case of switched network operation. In SNBU operation, this function will not take place.

When the 3276 is attached point to point or multipoint, and does not recognize any valid outbound frame for 8 seconds, the no-link activity timeout occurs. This timeout causes the Line Ready indicator to be turned off. The timer is reset to zero when the 3276 detects a valid outbound frame.

Bind Default

The following is suggested as a setting for the access method logmode table for LU type 1:

Byte	Binary Bits	Byte	Binary Bits
–	0123 4567	8	0000 0000
0	0011 0001	9	0000 0001
1	0000 0001	10	1000 0101
2	0000 0011	11	1000 0101
3	0000 0011	12, 13	0000 0000
4	1011 0001	14	0000 0001
5	1001 0000	15-17	0000 0000
6	0011 0000	18	1110 0001
7	1000 0000	19-26	0000 0000

The suggested settings for LU type 2 are the same as for LU type 1 except for:

Byte	Binary Bits	
–	0123 4567	
9	0000 0000	
10	1000 0111	
14	0000 0010	
18	0000 0000	
24	0000 0001	Model 1
24	0000 0010	Model 2

The suggested settings for LU type 3 are the same as for LU type 1 except for:

Byte	Binary Bits
–	0123 4567
9	0000 0000
14	0000 0011
18	0000 0000

Bind Check

The Bind parameters sent to the 3276 will be checked according to the following table:

Byte	Bit	LU Type 1		LU Type 2		LU Type 3	
		Check	Reject if	Check	Reject if	Check	Reject if
1	0–3	C	¬ X'0'	C	¬ X'0'	C	¬ X'0'
	4–7	C	¬ X'1'	C	¬ X'1'	C	¬ X'1'
2–3		C	¬ X'03'	C	¬ X'03'	C	¬ X'03'
4	0	NC		NC		NC	
	1	C	B'1'	C	B'1'	C	B'1'
	2–3	C	B'00'	C	B'00'	C	B'00' B'01'(3274)
	4,5	NC		NC		NC	
	6	C	B'1'	C	B'1'	C	B'1'
	7	C	B'0'	C	B'0'	C	B'0'
5	0	NC		C	B'0'	NC	
	1	NC		NC		NC	
	2–3	C save	B'00'	C save	B'00'	C save	B'00'
	4–7	NC		NC		NC	
6	0	NC		NC		NC	
	1	C	B'1'	C	B'1'	C	B'1'
	2	C	B'0'	C	B'0'	C	B'0'
	3	C	B'0'	C	B'0'	C	B'0'
	4	C	**	C	**	C	**
	5–7	NC		NC		NC	
7	0,1	C	¬ B'10'	C	¬ B'10'	C	¬ B'10'
	2	C	B'1'	C	B'1'	C	B'1'
	3	C	B'1'	C	B'1'	C	B'1'
	4–7	NC		NC		NC	
8		NC		NC		NC	
9	0,1	NC		NC		NC	
	2–7	NC		NC		NC	
10	0–7	C		C		NC	
11		C		NC		NC	
12,13		NC		NC		NC	
14		C	¬ correct device	C	¬ correct device	C	¬ correct device
15–19		NC		NC		NC	
20–23		NC		C*		C*	
24		NC		C save		C save Device Dep	
25		NC		NC		NC	
26†		C	¬ X'00'	C	¬ X'00'	C	¬ X'00'
27† All bytes ignored							

Notes:

† Bytes 26–35 are reserved for the Encrypt/Decrypt feature.
** If byte 24 bits 4–7 has X'E' or X'F', these bytes are checked.*
*** Feature dependent*
C – Check
NC – No check
B – Bit
¬ – Logical Not

Each major error code has modifiers for further description in sense byte 1. The modifier codes supported and the controller or terminal condition causing the negative response to be returned are described below.

Sense Byte 1 Description

Path Error X'80'

X'04' Unrecognized DAF'
Controller does not have a terminal adapter for the DAF address.

X'05' – NO SESSION

- A Bind has not been received or accepted by the 3276.
- A request other than Bind is sent to an SLU which has already accepted a Bind, and the OAF' is not X'00' or the OAF in the accepted Bind.

X'08' – PU NOT Active

The 3276 has not received or accepted an ACTPU, or a control condition caused an internally generated DACTPU.

X'09' – LU NOT Active

The 3276 has not received or accepted an ACTLU, or a control condition caused an internally generated DACTLU.

X'0F' – Invalid Address Combination

A request was addressed to the PU (DAF'=X'00'), and the OAF was not SSCP (OAF'=X'00').

RH Error X'40'

X'0F' – Format Indicator Not Allowed

An FM request received by the 3276 indicated formatted header included.

State Error X'20'

X'01' – Sequence Number Error

The sequence number of the normal flow request did not match the number expected by the 3276.

X'02' – Chaining Error

Chain elements were out of protocol sequence.

X'03' – Bracket State Error

A bracket state error occurred.

Sense
Byte 1 Description

X'05' – Data Traffic Reset

An FM or DFC request was received before an SDT was received or accepted.

X'09' – Session control protocol violation (Encrypt/Decrypt feature)

An FM request was received prior to a valid CRV.

Request Error X'10'

X'02' – RU Length Error

RU size exceeds Bind specification (LUT1 only).

X'03' – Function Not Supported.

- Unsupported Session Control Request
- Unsupported Data Flow Control Request
- SIGNAL Code is not X'00010000'
- Network Control Request
- FM Data Stream
- Invalid Command
  - Data following a Read, RM, RMA, or EAU command.
  - For LU type 3, any Read, RM, or RMA command.
- Unsupported FM Data, SSCP→SPU.

X'05' – Parameter Error

Invalid address following SBA, RA, or EUA order (SBA, RA, or EUA order without parameters), or SCS parameter error.

X'07' – Category Not Supported

- An FMD request from the SSCP was received by an SLU which has an attached device without a keyboard (3276).
- An unsupported network service message received.
- An unsupported FM Data command received.

Request Reject X'08'

X'01' – Resource Not Available

- For LU type 1, outbound pacing algorithm is overrun.

Sense
Byte 1 Description

X'02' – Intervention Required (on principal device).

- For LU type 2, security keylock is turned off.
- For LU type 1 or 3, printer condition such as end of form, paper jam, printer cover up, or hold time out.

X'05' – Session Limit Exceeded

A Bind was received whose OAF' differs from the PLU already bound.

X'07' – Subsidiary Device Temporarily Not Available

For LU type 2, a printer to be copied to is In Bracket on an LU type 1 or 3 session, or an operator has pressed DEVCNCL key.

X'0A' – Permission Rejected

Display or printer power is off. The SSCP will not be notified when the device powers on.

X'0C' – Procedure Not Supported

An unsupported REQMS type request was received.

X'11' – Break

Sent on LU type 1 when the operator presses the printer Hold Print key followed by Cancel key, if a chain has not completed printing.

X'13' – Bracket Bid Reject – (No RTR)

- Returned by LU types 1 and 2 to a BID or BID with Begin Bracket if the display has won contention and started a bracket.
- Returned by all LU types, when a BID or Begin Bracket was received, and INB state already exists. This may be a protocol error.

X'14' – Bracket Bid Reject – (RTR to follow).

For LU type 1 or 3, the printer is busy doing local copy from a display. RTR will be returned when the printer becomes not busy with local copy.

X'15' – Function Active

- Bind reject if the same OAF' already has an accepted Bind to the SLU.
- REQMS request is in process.

X'1B' – Receiver in Transmit Mode

- The SLU is between bracket but a data key has been depressed.
- An FM message was received from the SSCP while the display was owned by the PLU-SLU session or is in test mode.
- An SSCP FM message is rejected if local copy is taking place while the SSCP-SLU session owns the display.

Sense
Byte 1 Description

X'1C' – Request Not Executable

The 3276 has a nonrecoverable error.

X'21' – Invalid Session Parameters

- Bind parameters do not match the 3276 Bind checks.
- 3276 rejection of ACTPU or ACTLU if FM/TS profile byte is not X'01'.

X'29' – Change Direction Required

A 3276 read-type command was received without a Change Direction or an End Bracket.

X'2B' – Presentation Space Integrity Lost

- A temporary error has occurred; for example, parity check in device.
- An operator has cleared the display by switching to SSCP-SLU session or test mode and returned to PLU-SLU session.

X'2D' – SLU Busy

- LU type 2 Display is owned by SSCP-SLU session or test mode.
- LU type 2 Display is busy doing an operator-initiated local copy.

X'2E' – Intervention Required at Subsidiary Device.

For LU type 2, a printer being copied to from a host-initiated print has intervention-required type error. Refer X'0802'. Printer power off or not attached to the controller is included in this category.

X'2F' – Request Not Executable Because of LU Subsidiary Device.

For LU type 2, a printer being copied to has a nonrecoverable error.

X'31' – LU Component Disconnected

This response is returned if the device attached to the 3276 cannot be contacted by a device poll. This is due to device power off, cable detached from the controller port, or connecting cable broken.

Note: *This response is also returned on the SSCP-SLU session by the 3276.*

X'43' – Required Function Manager Synchronization Not Supplied

For LU type 2 or 3, chains having the print bit on, must be definite response or exception response chain must carry CD.

Logical Unit Status (LUSTAT)

LUSTAT provides a means for the SLU to report exception conditions or status when the SLU is not in Receive state (a negative response is used when the SLU is in Receive state). The following are the CD settings that accompany LUSTAT and the state changes, if any, that occur:

SLU State When LUSTAT Sent	CD Setting	State Change
BETB	CD may be set	None
ERP1	CD not set	None
Send	CD set for principal device	to Receive
	CD not set for subsidiary device	None

Inbound LUSTATs are sent with a definite response by the 3276. The 3276 must receive the response to an LUSTAT before it will send any further normal flow requests on that session; however, it will accept outbound requests prior to receiving the response to LUSTAT.

Programming Note: An LUSTAT that shows power off that is sent while in Send state carries CD. An LUSTAT that shows power on cannot be sent until the PLU causes an SLU state change to (S, *R).

The following status codes will be used by the 3276 to send information to the PLU, on the PLU-SLU session:

Value	Explanation
X'0001Z000'*	Device now available; presentation space not destroyed.
X'00020000'	Device has received CD, but has no input mechanism.
X'081CZ000'*	Component failure; permanent error.
X'082B0000'	Device available; presentation space integrity lost.
X'08310000'	Principal device is powered off or disconnected.
X'0801Z000'*	Printer has been removed from configured status.

*Where Z specifies whether the status refers to the principal or subsidiary device. (Refer to "SNA Printer Control Sessions" (Figure 4-15) for a description of principal and subsidiary devices.) The value of Z is defined as follows:

LU type 1 Principal	(printer)	Z=0
LU type 2 Principal	(display)	Z=D
LU type 2 Subsidiary	(printer)	Z=B
LU type 3 Principal	(printer)	Z=0

The priority of these status codes, in low to high order, is assigned as:

X'0002', X'0001', X'082B', X'0831', X'0801', X'081C'

The 3276 will send the highest level of priority status when an opportunity allows its transmission.

Definition: (S, *R) = Send state, ERP1 state, or BETB state.

The upper section of Figure 4-17 shows the LUSTAT codes that are returned to clear the negative response condition listed in the left column. The lower section lists the LUSTAT codes that are used to report an SLU error condition instead of a negative response. The X's show the sessions that use the code points.

LUSTAT Returned

Negative Response Code	LU Type			
	T1	T2	T3	SSCP
0802	00010000 082B0000 081C0000 08310000	0001D000 082B0000 081CD000 08310000	00010000 082B0000 081C0000 08310000	NA
0807	NA	0001B000 0801B000 081CB000 081CD000	NA	NA
082D	NA	0001D000 082B0000 081CD000	NA	NA
082E	NA	0001B000 0801B000 081CB000 081CD000	NA	NA
0831	082B0000 081C0000	082B0000 081CD000	082B0000 081C0000	NA NA

Sent By

	LU Type		
LUSTAT	T1	T2	T3
SEND			
BETB			
ERP.1			
00020000	X	X	X
081C0000	X		X
081CB000		X	
081CD000		X	
082B0000	X	X	X
08310000	X	X	X
0801B000		X	

Figure 4-17. Summary Table of LUSTATs

LUSTAT is used as follows:

For all LU types, when the 3276 has sent -RSP with X'0802' or X'082E' and this condition is reset, LUSTAT with X'0001P000' will be sent: Where the value P is X'0' for LU type 1 or 3, X'D' for LU type 2 principal (display), and X'B' for LU type 2 subsidiary device (printer).

If the presentation integrity is lost while an X'0802' condition exists, LUSTAT with X'082B0000' will be sent instead of X'0001P000' when the X'0802' condition is reset.

For LU type 2, when the 3276 SLU has sent -RSP with secondary component not available (X'0807') and this condition is reset, LUSTAT with X'0001B000' will be sent.

For all LU types supported by the 3276, the LUSTAT X'00020000' will be sent to the PLU when the 3276 accepts a normal flow request carrying CD, but no input components (keyboard, light pen, MSR, etc.) are attached to the device.

For all LU types, LUSTAT with X'082B0000' will be sent to the PLU when the 3276 SLU detects presentation integrity lost (for example, regeneration buffer parity error), and is in Send or Contention state for the 3276.

For LU type 2, when the 3276 has sent -RSP (Device Busy) (X'082D') to a PLU request because of session ownership change from PLU to SSCP or TEST, LUSTAT with X'082B0000' will be sent to the PLU when returning to PLU-SLU session.

For LU type 2, when 3276 has sent -RSP (Device Busy) (X'082D') to a PLU because the SLU is busy executing a local copy, the 3276 sends LUSTAT X'0001D000' component now available to the PLU when the busy condition clears.

For all LU types, if a principal device is powered off or unplugged from the controller port and a session exists which is in (S, *R) state, LUSTAT X'08310000' will be sent to the PLU.

For all LU types, when a principal device has sent -RSP or LUSTAT X'0831000' and then power is restored, LUSTAT with X'082B0000' will be sent to the PLU.

For all LU types, if 3276 finds a permanent error in the principal device and is in (S, *R) state, LUSTAT with X'081CP000' will be sent to the PLU. The value of P is the same as defined in item 1.

For LU type 2, the 3276 will send LUSTAT with X'081CB000' when it detects a permanent error in the subsidiary device and is in (S, *R) state only if an LUSTAT for the subsidiary device is owed.

The following sense codes are returned by a negative response or an LUSTAT. Suggested recovery procedures are indicated for each error code and must be evaluated for the needs of each user.

Negative Response Codes:

Error Code	Recovery Procedures (See "Recovery Notes")
Path errors	
X'80xx'	1
RH errors	
X'40xx'	2
State errors	
X'20xx'	2,3
Request errors	
X'10xx'	2,21

Request Reject:
X'08xx'

Hex 'xx'	LU Type 1	LU Type 2	LU Type 3	
01	5	5 or 6	5	See Recovery Note(s) indicated.
02	8	7	8	
05	4	4	4	
07	NA	07	NA	
0A	4	4	4	
11	9	NA	NA	
13	10,11	10,11	10,11	
14	12	NA	12	
15	4	4	4	
1B	NA	13	NA	
1C	3,4	3,4	3,4	
21	1	1	1	
29	3,4	3,4	3,4	
2A	NA	14	NA	
2B	16	16	16	
2D	NA	7	NA	
2E	NA	7	NA	
2F	NA	17	NA	
31	7	7,18	7	
43	NA	7,19	7,19	
45	1	1	1	

LUSTAT Sense Codes:

Hex Code	Recovery Procedure (See "Recovery Notes")
0001 0000	9a
0001 B000	9a
0001 D000	9a
0002 0000	21
082B 0000	16
081C 0000	3
081C B000	17
081C D000	3
0831 0000	7,18,20
0801 B000	6,17

Recovery Notes:

1. *No recovery action can be taken until the 'xx' condition reported is corrected.*

2. *Unbind and correct program code.*

3. *Retry the operation up to three times by sending Clear, SDT, and starting traffic at a program check-point restart. Terminate the operation if the retries are not successful.*

4. *No recovery; look for an alternate terminal or terminate the operation.*

5. *Unbind, and look for an alternate terminal, or terminate the operation.*

6. *Read the display, and save for later printout.*

7. *Wait for LUSTAT; recovery based on LUSTAT code.*

8. *Wait for LUSTAT; retransmit chain.*

9. *User options:*
   a. *Resend chain.*
   b. *Send next chain.*
   c. *Send query to printer operator for PA key response.*

10. *Check the input queue for inbound data with BB and CD.*

11. *Protocol error occurred. Retry without BID or BB.*

12. *Wait for RTR to begin bracket*

13. a. *Check the input queue, and wait for data.*
    b. *Send SIGNAL to get CD.*

14. *Retry with CD and not EB.*

15. *User options:*
    a. *Send Null or comment RU with CD to give control to operator.*
    b. *Send Read Modified command with CD to obtain display AIDs and modified data.*
    c. *Reformat display from check-point restart.*

16. *Reformat display or printer from check-point restart.*

17. *Retry the operation up to three times by use of Write command and WCC with Start Print bit set to 1. An alternate printer may become available.*

18. *Unbind to force user identification by entering new logon.*

19. *Retry with correct bit settings.*

20. *When received, the user must be sure the secondary logical unit is in ERP1 or Send state, to allow sending the LUSTAT which indicates a power-on condition. The 3276 requires user action to change state if it has sent LUSTAT 08310000 while BETB.*

21. *Program dependent:*
    a. *If input is required from terminal, unbind and select an alternate terminal.*
    b. *If input is not required, data output may continue. CD should be suppressed.*

Chapter 5. Screen Design

Introduction

To use whatever you have created for display or printing, your information must be communicated to a 3270 device by means of the 3270 data stream, which is made up of commands, control characters, orders, attributes, and data.

Commands control such things as whether you write to or read from a display and whether the screen is erased before new data is written.

Control characters are used with certain commands to perform such functions as sounding the audible alarm, formatting the printer, and restoring or enabling the keyboard.

Orders are instructions written to the 3270 to tell the display unit how to format your panel.

Attributes determine the characteristics of the fields and characters within a field.

Data is the information you are displaying or printing.

The 3270 data stream is based upon the presence of a mapped character buffer in the device. There is a fixed one-to-one relationship between each character storage location in the buffer and each character position on the display. For instance, consider a display for which the display screen is composed of 12 rows of 40 columns each. Row 1 maps to the first 40 character storage positions in the character buffer, row 2 maps to the second 40 character storage positions, and so on. This sequence is the same whether the display is 12 rows and 40 columns or up to 27 rows of 132 columns.

When an Erase/Write command is transmitted to the device, the character buffer is first cleared to nulls (X'00') and subsequent text is written into the character buffer sequentially. The format of the data stream is as follows:

Write Command	WCC	Orders and Text

Field Concept

People dealing with information see it as a collection of individual elements. For example, what we know about John Smith's employment may be a collection of individual elements: his name, serial number, location, and date of hire. The size of the element is the amount of data required to convey useful information. You do not think of J and O and H and N as useful individually, but collectively, as the name JOHN. You do not think of JOHNSMITH963981BOSTON070262 as being useful collectively, but see the elements individually: name: JOHN SMITH, serial number: 963981, location: BOSTON, date of hire: 07/02/62.

Each data element has its own characteristics. In this example, the serial number is six numeric digits and varies from employee to employee. The word NAME is 4 characters, is alphabetic, is all uppercase, and does not change. When people record these elements of data on paper, they take on such additional characteristics as position (where on the sheet of paper the item is written), color (what ink or medium is used), size of the letters, and writing style.

In the past, when information was handled by a data processing device, it was generally handled as an artificial entity called a *record.* The contents and characteristics of a record were primarily determined by device requirements, and little or no attention was given to the individual information elements. Data processing users had to adjust their thought pattern to conform to the machine requirements.

The IBM 3270 Information Display System recognizes that people deal with individual units of information. The system has been designed to conform to human needs and requirements, and enables you to deal with data by individual elements or fields, each with its own characteristics.

You may describe data to the 3270 on a field or character basis and specify the characteristics or attributes of each individual field or character. The 3270 then provides program and data control based on your individual field and character definitions.

What Attributes May Be Assigned to a Field

Besides length, which is controlled by the position of field attributes, you may specify the following additional characteristics with the attributes.

Protection

A field is either protected or unprotected. When it is protected, the operator cannot enter or modify data in any location within that field.

In an unprotected field, the operator can enter characters or can delete or modify characters that are already there. Headings, labels, titles, and formats are commonly specified as protected. Any field in which the 3270 operator should enter or modify data must be specified as unprotected.

In Figure 5-1, NAME: would most likely be specified as protected. JOHN B DOE would be specified as protected if it was written by the application program and is to remain unchanged. If JOHN B DOE is to be entered or modified by the operator, the attribute 2 must specify unprotected.

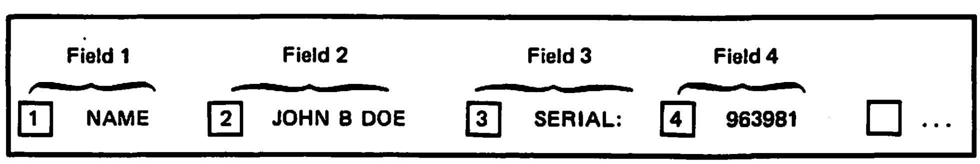


Figure 5-1. Example of Four Fields and Attribute Bytes

Color

If the device has the capability of displaying or printing in color, then the fields or individual characters may be defined in four colors, depending on whether the device has base-color capability.

Terminals with base color can display or print fields in one of four colors, depending on the definition of the field by the field-intensify/field-protection bits in the field attribute character. Base color can be produced on color displays using existing 3270 programs.

Character Content

A field is either alphameric, numeric, or user-defined symbols. An operator can enter alphameric, numeric, or special characters in an alphameric field.

The numeric attribute is more complex; it depends upon whether the numeric Lock feature is present and which keyboard is attached to the display. Figure 5-2 shows what characters may be entered with various combinations of keyboards and field types.

Keyboard Type	Keyboard Numeric Lock	Shift Key Pressed	Field Type	Protected	Resulting Characters In Buffer	Resulting Characters Displayed On Screen	Resulting Characters Read Into Storage
Typewriter	No	No	Alpha or Numeric	No	Lowercase	Uppercase	Lowercase
Typewriter	No	Yes	Alpha or Numeric	No	Uppercase	Uppercase	Uppercase
Typewriter	Yes	No	Alpha	No	Lowercase	Uppercase	Lowercase
Typewriter	Yes	Yes	Alpha	No	Uppercase	Uppercase	Uppercase
Typewriter	Yes	No	Numeric	No	Can only enter 0–9, period, and minus sign; any other characters lock keyboard.		
Typewriter	Yes	Yes	Numeric	No	Can only press dup key; any other action locks keyboard (3277). Shift key overrides on 3178, 3278, and 3279. See Numeric Lock discussion in Appendix C.		
Data Entry	No	–	Alpha	No	Alpha keys produce uppercase alpha characters. Numeric shift key produces numeric characters. Alpha shift key has no effect.		
Data Entry	No	–	Numeric	No	Numeric shift key has no effect. Alpha shift key overrides numeric specification and allows alpha character entry.		
Data Entry	Yes	–	Alpha	No	Alpha keys produce uppercase alpha characters. Numeric shift allows numeric character entry. Alpha shift key has no effect.		
Data Entry	Yes	–	Numeric	No	Can only enter 0–9, period, dup, and minus sign. Any other characters lock all keys except for RESET key. Numeric shift key allows numeric character entry. Alpha shift key allows alpha character entry.		

Figure 5-2. Results of Keyboard and Field Combinations

Visibility and Detectability

A field is either displayable or nondisplayable. When it is displayable and contains characters, those characters are visible to the operator. When it is nondisplayable, any characters within that field are not visible to the operator. The nondisplayable attribute is useful for entering classified or security information at a display unit that is in public view.

To maintain security, make sure that programs:

- Send a nondisplay attribute byte prior to sending the intended new nondisplayable data to preclude its momentary appearance on the screen.
- Do not overwrite a field attribute of nondisplay for the currently displayed image when partially changing field formats.

All characters within a displayable field can be displayed at regular brightness, at a high intensity, in color, so that they stand out among regular display fields. Color or high intensity can be used to call attention to error conditions or to emphasize protected fields or format fields. Normal intensity or underscore may be used for all input fields, so the terminal operator can tell at a glance which fields require operator action. When used on a color display, high intensity causes the field or character to be displayed in white. However, if the color property is defined, it will override the high-intensity property.

High intensity on a monochrome display results in the field or character so defined to be displayed at a brighter level than those defined as regular intensity.

You should not specify unprotected fields as high intensity since such fields may become selector-pen-detectable (if this feature is installed) if the operator enters a question mark, ampersand, or blank as the first input character. Fields are specified as either detectable or nondetectable. When a field is detectable, it can be used for selector-pen operations. A nondetectable field location cannot be detected by the selector-pen or cursor select. It is good practice to designate all detectable fields as protected to prevent the operator from changing the content of the sensitive field.

Transmission

The fields that have been entered or modified by the operator are sent to the application program by a Read Modified operation. The 3270 keeps track of such modifications and uses that information to select data to send to the application program. If you wish to pass a field into the computer regardless of modification, you may assign the "modified" or "modified data tag" (MDT) property. However, you should note that the operator can change the MDT property unless you also assign the protected property.

You can decide which combination of attributes you want within the limitations specified. Certain attribute combinations produce additional characteristics. For example, the numeric (limiting keyboard use) and protected (eliminating keyboard use) attributes seem contradictory, but, when specified together, automatically skip the cursor past the field.

You should also be aware that the application program is not limited by attributes. The application program can, for example, place alphabetic information in a field defined as numeric, or protected, or both. The operator does not have such liberty.

If you do not specify any combination of attributes, a field is assumed to have the following attributes:

- Alphameric
- Unprotected
- Displayable (at regular brightness)
- Nondetectable by the selector pen or cursor select
- Not modified

The field attribute in the 3270 data stream uses a single nondisplayed and protected character position on the screen and serves as a visual separation between successive fields.

Example of Field Definition

A typical sign-on procedure illustrates how you might define fields. Figure 5-3 illustrates a simple procedure in which the application program requests the operator to provide his name, location, and serial number.

Field 1: SIGN-ON PROCEDURE:

This field is a heading which the operator should not be able to alter. It is unnecessary for the words SIGN-ON PROCEDURE to be returned to the computer when the ENTER key is pressed. This field should be protected, alphameric, displayed at normal intensity, not detectable by the selector pen or cursor select, and not modified. All default attributes can be assumed, except that you must specify this field as protected.

Field 2: PLEASE ENTER . . . INFORMATION

You should specify this field as protected. Remember that the characteristics of a field are determined by the field attribute at the beginning of the field. Fields 1 and 2 have identical attributes and are adjacent to each other. You may choose to define them separately and use two field attributes, or you may choose to omit the field attribute at the beginning of field 2. In the latter case the two headings combine to become a single field of greater length.

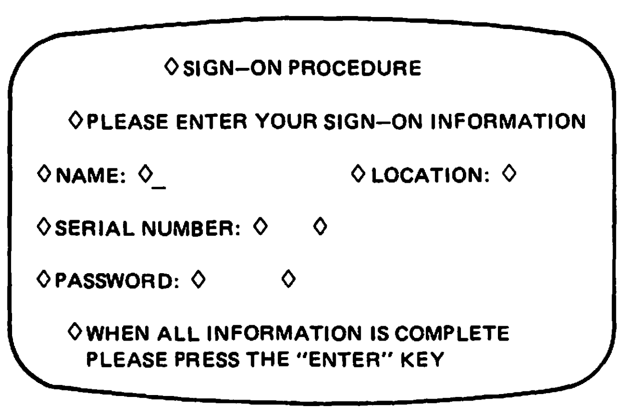


Figure 5-3. Example of Attribute Specification

Field 3: NAME:

This field should be protected, alphameric, not modified, and not detectable by the selector pen. The heading could be displayed at high intensity. Specify the protected and high-intensity attributes (the two deviations from the default attributes).

Field 4: The area following NAME:

The null area following NAME: is an input area for the operator and must therefore be unprotected. The display marks this field as modified if anything is entered into it, so you should not specify the modified attribute. The default attributes (alphameric, unprotected, displayable at normal intensity, not detectable by the selector pen or cursor select, and not modified) apply. Use a default attribute at the beginning of this field.

The maximum number of characters the operator can enter is determined by the length of this field. The length is equivalent to the number of nulls, or available positions on the screen, between the field attribute for field 4 and the field attribute for field 5.

Field 5: LOCATION:

The field attribute for this field is the same as that specified for field 3; protected and high intensity should be specified. This field attribute prevents the operator from keying a name longer than the maximum length desired. If the name is shorter than the maximum field size, the operator presses the TAB key when the name is complete. The TAB key automatically skips the cursor past protected fields, such as this one, and stops at the first character position in which data can be entered (the next unprotected field). In this example, the cursor would be positioned for entry of location. If the operator attempts to key too many characters (a name greater than 17 characters in the example), the cursor is positioned under this attribute for the 18th character. The next keystroke attempts to destroy this field attribute but fails to do so because field attributes are protected. The keyboard is inhibited, the clicker shuts off, and the "input inhibited" indicator is turned on. The operator's attention is assured since this condition requires pressing the RESET key to continue.

If the field attribute for this field were omitted, the word LOCATION: would become part of field 4 and would be normal intensity and unprotected. This is undesirable since the operator could continue entering name information beyond the desired maximum length and could modify the heading information by entering data in the screen locations occupied by LOCATION:.

Field 6: The area following LOCATION:

This field is for operator input and therefore must be unprotected. The rest of the default attribute values apply and so a default attribute may be used. You need specify only that a field is to begin following LOCATION: This field ends with the field attribute at the beginning of field 7, which determines the length of the field.

Field 7: SERIAL NUMBER:

This field, like NAME: and LOCATION:, should be specified as protected and high intensity. This also limits the location field length to 5 characters. Note that if field 6, the input field for location, were defined as always being a 5-character code, field 7, SERIAL NUMBER:, could be defined as auto-skip to save the operator from having to press TAB after filling in the location code.

Field 8: The area following SERIAL NUMBER:

The null area following SERIAL NUMBER: is an input area for the operator and must be unprotected. It should also be specified as numeric so that if the operator tries to enter alphabetic data in the field (and the keyboard has the Numeric Lock feature), the keyboard inhibits entry of the incorrect character, the keyboard clicker shuts off, and the DO NOT ENTER (X) indicator appears to notify the operator of the error. The improper character does not appear on the screen, and the correct digit may be entered after the operator presses the RESET key.

The serial number in the example always contains a fixed number of digits and is the last field entered. The maximum length of the field is determined by the location of the field attribute for the next field. But the next field in the example is too far away (PASSWORD).

By placing an additional field attribute following input field 8, the operator cannot enter a serial number that is too long.

Field 9: The area between the additional attribute described in field 8 and PASSWORD:

By definition, the additional field attribute you used to delimit the serial number field begins a new field. The protected attribute alone is sufficient for this field, and this attribute limits length for the serial number field. Normally, however, protected (output) fields that follow fixed-length input fields should be defined as *protected* and *numeric*. The protected and numeric field attribute defines a field as *auto-skip*. Auto-skip automatically positions the cursor at the location following the field attribute for the next unprotected field, which is the next place you want to key data. This technique saves keystrokes for the operator.

When the operator keys the last character of the preceding fixed-length field, the cursor normally enters the next field, which may be protected. But since the next field is auto-skip, the cursor skips this intervening protected field, and automatically positions itself for entry of the next field, without an extra keystroke.

Field 10: PASSWORD:

This would be exactly like the serial number field, protected and intensified.

Field 11: The input field for PASSWORD:

This field, like the input field for serial number, should be unprotected and numeric. But, one additional characteristic should be added, that of non-display. This allows the operator to input his or her password without anyone in the area being able to read it since it will not be visible on the display surface, thereby allowing for security. Again you would place an additional field attribute following input field 11 to ensure that the number entered would be of proper length.

This additional length check is used here because this is the last field to be entered. If you had another field to enter after PASSWORD:, it might be more advantageous to omit this length check, as explained in field 9.

Field 12: The area between the additional attribute described in field 11 and WHEN . . . COMPLETE:

This field should be protected since it is not an operator input area. The rest of the default values apply.

Field 13: WHEN ALL . . . KEY.

This field is a heading which the operator should not be allowed to change. It need not be high intensity and thus it may be defined as protected only. Field 13 does not automatically terminate when the last screen position is reached. The field definition continues from the bottom right screen position to the upper left screen position until the next field attribute is reached. This is called *wraparound.* Keep this in mind, particularly if you define the last field on a screen as *unprotected.*

Since fields 13 and 1 are adjacent (by wraparound) and both have the same attributes, they may be combined into a single field by the omission of field attributes before WHEN and SIGN-ON. The result is a single protected field beginning after the input area for password, wrapping around the screen, and terminating either at PLEASE or at NAME if fields 1 and 2 have been previously combined.

Combining fields in the above manner may be convenient but may cause confusion and error if you change the screen layout later. It is a better practice to specify separate fields in all cases.

The panel is completely formatted when the fields are positioned, the field attributes are all defined, and the cursor is placed. You must now begin the transition from the visual image, or human-oriented panel, to the detailed data necessary for the 3270 to implement your panel design.

Planning the Panel

You can think of a panel as a single display screen image created by your program. After an application program has been defined, the information that will be passed between the program and the terminal operator must be defined. This information can be thought of as output panels and input responses to panels. Usually, you will be able to approximate the sequence of the panels needed. The exact sequence of output panels often depends on the input responses to them.

Assuming you have a good understanding of the type of application program (such as data entry, order entry, or inquiry) and the kind of information that must be exchanged and processed (such as customer name, invoices, and check amounts), you can consider which panels come first. Suppose the first panel required is a sign-on panel, as shown in Figure 5-4. After sign-on, the next panel might allow the terminal operator to choose one of several different applications or procedures that he would use. But what if the name or word entered was not an authorized sign-on? Another panel might tell the terminal operator about this and ask him to reenter a sign-on name. Figure 5-4 illustrates a technique, sometimes called *block diagramming,* that may help in laying out a sequence of panels.

Using the Panel Layout Sheet

After block-diagramming the panels in the application or procedure, you are ready to decide on the exact contents of each panel: the fields that will be in the panel, what attributes each field will have, and what words will be displayed in the panel. This can be done on graph paper. The *IBM 3270 Information Display System Layout Sheet,* GX27-2951 or GX27-0014, is useful for layout.

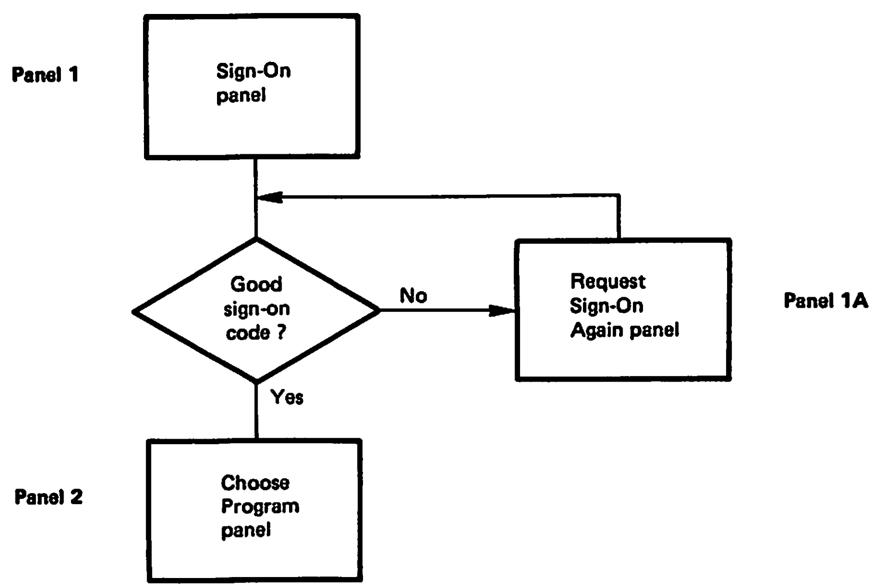


Figure 5-4. Block-Diagramming a Sequence of Panels.

One of these sheets can be used for each panel. After laying out a sequence of panels, you have a collection of panel layout sheets. Using the information on these sheets and the block diagram showing the relationship between panels, the program can be written to send the panels to a terminal and handle an operator's response to them.

An Example of Laying Out a Panel

To lay out a panel, consider the sign-on panel shown in Figure 5-3. You might jot down on a piece of paper the information required for the panel, or you might write it directly on the panel layout sheet. Figure 5-5 shows what the panel part of the layout sheet might look like after you put the text you wanted for your sign-on panel on the layout sheet. A 1920-character display is shown here.

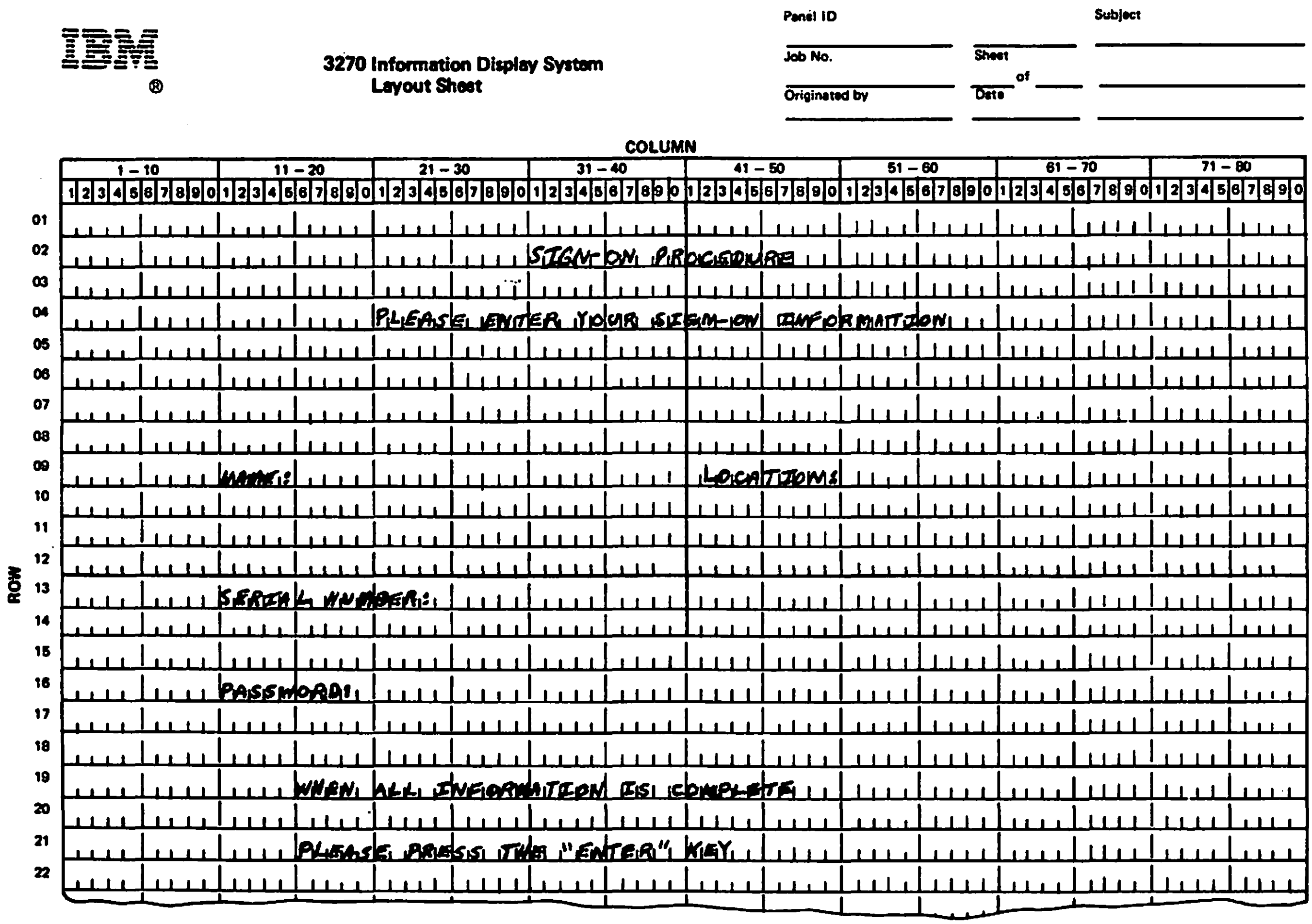


Figure 5-5. Sign-On Panel as Written Out on Layout Sheet

Now that you have written out what you want the terminal operator to see, you can define as fields the separate items of displayed text and spaces you are allowing for operator input. Remember that a field is always preceded by a field attribute. The field attribute occupies a space on the panel even though it appears as a blank space to the operator. Before deciding the attributes of a field, insert some character such as ◊ on the layout sheet to indicate the space for the field attribute. As you get used to creating panels, you may want to enter the ◊ at the same time you are laying out the text. You should also show the cursor location on the panel layout sheet to indicate to the operator where to start his response. The cursor position can be indicated by an underscore (_) under the space where you want it to appear, or you might enclose the space or characters in a rectangle. After the indications for field attributes and the cursor position have been added, the sign-on panel appears as shown in Figure 5-6.

You could have designed the panel as one long field (or even no field at all), but if you had you would not have been taking advantage of the 3270's capabilities. If you designate various items on the panel as fields, each field can have different attributes, as discussed earlier in 'What Attributes May Be Assigned to a Field."

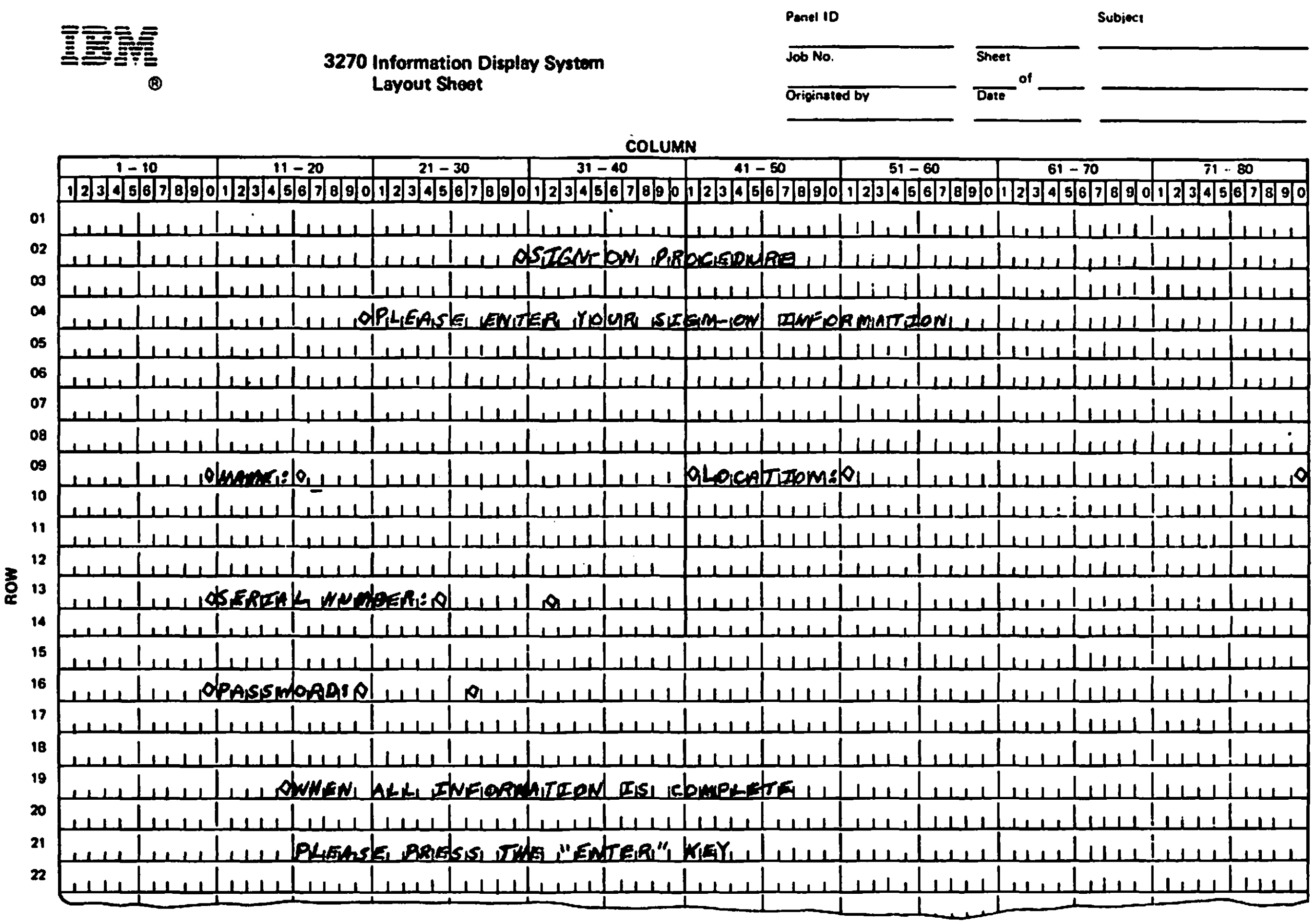


Figure 5-6. Panel Layout, Including Attribute and Cursor Positions

For example, you might want the fields NAME:, LOCATION:, and SERIAL NUMBER: to have high intensity to focus the operator's attention on them, because these fields indicate where the operator enters information. You might want to protect the fields other than the operator input fields so that the operator could not erase them; the operator input fields following NAME;, LOCATION:, SERIAL NUMBER:, and PASSWORD: should be unprotected so the operator can type in information. The operator input field following SERIAL NUMBER: and PASSWORD: can be numeric to allow some work station editing; the operator would not be allowed to accidentally enter any alphabetic characters. Field length can be defined by beginning a new field where you want the previous field to end (in some cases, this new field serves only to give a length attribute to a previous field).

Having decided on these attributes, you can use the columns on the back of the layout sheet to record the locations and attributes of the fields you have created. Your recording in these columns might appear as in Figure 5-7.

The use of these columns depends on whether the panel designer also codes the panels or only designs them. The information now on the layout sheet can be used to write a line of code that, when sent to the display, displays your panel with its specified field characteristics.

Adding Orders to the Panel Layout Sheets

The back of the panel layout sheet is used for writing the panel orders. The headings indicate what the columns should contain.

The first six columns, as shown in Figure 5-8, identify items in the text, their addresses, and the orders required to format them. The column headings are explained below:

- Item: Refers to any part of the panel that requires one or more orders to the control unit to format it. There are 14 items in the sign-on panel.

  1. SIGN-ON PROCEDURE
  2. PLEASE ENTER YOUR SIGN-ON INFORMATION
  3. NAME
  4. Input field
  5. LOCATION:
  6. Input field
  7. SERIAL NUMBER:
  8. Input field
  9. Field to limit size of serial number input
  10. PASSWORD
  11. Input field
  12. Field to limit the size of the password field
  13. WHEN ALL INFORMATION IS COMPLETE
  14. PLEASE PRESS THE ENTER KEY

	Display Printer		Buffer Address					
Item	Row	Col	Dec	Hex	Orders		Prot	No.
1	2	30			SBA			
					SF	ATT		
2	4	20			SBA			
					SF	ATT		
3	9	10			SBA			
					SF	ATT		
4	9	16			SF	ATT		
					IC			
5	9	41			SBA			
					SF	ATT		
6	9	51			SF	ATT		
	9	80			SBA			
					SF	ATT		
7	13	10			SBA			
					SF	ATT		
8	13	25			SF	ATT		
9	13	32			SBA			
					SF	ATT		
10	16	10			SBA			
					SF	ATT		
11	16	20			SF	ATT		
12	16	27			SBA			
					SF	ATT		
13	19	15			SBA			
					SF	ATT		
14	21	16			SBA			

Figure 5-7. Laying Out Field Attributes

Item	Display Printer		Buffer Address		Orders	
	Row	Col	Dec	Hex		

Figure 5-8. Text Items on Panel Layout Sheet

Since each field requires an SF order, there are always at least as many items as fields. There are more items than fields when, for example, the SBA order is used to space over unused positions within a single large field, as in Item 14.

- Row, Col: Contain the starting location (row, column) address of each item.
- Dec, Hex: Are for a different addressing format which you do not need if you use the row, column addressing format. Therefore, you may use these columns for any notes to yourself or leave them blank.
- Orders: Contains the orders you are writing, such as SBA, SF, SFE, and so forth.

As shown in Figure 5-9, the columns under the heading Attribute provide the field or character attributes that can be defined. The programmer checks the appropriate columns of the attributes he is changing from the default values. The meaning of abbreviations used in Figure 5-9 follows:

- Prot: Protected
- No.: Numeric
- High Int: High Intensity
- Sel Det: Selector-pen-detectable or cursor selectable
- Non-Disp Prt: Not displayed (nor printed at printer)
- MDT On: Modified data tag on

Note: The 3276 does not support the 7-Color Extended Highlighting, and PS sets.

No.	High Int	Sel Det	Non-Disp Prt	MDT On

Figure 5-9. Attributes

At the bottom of the columns (Figure 5-10) are the defaults automatically applied if you do not specify attributes. You must, however, specify a hexadecimal order value for the default attributes, as discussed under "Coding the Panel." The field attribute default values are:

Unpr: Unprotected

A/N: Alphameric (alphabetic and numeric)

Norm: Displayed at regular brightness

Non: Not detectable by the selector pen or cursor select

Norm: Displayed (at regular brightness)

Off: Not modified

Note: Extended field and character attributes are not supported.

A/N	Norm	Non	Norm	Off

---- Defaults ------

Figure 5-10. Attribute Default Values

You are now ready to add the required orders to the panel layout form. This may require that you rewrite the back of the form if it was originally prepared without regard to orders or if insufficient space was allowed.

Figure 5-11 shows a completed layout sheet containing all the orders to be sent with the sign-on panel.

Item	Display Printer Row	Display Printer Col	Buffer Address Dec	Buffer Address Hex	Orders		Prot	No.	High Int	Sel Det	Non-Disp Prt	MDT On
1	2	30			SBA							
					SF	ATT	✓					
2	4	20			SBA							
					SF	ATT	✓					
3	9	10			SBA							
					SF	ATT	✓		✓			
4	9	16			SF	ATT						
					IC							
5	9	41			SBA							
					SF	ATT	✓		✓			
6	9	51			SF	ATT						
	9	80			SBA							
					SF	ATT						
7	13	10			SBA							
					SF	ATT	✓		✓			
8	13	25			SF	ATT		✓				
9	13	32			SBA							
					SF	ATT	✓	✓				
10	16	10			SBA							
					SF	ATT	✓		✓			
11	16	20			SF	ATT	✓				✓	
12	16	27			SBA							
					SF	ATT	✓					
13	19	15			SBA							
					SF	ATT	✓					
14	21	16			SBA							

Figure 5-11. Completed Order and Information, No SFAP Capability

The layout sheet contains all the orders and attributes to be sent with the sign-on panel. The hexadecimal order values are discussed under "Coding the Panel" and are shown in Figure 5-12. Each item on the panel has been assigned an item number to help correlate the text with its associated orders. The sign-on panel will be designed for 1920-character screen capacity displays.

Order Sequence / Order	Byte 1 (Order Code)		Byte 2	Byte 3	Byte 4	Byte n	Byte n+1
	EBCDIC Hex	ASCII Hex					
Start Field (SF)	1D	1D	Attribute				
Set Buffer Address (SBA)	11	11	Address	Address			
Insert Cursor (IC)	13	13					
Program Tab (PT)	05	09					
Repeat To Address (RA)	3C	14	Address	Address	Char.[1]		
Erase Unprotected to Address (EUA)	12	12	Address	Address			

[1] When graphic escape is used in the RA order, the fourth byte will be the graphic escape character and the fifth byte will be the character to be repeated.

Figure 5-12. Buffer Control Orders and Order Codes

Coding the Panel

To write a panel in assembler language so that it can be part of the application program, you must transfer the panel's text and orders to the appropriate programming coding sheet or any other form you find suitable.

On the coding sheet (and in your program), a panel is represented by a series of programming statements, each with a name to which your program can refer. In the following example, SIGNPANL is the name of the sign-on panel. When the application program wants to send the sign-on panel to a display unit, it issues an Erase/Write or Erase/Write Alternate command and designates SIGNPANL as the panel for display.

The display orders must be written in the programming statements using the hexadecimal codes listed in Figure 5-12. Thus, SF is represented by 1D, SBA by 11, IC by 13, and so on.

Item 1. SIGN-ON PROCEDURE.

Begin coding the first item on the panel layout sheet: the title, SIGN-ON PROCEDURE. Start with the orders for the panel text which must precede the text itself so that the control unit knows what to do with the text. To write this title, you must tell the control unit:

- Where the title is displayed on the panel. The SBA order will be used to set the buffer address location to row 2 (R2) and column 30 (C30).
- That this location is the start of a field. The Start Field (SF) order tells the control unit that the location contains a field attribute and not a text character. The field attribute defines the properties of this field up to the next field attribute encountered. In this case, the field is protected. The remaining properties of the field are default attributes and do not have to be defined.

The first order for the title is the SBA order. Figure 5-12 shows that the SBA hexadecimal code is 11, so this code is used in the program statement. Figure 5-12 also shows that the SBA order must be accompanied by 2 bytes of address information. This address is also in hexadecimal.

Now refer to Appendix B for the address row 2 (R2) and column 30 (C30) that must follow the SBA order. The EBCDIC address is C16D and it follows the SBA code in the programming statement. This address should also be recorded on the layout sheet in the Buffer Address Hex column for future reference. If you prefer, you may look up all the addresses and record them in a similar manner before writing the programming statements. See Figure 5-13 for an example.

The next order for the title is the SF order, which is followed by the field attribute. Attribute definitions are shown in Figure 5-14. The SF code (1D) and the attribute code (60) are read from Figures 5-12 and 5-14 respectively and added to the programming statement. The data stream would be coded as follows:

SBA (11)	Address (C1)	Address (6D)	Order (1D)	Attribute (60)	S	I	G	N	-	O	N	

P	R	O	C	E	D	U	R	E

Note: *The meanings for the abbreviations used in these and the following data stream format examples are:*

```
SBA = Set Buffer Address
Addr = Address
Attr = Attribute
A/T = Attribute type
A/V = Attribute value
# Pairs = Number of attribute type/value pairs
Prot = protected
HI = High intensity
SF = Start Field
FA = Field Attribute
IC = Insert cursor
```

Each item contained in the block is 1 byte of the data in the program.

Item	Display Printer Row	Display Printer Col	Buffer Address Dec	Buffer Address Hex	Orders	
1	2	30		C16D	SBA	
					SF	ATT
2	4	20		C4C3	SBA	
					SF	ATT
3	9	10		4AC9	SBA	
					SF	ATT
4	9	16		4A4F	SF	ATT
					IC	
5	9	41		4AEB	SBA	
					SF	ATT
6	9	51			SF	ATT
	9	80		4B4F	SBA	
					SF	ATT
7	13	10		4FC9	SBA	
					SF	ATT
8	13	25			SF	ATT
9	13	32		4F5F	SBA	
					SF	ATT
10	16	10		D2F9	SBA	
					SF	ATT
11	16	20			SF	ATT
12	16	27		D34A	SBA	
					SF	ATT
13	19	15		D66E	SBA	
					SF	ATT
14	21	16		D94F	SBA	

Figure 5-13. Sign-On Procedure Panel with Buffer Addresses

Field Attribute Bit Definitions

ATTRIBUTE						
Prot	A/N	MDT ON	High Intens	Sel Det	Non Disp PRT	Hex
U						40
U		Y				C1
U				Y		C4
U		Y		Y		C5
U			H	Y		C8
U		Y	H	Y		C9
U			–	–	Y	4C
U		Y	–	–	Y	4D
U	N					50
U	N	Y				D1
U	N			Y		D4
U	N	Y		Y		D5
U	N		H	Y		D8
U	N	Y	H	Y		D9
U	N		–	–	Y	5C
U	N	Y	–	–	Y	5D
P						60
P		Y				61
P				Y		E4
P		Y		Y		E5
P			H	Y		E8
P		Y	H	Y		E9
P			–	–	Y	6C
P		Y	–	–	Y	6D
P	S					F0
P	S	Y				F1
P	S			Y		F4
P	S	Y		Y		F5
P	S		H	Y		F8
P	S	Y	H	Y		F9
P	S		–	–	Y	7C
P	S	Y	–	–	Y	7D

S = Skip Y = Yes
U = Unprotected H = High
P = Protected N = Numeric

Figure 5-14. Attribute Combinations in Hexadecimal

Item 2. PLEASE ENTER YOUR SIGN-ON INFORMATION

To write this information, the control unit must know only where the text is located. Therefore, you must use an SBA instruction followed by the address for R4, C20. This is also the beginning of a protected field, so you should include an SF order and a protected attribute.

The code for this field, except for the address, is identical with the code for field 1 SIGN-ON PROCEDURE and will not be repeated here.

Item 3. NAME:

As with Item 2, the location where the text is to be displayed must be identified. Write an SBA order followed by the EBCDIC buffer address X'hex 4AC9'. This is the beginning of a protected, high-intensity field. The data stream would be coded as follows:

SBA (11)	Address (4A)	Address (C9)	Order (SF) (1D)	Attribute (E8)	N	A	M	E	:

Item 4. Input Field for Operator's Name.

Since this item immediately follows Item 3, the control unit already knows the correct address. Therefore, there is no reason to issue an SBA order. Item 4 is the start of a new field, however, so you must issue an SF order to instruct the display to expect a field attribute next. The field attribute defines the input field as unprotected (U), alphameric (A), normal intensity, not detectable by selector pen, and no MDT on. Because these are the default attributes, you do not have to check anything in the attribute definition columns.

The cursor should follow the field attribute to indicate where the operator should begin to enter information. The Insert Cursor (IC) order displays the cursor at this current buffer address. After the display has stored the field attribute in location R9, C16, the new current address is R9, C17; this is the place where the cursor appears on the panel.

To code an input field that contains no text, such as the input field for NAME:, write just one programming statement that contains the orders for that field. The "hex" values are:

X'1D4013'

The data stream would be coded as follows:

Order SF (1D)	Attribute (40)	Order IC (13)

Item 5. LOCATION

The control unit must have two orders for this item which (1) give the starting buffer address (SBA) of the field as R9, C41, and (2) indicate that it is the start of a new field (SF), is protected, and has high intensity. The "hex" values are:

X'114AE81DE8'

The data stream would be coded as follows:

Order SBA (11)	Addr (4A)	Addr (E8)	Order SF (1D)	Attr (E8)	L	O	C	A	T	I	O	N	:

Note: *The previous examples show how the text follows in the data stream to be coded. Hereafter only the word "Text" will be used rather than the actual words.*

Item 6. Input Field for Operator's Location Code

This item immediately follows the text of the last item so there is no need to set the buffer address. Write only the SF order to indicate the start of a new unprotected field, and use default attributes.

If the field is not terminated by an additional SBA order at location column 80, the underscore property would wrap the line and continue until the next SF order is encountered. Since only the field for operator input should be underscored, terminate the field at the proper location using the proper orders. The "hex" values are:

X'1D40114B4F1D60'

The data stream would be coded as follows:

Order SF (1D)	Attr (40)	Order SBA (11)	Addr (4B)	Addr (4F)	Order SF (1D)	Addr (60)

Item 7. SERIAL NUMBER

This field requires an SBA order to location R13, C10 and an SF order to begin a new field. The attributes are specified the same as that for Item 5. The "hex" values are:

X'114FC91DE8, text "SERIAL NUMBER:"

The data stream would be coded as follows:

Order SBA (11)	Addr (4F)	Addr (C9)	Order SF (1D)	Attr (E8)	Text

Item 8. Input Field for Serial Number.

Because the field attribute for this input field immediately follows the last character of the previous field, an SBA is not required. The attribute is numeric. The "hex" values are:

X'1D50'

The data stream would be coded as follows:

Order SF (1D)	Attribute (50)

Item 9. An Extra Field Created to Limit the Size of the Serial Number Input Field

This field follows the input field and is protected and numeric. This field is an auto skip field which will place the cursor at the next location for operator input, the password input field. An SBA is required for location R13, C32, for proper placement of the attribute. The "hex" values are:

X'114F5F1DF0'

The data stream would be coded as follows:

Order SBA (11)	Addr (4F)	Addr (5F)	Order SF (1D)	Attribute (F0)

Note: *The previous format examples are representative of all those for the sign-on panel fields. The following item explanations will not show formats unless there are differences.*

Item 10. PASSWORD

The control unit must have two orders for this item: an SBA order that gives the starting address of R16, C10 and an SF order to indicate that it is the start of a new field. The field attribute defines a protected and intensified field. The "hex" values are:

X'11D2F91DE8'

Item 11. Input Field for Password

Because the field attribute for this input field immediately follows the last character of the previous field, an SBA order is not required. The attribute is numeric and non-displayable. The "hex" values are:

X'1D5C'

Item 12. Extra Field

This is another extra field created to limit the size of the password, identical with the field created to limit the size of the serial number. This follows an input field and is protected only. An SBA is required for location R16, C27, for proper placement of the field attribute byte. The "hex" values are:

X'11D34A1D60'

Item 13. "WHEN ALL INFORMATION IS COMPLETE"

The control unit must have two orders for this item: an SBA order that gives the starting address of R19, C16 and an SF order to indicate that it is the start of a new field. The field attribute defines a protected field, and the rest of the field attributes take the default value. The "hex" values are:

X'11D66E1D60'

Following each of the above statements in the program would be the text "WHEN ALL INFORMATION IS COMPLETE."

Item 14. "PLEASE PRESS THE 'ENTER' KEY."

All the words from "WHEN ALL . . . . ." through "KEY" could have been treated as a single item, but the proper number of blank spaces would have to be sent between "COMPLETE" and "PLEASE" to position "PLEASE" properly at R21, C16. It is easier, and with less chance of error, to use the three characters required for the SBA order and its associated address, breaking the field into two items, to position "PLEASE" at R21, C16. The "hex" values are: X'11D94F' followed by the text "PLEASE PRESS THE 'ENTER' KEY."

Each item from the panel layout sheet is coded in this fashion. Figure 5-15 shows the assembler language code required to display the sign-on panel for a display with no SFAP capability. Except for one control character, it consists entirely of the panel text, preceded by the display orders for that text.

The SIGN-ON panel is now complete and can be sent to the display unit by the application program.

```
SIGNPANL (NO EXTENDED ATTRIBUTES)
JOHN DOE                         11/11/80


SIGNPANL DC  X'F5'                                 ERASE/WRITE
         DC  X'C7'                                 WCC
         DC  X'11C16E1D60'                         SBA R2C31 ATT P
         DC  C'SIGN-ON PROCEDURE'
         DC  X'11C4C41D60'                         SBA R4C21 ATT P
         DC  C'PLEASE ENTER YOUR SIGN-ON INFORMATION'
         DC  X'114A4A1DE8'                         SBA R9C11 ATT P HI-INT
         DC  C'NAME:'
         DC  X'1D4013'                                       ATT U CURSOR
         DC  X'114AE91DE8'                         SBA R9C42 ATT P HI-INT
         DC  C'LOCATION:'                                    ATT U
         DC  X'1D40'
         DC  X'114F4A1DE8'                         SBA R13C11 ATT P HI-INT
         DC  C'SERIAL NUMBER:'
         DC  X'1D50'                                          ATT U NUM
         DC  X'114FSF1D60'                         SBA R13C32 ATT P
         DC  X'11D27A1DE8'                         SBA R16C11 ATT P HI-INT
         DC  C'PASSWORD:'
         DC  X'1D5C'                               ATT U NUM NON-DISPLAY
         DC  X'11D34A1D60'                         SBA R16C27 ATT P
         DC  X'11D66F1D60'                         SBA R19C16 ATT P
         DC  C'WHEN ALL INFORMATION IS COMPLETE'
         DC  X'11D94F'                             SBA R21C16
         DC  C'PLEASE PRESS THE "ENTER" KEY'
```

Figure 5-15. Assembler Language Statements for Sign-On Panel

Using the Repeat to Address Order

The Repeat to Address (RA) order stores a specified alphameric or null character in buffer locations, starting at the current buffer address and ending at (but not including) the specified stop address. The specified stop address then becomes the current buffer address.

RA is 3C in hexadecimal. RA can repeat null characters and can erase selected parts of the screen. You may also use it to repeat any other character. To put a row of asterisks under the last title in the sign-on panel, after the DC statement for "PLEASE PRESS THE ENTER KEY," you specify an SBA for R22, C1. The RA order should repeat the asterisk character to location R1, C1 (the address after the last *). This is noted on the layout form as shown in Figure 5-16.

The order in the example is coded as:

```
DC  X'3C4040'
DC  C'*'
```

If you want to delete a field already on the screen, repeat the "null" character.

Item	Display Printer Row	Display Printer Col	Buffer Address Dec	Buffer Address Hex	Orders	
1	2	30		C16D	SBA	
					SF	ATT
2	4	20		C4C3	SBA	
					SF	ATT
3	9	10		4AC9	SBA	
					SF	ATT
4	9	16		4A4F	SF	ATT
					IC	
5	9	41		4AEB	SBA	
					SF	ATT
6	9	51			SF	ATT
	9	80		4B4F	SBA	
					SF	ATT
7	13	10		4FC9	SBA	
					SF	ATT
8	13	25			SF	ATT
9	13	32		4F5F	SBA	
					SF	ATT
10	16	10		D2F9	SBA	
					SF	ATT
11	16	20			SF	ATT
12	16	27		D34A	SBA	
					SF	ATT
13	19	15		D66E	SBA	
					SF	ATT
14	21	16		D94F	SBA	
15	23	01		5B60	RA	*

Figure 5-16. An Example of the RA Order

Using the Write Control Character (WCC)

The control unit to which the display unit is attached uses the orders to format the panel. One control character for the control unit must be included following the write commands for every panel you write: the Write Control Character (WCC). The WCC is a hexadecimal code that provides control information for the control unit and defines printer information for printing panels. The other information in the WCC specifies:

- Whether to sound the audible alarm. The audible alarm is an optional display unit and printer feature that sounds a tone at the display unit upon program request. You can request this function by selecting the appropriate WCC hexadecimal code. If this feature is not installed on a display unit, the request is ignored.
- Whether to restore the keyboard at the end of your panel operation. If this option is requested, the keyboard, which locks when the operator completes a panel operation, is automatically unlocked when the program has finished processing the operator's input. Keyboard restoration means the operator does not have to press the RESET key.

  You might not want to unlock the keyboard after each panel is displayed. For example, if you plan to write out another panel before you want to accept input, locking the keyboard prevents the operator from entering data before it is needed. Also, after writing an incorrect panel, you may want to force the operator to press the RESET key to make sure you have gained his attention.
- Whether to reset the modified data tag (MDT). If this option is specified, the field attributes of all modified fields are reset. This function resets all input fields to their original (unmodified) status when an operation is completed so they are ready for the next operation.

Each panel written to a display unit or printer must begin with the WCC to identify whether these functions are requested. The hexadecimal code for each possible WCC combination is shown in Figure 5-17. See the *IBM 3270 Data Stream Programmer's Reference Manual,* GA23-0059, for more information on the WCC.

WCCs for the Display

Start Printer	Sound Audible Alarm	Restore Keyboard	Reset MDTs	Code This Hex Value
No	Yes	Yes	Yes	C7
No	Yes	Yes	No	C6
No	Yes	No	Yes	C5
No	Yes	No	No	C4
No	No	Yes	Yes	C3
No	No	Yes	No	C2
No	No	No	Yes	C1
No	No	No	No	40

WCCs for the Printer

Start Printer	Sound Audible Alarm	Restore Keyboard	Reset MDTs	Code This Hex Value If You Want			
				NL and EM Codes Honored	40-Char. Line	64-Char. Line	80-Char. Line
Yes	Yes	Yes	Yes	4F	5F	6F	7F
Yes	Yes	Yes	No	4E	5E	6E	7E
Yes	Yes	No	Yes	4D	5D	6D	7D
Yes	Yes	No	No	4C	5C	6C	7C
Yes	No	Yes	Yes	4B	5B	6B	7B
Yes	No	Yes	No	4A	5A	6A	7A
Yes	No	No	Yes	C9	D9	E9	F9
Yes	No	No	No	C8	D8	E8	F8

Figure 5-17. WCC Hexadecimal Codes

An Example of a Sequence of Panels

Assume you are given the assignment of designing the panels for an accounts receivable application. You are to create the panels that will allow a terminal operator to post a customer payment against his unpaid invoices. The terminal operator will be sitting at a 3270 work station, removing checks and invoice copies from envelopes. If the invoice copies are returned with the check, the terminal operator will for each invoice enter the customer number, payment, and invoice number. If the invoice copies are not returned, the terminal operator will have to find the customer number based on the customer name and then decide which open invoices to apply the payment against. It will be helpful if the operator has some way of adding various open invoices to find a combination that totals the payment. The 1920-character panels that follow show one possible solution.

The first panel in the application is shown in Figure 5-18. If the invoice copies come with the check, the terminal operator can enter the customer number, amount, and invoice number, and press the ENTER key. This posts the payment against the specified invoice. The terminal operator can then post the next payment and so forth; as long as the customer number and invoice number are known, only Panel is displayed.

If, however, no invoice is returned and the customer number is not known, the customer name can be entered. The name need not be the complete name of the company; it can be the first name of the company. In our example, the check says only CAPITOL so that is what the operator enters. When the name has been entered, the terminal operator presses the ENTER key. The customer number is missing, so Panel 2 is displayed.

Panel 2, shown in Figure 5-19, shows all customers and customer numbers phonetically similar to the name entered in response to Panel 1. Item numbers in Panel 2 allow the terminal operator to select one by using the corresponding Program Function (PF) key (see "Program Function Keys" later in this chapter).

As a result of terminal operator response to Panel 2, Panel 3 (shown in Figure 5-20) displays all open invoices for the identified customer. The terminal operator can now use the selector pen or cursor select to specify the open invoices to which the payment applies. He does this by touching the selector pen to the question mark adjacent to each desired invoice number or by positioning the cursor in the invoice number field and processing the cursor select keys; selection is verified immediately by the question mark changing to a > character. To post the payment against the selected invoice numbers, the operator can select APPLY. If, however, the operator cannot easily tell the invoices to which the payment is applied, he can select CALC instead of APPLY.

Selecting CALC displays Panel 4 (Figure 5-21); this is the same as Panel 3 except that ACCOUNTS RECEIVABLE, which was high intensity in Panel 3, is now normal intensity in Panel 4. A new line with CALCULATOR in high intensity indicates the screen mode and explains the functions of the PF keys. The terminal operator can now use the lower right quadrant of the screen as a scratch pad to figure out a combination of open invoices that will total the payment check. This use of one part of the screen for a separate function is sometimes called a *split-screen capability.*

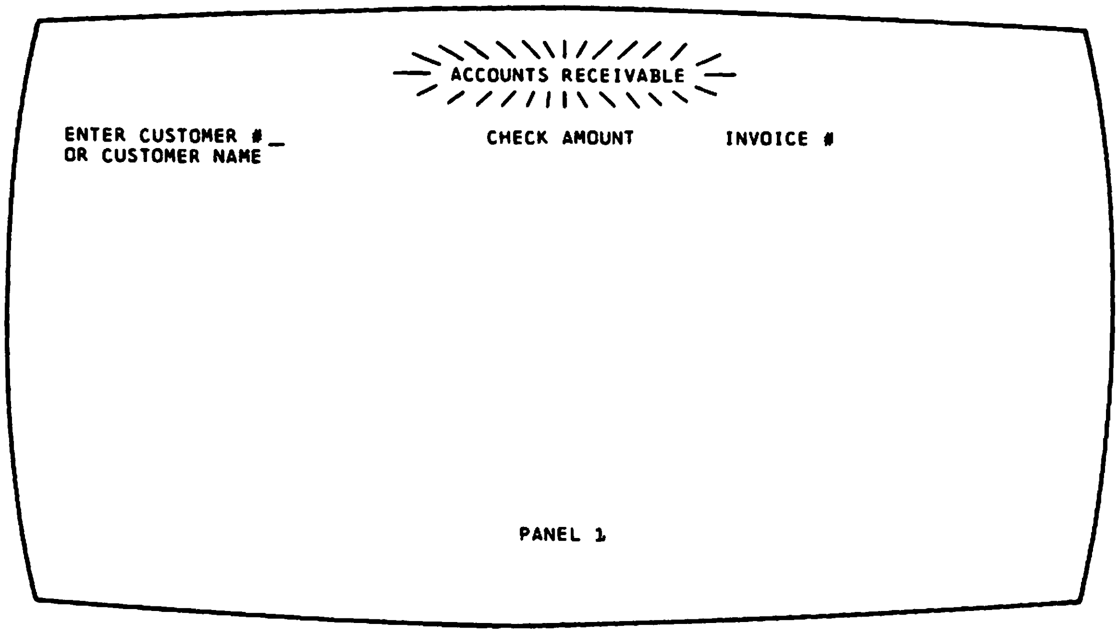
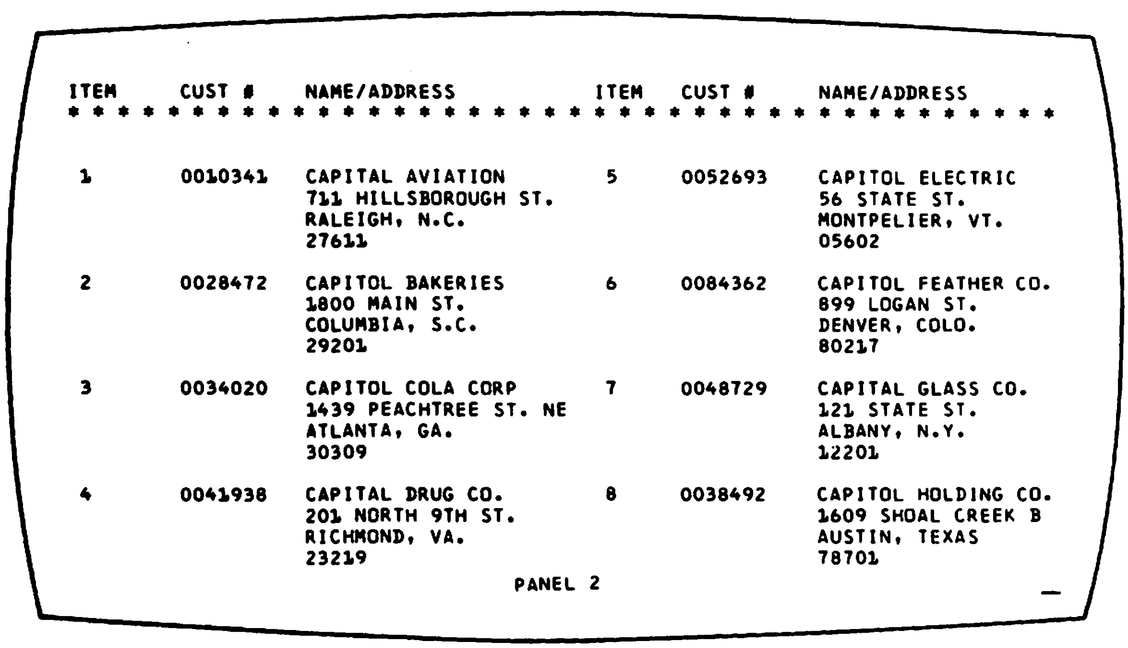


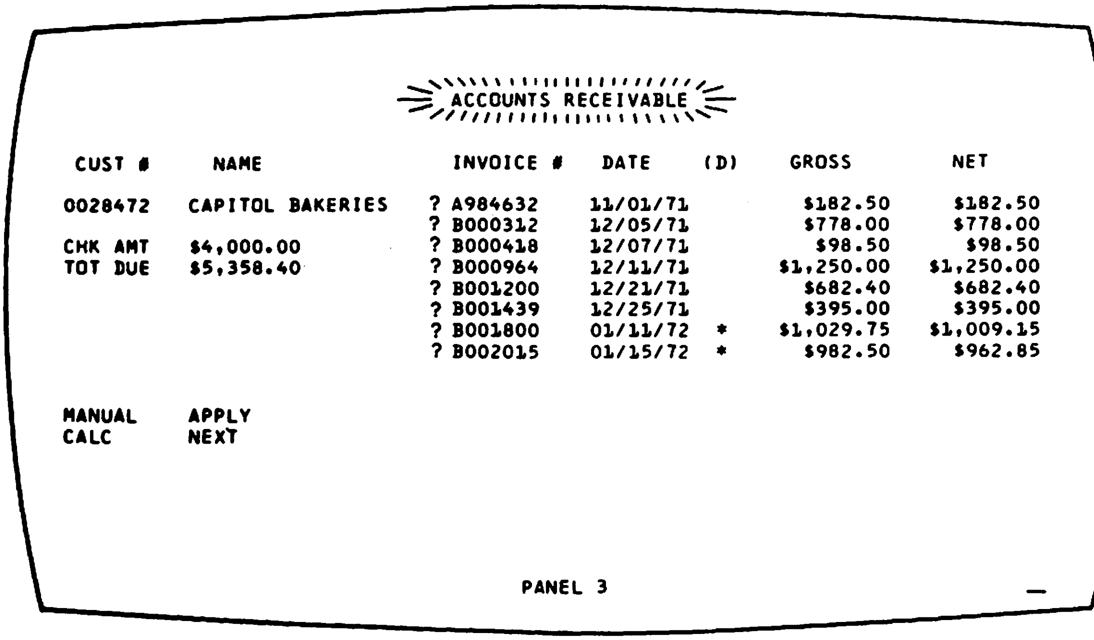
Figure 5-18. Panel 1 of an Accounts Receivable Application



ITEM	CUST #	NAME/ADDRESS	ITEM	CUST #	NAME/ADDRESS
1	0010341	CAPITAL AVIATION 711 HILLSBOROUGH ST. RALEIGH, N.C. 27611	5	0052693	CAPITOL ELECTRIC 56 STATE ST. MONTPELIER, VT. 05602
2	0028472	CAPITOL BAKERIES 1800 MAIN ST. COLUMBIA, S.C. 29201	6	0084362	CAPITOL FEATHER CO. 899 LOGAN ST. DENVER, COLO. 80217
3	0034020	CAPITOL COLA CORP 1439 PEACHTREE ST. NE ATLANTA, GA. 30309	7	0048729	CAPITAL GLASS CO. 121 STATE ST. ALBANY, N.Y. 12201
4	0041938	CAPITAL DRUG CO. 201 NORTH 9TH ST. RICHMOND, VA. 23219	8	0038492	CAPITOL HOLDING CO. 1609 SHOAL CREEK B AUSTIN, TEXAS 78701

PANEL 2

Figure 5-19. Panel 2, Showing the Results of a Search on a Customer Name



ACCOUNTS RECEIVABLE

CUST #	NAME	INVOICE #	DATE	(D)	GROSS	NET
0028472	CAPITOL BAKERIES	? A984632	11/01/71		$182.50	$182.50
		? B000312	12/05/71		$778.00	$778.00
CHK AMT	$4,000.00	? B000418	12/07/71		$98.50	$98.50
TOT DUE	$5,358.40	? B000964	12/11/71		$1,250.00	$1,250.00
		? B001200	12/21/71		$682.40	$682.40
		? B001439	12/25/71		$395.00	$395.00
		? B001800	01/11/72	*	$1,029.75	$1,009.15
		? B002015	01/15/72	*	$982.50	$962.85

MANUAL APPLY
CALC NEXT

PANEL 3

Figure 5-20. Panel 3, Showing the Customer's Open Invoices

The calculator could be programmed a number of different ways. It could, as our example illustrates, show in one column in the CALCULATOR quadrant all invoice numbers selected (shown with > in Figure 5-21) prior to selecting CALC and in another column show any balance remaining from the check amount after subtracting the selected invoice numbers. In Figure 5-21, Panel 4 is shown as it would appear if the terminal operator had first selected four invoice numbers and then selected CALC. In this example, the selected invoices equal the check amount so .00 is shown as the balance after subtracting the selected invoices.

Panel 4 shows that the CALCULATOR could also allow the operator to key in amounts and to add or subtract them from the check amount (pressing PF1 in our example adds keyed-in amounts; PF2 subtracts one keyed-in amount from another). To start over at any point, the operator can press PF3 to clear the calculator quadrant. In our example, the selected invoices equal the check amount, so they can now be posted. But first the terminal operator must leave the CALCULATOR routine by pressing PF4 (RETURN). This displays Panel 5, shown in Figure 5-22.

Panel 5 is the same as Panel 4 except that, with the operator having signaled completion of the CALCULATOR, the word now appears in normal intensity and ACCOUNTS RECEIVABLE once again appears in high intensity. The terminal operator can now, using the selector pen or cursor select, select the invoices against which to apply the payment and then select APPLY to post the payment.

Panel 6 (in Figure 5-23) shows the ACCOUNTS RECEIVABLE file for the customer after posting the payment, with the new balance and the total amount applied. To continue to the next customer, the operator selects NEXT and returns to Panel 1.

Not all the 3270's possibilities are shown in these six panels, and not all users will have the selector pen or cursor select; this example was designed to show only what panels are and how the 3270 can be used.

Note that, in the above example, the terminal operator does not see as many panels as the programmer must create; not all panels necessarily appear to the operator in any given application. What the programmer regards as separate panels may appear to the terminal operator as one changing panel.

In the above example, a number of additional panels or variations to the panels shown would be required. For example, if the terminal operator presses an invalid PF key, a variation of the panel would be required to send a message to the operator over the panel presently at his display. In programming panels that are variations of one main panel, it may be useful to assign panel designations (for example, Panel 4A, 4B, and so forth) for variations of Panel 4.

```
                                    ACCOUNTS RECEIVABLE
 CUST #      NAME                   INVOICE #   DATE     (D)     GROSS         NET

0028472    CAPITOL BAKERIES       ? A984632    11/01/71           $182.50      $182.50
                                  > B000312    12/05/71           $778.00      $778.00
CHK AMT    $4,000.00              ? B000418    12/07/71            $98.50       $98.50
TOT DUE    $5,358.40              > B000964    12/11/71         $1,250.00    $1,250.00
                                  ? B001200    12/21/71           $682.40      $682.40
                                  ? B001439    12/25/71           $395.00      $395.00
                                  > B001800    01/11/72   *     $1,029.75    $1,009.15
                                  > B002015    01/15/72   *       $982.50      $962.85

MANUAL     APPLY
CALC       NEXT                     CALCULATOR  PF1= + PF2= -   PF3= CLEAR PF4= RET

                                                 $778.00     .00
                                               $1,250.00
                                               $1,009.15
                                                 $962.85

                                        PANEL 4                                    _
```

Figure 5-21. Panel 4, Showing Use of the Calculator

```
                         ACCOUNTS RECEIVABLE

 CUST #     NAME                 INVOICE #   DATE     (D)    GROSS        NET

 0028472    CAPITOL BAKERIES   ? A984632    11/01/71          $182.50     $182.50
                               > B000312    12/05/71          $778.00     $778.00
 CHK AMT    $4,000.00          ? B000418    12/07/71           $98.50      $98.50
 TOT DUE    $5,358.40          > B000964    12/11/71        $1,250.00   $1,250.00
                               ? B001200    12/21/71          $682.40     $682.40
                               ? B001439    12/25/71          $395.00     $395.00
                               > B001800    01/11/72   *    $1,029.75   $1,009.15
                               > B002015    01/15/72   *      $982.50     $962.85


 MANUAL     APPLY
 CALC       NEXT                 CALCULATOR  PF1= + PF2= -   PF3= CLEAR PF4= RET


                                             $778.00        .00
                                           $1,250.00
                                           $1,009.15
                                             $962.85
                                    PANEL 5                                  _
```

Figure 5-22. Panel 5, Showing Selection of Invoices after Using the Calculator.

```
                         ACCOUNTS RECEIVABLE

 CUST #     NAME                 INVOICE #   DATE     (D)    GROSS        NET

 0028472    CAPITOL BAKERIES   ? A984632    11/01/71          $182.50     $182.50
 CHK AMT    $4,000.00          ? B000418    12/07/71           $98.50      $98.50
 TOT DUE    $5,358.40          ? B001200    12/21/71          $682.40     $682.40
 NEW BAL    $1,358.40          ? B001439    12/25/71          $395.00     $395.00

 SEL INV    $4,000.00


 MANUAL     APPLY
 CALC       NEXT




                                    PANEL 6                                  _
```

Figure 5-23. Panel 6, Showing New Balance after Posting

Analyzing Input Data

The Operator's Response

When a sign-on panel is displayed, the operator responds by entering name, location, and serial number as shown in Figure 5-24. As the operator keys this information, the entered data characters are stored in the display unit's buffer and are displayed as part of the panel. Data that is entered in a nondisplayable field is stored in the buffer, but does not appear on the panel.

```
              SIGN-ON PROCEDURE

   PLEASE ENTER YOUR SIGN-ON INFORMATION

 NAME: JOHN SMITH          LOCATION: BOSTN

 SERIAL NUMBER: 963981

       WHEN ALL INFORMATION IS COMPLETE
       YOU MAY PRESS THE ENTER KEY
```

Figure 5-24. Sign-On Panel with Operator's Input

When the operator finishes entering the requested data, he indicates the end of this operation by pressing the ENTER key, which causes an automatic Read Modified command execution and sends the following information to your program:

- An attention code to identify that the ENTER key was pressed.
- The address of the cursor's location.
- The start buffer address code to identify the next 2 characters as addresses.
- The starting addresses of every modified field, followed by the data in the modified field.

Figure 5-25 shows this sequence of input data, which is explained below.

AID for Enter	Cursor address	SBA	Addr first modified field	Text from first modified field	SBA	Addr of second modified field	Text from second modified field	

Figure 5-25. Input Data Sequence

Attention Identifier (AID)

The Attention Identifier (AID) is a hexadecimal code. By identifying this code, your program can determine in which of several possible ways the operator contacted the program and determine what request is being made. For example, pressing the ENTER key requests "Please enter this data."

For a Read Modified, the AID code is followed by the cursor address, which is the hexadecimal code for the row and column location of the cursor when the operator contacted your program.

Input Data

All the modified fields from the panel follow the AID code and the cursor address. A modified field is any field whose field attribute has the MDT on. A modified field can be one that was modified by the operator or one that was defined by you in your program with the MDT on in its field attribute.

When any character location in an input field is modified by the operator, the MDT in the field attribute for that field is automatically turned on. An input field is not necessarily a modified field. If the operator made no entry in the SERIAL field, for example, only his name, location, and the date would be sent as modified fields to your program.

The display unit sends all the data in a modified field except nulls. When an operator finishes an operation, the display unit reads through the buffer for every field attribute whose code indicates its MDT is on. Each time one is found, the display unit provides an SBA code and the starting address (the field attribute's address plus 1) of the modified field. The SBA code identifies to your program that an address follows. It is the same X'11' code that you coded in your panel to identify the starting locations of the panel's text.

SBA Codes

SBA codes identify the incoming data by cross-referencing it to the correct input field.

For the sign-on panel, your program knows that row 6, column 8 (X'C34F') is the start of the name input field. When it receives the first SBA code (X'11'), it checks the address that follows to see if it is (X'C34F'). If it is, your program knows the text that follows it (until the next SBA code) is the operator's name and can process the input accordingly.

The first part of the input from the sign-on panel is as follows:

7D	C4	C6	11	C3	4F	J	O	H	N		S	M	I	T	H	...

The hexadecimal codes are:

7D:	The AID code for the ENTER key
C4C6:	The cursor address R7, C23. The cursor is at the next character location after the entered serial number.
11:	The SBA order code which tells the program the next 2 characters are addresses. (See Figure 5-12.)
C34F:	The location (R6, C8) where the following text is located on the panel.
JOHN SMITH.:	The first modified field containing the operator's name.

Program Attention (PA) Keys

Each 3270 keyboard has at least one program attention (PA) key that the operator can use to request program attention without sending any input data.

The AID codes for the PA keys are shown under a separate heading in Figure 1-17, because they are not followed by input data even though there may be modified fields on the panel when a PA key is pressed. All four short read codes consist of just the AID code.

Your program should use these keys for operator requests for immediate action such as trouble alerts or requests for termination. For example, the assignment of several PA keys might be:

PA1: Terminate current application
PA2: Return to starting (master) panel
PA3: Explain system message

Program Function (PF) Keys

Program function (PF) keys are a keyboard feature. Your program defines the function that each key requests when it is pressed by the operator.

There is a separate AID code for each PF key so that your program can quickly identify which key was pressed and consequently which function was requested. When a PF key is pressed, all modified fields on the panel and their addresses are sent with the AID code and cursor address, the same as the ENTER key. For this reason, a PF key can be a valuable time-saving device for the operator. For example, the assignment of several PF keys might be:

PF1: Return to previous panel
PF2: Clear (without using data) and repeat current panel
PF3: Set up next panel
PF4: Page forward
PF5: Page backward
PF6: Return to page 1

Selector Pen and Cursor Select Input and Output

Positioning data for selector pen (optional feature) or cursor select (basic feature on the 3276, 3278, and 3279) use and setting the attribute characters are the same as for any other type of data, but the select function has additional data-stream requirements.

Selector Field Format

A field for selector pen operations must be defined as shown in Figure 5-26. The cursor select does not require the three-part character that must precede the selector pen field, although it can be present. Also, the cursor selection can be on any character in the field.

The field attribute, the designator character (described in the next section), and displayed alphameric characters must be on the same line. If the field is longer than one line, only those characters on the same line as the field attribute can be detected by the selector.

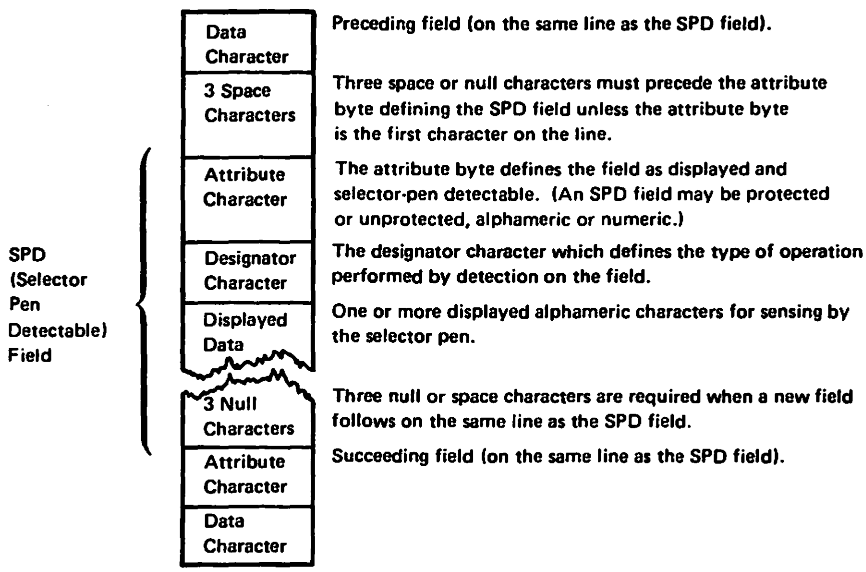


Figure 5-26. Definition of Field for Selector Pen Operation

Designator characters define three types of selector fields: selection and two types of attention. Each type of field performs a different operation.

The selection field is defined by a question mark (?) designator character. When the selector can detect a selection field, the MDT bit in the field attribute for that field is set in the display buffer. Also, the designator character is automatically changed on the screen to a greater than (>) sign to provide a visible indication to the operator that the detection was successful. If a mistake was made and the operator again detects on that same field, the > reverts to a ? and the MDT is reset. The first type of attention field is defined by a space or null designator character. Probing an attention field or selecting it with the cursor is similar to using an ENTER key. The input information is released to be read by the application program when it is ready to do so. The second type of attention field is the ampersand (&) with the 3276. Probing this field causes the program to issue a Read Modified command and obtain both the address and data of each field.

Figure 5-27 shows a sample selector pen panel that illustrates some of the special input and output data stream considerations.

For output, an Erase/Write creates the panel. In the WCC, you enable input and optionally reset the MDTs. Next you specify an SBA sequence to get you to R1, C7, followed by an SF with a protected attribute.

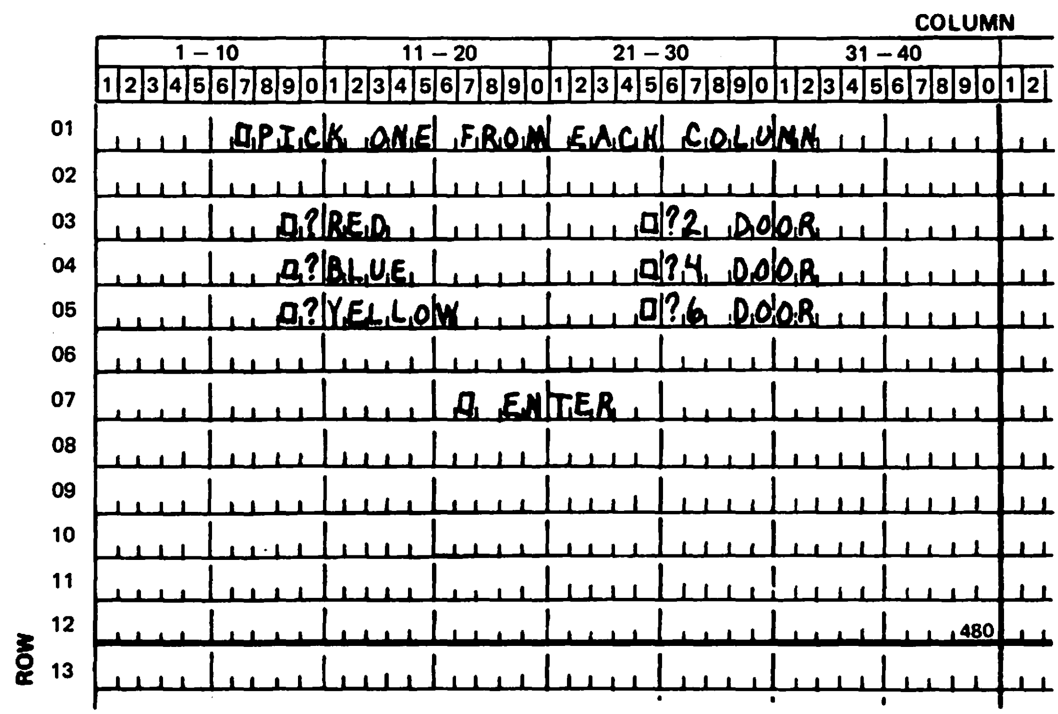


Figure 5-27. Sample Panel for Selector Pen or Cursor Select Detection

This should be followed by the heading "PICK . . . COLUMN" and another SBA to R3, C9. Then specify an SF order, followed by a protected (detectable fields may be protected) and detectable attribute. Next you need the designator "?" followed by "RED":

C	O	L	U	M	N	SBA	R3	C9	SF	P+D	?	R	E	D

An SBA after "RED" to R3, C25, provides more than the three required null characters and positions the SF field attribute and designator for "2 DOOR." This type of sequence is repeated for the remaining fields to location R7, C28. The designator here must be a null or a blank so that probing or selecting by the cursor causes the "ENTER" field to release the selection to the application program.

As the operator uses the selector pen or cursor select, the program correlates the address of each selector pen detectable field with the data associated with it.

To combine selector pen or cursor select detectable input with keyboard or cursor select input, use the keyboard to release the data to the application program by pressing the ENTER key or a PF key. Use of the selector pen or cursor select to release the data, such as by selecting "ENTER" in our example, transmits only the addresses of the fields in which the MDT was set unless you are using a 3276 control unit, in which case the address and data are transmitted.

In the example, if you pick RED and 4 DOOR, the symbolic input would appear as follows.

Pen AID	Cursor ADDR	SBA	R3	C10	SBA	R4	C26

Shortening transmissions by eliminating unnecessary data requires some caution. If you design a panel requiring both pen selection and keyboard entry, do not put an attention designator (space or null) on the panel. An attention designator after keyboard entry transmits only the address of the keyboard input field and causes the loss of its contents. Not having an attention designator on the panel assures you that an ENTER or PF key will be used, and the modified field contents will be transmitted (and the words "RED" and "4 DOOR" in the example).

The Relationship of One Data Stream to Another

The examples used so far have assumed that you started with a blank screen and that you built the entire panel into your data stream with Erase or Write commands. This approach may lead to tedious work and lengthy data streams, which you can avoid if the panel you wish to display differs only slightly from the one that is presently displayed. The following discussion deals with modification of existing data streams.

Modifying Existing Panels

Suppose the displayed panel is the sign-on panel in the previous sections. If the operator keys an invalid serial number, you may wish to notify him of his error and request reentry of the serial number field only. You could create a new error message panel, write it to the display, require that the operator acknowledge its receipt, create a special serial number entry panel, write it, and finally read the corrected serial number. A better way might be to use the existing sign-on panel.

After the operator has keyed the data and it has been read into the computer, the screen appears as shown in Figure 5-28. You would like the screen to look like Figure 5-29. Most of the information you want displayed is already there. An Erase/Write or Erase/Write Alternate command would clear the screen and require writing a data stream containing all the information for the new panel. You could use a Write command which modifies existing data in the 3270's buffer.

To change the panel in Figure 5-28 to look like Figure 5-29:

1. Position the cursor at R7, C17.

2. Replace the message beginning at R10, C5 with the error message.

3. Change the attribute at R10, C4 to high intensity for the error message.

To make these changes, the right side of your panel layout for the error panel might (in abbreviated form) look like Figure 5-30:

Item 1. Repositions the cursor to R7, C17.

Item 2. Changes the attribute at R10, C4 to protected and high intensity.

Note: *If the designer of the sign-on panel had combined the original field at this location with the previous field, with the field "SIGN-ON PROCEDURE," and with the following field by omitting the attributes at R10, C4; R2, C11; and R4, C2, the result would be undesirable. The attribute placed at R10, C4 would begin a new field. This would not affect the preceding field but, by wraparound, would cause "SIGN-ON PROCEDURE" and "PLEASE . . . INFORMATION" to be high intensity even though they were neither intended to be so nor rewritten. For this reason you should adhere closely to the "Field Concept" and not combine fields unless necessary for efficiency; if you must combine fields, be very careful to avoid undesired results.*

Item 3. Repositions the data flow to correctly place the second line of the error message. Three characters are used instead of 6 null characters.

Item 4. Repositions the data flow for the third line of the error message.

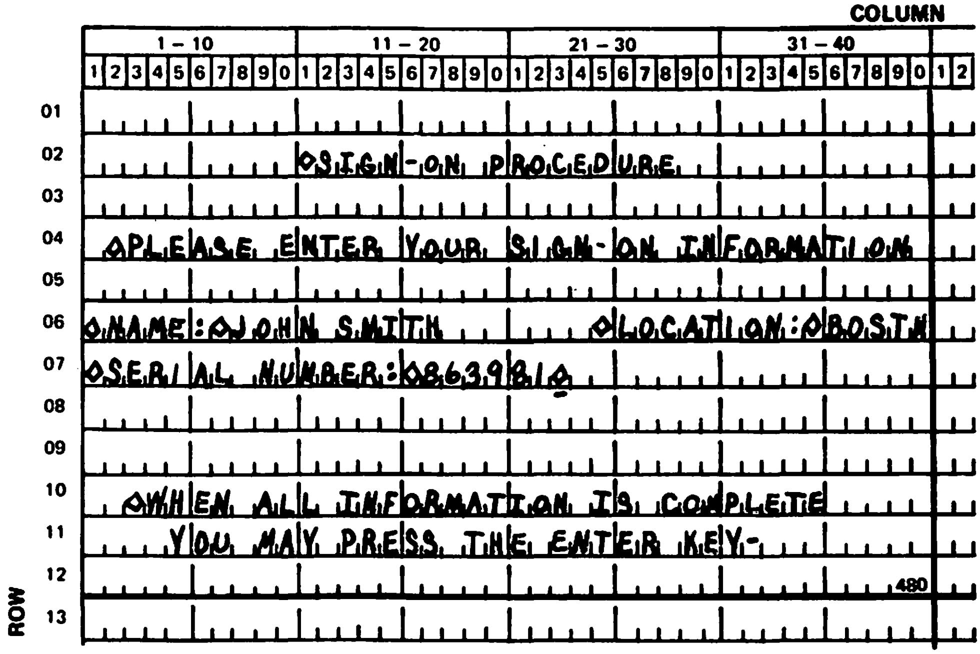


Figure 5-28. Modifying an Existing Panel, Basic Panel

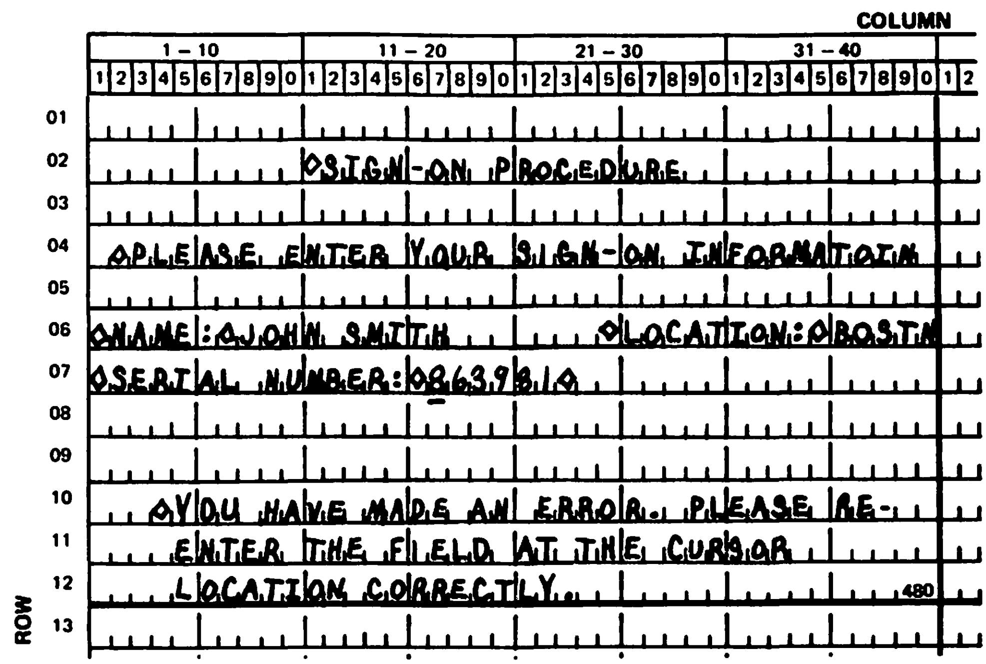


Figure 5-29. Existing Panel with Error Message

Item	Display Printer Row	Col	Buffer Address Dec	Hex	Orders		Prot	No.	High Int	Sel Det	Non-Disp Prt	MDT On
1	07	17			SBA							
					IC							
2	10	04			SBA							
					SF							
		"LINE 1 OF ERROR MESSAGE"										
3	11	05			SBA							
		"LINE 2 OF ERROR MESSAGE"										
4	12	05			SBA							
		"LINE 3 OF ERROR MESSAGE"										

Figure 5-30. Panel Layout Changes for Error Message (Keyed to Text)

Since there are two different types of Write commands for the 3276 you must tell the I/O portion of your program which type to use for the data stream. You may want to indicate the type you want in a comment in the data stream. It is suggested that you establish some convention for indicating command selection by discussing it at your installation with the people responsible for the I/O portion of the program.

In Figure 5-28, assume that the operator now keys "9" and presses the ENTER key. The "9" corrects the original entry error and the serial number field now reads "963981." What goes into the application program? The prior discussion of input data streams shows the basic format, but which fields can you expect? You know that the serial number input field will be received in its entirety, since keying the "9" caused the display to turn on the MDT for this field, and any field which has been modified is transmitted in its entirety (except nulls).

The input field MDTs for NAME, LOCATION, and SERIAL NUMBER were all turned on by the data entered into those fields in the sign-on panel. While an Erase/Write or Erase/Write Alternate resets all MDTs, a Write does not: therefore, if you do not reset them, all three input fields are returned to the application program. Because not all of them have changed, all three should not return to the application program. You may specify in the WCC that all MDTs in the device are reset "off" or "not modified" (you should do so here).

You may also want to sound the audible alarm, if you have one, with the error panel. A WCC to reset the keyboard, reset all MDTs, and sound the alarm is defined as DC X'C7' (see Figure 5-17). You can now use the Write command to change the sign-on panel into the error message panel.

Warning: As you have seen, the Write command allows you to modify an existing screen image while retaining all or a portion of the information already displayed. With the Write command, you can treat the 3270 as a typewriter-type terminal and write your panel line by line or field by field. Using multiple Write commands to create a panel, while technically possible, may create problems.

The operator might start keying data into the panel before you have finished writing it all to the screen. You can prevent this problem by not enabling the keyboard until the last Write in the series.

Using successive Write commands to accomplish what one Write command can do is an inefficient use of the communication line on remote 3270s and unnecessary I/O overhead on local 3270s.

Wherever possible, use a single Write command to avoid the inconveniences noted above.

Using Erase Unprotected to Address (EUA)

The error panel shown in Figure 5-29 displayed the erroneous serial number. All the operator had to do was to key over the incorrect digits. This may sometimes be confusing. You might instead want to erase only the serial number input field as shown in Figure 5-31.

Begin again with the desired WCC. Place the cursor at R7, C17 with an SBA to R7, C17, followed by an IC order. To erase what was entered in the serial number input field, use the EUA order (watch the sequence of these letters so you do not confuse them with EAU, which is discussed next). EUA inserts nulls (erases all unprotected positions) from the current buffer address up to, but not including, the specified stop address. It will also set any character attributes of the nulled characters to X‘00’.

The specified stop address then becomes the current buffer address. The format of the order is similar to an SBA; the code for the order itself (X‘12’ for EUA) is immediately followed by a row and column address.

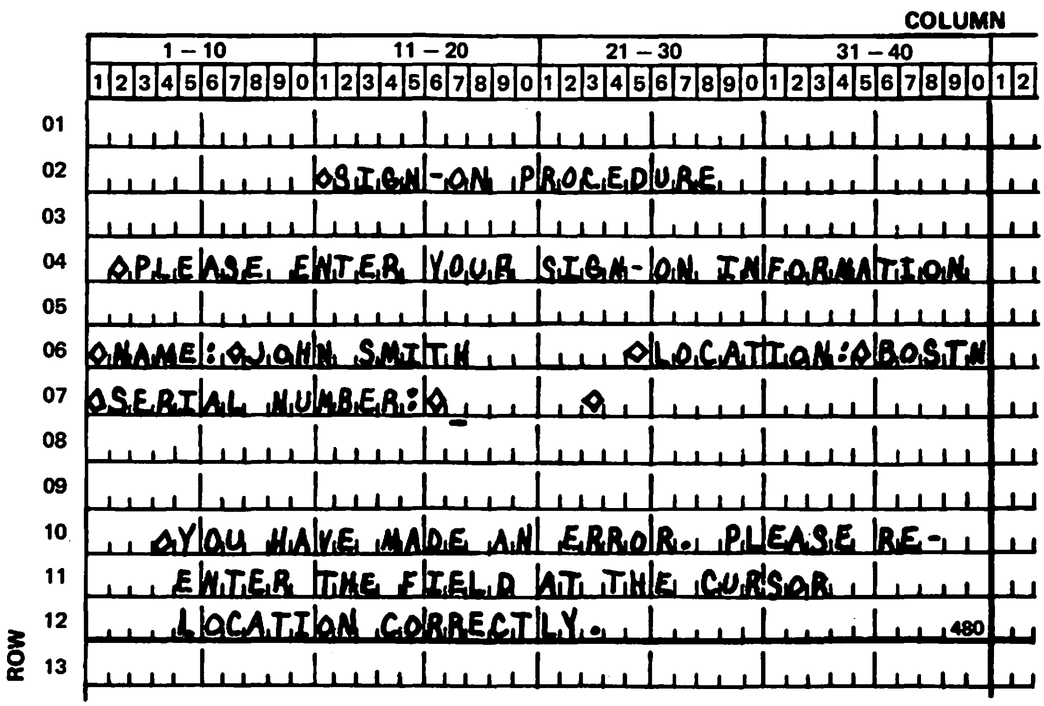


Figure 5-31. Error Message Panel with Serial Number Field Erased

At the first position to be erased (a result of prior operation), you should include an EUA order. For a terminating address, you may use R7, C23 (the first position after the last to be erased). There is a better stop address, however. Since EUA erases only unprotected fields, and since the field beginning at R7, C23 is protected, it can be included in the range covered by the EUA. If R10, C4 is used as the stop address, nothing additional is erased, but you can then write the next attribute without using an SBA, saving three characters of transmission (see Figure 5-32). The current buffer address is the stop address. Any data or SF order that follows goes into the buffer at this address.

EUA erases all unprotected fields within its range and can erase multiple fields. Suppose you wanted all three input fields erased on the error panel, as shown in Figure 5-33. First place the cursor at R7, C17; then "back up" with an SBA to R6, C8 (the name input field) before issuing the EUA to R10, C4 (see Figure 5-34).

You could have started at R6, C8 with an SBA to R6, C8, followed by the EUA to R10, C4. However, sometime later in the data stream you would have had to "back up," probably with an SBA to insert the cursor.

Using Erase All Unprotected (EAU) Command

In the preceding example, you wanted to erase all unprotected data, reposition the cursor, and add some new titles to the sign-on panel to make it an error panel. The EAU command:

- Clears all unprotected character locations and associated character attributes to nulls.
- Resets MDTs in all unprotected fields.
- Unlocks the keyboard.
- Resets the AID.
- Repositions the cursor to the first character of the first unprotected field.

	Display Printer		Buffer Address									Non-	
Item	Row	Col	Dec	Hex	Orders		Prot	No.	High Int	Sel Det	Disp Prt	MDT On	
1	07	17			SBA								
					IC								
	10	04			EUA								
2					SF	ATT	✓		✓				
	"LINE		1	OF	ERROR		MESSAGE"						
					•								
					•								
					•								

Figure 5-32. Example of EUA Use

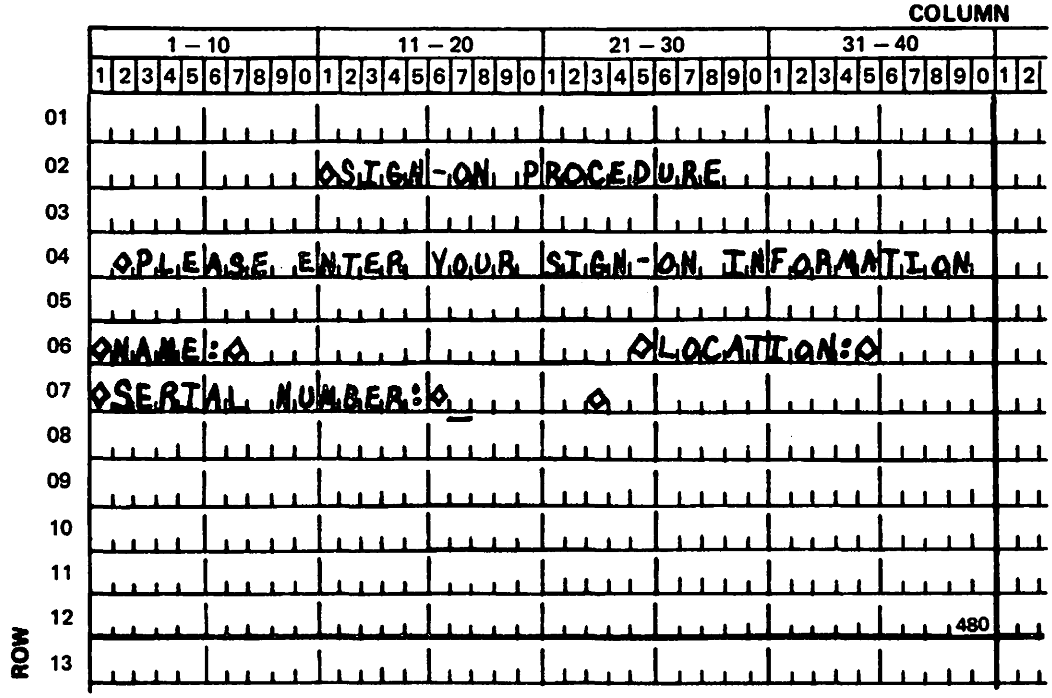


Figure 5-33. Sign-On Panel with Three Erased Fields

Item	Display Printer Row	Display Printer Col	Buffer Address Dec	Buffer Address Hex	Orders		Prot	No.	High Int	Sel Det	Non-Disp Prt	MDT On
01	07	17			SBA							
					IC							
	06	08			SBA							
	10	04			EUA							
					SF	ATT	✓		✓			
					•							
					•							
					•							

Figure 5-34. Erasing Multiple Fields with EUA

This command appears to do what you want (it even does what the WCC would have done), but it does not write any data to the screen. You could issue an EAU command before the Write command. Then you would just write the new titles in their proper positions. You have then issued two commands to create one panel. What, then, is EAU for? It logically resets the panel for repetitive input using the same panel. Do not use EAU to change panels.

Data Entry Example: You can use the EAU command to change a sign-on panel slightly and make it a data entry panel. Then the operator just keys in NAME, LOCATION, and SERIAL NUMBER for the first employee. If an error is made, an error panel is shown. If there is no error, you may want to clear the input, reset the MDTs, unlock the keyboard, and reposition the cursor. The data entry panel might appear as shown in Figure 5-35.

The operator keys JOHN SMITH, presses TAB, keys BOSTN, presses TAB, keys 963981, and presses ENTER (Figure 5-36). You simply send the 3270 an EAU command to unlock the keyboard. The operator then sees the same panel as in Figure 5-35 and may now key data for the next employee. You have used your knowledge of what is already displayed to arrive at the next panel or to re-create the present panel.

Repetitive Output

In the data entry example, you used one panel repetitively for input of employee information. You can reverse the requirement and design an employee data screen. For this example, assume the application is inquiry with "browsing" capability. Assume also that the operator has previously used another panel to request the information for employee number 963981. The display might appear as shown in Figure 5-37.

At the bottom of the panel, the operator is instructed to use the PA1 key to see the next employee page, probably number 963982. The PA2 key is assigned to page backward. Remember, PA keys are assigned by the program. Program attention keys cause a short transmission; they do not even transmit the contents of changed fields. For an inquiry and browsing application, there should be no input. The PA key assures there is no input even if the operator changes one of the unprotected fields, so its use is preferred to the ENTER or PF keys.

Using the Program Tab (PT)

The input fields in the previous examples are output fields in this example. You could designate them as *protected,* but if you did, you could not use another 3270 function called *Program Tab.* The Program Tab (PT) order advances the current buffer address to the address of the first character location of the next unprotected field. When the PT order immediately follows an alphameric or null character (not another order) in the Write data stream (other than the character specified by the Repeat to Address order, which is discussed earlier), it also inserts nulls in all the character positions from the current buffer address to the end of the current field. The PT order can be used to page through the employee data file.

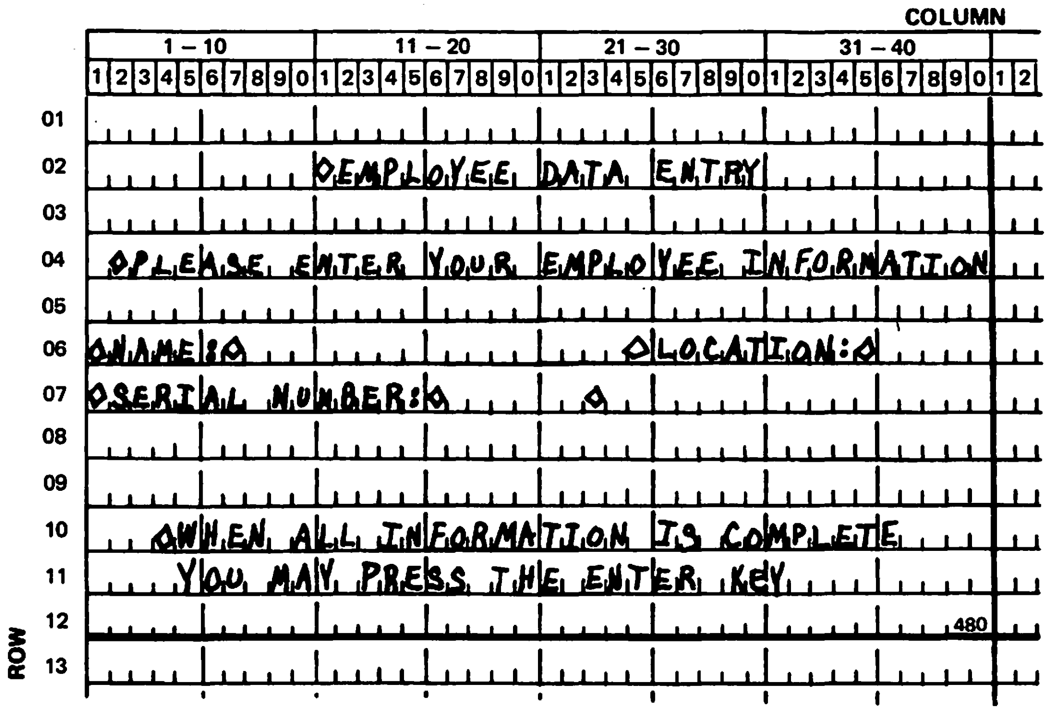


Figure 5-35. Example of Data Entry Panel

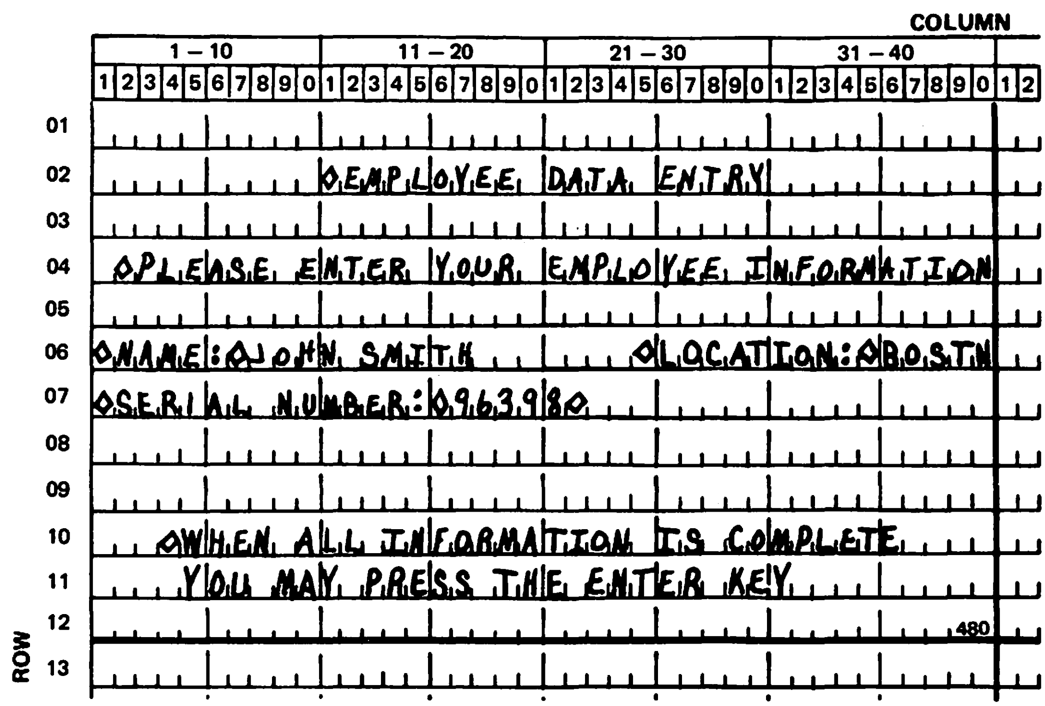


Figure 5-36. Data Entry Panel with Entered Data

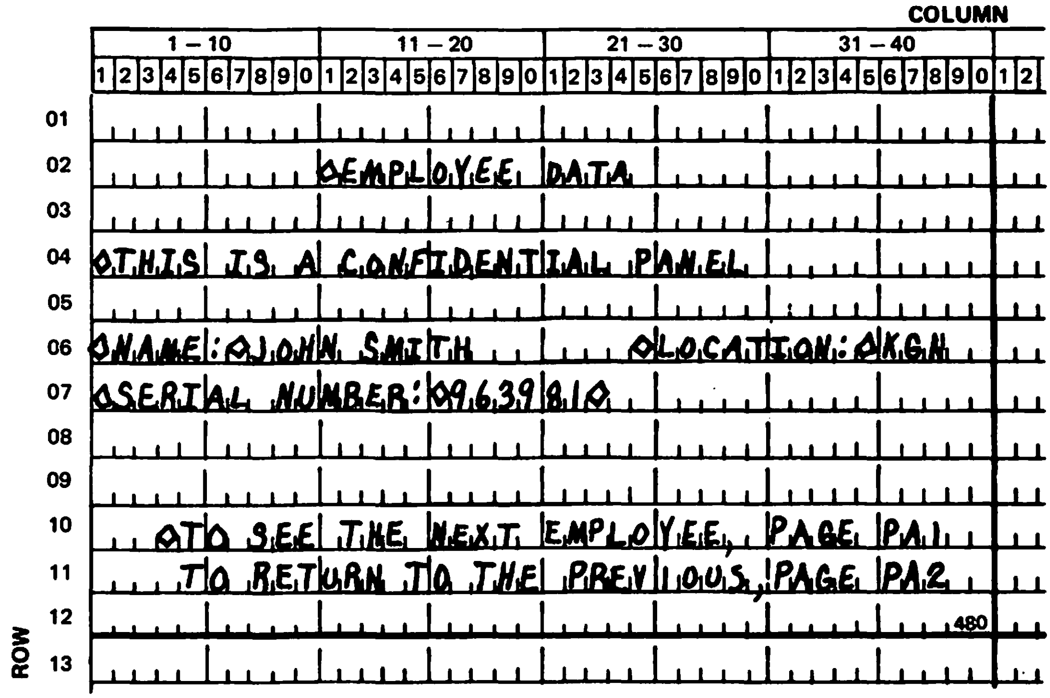
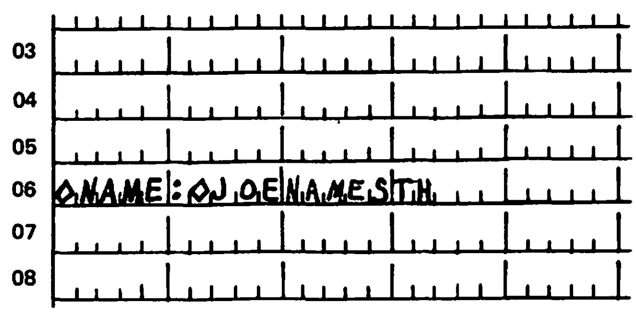


Figure 5-37. Employee Data Panel

When ready to view the information for the next employee, press the PA1 key. Since you want to modify only the present panel, not erase it or blank the unprotected fields, you request a Write command with a WCC to unlock the keyboard. Because you are not sure of the present buffer address, you might begin with an SBA order to R6, C8, followed by the next employee name from the disk file – JOE AMES. Because this name contains fewer characters than JOHN SMITH, the screen would look like this if you did not clear the remainder of the field:



You must also place the location code at location R6, C36. You could use blanks after the name and an SBA sequence, or EUA with its associated address. Use PT instead. Insert a PT order after the "S" in "AMES". The single PT order clears the remainder of the unprotected name field to nulls and positions for the location code. PT should also follow the location code to position for the serial number. The data stream might look like this:

WCC	SBA	R6	C36	J	C	E	A	M	E	S	PT	K	N	G	S	T	PT	9	3	9	8	2

The screen would appear as shown in Figure 5-38.

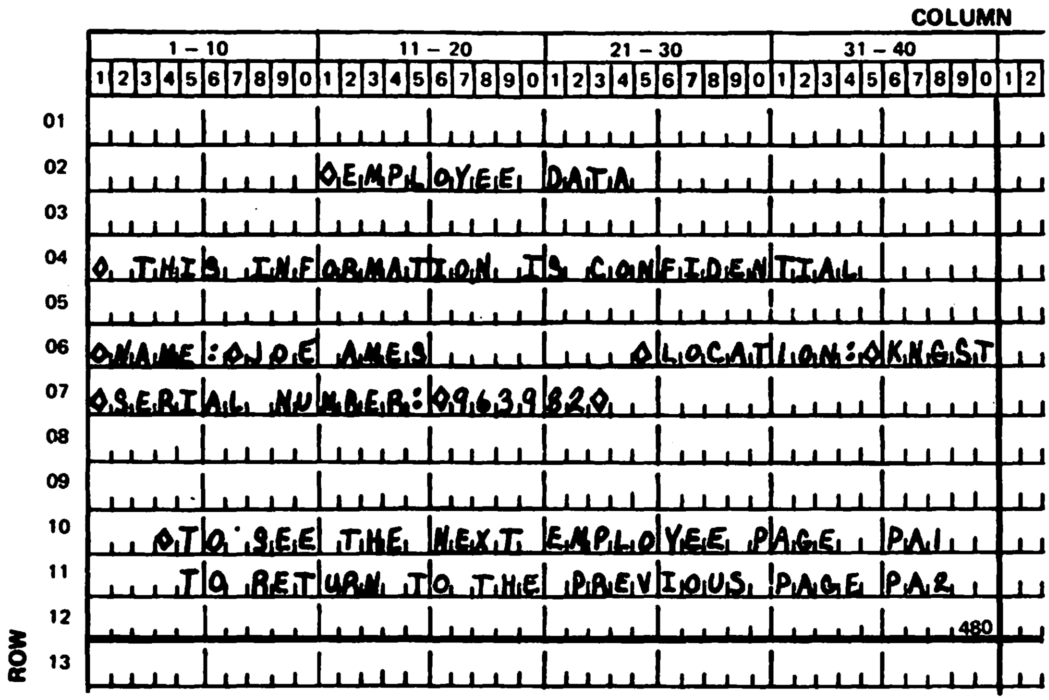


Figure 5-38. Panel Defined with Program Tab

Chapter 6. Screen Management

A screen management program module is a set of subroutines physically separate from application programs and from the telecommunications management program module of an online 3270 system. Figure 6-1 illustrates this relationship.

Support functions in a screen management program may reduce the amount of detail work required by the application programs and effectively use the features of the 3270. The separation of screen management from the other programs also allows screen management to be modified with little or no impact on application programs or on telecommunications management programs.

Screen management might include:

- Decoding input data streams.
- Dynamic building of output data streams.
- Generating multiple I/O requests to the Line Control Module based upon a single request from an application program (that is, WRITE then READ).
- Automatic paging; the application program passes multiple pages to screen management, which asks the line control module to write a particular page to a display, depending on the display operator's request.
- Automatic copying (providing a hard copy of a display image).

The BSC COPY function supports data movement between any types of device attached to the same control unit: display to display, display to printer, printer to display, and printer to printer. To prevent copying information from an unauthorized device, the control unit provides a program-controlled copy-lock for devices attached to it. If the first position of a device buffer contains a field attribute character with the protected option, the control unit rejects any attempt to copy from that device.

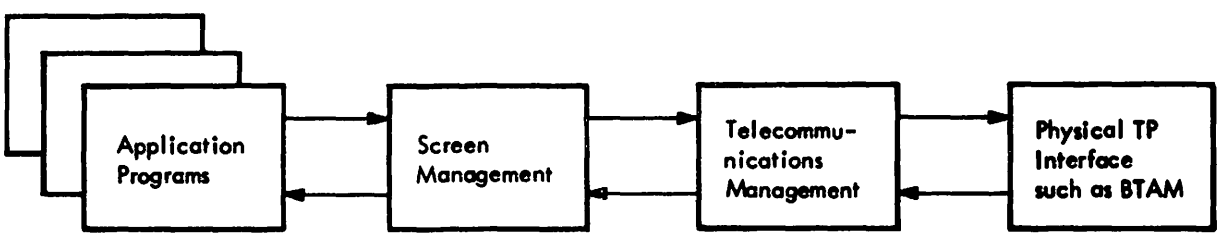


Figure 6-1. Relationship of Screen Management to Telecommunications Management and Application Program

Decoding and Generating Data Streams

The data streams sent between application programs and the 3270 contain unique orders that request particular operations by the 3270 displays and printers. Generalized subroutines can be written to assist the application programmer's interface with the 3270 system, and an interface can be built to simplify online programs.

This chapter discusses several approaches to the development of a screen management module whose functions can be used by the application programmer to prepare output data streams and to decode input data streams. The approaches demonstrate how some 3270 device-dependent considerations can be removed from the application programmer's responsibility. The different techniques for 3270 input or output data stream manipulation can be used in various combinations to suit the needs of the installation.

This discussion assumes that the device management routines (line control) make the local and remote 3270 transparent to the application program. Therefore, discussion of data streams in this chapter ignores all header data in the input stream up to and including the AID character and all header data in the output stream up to but not including the Write Control Character (WCC).

Decoding Read Modified Input Data Stream

A Read Modified command for a display station with a formatted screen (a screen with at least one attribute character defined) produces a data stream consisting of the data from each field whose modified data tag has been turned on (either by program control or by data entered in the field). Each transmitted data field is preceded by the 3270 buffer address where that data is located on the display. The order of the fields transmitted from the screen is from left to right for each line, starting at the top of the screen and ending at the bottom of the screen. All null characters in a transmitted field are stripped out by the control unit during transmission.

The data stream, ignoring the header information up to and including the AID character, appears as:

S B A	A 1	A 2	DATA	S B A	A 1	A 2	. . .

If the data entered in a field is of variable length or if a field can be skipped by the terminal operator, the data from a particular field on a given panel can appear in a different location within the data stream for each set of operator input. A Read Modified command produces a variable-length data stream of fixed-length fields and variable-length fields concatenated together.

Each 2-character screen address in the data stream is immediately preceded by a Set Buffer Address (SBA) order. The detection of each SBA order in the data stream identifies the next 2 characters in the stream as a 3270 screen address and also indicates the end of the preceding data field. The System/360 and System/370 Translate and Test instruction (TRT) can be used to scan the data stream and to stop at each main storage address containing an SBA order. If the detected main storage address of the current SBA order is known, the following calculations can be performed for a given data stream:

```
SBA(1), ADD(1A), ADD(1B), DATA FIELD (1),
SBA(2), ADD(2A), ADD(2B), DATA FIELD (2),
SBA(3),
```

The numbers in parentheses are used as subscripts to provide unique identification:

- The length of data field (1) = (Address of SBA(2) - Address of SBA(1)) -3.
- The 2-character 3270 screen address of data field (1) can be found at the address of SBA(1) +1.
- The length of data field(2) = (Address of SBA(3) - Address of SBA(2)) -3.
- The 2-character screen address of data field(2) can be found at the address of SBA(2) +1.

The 2-character 3270 screen address as it appears in the input stream does not provide a direct decimal or binary numeric value that can be used to calculate the relative position in the 3270 buffer from which the data was read. However, you can use the following routine to convert the 3270 address as it appears in the input data stream to a binary value which directly indicates the position (relative to zero) of the data in the 3270 buffer.

Assume that R3 contains the address of SBA(1) and that R4 and R5 are work registers. R5 will contain the result at the end of the routine.

```
ADDCNVRT  EQU *
          SR  R4, R4        CLEAR WORK REG
          SR  R5, R5        CLEAR WORK REG
          IC  R4, 0(R3)     GET FIRST ADDRESS CHAR (ADD (1A))
          N   R4, = F'63'   TURN OFF ALL BITS EXCEPT LAST SIX
          IC  R5, 1(R3)     GET SECOND ADDRESS CHAR (ADD (1B))
          N   R5, = F'63    TURN OFF ALL BITS EXCEPT LAST SIX
          SLL R4, 6         SHIFT FIRST ADDRESS SIX BITS TO THE LEFT
          AR  R5, R4        ADD THE RESULTS TOGETHER
```

By using the above technique, several approaches may be developed to a general purpose subroutine that decodes the variable-field-length data stream for the application program, and returns the data in a more easily processed format.

DISPLAY BUFFER IMAGE TECHNIQUE: By using the Read Buffer command you can use the display buffer image technique to return to the application program a main storage buffer area the same size as the display buffer (480, 960, 1920, 2560, 3440). The data read from the display is placed in the same relative position in the main storage buffer as it occupied in the display buffer, with all other positions in the returned buffer cleared to spaces.

For this technique, use the TRT instruction and the 3270 address conversion routine. You must know the relative locations in the display buffer where data can be entered by the operator, so that the decoded buffer can be processed when returned by the mapping subroutine. The completed layout sheet for the panel in which the operator enters data will give you the required addresses relative to the respective buffers.

Using the image technique, all data received from the 3270 is left-justified in its respective fields. This has no effect on fixed-length fields, variable-length alphameric fields (which are normally left-justified), or on omitted input fields. However, you must be aware of variable-length numeric fields where the operator can omit leading zeros.

Although the image technique requires little main storage for the mapping subroutine, main storage can be wasted if the routine returns a complete buffer with little data. To help overcome this problem, the decoding routine can pass back to the application program a field at the beginning of the buffer. The field indicates the total length of the buffer, which allows the decoding routine to use a buffer area just large enough to accommodate the relative address of the last data field read.

MAPPING FROM A TABLE OF REQUIREMENTS: This mapping technique requires a table assembly for each unique input panel that the mapping subroutine decodes for the application program. The table provides information to the subroutine so that the input data stream in one main storage buffer can be decoded a field at a time and moved to a specified relative offset in another main storage buffer (the target buffer) according to the directions assembled in the table. The preassembled table could be used to specify the following information to the mapping subroutines:

1. The 3270 buffer address preceding each field, which could be read from a particular panel. This is the buffer address as it appears in the data stream which corresponds to the first data position in a field, not to the buffer location of the field attribute byte that defines the field. Any data fields in the 3270 input stream that do not have a matching buffer address in the table would be ignored by the typical mapping routine using the table approach.

2. An offset relative to zero that provides the starting position of each field in the target buffer. This information allows the application programmer to order the fields in the target buffer in a sequence that may or may not agree with the field sequence in the transmitted data stream.

3. A value that indicates the maximum length of each field in the target buffer. This information allows the mapping routine to truncate data stream fields that are too long for the target fields. The maximum field length value is also required if the mapping routine supports right-justification of fields during mapping.

4. A flag byte consisting of bit switches that could indicate:

   - Whether left justification with low-order blank padding is requested.
   - Whether right justification with high-order zero fill is requested.
   - Whether the field should be translated to ensure uppercase characters only.
   - Any additional functions the installation wishes to implement in the mapping routine.

Figure 6-2 shows some typical logical contents of the table. The order of the elements within each table entry is optional.

Assume that you map the following input data stream in hexadecimal using the sample table in Figure 6-2:

```
1140D4F1F2F31140E8818283848511C1C6E385A7A3
```

The following target buffer, also in hexadecimal, would be returned to the application program:

```
C1C2C3C4C54040404040F0F0F1F2F3E385A7A34040
```

This approach to mapping makes the application program's input processing routine device-independent.

Instead of the mapping table, you could write a macro instruction to prepare the table; the macro would convert written requests into the proper machine language constants.

A typical format for a macro instruction to build the sample table shown in Figure 6-2 might be:

```
MAP      NAME=TABLE,MODEL=2
MAP      ADD=(1,21),OFFSET=11,MAXL=5,JUST=RIGHT
MAP      ADD=(1,41),OFFSET=1,MAXL=10,JUST=LEFT,TRAN=YES
MAP      ADD=(1,71),OFFSET=16,MAXL=6,JUST=LEFT
```

Note: *The ADD parameter specifies the 3270 buffer in row and column notation relative to 1. For example, buffer position 0 equals row 1, column 1. The offset values are expressed relative to 1. The macro instruction can have default options; for example, if JUST=RIGHT is not specified, JUST=LEFT can be assumed.*

```
TABLE      DS OH
ENTRY 1    DC X'40D4'  ACTUAL 3270 ADDRESS FOR POS 20
           DC H'10'    RELATIVE OFFSET IN TARGET BUFFER
           DC HL1'5'   MAX FIELD LENGTH OF TARGET FIELD
           DC X'80'    RIGHT JUSTIFY, NO TRANSLATE FLAG
ENTRY 2    DC X'40E8'  ACTUAL 3270 ADDRESS FOR POS 40
           DC H'0'     RELATIVE OFFSET IN TARGET BUFFER
           DC HL1'10'  MAX FIELD LENGTH OF TARGET FIELD
           DC X'40'    LEFT JUSTIFY, TRANSLATE FLAG
ENTRY 3    DC X'C1C6'  ACTUAL 3270 ADDRESS FOR POS 70
           DC H'15'    RELATIVE OFFSET IN TARGET BUFFER
           DC HL1'6'   MAX FIELD LENGTH OF TARGET FIELD
           DC X'00'    LEFT JUSTIFY, NO TRANSLATE FLAG
ENDOLIST   DC X'FF'    END OF LIST INDICATOR
```

Note: *3270 buffer addresses in the table are shown relative to buffer location 0; relative offsets in the target buffer are shown relative to 0.*

Figure 6-2. Table of Requirements

The following example shows the logic flow for a table-driven input mapping technique:

1. Find the 3270 buffer address of a data field to be processed in the input data stream using the TRT instruction.

2. Determine the length of the data field in the data stream using the techniques discussed in this section.

3. Search the table of requirements, using the 3270 buffer address found in step 1 as a search argument to find a matching entry.

4. Add the offset value from the entry found in the table to the starting address of the main storage map buffer to produce the main storage address of the start of the receiving field.

5. If the length of the data field determined in step 2 is greater than the maximum field length value in the entry found in the table, go to step 10.

6. Check the flag byte in the entry found in the table. If left justification is requested, go to step 10. Otherwise proceed to step 7 for right justification.

7. Move zoned decimal zeros to the receiving field, using the field starting address determined in step 4. Use the maximum field length value in the entry found in the table as the length for the move.

8. Develop a new main storage address for the start of the receiving field to accommodate the request for right justification. The right-justified starting address for the receiving field = (field starting address determined in step 4 + maximum field length value in the entry found in the table) - length of the data field in the data stream found in step 2.

9. Move the data field from the data stream to the main storage address developed in step 8, using the length of the data in the data stream determined in step 2. Return to the start of this routine to find the next data field in the data stream.

10. Move blanks to the receiving field using the starting address of the field as determined in step 4. Use the maximum field length value in the entry found in the table as the length for the move.

11. Move the data field from the data stream to the receiving field using the field address determined in step 4. Use the length of the data in the data stream (determined in step 2) as the length for the move.

12. Check the flag byte in the entry found in the table to determine if uppercase translation is requested. If it is not requested, return to the start of this routine to find the next data field in the data stream.

13. Translate the data in the receiving field to uppercase; then return to the start of this routine to find the next data field in the data stream. The translation can be done in two ways:

    - Use the Translate instruction with the translation table built to convert lowercase alphabetic characters to uppercase.
    - Use the OR instruction to place blanks in the field. This will change the DUP and FM characters. The FM appears as a semi-colon (;) on the screen, but appears in the data stream as X'1E'. It will be converted to a true semi-colon (;), that is, X'5E'. The DUP appears as an asterisk (*) on the screen, but appears in the data stream as X'1C'. It will be converted to a true asterisk (*), (X'5C').

Immediate Selector Pen or Cursor Select Data Stream

When a Read Modified command is executed for a display station as a result of an immediate detection by the selector pen or cursor select, the resulting data stream consists of address strings that identify which fields on the screen have the modified data tag set; the 3276 control unit also transmits the modified data if the proper designator character is used.

The data stream, ignoring the header information up to and including the AID character, appears as:

SBA	A1 A2	SBA	A1 A2	. . .

If the operator keys into a field and an immediate selector field is selected, the keyed data is not transmitted. However, if keyed data is entered by the operator, delayed selector fields are selected, and the ENTER key or a PF key is pressed; then the address and data for all fields, whether selected or keyed, are included in the data stream.

You can use a subroutine to free the application program from determining which fields were selected on a panel. A table can be built that consists of the 3270 buffer addresses, giving the location of each selectable field on a panel. The mapping routine can then compare the addresses in the table, and return to the application program a list of indicators that identifies the selected fields.

The list of indicators can be returned to the application program. A string of one-position fields can be used, and each position can indicate with a unique character that a field was selected. The first position in the returned list can be marked if a field in the data stream has the same address as the first element in the address table; the second position in the returned list can be marked if a field in the data stream has the same address as the second element in the address table. The application program can then determine which relative positions in the list have been marked to determine which fields have been selected by the operator.

Because the input from a display using selector pen or cursor select detection is a series of fixed-length addresses, the mapping routine can analyze the input stream and decode it.

For example, using the selector panel illustration in Figure 6-3, assume that the operator has selected the delayed-detectable fields located at row 5, column 10 and row 3, column 26 and the immediate-detectable field located at row 7, column 18. The input data stream transmitted in hexadecimal from the display would be:

```
11C1E911C2E911C4C1
```

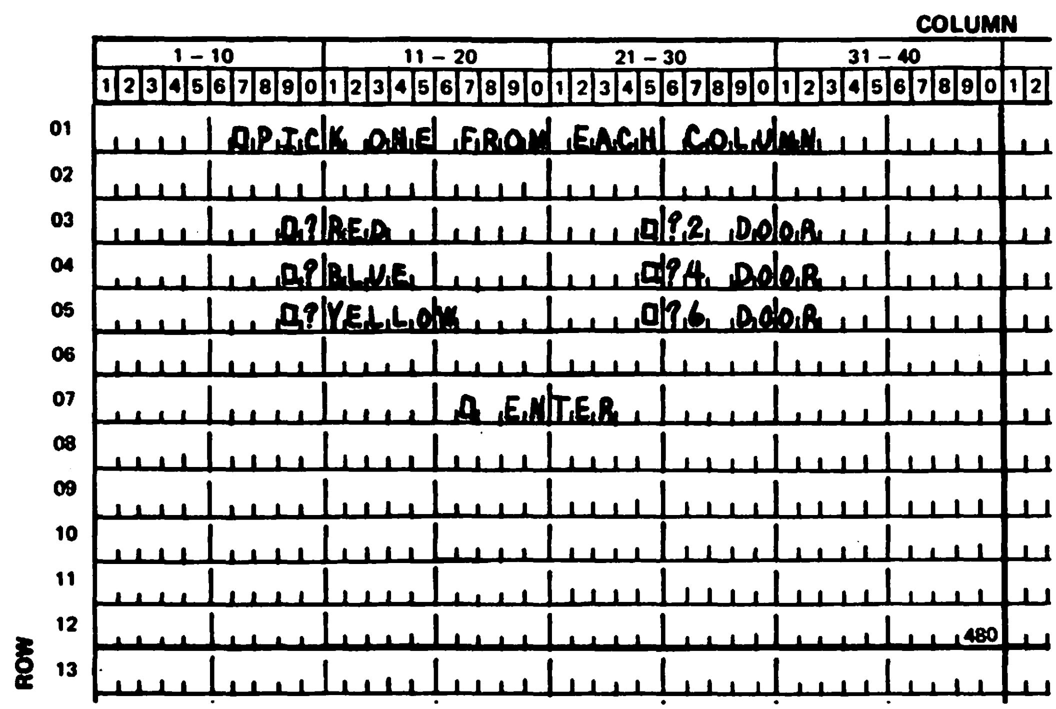


Figure 6-3. Example of Selector Pen Panel

Using the sample table in Figure 6-4, the mapping routine returns a list in hexadecimal to the application program:

```
406F40406F406F
```

This list indicates that the second, fifth, and seventh fields were selected. Note that the addresses of the selected fields appear in the data stream in the same sequence as the fields appear in the display buffer. When a selector pen panel is designed by columns, the address of the field selected from the first column may not occur before the address of the field selected from the second column in the input data stream.

You can write a macro instruction similar to the one used to build the table in Figure 6-2 to build the selector pen table:

```
MAP      NAME=SELTABLE,MODEL=1
MAP      ADD=(3,10)
MAP      ADD=(3,26)
MAP      ADD=(4,10)
. . .
```

```
SELTABLE   EQU *              FOR MODEL 1 DISPLAY
           DC  X'C1D9'        ROW 3 COL 10
           DC  X'C1E9'        ROW 3 COL 26
           DC  X'C2C1'        ROW 4 COL 10
           DC  X'C2D1'        ROW 4 COL 26
           DC  X'C2E9'        ROW 5 COL 10
           DC  X'C2F9'        ROW 5 COL 26
           DC  X'C4C1'        ROW 7 COL 18
           DC  X'FF'          TABLE STOP INDICATOR
```

Note: *The 3270 addresses used in the above table correspond to the buffer position of the Selector Pen designator character in a field, not to the location of the field attribute character which defines the field.*

Figure 6-4. Sample Mapping Table

Mixed Read Modified Input Data Streams

When some keyed input and some delayed selector pen or cursor select detection occur in a panel during the same input operation from a display, you can use the table-driven mapping technique for nonselector pen or cursor select panels. Specify the table elements so that all delayed selector fields have a maximum length of one character. The mapping routine places the first character from the appropriate data stream field into the target field. The first character in a delayed selector field that has been selected is always a (>); that is, X'6E'. The application program can examine the target buffer for that character in the proper target field to determine if the field has been selected.

Building Output Data Streams

The 3270 requires specific bit patterns for order sequences, control characters, and buffer addressing. The data streams can be prepared in several different ways. A data stream to build a static panel (a panel which will always be displayed in exactly the same manner) can be assembled in an application program as a set of data constants. A semidynamic panel, which may occasionally be modified or added to, can have the static portion assembled in the application program and have the program dynamically modify or add to the data stream. A data stream for a dynamic panel (a panel with a high degree of change) must be created or assembled as a unit at execution. This section discusses how to reduce the considerations of device-dependency required to support static, semidynamic, and dynamic output data streams.

Static Data Streams

You can write macro instructions to simplify the preparation of static data streams for the 3270. One approach is to write a set of macro instructions in which each macro instruction prepares a single order sequence. Another approach is to write one macro instruction that can prepare all types of order sequences, but prepares only one sequence for each execution of the macro instruction in a program.

A sample macro instruction of the first type might be:

```
$MOD MODEL = 1, 2, 3, 4, 5
```

This macro instruction sets a global value so that the specified model number is used until another $MOD macro instruction is encountered. The model number is required to correctly calculate 3270 buffer addresses. The buffer address X'C2D5' represents column 4, row 30 for a Model 1 display, and column 2, row 70 for a Model 2 display.

The following are also examples of the first type of macro instruction:

$SBA (1, 10) generates the SBA order sequence X'1140C9'

$SF (PROT,NUM,SKIP,MDT,HI,DET,NONDISP)

generates an SF order (X'1D') followed by the appropriate attribute character defined by the options selected in parentheses. Notice that if PROT is not specified, unprotected is assumed; if numeric is not specified, alphameric is assumed.

$RA (1,10,'*') generates the RA order sequence X'3C40C95C'.

$EUA (1, 10) generates an EUA order sequence X'1240C9'.

$WCC (RESET,RESTORE,ALARM,PRINT,40CHAR,64CHAR,80CHAR,NLEM)

generates the proper WCC, depending on the options selected in parentheses.

$CCC (PRINT,40CHAR,64CHAR,80CHAR,ALARM,ATTR,UNPROT,PROT,ALL)

generates the proper copy control character (CCC), depending on the options selected in parentheses. (The CCC identifies the type of data to be copied.)

$IC generates X'13'

$KBD KEYBOARD = APL or Text

Used with the Data Analysis feature to identify the keyboard providing 3277-2 display input.

$SI generates the Suppress Index character, valid for the 3289 printer. Other printers receive | (the or bar) in place of the Suppress Index character.

After you have defined the macro instruction, the data stream required to build the sign-on panel shown in Figure 5-6 could be created as follows:

```
SIGNON     $MOD     MODEL=1
           $WCC     (RESET,RESTORE)
           $SBA     (2,11)
           $SF      (PROT)
           DC       C'SIGN-ON PROCEDURE'
           $SBA     (4,2)
           $SF      (PROT)
           DC       C'PLEASE ENTER YOUR SIGN-ON INFORMATION'
           $SBA     (6,1)
           $SF      (PROT,HI)
           DC       C'NAME:'
           $SF
           $IC
           $SBA     (6,25)
           $SF      (PROT,HI)
           DC       C'LOCATION:'
           $SF
           $SBA     (7,1)
           $SF      (PROT,HI)
           DC       C'SERIAL NUMBER:'
           $SF      (NUM)
           $SBA     (7,23)
           $SF      (PROT)
           $SBA     (10,4)
           $SF      (PROT)
           DC       C'WHEN ALL ... ENTER KEY'
```

You could also write the second type of instruction, a single 3270 data stream macro instruction, which might have the format:

```
                            ,(attributes)                              1
 [symbol] $MAC op-type ,(row,column) [,character] ,MODEL= 2
                                                                       3
                                                                       4
```

symbol
specifies a symbol that refers to the data stream

op-type
specifies the type of screen control operation to generate. Valid values are: SF, SBA, IC, RA, EUA, WCC, and CCC.

(row,column)
specifies the row (1 to 43) and column (1 to 132) where the operation starts or ends (depending on the op-type). This parameter is required for op-types SBA, RA, and EUA.

(attributes)
indicates attributes or control bits for SF, WCC, and CCC.
Some valid values for SF are: PROT, SKIP, NUM, MDT, HI, DET, NONDISP.

Some valid values for WCC are: RESET, RESTORE, ALARM, PRINT, 40CHAR, 64CHAR, 80CHAR, NLEM.

Some valid values for CCC are: PRINT, 40CHAR, 64CHAR, 80CHAR, ALARM, ATTR, UNPROT, PROT, ALL.

character
specifies the character used in the RA function.

MODEL=
indicates the model of 3270. This model number is used to calculate the buffer address. This parameter is specified once in the first macro instruction of a data stream series or whenever the data stream to be generated is for a different model than the preceding series. Model numbers 3 and 4 can be specified only for the 3278 Display Station.

After you have defined the macro instruction, the data stream required to create the sign-on panel shown in Figure 5-6 could be as follows:

```
SIGNON    $MAC    WCC,(RESET,RESTORE),MODEL=1
          $MAC    SBA,(2,11)
          $MAC    SF,(PROT)
          DC      C'SIGN-ON PROCEDURE'
          $MAC    SBA,(4,2)
          $MAC    SF,(PROT)
          DC      C'PLEASE ENTER YOUR SIGN-ON INFORMATION'
          $MAC    SBA,(6,1)
          $MAC    SF,(PROT,HI)
          DC      C'NAME:'
          $MAC    SF
          $MAC    IC
          $MAC    SBA,(6,25)
          $MAC    SF,(PROT,HI)
          DC      C'LOCATION:'
          $MAC    SF
          $MAC    SBA,(7,1)
          $MAC    SF,(PROT,HI)
          DC      C'SERIAL NUMBER:'
          $MAC    SF(NUM)
          $MAC    SBA,(7,23)
          RMAC    SF,(PROT)
          $MAC    SBA,(10,4)
          $MAC    SF,(PROT)
          DC      C'WHEN ALL ... ENTER KEY'
```

These two types of macro instructions can generate either a total static data stream or static sections of data streams that can be dynamically assembled at execution by the application program.

Semidynamic Output Streams

A semidynamic panel requires some dynamic modification. Perhaps an error message must be written to a particular part of the panel and the cursor must be moved to the input field in which an error was detected during editing. The application program can concatenate preassembled static data stream segments into the program, such as field error messages. The same macro instructions that build static data streams can build partial static streams. As the input from a panel is edited, the standard error message for each field can be assembled in the output buffer, thus allowing multiple brief messages to be sent to the display in one operation.

You may have to change one or two attribute characters from high intensity to low intensity and erase the unprotected fields on a display. For example, an error message segment may have changed a field to high intensity to call the operator's attention to the field; the operator has recognized the error and reentered the correct information. The display must now be made ready for the next input on the panel. Concatenate the order stream segments to change the attribute characters and use the Erase Unprotected to Address (EUA) order to restore the panel; do not transmit all the data and orders to completely refresh the panel.

Dynamic Output Streams

It may become physically impossible to hold in main storage all possible output data and order stream combinations that could occur during execution of an application. You can incorporate a subroutine into screen management to accept parameters from an application program to decode the parameters and to create the data stream. You can also write for the application program a macro instruction that builds a parameter list inline from entries you specify in the macro instruction, and then branches to the screen management routine to build the required orders and data in the buffer area.

The macro instruction could appear as follows:

```
$BUILD ADD=ADDFIELD,ATTR=(R3),DATA=(R4),LEN=(R5)
```

The ADDFIELD contains the 3270 buffer address in either row-column format, binary offset, or 3270 address form. R3 contains the address of the attribute byte, R4 contains the address of the data to be entered in the field, and R5 contains the length of the data. The attribute character parameter is optional.

The subroutine could convert row and column buffer addresses relative to 1 to decimal offsets relative to 0 with the following formula:

```
Model 1 Buffer:           ((R-1)X40)+(C-1)
Model 2, 3, 4 Buffer:     ((R-1)X80)+(C-1)
```

If the row and column buffer addresses relative to 1 are in two single-byte areas in binary, the conversion to binary offsets relative to 0 can be coded as follows:

```
SR      R3,R3
IC      R3,COLUMN
BCTR    R3,C
SR      R4,R4
IC      R4,ROW
BCTR    R4,0
MH      R4,=H'40' USE VALUE OF 80 FOR MODEL 2
AR      R4,R3   RESULT IN R4
```

The following subroutine converts a binary halfword that represents the offset relative to 0 of a position in a 3270 buffer to an equivalent 2-character 3270 address. R3 is a work register, and R4 points to the binary halfword to be converted. The converted result is found at ANSWER.

```
         LH      R3,0(R4)
         STC     R3,ANSWER+1
         SRL     R3,6
         STC     R3,ANSWER
         NI      ANSWER+1,X'3F'
         TR      ANSWER(2),TAB
         .
         .
         .
ANSWER   DC      X'0000'
TAB      DC      X'40C1C2C3C4C5C6C7C8C94A4B'
         DC      X'4C4D4F4F50D1D2D3D4D5D6D7'
         DC      X'D8D95A5B5C5D5E5F6061E2E3'
         DC      X'E4E5E6E7E8E96A6B6C6D6E6F'
         DC      X'F0F1F2F3F4F5F6F7F8F97A'
         DC      X'7B7C7D7E7F'
```

Large Screen Size

Application programs written for systems that use 480- or 1920-character screen size will run on large screen displays with the same width but with a greater number of lines. Terminals with large screen capacity (960, 2560, and 3440 characters) will automatically default to smaller screen size unless the large screen size has been specified explicitly by the application program. The Erase Write Alternate command is used to switch a display into large screen mode.

Since buffer address wrapping is screen-size-dependent, application programs should not depend on buffer wrap during write operations. Also, field attributes must be appropriately placed to delimit the end of the screen image.

Appendix A. Indicators and Controls

This appendix describes the function of switches, controls, and lights on the operator's panel, and symbols displayed in the Operator Information Area (Figure A-1 and A-2).

3276 Switches, Controls, and Lights

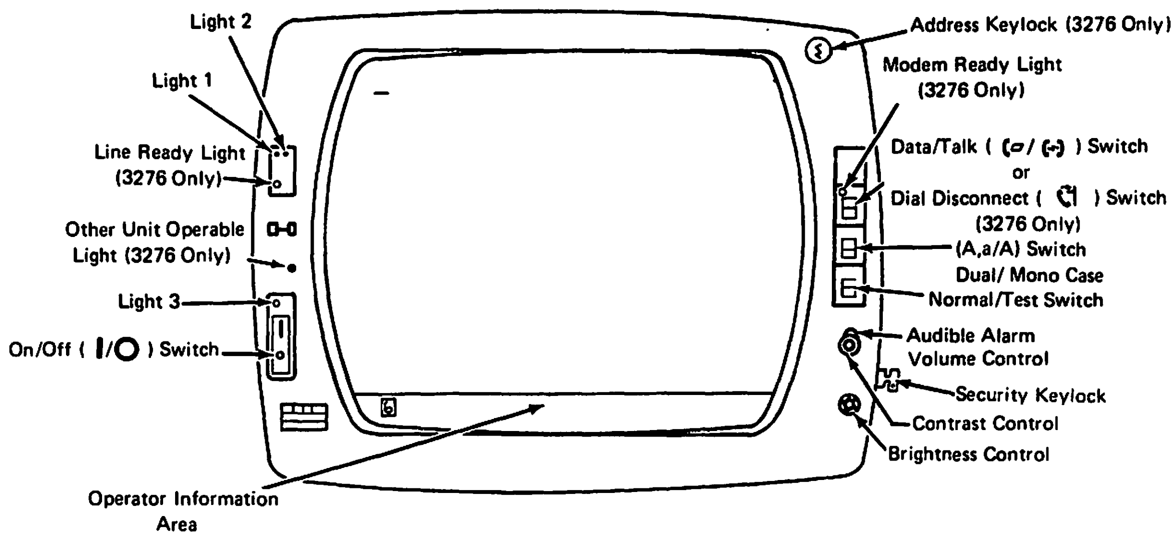


Figure A-1. 3276 Operator Panel

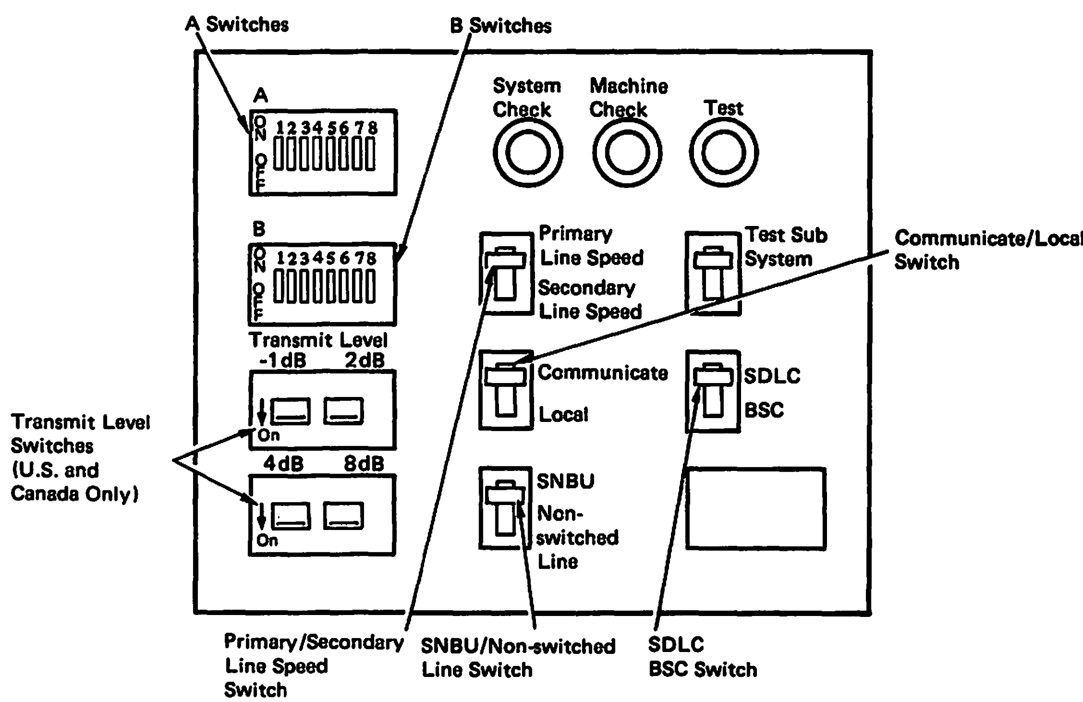


Figure A-2. 3276 Operator Drawer Panel

The following listing describes the function of the indicators and controls shown in Figure A-1 and Figure A-2.

Audible Alarm Volume Control. This control allows adjustment of the audible alarm, when the Audible Alarm feature has been installed on the 3276. The audible alarm tone amplifier control is attached to the Contrast Control, located below the Normal/Test switch near the lower-right corner of the CRT.

Brightness Control. This is a dual-function control. Rotating the control clockwise increases CRT brightness. On the 3276 display, rotating the control completely clockwise and holding the control places the control in test intensity override position, which unblanks the CRT screen. The Brightness Control is located near the lower-right corner of the CRT.

Contrast Control. The Contrast Control controls CRT contrast and is located above the Brightness Control.

Modem Ready Light. This indicator is turned on when the Data Set Ready signal is received from the modem. It is located above the Dial Disconnect switch or Data/Talk switch on the right side of the CRT.

Data/Talk. This switch, located on the right side of the CRT, is used to switch from talk mode to data mode, or vice versa, when integrated modem is operated in the switched network.

Dial Disconnect Switch. This switch is used to terminate a switched network call.

Dual/Mono Case Switch. When in the Mono Case (A) position, only uppercase characters are displayed. When in the Dual Case (A,a) position, uppercase and lowercase characters can be displayed. This switch is located on the right side of the CRT.

Light 2. This indicator should light after power is applied. It is located in the upper-right position on the left side of the CRT.

Line Ready: This indicator lights when the communication line is functioning correctly. In BSC operating mode, it is turned on when a polling or selection sequence is received and is turned off after 3 seconds if character synchronization is not achieved in control mode, or when a Machine Check condition caused by an integrated modem is detected. The Line Ready indicator is located below the Light 2 indicator on the left side of the CRT. The light is turned off when 8 seconds elapse without receiving the controller address. It is also turned off by depressing Test Subsystem (3276), or when a Machine Check condition is caused by MC/PC error except DTA card or caused by CCA or a modem error.

Normal/Test. This switch, when placed in the Test position, disconnects the 3178, 3278, 3279, or 3276 display from the attached 3276 to allow testing operations. The Normal/Test switch is located on the right side of the CRT.

Light 3. This indicator lights when normal power is available in the unit. It is located above the Power On/Power Off switch on the left side of the CRT.

Other Units Operable. This indicator lights when at least one display station or terminal printer, attached to the 3276 is operable. The indicator is turned off when all attached devices are powered off or are disconnected from the 3276 because of a malfunction.

Power On/Power Off. This switch applies and removes internal power.

Light 1. This indicator is located to the left of the Light 2 indicator on the upper-left side of the CRT. It should light after power is applied.

The following listing describes the function of the indicators and controls shown in Figure A-2:

BSC Address. The BSC address is established by the setting of five switches, positions 1 through 5 of switch B on the operator panel drawer on the right side of the CRT.

Communicate/Local. This switch connects or disconnects the 3276 to or from a loop of the 8100 Information System and the 4300 Processor. When this switch is in the Communicate position, the 3276 is connected to a loop; when it is in Local, the 3276 is disconnected from a loop and the Line Ready indicator turns off. Whenever this switch is turned to Communicate, a wrap test is performed for the 3276 loop adapter.

Half Duplex/Full Duplex. This switch is located at position 6 of switch B on the operator drawer. With the switch in the on position (full duplex), the Request to Send (RTS) signal is held on in SNA/SDLC operation. When operating in BSC, the RTS signal is turned on when the End of Transmission (EOT) signal is received and is turned off upon transmission of the EOT signal. When the switch is in the off position (half duplex), the RTS signal is turned on when the EOT signal is received, and is turned off at the beginning and end of each transmission of the text block.

Machine Check. This indicator lights when a nonprogramming recoverable error is detected in the 3276.

NRZ/NRZI. This switch is located at position 8 of switch B on the operator drawer. When the switch is in the ON position, transmission and reception are in NRZ mode. When in the OFF position, NRZI mode is used.

Primary Line Speed/Secondary Line Speed. This switch is used to select secondary speed.

SDLC Address. The SDLC address is established by the setting of eight switches, positions 1 through 8, on switch panel A, on the operator panel drawer.

SDLC/BSC. This switch is set according to the line discipline selected.

Set Primary/Secondary Loop Speeds. The primary and secondary loop speeds are set by the combination of the settings of switch positions 6 through 8, which are located on switch panel B on the operator panel drawer. The following combinations are provided:

Switch Position			
6	**7**	**8**	**Primary/Secondary Speeds**
OFF	OFF	OFF	9600/4800
OFF	ON	OFF	9600/2400
OFF	OFF	ON	4800/2400
ON	OFF	OFF	2400/1200
ON	OFF	ON	1200/ 600

Note: *Loop speeds depend on the 8100 system. A decal is provided below switch panel B.*

Switched Network Backup (SNBU). This switch is used to switch from nonswitched line operation to switched line backup mode.

System Check. This indicator is turned on when a program check or communication check is detected.

Test Indicator. This indicator lights under two conditions:

1. When the 3276 and attached devices have been placed in test mode, the Test indicator lights and remains on while in test mode.

2. When the 3276 detects a loss-of-carrier condition, the Test indicator blinks and the Line Ready indicator goes off.

Test Subsystem. This momentary switch is used to test the 3276 subsystem. Pressing and releasing the switch starts the subsystem test.

Transmit Level (U.S. and Canada only). These four switches are used to match the transmit level between an integrated modem and the protective coupler that is attached to the telephone line.

The following lists and explains the symbols displayed in the 3178, 3276, 3278, and 3279 Operator Information Area.

Readiness and System Connection Symbols (locations 1 through 6)

Symbol	Name	Explanation
$\boxed{6}$	3276 Ready	The appropriate ready symbol is displayed in location 1 of the Operator Information Area when the 3276 $\boxed{6}$ control unit to which the display is attached is ready (functional) and the display is ready.
$\underline{A}$ $\underline{B}$	Online A Online B	The Online $\underline{A}$ and Online $\underline{B}$ symbols govern transactions with the host system. Certain keyboard functions and the meaning of some Operator Information Area symbols differ depending upon which set of rules are applicable. <u>Online A</u>. The control unit is connected to the system under $\underline{A}$ rules. The $\underline{A}$ symbol appears in remote systems using BSC protocol. It is turned on by receipt of the following commands: Write, Erase/Write, Erase All Unprotected, Copy, Read Modified, and Read Buffer. The $\underline{A}$ symbol is turned off when: 1. An operator action causes host communication. 2. The display station is turned off. 3. The Normal/Test switch is placed in Test. <u>Online B</u>. The control unit is connected to the system under $\underline{B}$ rules. The $\underline{B}$ symbol appears in systems that use SNA protocol. It is turned on by completion of an ACTPU/ACTLU command sequence, and is turned off by execution of DACTPU or DACTLU, including an internal DACTPU sequence, and when the Normal/Test switch is placed in Test or the TEST key is pressed.
█	My Job	The display station is connected to the operator's application program. This symbol is displayed in position 3. This symbol appears in systems that use BSC or SNA protocol. In systems using BSC, it is turned on with the $\underline{A}$ symbol, and is turned off when power is removed, and when the Normal/Test switch is placed in Test. When using SNA protocol, it is turned on when the operator's application session owns the screen.
	System Operator	This symbol is used with SNA protocol and indicates that the system operator (SSCP Control Program) session owns the display screen. Except for the ENTER key, the Program Attention keys are not functional when this symbol is displayed.
$\boxed{?}$	Unowned	The display station is connected to the system (using SNA only), but not to the operator's application program or to the system operator (control program). The SYS REQ key is used if LOGON is required. This symbol is displayed in position 3.
TEST	Test	The display station is in test mode. Test mode is initiated or terminated by pressing the TEST key while holding the ALT key. TEST is displayed in positions 3 through 6. Test procedures are described in the *IBM 3270 Information Display System: IBM 3178 Display Station Operator Reference Guide*, GA18-2128, *3276 Control Unit Display Station; Problem Determination Guide*, GA18-2014, the *IBM 3270 Information Display System: 3278 Display Station; Problem Determination Guide*, GA27-2639, and the *IBM 3270 Information Display System: Problem Determination Guide*, GA33-3051.

Do Not Enter (Input Inhibited), locations 9 through 17: All these symbols contain an "X" in position 9 (do not enter), combined with other symbols in positions 11 through 17, which define why input is disabled. The keyboard does not lock mechanically, but a change in state of the keyboard clicker (on to off, or off to on) indicates that the keyboard is disabled.

The following keys are not disabled: RESET, SYS REQ, ATTN, TEST, DEV CNCL, shift keys, ALT CURSR, CURSR BLINK, and Click keys.

During buffer transfer while executing a BSC Copy command (3274 and 3276), a limited number of keystrokes will be accepted for processing, and input is not disabled. The 3276 will queue at least two keystrokes and, if the queue is not exceeded, the keystrokes will be processed when communication with the keyboard is restored. In either case, if the capacity of the queue is exceeded, all queued keystrokes will be discarded and the What symbol is displayed.

RESET will remove the input disabled condition and restore the keyboard except when the following symbols are displayed: Time, Printer Busy, Printer Very Busy, Printer Not Working, and Security Key.

For a 3278 or 3279 display without a keyboard, a selector-light-pen or MSR operation will remove the same input disabled conditions as the RESET key. A selector-light-pen or MSR operation will not cause a reset on a 3278 or 3279 display that has a keyboard attached.

The following symbols are arranged in order of probability.

Symbol	Name	Explanation
X 	Time	Time is required for the system to perform a function. This symbol is displayed due to: 1. Line protocol requirements. 2. A keyboard that has been locked by the host; for example, during a host-initiated print operation. 3. Internal processing constraints of the control unit. When operating with SNA protocol, the keyboard will be restored and the Time symbol is removed by a WCC which contains the keyboard restore bit set to 1. If a "Change Direction" was also received, the 3276 will enter send state. However, if a CD was not received, the session will remain in receive. state when the WCC contains the Keyboard Restore bit set to 1. In this state, all keys can be used except the Program Attention and Print keys. Use of a Program Attention key will result in display of the Minus Function symbol. If a WCC which contains a Keyboard Restore bit set is not received, display of the Time symbol is determined by whether the CD has been received as follows: 1. If CD has not been received, the session will remain in receive state and the Time symbol remains displayed with keyboard locked. 2. If CD has been received, the 3274 and 3276 will enter send state; and, if the keyboard was unlocked prior to receipt of the command, the Time symbol is removed and the keyboard is restored. Otherwise, the Time symbol is replaced by the System Lock symbol. If End Bracket is received, the Time symbol is removed, the session enters contention state, and the keyboard is restored regardless of the WCC setting.

Symbol	Name	Explanation
		When using BSC protocol, the keyboard will be unlocked, and the Time symbol removed, if the WCC keyboard restore bit is on, or if the keyboard had been unlocked prior to receipt of the command. Otherwise, Time will be replaced by the System Lock symbol.
X SYSTEM	System Lock	The program has disabled the keyboard following an entry. The operator may receive a message and then press RESET to restore the keyboard. In systems that use SNA protocol, the System Lock symbol appears when the application program has replied to the last message sent by the operator and is requesting the operator to send the next message. At this time, however, the host has not unlocked the keyboard. (The Keyboard Restore bit is not set in any WCC that follows the last message from the operator.) When the System Lock symbol appears in BSC systems, the host is notified of the last AID generated.
X nn	Machine Check	The display station is not working properly. The symbol is accompanied by up to two digits [nn(3178, 3276, 3278, or 3279 attached to 3276)], which define the probable cause of the problem. Recovery procedures depend upon the type of error. Refer to Appendix C for a description of the machine-check codes. Machine check symbols are almost always reset by the operator using the RESET, SYS REQ (SNA only), or TEST keys. If the 3278 or 3279 does not have a keyboard, a selector light pen, an MSR, can be used to reset the Machine Check symbol.
X nn	Communication	An attempt is made to cause host communication or to use the MSR, or selector light pen that causes host communication, and a communication link error was detected while the Communications Reminder is displayed. Data cannot be sent. The RESET, TEST, or SYS REQ (SNA) key should be pressed. This symbol is accompanied by up to two digits [nn (3178, 3276, 3278, or 3279 attached to 3276)], which define the probable cause of the problem. (The Communication Reminder symbol is displayed as long as the condition exists.) Refer to Appendix C for a description of the communication-check codes.
X PROGnn	Program	A programming error was detected in the data received by the control unit. RESET should be pressed and the operation should be retried. This symbol is accompanied by up to two digits [nn (3178, 3276, 3278, or 3279 attached to 3276)], which define the probable cause of the problem. Refer to Appendix C for a description of the program-check codes.
X ?+	What?	The last input was not accepted. The What symbol appears when: 1. Keystrokes are being queued during an unsolicited write or buffer transfer, and the capacity of the queue is exceeded. (The queue is not processed in this case.) 2. SYS REQ was pressed while inbound processing was queued for the device.

Symbol | **Name** | **Explanation**

3. ATTN, SYS REQ, or TEST was pressed during a Time condition which was caused by internal processing constraints of the 3276.
4. The operator continued to key while the Time, Printer Busy, or Printer Not Working symbol was displayed.
5. Two conflicting operations have been attempted "simultaneously" with one operation not serviced. (For example, CLEAR and selector light pen.)
6. A dead key operation has been aborted, and a standalone accent created at the cursor location.
7. Print ID mode has been aborted. The RESET key restores the keyboard.

Because of uncertainty about what was accepted, the operator should check the contents of the screen before repeating the operation. In addition:

1. If ALT or a shift key was used, press the key again and then press RESET and retry the operation.
2. When retrying SYS REQ or ATTN, repeated use of these keys may be necessary if inbound processing is queued.

X -f — Minus Function

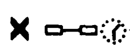
A currently unavailable function was requested. RESET should be pressed to restore the keyboard. Conditions that cause a Minus Function are:

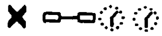
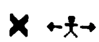
1. Use of an ATTN, PF, or PA key while in SSCP session or in " unowned state, " or prior to ACTLU. Also use of the ENTER key in the "unowned state" or prior to ACTLU.
2. Use of SYS REQ prior to receipt of ACTLU in SNA.
3. Any of the following actions in receive state with the keyboard unlocked: Print and all AID generating keys.
4. Use of ATTN while operating with remote systems that use BSC.
5. Use of SYS REQ, ATTN, and any PA or PF key that is not specified for test mode.
6. When invoking concurrent test 0, the control terminal is not the test terminal and the latter is either in session (SNA), or has the Time indicator on in systems that use BSC.
7. When using the IDENT key during a printing operation.
8. MSR in "receive state" or in "unowned state."
9. MSR in SSCP-LU session with 10-character set.

The security key is turned off and no operator input

X -f x — Minus Function Operator Unauthorized

This symbol means that the display operator has tried to change the Programmed Symbols, Color, or Extended Highlighting attributes when disallowed by the host program. The keyboard is locked as a result. Pressing the Reset key restores the keyboard.

Symbol	Name	Explanation
		The indocator is also displayed when a Programmed Symbols terminal storage is referenced (PS-A – PS-F attribute keys) but the storage has no symbol set currently associated with it, or the symbol set is marked not keyboard-selectable.
	Security Key	The security key is turned off and no operator input can be accepted. When the key is turned on, this symbol disappears, but any other pre-existing do-not-enter condition may then be displayed. RESET does not remove the Security Key symbol. The Shift key, ALT CURSR, CURSR BLINK, and Click key, and associated symbols, and all other noninput disabled symbols will function when the Security Key symbol is displayed. The Security Key has priority over other input disabled symbols except when machine checks prevent communication between the control unit and the terminal.
	Printer Not Working	The printer assigned to the display station is not functioning, and no other printers in the class are available. If this symbol appears after the Print key was pressed, and if the Printer Failure symbol is not displayed, the printer assigned to the display (or the most available printer in the class) is not functional. The print request is canceled, and the DEV CNCL key should be pressed to restore the keyboard. (RESET has no effect.) Restoration of the printer will not automatically remove the Printer Not Working symbol. If the Printer Failure symbol is displayed in the printer status area, the printer stopped during the last print operation. If the print operation was initiated by the Print key, DEV CNCL should be pressed to restore the keyboard. The display terminal indicator may precede a comparable indicator on the printer by as much as 2 minutes. The Printer Not Working symbol may also appear for a host-initiated print operation. Operators are not instructed to use DEV CNCL, but, if used, the Printer Not Working symbol is replaced with the Time symbol, and the host must continue the operation. Subsequent receipt of outbound FM data will remove the Printer Not Working symbol.
	Printer Busy	The printer assigned to the display station is busy. The operator may either wait for the printer to become available or press the DEV CNCL key. For print requests initiated by the Print key, DEV CNCL will cancel the request, remove the Device Busy symbol, and restore the keyboard. For host-initiated requests, DEV CNCL will cause Device Busy to be replaced by the Wait symbol, and a negative response will be sent to the host. If the Print key was used, it may be possible to select another printer.

Symbol	Name	Explanation
	Printer Very Busy	This symbol applies only to operator-initiated requests via the Printer key and means the same as Printer Busy except that more time than usual is anticipated before the print request is accepted. It is displayed when the requested printer is allocated to the host as follows: 1. If $\boxed{6}$ B is displayed, the printer is currently "in bracket" with a host PLU. 2. If $\boxed{6}$ A is displayed, a host Write, Erase/Write, or Copy command has been addressed to the printer, and the print operation has not yet been started by the host (via a command with the Start Print bit on in the WCC).
	Operator Unauthorized	This symbol means that the operator has requested a printer for which the terminal or attached device is not authorized. RESET should be pressed to restore the keyboard. This symbol appears when: 1. The Print key is pressed while the Printer Assignment columns of the Operator Information Area show no printer assignment or show question marks. 2. During a print ID sequence, the operator enters a number which is in the printer authorization matrix, but is not authorized for the display. 3. During a local print operation initiated by the Print key, the "printer" assigned is really a display. This can occur if an invalid device description is loaded into the printer authorization matrix. 4. The print buffer is unable to store the contents of a display buffer (for example when the display buffer is too large) during an operator-initiated local copy operation.
	Go Elsewhere	An action has been attempted which is invalid for the display screen location. RESET should be pressed and either the cursor should be moved or some other action taken. The Go Elsewhere symbol appears when: 1. An attempt has been made to enter, insert, erase, or delete a character when the cursor is in a protected field or at an attribute location. 2. An attempt has been made to use the CURSR SEL key while the cursor is not in a cursor select or selector-light-pen field.
	More Than	This symbol means that the operator has attempted to enter too much information into a field. RESET should be pressed to restore the keyboard, and the operation should be retried and the entry corrected.
	Numeric	This symbol appears when the Numeric Lock feature is installed. A non-numeric entry was made at a display screen location reserved for numeric information. RESET should be pressed to restore the keyboard, and the operation should be retried.

Symbol	Name	Explanation
X 🯅#?	What Number	The operator has entered a number which is unacceptable at the display screen location. This message appears when a selected print ID is not numeric or is not in the matrix, or an incorrect entry is made in test mode. (Refer to description of IDENT key in Chapter 3 for further information.) RESET should be pressed to restore the keyboard and to make the correct entry.
X 🯅▭?	Questionable Card	The operator tried to read an inappropriate magnetic stripe card. RESET should be pressed and the correct MSR card should be used. If a keyboard is not available, repeat the operation using a valid MSR card. This symbol will also appear if the End of Inquiry (EOI) character is present on the magnetic card. Cards with EOI are applicable to the operator identification card reader for the 3275 and 3277 only.
X 🯅´+? X 🯅`+? X 🯅^+? X 🯅¨+? X 🯅¸+?	Accent Plus What	These messages indicate that an invalid dead key/character key combination was entered (Canadian French keyboard only). RESET should be pressed to restore the keyboard, and a valid dead key/character key combination should be entered. Valid combinations are as follows: ` à À è È ù Ù ´ é É ^ â Â ê Ê î Î ô Ô û Û ¨ ë Ë ï Ï ü Ü ç Ç ¸ For further information, refer to "Dead Keys, Canadian French Keyboards" in Chapter 2.
X -S	Minus Symbol	The symbol keyed is not available. The RESET key should be pressed to restore the keyboard.
X ▭←▣	Message Received	A message from the system operator (SSCP control program) was received and rejected. RESET should be pressed to restore the keyboard. This symbol appears only on displays attached to a 3276 unit that uses SNA protocol.

Reminders (locations 21 through 27)

Symbol	Name	Explanation
—↯_nn	Communication	The communication link connecting the control unit to the system is producing errors. Refer to Appendix C for a description of the error codes. The Communication Reminder appears when: 1. The control unit detects a permanent error condition in the connection to the host. (Attempts to retry have ceased.) In this case, the reminder symbol is sent to all terminals attached to the control unit. 2. In BSC mode, a line error is detected which results in the original contents of the screen being restored and a request for retransmission made to the host. In this case, the reminder symbol is sent only to the affected terminal.
| ▭←▣	Reserved	This symbol (3178, 3276, 3278, or 3279 attached to a 3276 only) is reserved for future use and should be ignored if it is displayed.

Shifts and Modes (locations 37 through 41):

Note: *Display stations that support the Extended Data Stream feature use locations 36 through 44 for Shifts and Modes and the insert-mode symbol transfers to location 52.*

NUM	Numeric	The Numeric Lock feature is installed and the keyboard is in numeric shift, which allows use of the 0 through 9 keys, and the decimal sign, minus (–), and DUP keys only.
⇧	Upshift	The keyboard is in upshift.
^	Insert	The keyboard is in insert mode. A character may be inserted at the cursor location. Characters beyond the cursor position move to make room for the inserted character.
APL		The keyboard is in APL mode.
TEXT		The keyboard is in TEXT mode.

Printer Status (locations 60 through 64)

□–□nn	Printer Assignment	The display station is authorized to use printer address number nn. Individual printers may be assigned address numbers 1 through 7 when attached to the 3276.
□–□??	What Printer	The printer IDENT has changed. Pressing the IDENT key causes display of a new printer assignment.
□–■nn	Printer Printing	The printer identified by nn is printing information from the display station.
□–⚡nn	Printer Failure	The printer identified by nn has stopped while printing information from the display station. This symbol will remain on until: 1. The condition is cleared following operator intervention. 2. The operator uses DEV CNCL following a printer-not-functional condition. 3. Receipt of outbound FM data. 4. Printer assignment is changed because power is applied to another printer (3276 default printer authorization matrix).
□–□_ _	Assign Printer	When the operator changes the assigned printer using the IDENT key, the two numbers appear in the assignment columns, replacing the underlines.
	(nothing displayed)	If the display is attached to a 3276 (6 displayed in location 1), there is no automatic printer authorization. The operator may be able to assign a printer using the IDENT key.

Appendix B. Buffer Address I/O Interface Codes

40 Col		80 Col		Position		Buffer Address (Hex)			
R	C	R	C	Dec	Hex	EBCDIC		ASCII	
01	01	01	01	0000	000	40	40	20	20
01	02	01	02	0001	001	40	C1	20	41
01	03	01	03	0002	002	40	C2	20	42
01	04	01	04	0003	003	40	C3	20	43
01	05	01	05	0004	004	40	C4	20	44
01	06	01	06	0005	005	40	C5	20	45
01	07	01	07	0006	006	40	C6	20	46
01	08	01	08	0007	007	40	C7	20	47
01	09	01	09	0008	008	40	C8	20	48
01	10	01	10	0009	009	40	C9	20	49
01	11	01	11	0010	00A	40	4A	20	5B
01	12	01	12	0011	00B	40	4B	20	2E
01	13	01	13	0012	00C	40	4C	20	3C
01	14	01	14	0013	00D	40	4D	20	28
01	15	01	15	0014	00E	40	4E	20	2B
01	16	01	16	0015	00F	40	4F	20	21
01	17	01	17	0016	010	40	50	20	26
01	18	01	18	0017	011	40	D1	20	4A
01	19	01	19	0018	012	40	D2	20	4B
01	20	01	20	0019	013	40	D3	20	4C
01	21	01	21	0020	014	40	D4	20	4D
01	22	01	22	0021	015	40	D5	20	4E
01	23	01	23	0022	016	40	D6	20	4F
01	24	01	24	0023	017	40	D7	20	50
01	25	01	25	0024	018	40	D8	20	51
01	26	01	26	0025	019	40	D9	20	52
01	27	01	27	0026	01A	40	5A	20	5D
01	28	01	28	0027	01B	40	5B	20	24
01	29	01	29	0028	01C	40	5C	20	2A
01	30	01	30	0029	01D	40	5D	20	29
01	31	01	31	0030	01E	40	5E	20	3B
01	32	01	32	0031	01F	40	5F	20	5E
01	33	01	33	0032	020	40	60	20	2D
01	34	01	34	0033	021	40	61	20	2F
01	35	01	35	0034	022	40	E2	20	53
01	36	01	36	0035	023	40	E3	20	54
01	37	01	37	0036	024	40	E4	20	55
01	38	01	38	0037	025	40	E5	20	56
01	39	01	39	0038	026	40	E6	20	57
01	40	01	40	0039	027	40	E7	20	58
02	01	01	41	0040	028	40	E8	20	59
02	02	01	42	0041	029	40	E9	20	5A
02	03	01	43	0042	02A	40	6A	20	7C
02	04	01	44	0043	02B	40	6B	20	2C
02	05	01	45	0044	02C	40	6C	20	25
02	06	01	46	0045	02D	40	6D	20	5F
02	07	01	47	0046	02E	40	6E	20	3E
02	08	01	48	0047	02F	40	6F	20	3F
02	09	01	49	0048	030	40	F0	20	30
02	10	01	50	0049	031	40	F1	20	31
02	11	01	51	0050	032	40	F2	20	32
02	12	01	52	0051	033	40	F3	20	33
02	13	01	53	0052	034	40	F4	20	34
02	14	01	54	0053	035	40	F5	20	35
02	15	01	55	0054	036	40	F6	20	36
02	16	01	56	0055	037	40	F7	20	37
02	17	01	57	0056	038	40	F8	20	38
02	18	01	58	0057	039	40	F9	20	39
02	19	01	59	0058	03A	40	7A	20	3A

40 Col R	40 Col C	80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		ASCII	
02	20	01	60	0059	03B	40	7B	20	23
02	21	01	61	0060	03C	40	7C	20	40
02	22	01	62	0061	03D	40	7D	20	27
02	23	01	63	0062	03E	40	7E	20	3D
02	24	01	64	0063	03F	40	7F	20	22
02	25	01	65	0064	040	C1	40	41	20
02	26	01	66	0065	041	C1	C1	41	41
02	27	01	67	0066	042	C1	C2	41	42
02	28	01	68	0067	043	C1	C3	41	43
02	29	01	69	0068	044	C1	C4	41	44
02	30	01	70	0069	045	C1	C5	41	45
02	31	01	71	0070	046	C1	C6	41	46
02	32	01	72	0071	047	C1	C7	41	47
02	33	01	73	0072	048	C1	C8	41	48
02	34	01	74	0073	049	C1	C9	41	49
02	35	01	75	0074	04A	C1	4A	41	5B
02	36	01	76	0075	04B	C1	4B	41	2E
02	37	01	77	0076	04C	C1	4C	41	3C
02	38	01	78	0077	04D	C1	4D	41	28
02	39	01	79	0078	04E	C1	4E	41	2B
02	40	01	80	0079	04F	C1	4F	41	21
03	01	02	01	0080	050	C1	50	41	26
03	02	02	02	0081	051	C1	D1	41	4A
03	03	02	03	0082	052	C1	D2	41	4B
03	04	02	04	0083	053	C1	D3	41	4C
03	05	02	05	0084	054	C1	D4	41	4D
03	06	02	06	0085	055	C1	D5	41	4E
03	07	02	07	0086	056	C1	D6	41	4F
03	08	02	08	0087	057	C1	D7	41	50
03	09	02	09	0088	058	C1	D8	41	51
03	10	02	10	0089	059	C1	D9	41	52
03	11	02	11	0090	05A	C1	5A	41	5D
03	12	02	12	0091	05B	C1	5B	41	24
03	13	02	13	0092	05C	C1	5C	41	2A
03	14	02	14	0093	05D	C1	5D	41	29
03	15	02	15	0094	05E	C1	5E	41	3B
03	16	02	16	0095	05F	C1	5F	41	5E
03	17	02	17	0096	060	C1	60	41	2D
03	18	02	18	0097	061	C1	61	41	2F
03	19	02	19	0098	062	C1	E2	41	53
03	20	02	20	0099	063	C1	E3	41	54
03	21	02	21	0100	064	C1	E4	41	55
03	22	02	22	0101	065	C1	E5	41	56
03	23	02	23	0102	066	C1	E6	41	57
03	24	02	24	0103	067	C1	E7	41	58
03	25	02	25	0104	068	C1	E8	41	59
03	26	02	26	0105	069	C1	E9	41	5A
03	27	02	27	0106	06A	C1	6A	41	7C
03	28	02	28	0107	06B	C1	6B	41	2C
03	29	02	29	0108	06C	C1	6C	41	25
03	30	02	30	0109	06D	C1	6D	41	5F
03	31	02	31	0110	06E	C1	6E	41	3E
03	32	02	32	0111	06F	C1	6F	41	3F
03	33	02	33	0112	070	C1	F0	41	30
03	34	02	34	0113	071	C1	F1	41	31
03	35	02	35	0114	072	C1	F2	41	32
03	36	02	36	0115	073	C1	F3	41	33
03	37	02	37	0116	074	C1	F4	41	34
03	38	02	38	0117	075	C1	F5	41	35
03	39	02	39	0118	076	C1	F6	41	36
03	40	02	40	0119	077	C1	F7	41	37
04	01	02	41	0120	078	C1	F8	41	38
04	02	02	42	0121	079	C1	F9	41	39

40 Col R	40 Col C	80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		ASCII	
04	03	02	43	0122	07A	C1	7A	41	3A
04	04	02	44	0123	07B	C1	7B	41	23
04	05	02	45	0124	07C	C1	7C	41	40
04	06	02	46	0125	07D	C1	7D	41	27
04	07	02	47	0126	07E	C1	7E	41	3D
04	08	02	48	0127	07F	C1	7F	41	22
04	09	02	49	0128	080	C2	40	42	20
04	10	02	50	0129	081	C2	C1	42	41
04	11	02	51	0130	082	C2	C2	42	42
04	12	02	52	0131	083	C2	C3	42	43
04	13	02	53	0132	084	C2	C4	42	44
04	14	02	54	0133	085	C2	C5	42	45
04	15	02	55	0134	086	C2	C6	42	46
04	16	02	56	0135	087	C2	C7	42	47
04	17	02	57	0136	088	C2	C8	42	48
04	18	02	58	0137	089	C2	C9	42	49
04	19	02	59	0138	08A	C2	4A	42	5B
04	20	02	60	0139	08B	C2	4B	42	2E
04	21	02	61	0140	08C	C2	4C	42	3C
04	22	02	62	0141	08D	C2	4D	42	28
04	23	02	63	0142	08E	C2	4E	42	2B
04	24	02	64	0143	08F	C2	4F	42	21
04	25	02	65	0144	090	C2	50	42	26
04	26	02	66	0145	091	C2	D1	42	4A
04	27	02	67	0146	092	C2	D2	42	4B
04	28	02	68	0147	093	C2	D3	42	4C
04	29	02	69	0148	094	C2	D4	42	4D
04	30	02	70	0149	095	C2	D5	42	4E
04	31	02	71	0150	096	C2	D6	42	4F
04	32	02	72	0151	097	C2	D7	42	50
04	33	02	73	0152	098	C2	D8	42	51
04	34	02	74	0153	099	C2	D9	42	52
04	35	02	75	0154	09A	C2	5A	42	5D
04	36	02	76	0155	09B	C2	5B	42	24
04	37	02	77	0156	09C	C2	5C	42	2A
04	38	02	78	0157	09D	C2	5D	42	29
04	39	02	79	0158	09E	C2	5E	42	3B
04	40	02	80	0159	09F	C2	5F	42	5E
05	01	03	01	0160	0A0	C2	60	42	2D
05	02	03	02	0161	0A1	C2	61	42	2F
05	03	03	03	0162	0A2	C2	E2	42	53
05	04	03	04	0163	0A3	C2	E3	42	54
05	05	03	05	0164	0A4	C2	E4	42	55
05	06	03	06	0165	0A5	C2	E5	42	56
05	07	03	07	0166	0A6	C2	E6	42	57
05	08	03	08	0167	0A7	C2	E7	42	58
05	09	03	09	0168	0A8	C2	E8	42	59
05	10	03	10	0169	0A9	C2	E9	42	5A
05	11	03	11	0170	0AA	C2	6A	42	7C
05	12	03	12	0171	0AB	C2	6B	42	2C
05	13	03	13	0172	0AC	C2	6C	42	25
05	14	03	14	0173	0AD	C2	6D	42	5F
05	15	03	15	0174	0AE	C2	6E	42	3E
05	16	03	16	0175	0AF	C2	6F	42	3F
05	17	03	17	0176	0B0	C2	F0	42	30
05	18	03	18	0177	0B1	C2	F1	42	31
05	19	03	19	0178	0B2	C2	F2	42	32
05	20	03	20	0179	0B3	C2	F3	42	33
05	21	03	21	0180	0B4	C2	F4	42	34
05	22	03	22	0181	0B5	C2	F5	42	35
05	23	03	23	0182	0B6	C2	F6	42	36
05	24	03	24	0183	0B7	C2	F7	42	37

40 Col R	40 Col C	80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		ASCII	
05	25	03	25	0184	0B8	C2	F8	42	38
05	26	03	26	0185	0B9	C2	F9	42	39
05	27	03	27	0186	0BA	C2	7A	42	3A
05	28	03	28	0187	0BB	C2	7B	42	23
05	29	03	29	0188	0BC	C2	7C	42	40
05	30	03	30	0189	0BD	C2	7D	42	27
05	31	03	31	0190	0BE	C2	7E	42	3D
05	32	03	32	0191	0BF	C2	7F	42	22
05	33	03	33	0192	0C0	C3	40	43	20
05	34	03	34	0193	0C1	C3	C1	43	41
05	35	03	35	0194	0C2	C3	C2	43	42
05	36	03	36	0195	0C3	C3	C3	43	43
05	37	03	37	0196	0C4	C3	C4	43	44
05	38	03	38	0197	0C5	C3	C5	43	45
05	39	03	39	0198	0C6	C3	C6	43	46
05	40	03	40	0199	0C7	C3	C7	43	47
06	01	03	41	0200	0C8	C3	C8	43	48
06	02	03	42	0201	0C9	C3	C9	43	49
06	03	03	43	0202	0CA	C3	4A	43	5B
06	04	03	44	0203	0CB	C3	4B	43	2E
06	05	03	45	0204	0CC	C3	4C	43	3C
06	06	03	46	0205	0CD	C3	4D	43	28
06	07	03	47	0206	0CE	C3	4E	43	2B
06	08	03	48	0207	0CF	C3	4F	43	21
06	09	03	49	0208	0D0	C3	50	43	26
06	10	03	50	0209	0D1	C3	D1	43	4A
06	11	03	51	0210	0D2	C3	D2	43	4B
06	12	03	52	0211	0D3	C3	D3	43	4C
06	13	03	53	0212	0D4	C3	D4	43	4D
06	14	03	54	0213	0D5	C3	D5	43	4E
06	15	03	55	0214	0D6	C3	D6	43	4F
06	16	03	56	0215	0D7	C3	D7	43	50
06	17	03	57	0216	0D8	C3	D8	43	51
06	18	03	58	0217	0D9	C3	D9	32	52
06	19	03	59	0218	0DA	C3	5A	43	5D
06	20	03	60	0219	0DB	C3	5B	43	24
06	21	03	61	0220	0DC	C3	5C	43	2A
06	22	03	62	0221	0DD	C3	5D	43	29
06	23	03	63	0222	0DE	C3	5E	43	3B
06	24	03	64	0223	0DF	C3	5F	43	5E
06	25	03	65	0224	0E0	C3	60	43	2D
06	26	03	66	0225	0E1	C3	61	43	2F
06	27	03	67	0226	0E2	C3	E2	43	53
06	28	03	68	0227	0E3	C3	E3	43	54
06	29	03	69	0228	0E4	C3	E4	43	55
06	30	03	70	0229	0E5	C3	E5	43	56
06	31	03	71	0230	0E6	C3	E6	43	57
06	32	03	72	0231	0E7	C3	E7	43	58
06	33	03	73	0232	0E8	C3	E8	43	59
06	34	03	74	0233	0E9	C3	E9	43	5A
06	35	03	75	0234	0EA	C3	6A	43	7C
06	36	03	76	0235	0EB	C3	6B	43	2C
06	37	03	77	0236	0EC	C3	6C	43	25
06	38	03	78	0237	0ED	C3	6D	43	5F
06	39	03	79	0238	0EE	C3	6E	43	3E
06	40	03	80	0239	0EF	C3	6F	43	3F
07	01	04	01	0240	0F0	C3	F0	43	30
07	02	04	02	0241	0F1	C3	F1	43	31
07	03	04	03	0242	0F2	C3	F2	43	32
07	04	04	04	0243	0F3	C3	F3	32	33
07	05	04	05	0244	0F4	C3	F4	43	34
07	06	04	06	0245	0F5	C3	F5	43	35

40 Col R	40 Col C	80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		ASCII	
07	07	04	07	0246	0F6	C3	F6	43	36
07	08	04	08	0247	0F7	C3	F7	43	37
07	09	04	09	0248	0F8	C3	F8	43	38
07	10	04	10	0249	0F9	C3	F9	43	39
07	11	04	11	0250	0FA	C3	7A	43	3A
07	12	04	12	0251	0FB	C3	7B	43	23
07	13	04	13	0252	0FC	C3	7C	43	40
07	14	04	14	0253	0FD	C3	7D	43	27
07	15	04	15	0254	0FE	C3	7E	43	3D
07	16	04	16	0255	0FF	C3	7F	43	22
07	17	04	17	0256	100	C4	40	44	20
07	18	04	18	0257	101	C4	C1	44	41
07	19	04	19	0258	102	C4	C2	44	42
07	20	04	20	0259	103	C4	C3	44	43
07	21	04	21	0260	104	C4	C4	44	44
07	22	04	22	0261	105	C4	C5	44	45
07	23	04	23	0262	106	C4	C6	44	46
07	24	04	24	0263	107	C4	C7	44	47
07	25	04	25	0264	108	C4	C8	44	48
07	26	04	26	0265	109	C4	C9	44	49
07	27	04	27	0266	10A	C4	4A	44	5B
07	28	04	28	0267	10B	C4	4B	44	2E
07	29	04	29	0268	10C	C4	4C	44	3C
07	30	04	30	0269	10D	C4	4D	44	28
07	31	04	31	0270	10E	C4	4E	44	2B
07	32	04	32	0271	10F	C4	4F	44	21
07	33	04	33	0272	110	C4	50	44	26
07	34	04	34	0273	111	C4	D1	44	4A
07	35	04	35	0274	112	C4	D2	44	4B
07	36	04	36	0275	113	C4	D3	44	4C
07	37	04	37	0276	114	C4	D4	44	4D
07	38	04	38	0277	115	C4	D5	44	4E
07	39	04	39	0278	116	C4	D6	44	4F
07	40	04	40	0279	117	C4	D7	44	50
08	01	04	41	0280	118	C4	D8	44	51
08	02	04	42	0281	119	C4	D9	44	52
08	03	04	43	0282	11A	C4	5A	44	5D
08	04	04	44	0283	11B	C4	5B	44	24
08	05	04	45	0284	11C	C4	5C	44	2A
08	06	04	46	0285	11D	C4	5D	44	29
08	07	04	47	0286	11E	C4	5E	44	3B
08	08	04	48	0287	11F	C4	5F	44	5E
08	09	04	49	0288	120	C4	60	44	2D
08	10	04	50	0289	121	C4	61	44	2F
08	11	04	51	0290	122	C4	E2	44	53
08	12	04	52	0291	123	C4	E3	44	54
08	13	04	53	0292	124	C4	E4	44	55
08	14	04	54	0293	125	C4	E5	44	56
08	15	04	55	0294	126	C4	E6	44	57
08	16	04	56	0295	127	C4	E7	44	58
08	17	04	57	0296	128	C4	E8	44	59
08	18	04	58	0297	129	C4	E9	44	5A
08	19	04	59	0298	12A	C4	6A	44	7C
08	20	04	60	0299	12B	C4	6B	44	2C
08	21	04	61	0300	12C	C4	6C	44	25
08	22	04	62	0301	12D	C4	6D	44	5F
08	23	04	63	0302	12E	C4	6E	44	3E
08	24	04	64	0303	12F	C4	6F	44	3F
08	25	04	65	0304	130	C4	F0	44	30
08	26	04	66	0305	131	C4	F1	44	31
08	27	04	67	0306	132	C4	F2	44	32
08	28	04	68	0307	133	C4	F3	44	33

40 Col R	40 Col C	80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		ASCII	
08	29	04	69	0308	134	C4	F4	44	34
08	30	04	70	0309	135	C4	F5	44	35
08	31	04	71	0310	136	C4	F6	44	36
08	32	04	72	0311	137	C4	F7	44	37
08	33	04	73	0312	138	C4	F8	44	38
08	34	04	74	0313	139	C4	F9	44	39
08	35	04	75	0314	13A	C4	7A	44	3A
08	36	04	76	0315	13B	C4	7B	44	23
08	37	04	77	0316	13C	C4	7C	44	40
08	38	04	78	0317	13D	C4	7D	44	27
08	39	04	79	0318	13E	C4	7E	44	3D
08	40	04	80	0319	13F	C4	7F	44	22
09	01	05	01	0320	140	C5	40	45	20
09	02	05	02	0321	141	C5	C1	45	41
09	03	05	03	0322	142	C5	C2	45	42
09	04	05	04	0323	143	C5	C3	45	43
09	05	05	05	0324	144	C5	C4	45	44
09	06	05	06	0325	145	C5	C5	45	45
09	07	05	07	0326	146	C5	C6	45	46
09	08	05	08	0327	147	C5	C7	45	47
09	09	05	09	0328	148	C5	C8	45	48
09	10	05	10	0329	149	C5	C9	45	49
09	11	05	11	0330	14A	C5	4A	45	5B
09	12	05	12	0331	14B	C5	4B	45	2E
09	13	05	13	0332	14C	C5	4C	45	3C
09	14	05	14	0333	14D	C5	4D	45	28
09	15	05	15	0334	14E	C5	4E	45	2B
09	16	05	16	0335	14F	C5	4F	45	21
09	17	05	17	0336	150	C5	50	45	46
09	18	05	18	0337	151	C5	D1	45	4A
09	19	05	19	0338	152	C5	D2	45	4B
09	20	05	20	0339	153	C5	D3	45	4C
09	21	05	21	0340	154	C5	D4	45	4D
09	22	05	22	0341	155	C5	D5	45	4E
09	23	05	23	0342	156	C5	D6	45	4F
09	24	05	24	0343	157	C5	D7	45	50
09	25	05	25	0344	158	C5	D8	45	51
09	26	05	26	0345	159	C5	D9	45	52
09	27	05	27	0346	15A	C5	5A	45	5D
09	28	05	28	0347	15B	C5	5B	45	24
09	29	05	29	0348	15C	C5	5C	45	2A
09	30	05	30	0349	15D	C5	5D	45	29
09	31	05	31	0350	15E	C5	5E	45	3B
09	32	05	32	0351	15F	C5	5F	45	5E
09	33	05	33	0352	160	C5	60	45	2D
09	34	05	34	0353	161	C5	61	45	2F
09	35	05	35	0354	162	C5	E2	45	53
09	36	05	36	0355	163	C5	E3	45	54
09	37	05	37	0356	164	C5	E4	45	55
09	38	05	38	0357	165	C5	E5	45	56
09	39	05	39	0358	166	C5	E6	45	57
09	40	05	40	0359	167	C5	E7	45	58
10	01	05	41	0360	168	C5	E8	45	59
10	02	05	42	0361	169	C5	E9	45	5A
10	03	05	43	0362	16A	C5	6A	45	7C
10	04	05	44	0363	16B	C5	6B	45	2C
10	05	05	45	0364	16C	C5	6C	45	25
10	06	05	46	0365	16D	C5	6D	45	5F
10	07	05	47	0366	16E	C5	6E	45	3E
10	08	05	48	0367	16F	C5	6F	45	3F
10	09	05	49	0368	170	C5	F0	45	30
10	10	05	50	0369	171	C5	F1	45	31

40 Col		80 Col		Position		Buffer Address (Hex)			
R	C	R	C	Dec	Hex	EBCDIC		ASCII	
10	11	05	51	0370	172	C5	F2	45	32
10	12	05	52	0371	173	C5	F3	45	33
10	13	05	53	0372	174	C5	F4	45	34
10	14	05	54	0373	175	C5	F5	45	35
10	15	05	55	0374	176	C5	F6	45	36
10	16	05	56	0375	177	C5	F7	45	37
10	17	05	57	0376	178	C5	F8	45	38
10	18	05	58	0377	179	C5	F9	45	39
10	19	05	59	0378	17A	C5	7A	45	3A
10	20	05	60	0379	17B	C5	7B	45	23
10	21	05	61	0380	17C	C5	7C	45	40
10	22	05	62	0381	17D	C5	7D	45	27
10	23	05	63	0382	17E	C5	7E	45	3D
10	24	05	64	0383	17F	C5	7F	45	22
10	25	05	65	0384	180	C6	40	46	20
10	26	05	66	0385	181	C6	C1	46	41
10	27	05	67	0386	182	C6	C2	46	42
10	28	05	68	0387	183	C6	C3	46	43
10	29	05	69	0388	184	C6	C4	46	44
10	30	05	70	0389	185	C6	C5	46	45
10	31	05	71	0390	186	C6	C6	46	46
10	32	05	72	0391	187	C6	C7	46	47
10	33	05	73	0392	188	C6	C8	46	48
10	34	05	74	0393	189	C6	C9	46	49
10	35	05	75	0394	18A	C6	4A	46	5B
10	36	05	76	0395	18B	C6	4B	46	2E
10	37	05	77	0396	18C	C6	4C	46	3C
10	38	05	78	0397	18D	C6	4D	46	28
10	39	05	79	0398	18E	C6	4E	46	2B
10	40	05	80	0399	18F	C6	4F	46	21
11	01	06	01	0400	190	C6	50	46	26
11	02	06	02	0401	191	C6	D1	46	4A
11	03	06	03	0402	192	C6	D2	46	4B
11	04	06	04	0403	193	C6	D3	46	4C
11	05	06	05	0404	194	C6	D4	46	4D
11	06	06	06	0405	195	C6	D5	46	4E
11	07	06	07	0406	196	C6	D6	46	4F
11	08	06	08	0407	197	C6	D7	46	50
11	09	06	09	0408	198	C6	D8	46	51
11	10	06	10	0409	199	C6	D9	46	52
11	11	06	11	0410	19A	C6	5A	46	5D
11	12	06	12	0411	19B	C6	5B	46	24
11	13	06	13	0412	19C	C6	5C	46	2A
11	14	06	14	0413	19D	C6	5D	46	29
11	15	06	15	0414	19E	C6	5E	46	3B
11	16	06	16	0415	19F	C6	5F	46	5E
11	17	06	17	0416	1A0	C6	60	46	2D
11	18	06	18	0417	1A1	C6	61	46	2F
11	19	06	19	0418	1A2	C6	E2	46	53
11	20	06	20	0419	1A3	C6	E3	46	54
11	21	06	21	0420	1A4	C6	E4	46	55
11	22	06	22	0421	1A5	C6	E5	46	56
11	23	06	23	0422	1A6	C6	E6	46	57
11	24	06	24	0423	1A7	C6	E7	46	58
11	25	06	25	0424	1A8	C6	E8	46	59
11	26	06	26	0425	1A9	C6	E9	46	5A
11	27	06	27	0426	1AA	C6	6A	46	7C
11	28	06	28	0427	1AB	C6	6B	46	2C
11	29	06	29	0428	1AC	C6	6C	46	25
11	30	06	30	0429	1AD	C6	6D	46	5F
11	31	06	31	0430	1AE	C6	6E	46	3E
11	32	06	32	0431	1AF	C6	6F	46	3F
11	33	06	33	0432	1B0	C6	F0	46	30

40 Col R	40 Col C	80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		ASCII	
11	34	06	34	0433	1B1	C6	F1	46	31
11	35	06	35	0434	1B2	C6	F2	46	32
11	36	06	36	0435	1B3	C6	F3	46	33
11	37	06	37	0436	1B4	C6	F4	46	34
11	38	06	38	0437	1B5	C6	F5	46	35
11	39	06	39	0438	1B6	C6	F6	46	36
11	40	06	40	0439	1B7	C6	F7	46	37
12	01	06	41	0440	1B8	C6	F8	46	38
12	02	06	42	0441	1B9	C6	F9	46	39
12	03	06	43	0442	1BA	C6	7A	46	3A
12	04	06	44	0443	1BB	C6	7B	46	23
12	05	06	45	0444	1BC	C6	7C	46	40
12	06	06	46	0445	1BD	C6	7D	46	27
12	07	06	47	0446	1BE	C6	7E	46	3D
12	08	06	48	0447	1BF	C6	7F	46	22
12	09	06	49	0448	1C0	C7	40	47	20
12	10	06	50	0449	1C1	C7	C1	47	41
12	11	06	51	0450	1C2	C7	C2	47	42
12	12	06	52	0451	1C3	C7	C3	47	43
12	13	06	53	0452	1C4	C7	C4	47	44
12	14	06	54	0453	1C5	C7	C5	47	45
12	15	06	55	0454	1C6	C7	C6	47	46
12	16	06	56	0455	1C7	C7	C7	47	47
12	17	06	57	0456	1C8	C7	C8	47	48
12	18	06	58	0457	1C9	C7	C9	47	49
12	19	06	59	0458	1CA	C7	4A	47	5B
12	20	06	60	0459	1CB	C7	4B	47	2E
12	21	06	61	0460	1CC	C7	4C	47	3C
12	22	06	62	0461	1CD	C7	4D	47	28
12	23	06	63	0462	1CE	C7	4E	47	2B
12	24	06	64	0463	1CF	C7	4F	47	21
12	25	06	65	0464	1D0	C7	50	47	26
12	26	06	66	0465	1D1	C7	D1	47	4A
12	27	06	67	0466	1D2	C7	D2	47	4B
12	28	06	68	0467	1D3	C7	D3	47	4C
12	29	06	69	0468	1D4	C7	D4	47	4D
12	30	06	70	0469	1D5	C7	D5	47	4E
12	31	06	71	0470	1D6	C7	D6	47	4F
12	32	06	72	0471	1D7	C7	D7	47	50
12	33	06	73	0472	1D8	C7	D8	47	51
12	34	06	74	0473	1D9	C7	D9	47	52
12	35	06	75	0474	1DA	C7	5A	47	5D
12	36	06	76	0475	1DB	C7	5B	47	24
12	37	06	77	0476	1DC	C7	5C	47	2A
12	38	06	78	0477	1DD	C7	5D	47	29
12	39	06	79	0478	1DE	C7	5E	47	3B
12	40	06	80	0479	1DF	C7	5F	47	5E
		07	01	0480	1E0	C7	60	47	2D
		07	02	0481	1E1	C7	61	47	2F
		07	03	0482	1E2	C7	E2	47	53
		07	04	0483	1E3	C7	E3	47	54
		07	05	0484	1E4	C7	E4	47	55
		07	06	0485	1E5	C7	E5	47	56
		07	07	0486	1E6	C7	E6	47	57
		07	08	0487	1E7	C7	E7	47	58
		07	09	0488	1E8	C7	E8	47	59
		07	10	0489	1E9	C7	E9	47	5A
		07	11	0490	1EA	C7	6A	47	7C
		07	12	0491	1EB	C7	6B	47	2C
		07	13	0492	1EC	C7	6C	47	25
		07	14	0493	1ED	C7	6D	47	5F
		07	15	0494	1EE	C7	6E	47	3E

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
07	16	0495	1EF	C7	6F	47	3F
07	17	0496	1F0	C7	F0	47	30
07	18	0497	1F1	C7	F1	47	31
07	19	0498	1F2	C7	F2	47	32
07	20	0499	1F3	C7	F3	47	33
07	21	0500	1F4	C7	F4	47	34
07	22	0501	1F5	C7	F5	47	35
07	23	0502	1F6	C7	F6	47	36
07	24	0503	1F7	C7	F7	47	37
07	25	0504	1F8	C7	F8	47	38
07	26	0505	1F9	C7	F9	47	39
07	27	0506	1FA	C7	7A	47	3A
07	28	0507	1FB	C7	7B	47	23
07	29	0508	1FC	C7	7C	47	40
07	30	0509	1FD	C7	7D	47	27
07	31	0510	1FE	C7	7E	47	3D
07	32	0511	1FF	C7	7F	47	22
07	33	0512	200	C8	40	48	20
07	34	0513	201	C8	C1	48	41
07	35	0514	202	C8	C2	48	42
07	36	0515	203	C8	C3	48	43
07	37	0516	204	C8	C4	48	44
07	38	0517	205	C8	C5	48	45
07	39	0518	206	C8	C6	48	46
07	40	0519	207	C8	C7	48	47
07	41	0520	208	C8	C8	48	48
07	42	0521	209	C8	C9	48	49
07	43	0522	20A	C8	4A	48	5B
07	44	0523	20B	C8	4B	48	2E
07	45	0524	20C	C8	4C	48	3C
07	46	0525	20D	C8	4D	48	28
07	47	0526	20E	C8	4E	48	2B
07	48	0527	20F	C8	4F	48	21
07	49	0528	210	C8	50	48	26
07	50	0529	211	C8	D1	48	4A
07	51	0530	212	C8	D2	48	4B
07	52	0531	213	C8	D3	48	4C
07	53	0532	214	C8	D4	48	4D
07	54	0533	215	C8	D5	48	4E
07	55	0534	216	C8	D6	48	4F
07	56	0535	217	C8	D7	48	50
07	57	0536	218	C8	D8	48	51
07	58	0537	219	C8	D9	48	52
07	59	0538	21A	C8	5A	48	5D
07	60	0539	21B	C8	5B	48	24
07	61	0540	21C	C8	5C	48	2A
07	62	0541	21D	C8	5D	48	29
07	63	0542	21E	C8	5E	48	3B
07	64	0543	21F	C8	5F	48	5E
07	65	0544	220	C8	60	48	2D
07	66	0545	221	C8	61	48	2F
07	67	0546	222	C8	E2	48	53
07	68	0547	223	C8	E3	48	54
07	69	0548	224	C8	E4	48	55
07	70	0549	225	C8	E5	48	56
07	71	0550	226	C8	E6	48	57
07	72	0551	227	C8	E7	48	58
07	73	0552	228	C8	E8	48	59
07	74	0553	229	C8	E9	48	5A
07	75	0554	22A	C8	6A	48	7C
07	76	0555	22B	C8	6B	48	2C
07	77	0556	22C	C8	6C	48	25

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
07	78	0557	22D	C8	6D	48	5F
07	79	0558	22E	C8	6E	48	3E
07	80	0559	22F	C8	6F	48	3F
08	01	0560	230	C8	F0	48	30
08	02	0561	231	C8	F1	48	31
08	03	0562	232	C8	F2	48	32
08	04	0563	233	C8	F3	48	33
08	05	0564	234	C8	F4	48	34
08	06	0565	235	C8	F5	48	35
08	07	0566	236	C8	F6	48	36
08	08	0567	237	C8	F7	48	37
08	09	0568	238	C8	F8	48	38
08	10	0569	239	C8	F9	48	39
08	11	0570	23A	C8	7A	48	3A
08	12	0571	23B	C8	7B	48	23
08	13	0572	23C	C8	7C	48	40
08	14	0573	23D	C8	7D	48	27
08	15	0574	23E	C8	7E	48	3D
08	16	0575	23F	C8	7F	48	22
08	17	0576	240	C9	40	49	20
08	18	0577	241	C9	C1	49	41
08	19	0578	242	C9	C2	49	42
08	20	0579	243	C9	C3	49	43
08	21	0580	244	C9	C4	49	44
08	22	0581	245	C9	C5	49	45
08	23	0582	246	C9	C6	49	46
08	24	0583	247	C9	C7	49	47
08	25	0584	248	C9	C8	49	48
08	26	0585	249	C9	C9	49	49
08	27	0586	24A	C9	4A	49	5B
08	28	0587	24B	C9	4B	49	2E
08	29	0588	24C	C9	4C	49	3C
08	30	0589	24D	C9	4D	49	28
08	31	0590	24E	C9	4E	49	2B
08	32	0591	24F	C9	4F	49	21
08	33	0592	250	C9	50	49	26
08	34	0593	251	C9	D1	49	4A
08	35	0594	252	C9	D2	49	4B
08	36	0595	253	C9	D3	49	4C
08	37	0596	254	C9	D4	49	4D
08	38	0597	255	C9	D5	49	4E
08	39	0598	256	C9	D6	49	4F
08	40	0599	257	C9	D7	49	50
08	41	0600	258	C9	D8	49	51
08	42	0601	259	C9	D9	49	52
08	43	0602	25A	C9	5A	49	5D
08	44	0603	25B	C9	5B	49	24
08	45	0604	25C	C9	5C	49	2A
08	46	0605	25D	C9	5D	49	29
08	47	0606	25E	C9	5E	49	3B
08	48	0607	25F	C9	5F	49	5E
08	49	0608	260	C9	60	49	2D
08	50	0609	261	C9	61	49	2F
08	51	0610	262	C9	E2	49	53
08	52	0611	263	C9	E3	49	54
08	53	0612	264	C9	E4	49	55
08	54	0613	265	C9	E5	49	56
08	55	0614	266	C9	E6	49	57
08	56	0615	267	C9	E7	49	58
08	57	0616	268	C9	E8	49	59
08	58	0617	269	C9	E9	49	5A
08	59	0618	26A	C9	6A	49	7C
08	60	0619	26B	C9	6B	49	2C

80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		ASCII	
08	61	0620	26C	C9	6C	49	25
08	62	0621	26D	C9	6D	49	5F
08	63	0622	26E	C9	6E	49	3E
08	64	0623	26F	C9	6F	49	3F
08	65	0624	270	C9	F0	49	30
08	66	0625	271	C9	F1	49	31
08	67	0626	272	C9	F2	49	32
08	68	0627	273	C9	F3	49	33
08	69	0628	274	C9	F4	49	34
08	70	0629	275	C9	F5	49	35
08	71	0630	276	C9	F6	49	36
08	72	0631	277	C9	F7	49	37
08	73	0632	278	C9	F8	49	38
08	74	0633	279	C9	F9	49	39
08	75	0634	27A	C9	7A	49	3A
08	76	0635	27B	C9	7B	49	23
08	77	0636	27C	C9	7C	49	40
08	78	0637	27D	C9	7D	49	27
08	79	0638	27E	C9	7E	49	3D
08	80	0639	27F	C9	7F	49	22
09	01	0640	280	4A	40	5B	20
09	02	0641	281	4A	C1	5B	41
09	03	0642	282	4A	C2	5B	42
09	04	0643	283	4A	C3	5B	43
09	05	0644	284	4A	C4	5B	44
09	06	0645	285	4A	C5	5B	45
09	07	0646	286	4A	C6	5B	46
09	08	0647	287	4A	C7	5B	47
09	09	0648	288	4A	C8	5B	48
09	10	0649	289	4A	C9	5B	49
09	11	0650	28A	4A	4A	5B	5B
09	12	0651	28B	4A	4B	5B	2E
09	13	0652	28C	4A	4C	5B	3C
09	14	0653	28D	4A	4D	5B	28
09	15	0654	28E	4A	4E	5B	2B
09	16	0655	28F	4A	4F	5B	21
09	17	0656	290	4A	50	5B	26
09	18	0657	291	4A	D1	5B	4A
09	19	0658	292	4A	D2	5B	4B
09	20	0659	293	4A	D3	5B	4C
09	21	0660	294	4A	D4	5B	4D
09	22	0661	295	4A	D5	5B	4E
09	23	0662	296	4A	D6	5B	4F
09	24	0663	297	4A	D7	5B	50
09	25	0664	298	4A	D8	5B	51
09	26	0665	299	4A	D9	5B	52
09	27	0666	29A	4A	5A	5B	5D
09	28	0667	29B	4A	5B	5B	24
09	29	0668	29C	4A	5C	5B	2A
09	30	0669	29D	4A	5D	5B	29
09	31	0670	29E	4A	5E	5B	3B
09	32	0671	29F	4A	5F	5B	5E
09	33	0672	2A0	4A	60	5B	2D
09	34	0673	2A1	4A	61	5B	2F
09	35	0674	2A2	4A	E2	5B	53
09	36	0675	2A3	4A	E3	5B	54
09	37	0676	2A4	4A	E4	5B	55
09	38	0677	2A5	4A	E5	5B	56
09	39	0678	2A6	4A	E6	5B	57
09	40	0679	2A7	4A	E7	5B	58
09	41	0680	2A8	4A	E8	5B	59
09	42	0681	2A9	4A	E9	5B	5A

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
09	43	0682	2AA	4A	6A	5B	7C
09	44	0683	2AB	4A	6B	5B	2C
09	45	0684	2AC	4A	6C	5B	25
09	46	0685	2AD	4A	6D	5B	5F
09	47	0686	2AE	4A	6E	5B	3E
09	48	0687	2AF	4A	6F	5B	3F
09	49	0688	2B0	4A	F0	5B	30
09	50	0689	2B1	4A	F1	5B	31
09	51	0690	2B2	4A	F2	5B	32
09	52	0691	2B3	4A	F3	5B	33
09	53	0692	2B4	4A	F4	5B	34
09	54	0693	2B5	4A	F5	5B	35
09	55	0694	2B6	4A	F6	5B	36
09	56	0695	2B7	4A	F7	5B	37
09	57	0696	2B8	4A	F8	5B	38
09	58	0697	2B9	4A	F9	5B	39
09	59	0698	2BA	4A	7A	5B	3A
09	60	0699	2BB	4A	7B	5B	23
09	61	0700	2BC	4A	7C	5B	40
09	62	0701	2BD	4A	7D	5B	27
09	63	0702	2BE	4A	7E	5B	3D
09	64	0703	2BF	4A	7F	5B	22
09	65	0704	2C0	4B	40	2E	20
09	66	0705	2C1	4B	C1	2E	41
09	67	0706	2C2	4B	C2	2E	42
09	68	0707	2C3	4B	C3	2E	43
09	69	0708	2C4	4B	C4	2E	44
09	70	0709	2C5	4B	C5	2E	45
09	71	0710	2C6	4B	C6	2E	46
09	72	0711	2C7	4B	C7	2E	47
09	73	0712	2C8	4B	C8	2E	48
09	74	0713	2C9	4B	C9	2E	49
09	75	0714	2CA	4B	4A	2E	5B
09	76	0715	2CB	4B	4B	2E	2E
09	77	0716	2CC	4B	4C	2E	3C
09	78	0717	2CD	4B	4D	2E	28
09	79	0718	2CE	4B	4E	2E	2B
09	80	0719	2CF	4B	4F	2E	21
10	01	0720	2D0	4B	50	2E	26
10	02	0721	2D1	4B	D1	2E	4A
10	03	0722	2D2	4B	D2	2E	4B
10	04	0723	2D3	4B	D3	2E	4C
10	05	0724	2D4	4B	D4	2E	4D
10	06	0725	2D5	4B	D5	2E	4E
10	07	0726	2D6	4B	D6	2E	4F
10	08	0727	2D7	4B	D7	2E	50
10	09	0728	2D8	4B	D8	2E	51
10	10	0729	2D9	4B	D9	2E	52
10	11	0730	2DA	4B	5A	2E	5D
10	12	0731	2DB	4B	5B	2E	24
10	13	0732	2DC	4B	5C	2E	2A
10	14	0733	2DD	4B	5D	2E	29
10	15	0734	2DE	4B	5E	2E	3B
10	16	0735	2DF	4B	5F	2E	5E
10	17	0736	2E0	4B	60	2E	2D
10	18	0737	2E1	4B	61	2E	2F
10	19	0738	2E2	4B	E2	2E	53
10	20	0739	2E3	4B	E3	2E	54
10	21	0740	2E4	4B	E4	2E	55
10	22	0741	2E5	4B	E5	2E	56
10	23	0742	2E6	4B	E6	2E	57
10	24	0743	2E7	4B	E7	2E	58

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
10	25	0744	2E8	4B	E8	2E	59
10	26	0745	2E9	4B	E9	2E	5A
10	27	0746	2EA	4B	6A	2E	7C
10	28	0747	2EB	4B	6B	2E	2C
10	29	0748	2EC	4B	6C	2E	25
10	30	0749	2ED	4B	6D	2E	5F
10	31	0750	2EE	4B	6E	2E	3E
10	32	0751	2EF	4B	6F	2E	3F
10	33	0752	2F0	4B	F0	2E	30
10	34	0753	2F1	4B	F1	2E	31
10	35	0754	2F2	4B	F2	2E	32
10	36	0755	2F3	4B	F3	2E	33
10	37	0756	2F4	4B	F4	2E	34
10	38	0757	2F5	4B	F5	2E	35
10	39	0758	2F6	4B	F6	2E	36
10	40	0759	2F7	4B	F7	2E	37
10	41	0760	2F8	4B	F8	2E	38
10	42	0761	2F9	4B	F9	2E	39
10	43	0762	2FA	4B	7A	2E	3A
10	44	0763	2FB	4B	7B	2E	23
10	45	0764	2FC	4B	7C	2E	40
10	46	0765	2FD	4B	7D	2E	27
10	47	0766	2FE	4B	7E	2E	3D
10	48	0767	2FF	4B	7F	2E	22
10	49	0768	300	4C	40	3C	20
10	50	0769	301	4C	C1	3C	41
10	51	0770	302	4C	C2	3C	42
10	52	0771	303	4C	C3	3C	43
10	53	0772	304	4C	C4	3C	44
10	54	0773	305	4C	C5	3C	45
10	55	0774	306	4C	C6	3C	46
10	56	0775	307	4C	C7	3C	47
10	57	0776	308	4C	C8	3C	48
10	58	0777	309	4C	C9	3C	49
10	59	0778	30A	4C	4A	3C	5B
10	60	0779	30B	4C	4B	3C	2E
10	61	0780	30C	4C	4C	3C	3C
10	62	0781	30D	4C	4D	3C	28
10	63	0782	30E	4C	4E	3C	2B
10	64	0783	30F	4C	4F	3C	21
10	65	0784	310	4C	50	3C	26
10	66	0785	311	4C	D1	3C	4A
10	67	0786	312	4C	D2	3C	4B
10	68	0787	313	4C	D3	3C	4C
10	69	0788	314	4C	D4	3C	4D
10	70	0789	315	4C	D5	3C	4E
10	71	0790	316	4C	D6	3C	4F
10	72	0791	317	4C	D7	3C	50
10	73	0792	318	4C	D8	3C	51
10	74	0793	319	4C	D9	3C	52
10	75	0794	31A	4C	5A	3C	5D
10	76	0795	31B	4C	5B	3C	24
10	77	0796	31C	4C	5C	3C	2A
10	78	0797	31D	4C	5D	3C	29
10	79	0798	31E	4C	5E	3C	3B
10	80	0799	31F	4C	5F	3C	5E
11	01	0800	320	4C	60	3C	2D
11	02	0801	321	4C	61	3C	2F
11	03	0802	322	4C	E2	3C	53
11	04	0803	323	4C	E3	3C	54
11	05	0804	324	4C	E4	3C	55
11	06	0805	325	4C	E5	3C	56

80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		Buffer Address (Hex) ASCII	
11	07	0806	326	4C	E6	3C	57
11	08	0807	327	4C	E7	3C	58
11	09	0808	328	4C	E8	3C	59
11	10	0809	329	4C	E9	3C	5A
11	11	0810	32A	4C	6A	3C	7C
11	12	0811	32B	4C	6B	3C	2C
11	13	0812	32C	4C	6C	3C	25
11	14	0813	32D	4C	6D	3C	5F
11	15	0814	32E	4C	6E	3C	3E
11	16	0815	32F	4C	6F	3C	3F
11	17	0816	330	4C	F0	3C	30
11	18	0817	331	4C	F1	3C	31
11	19	0818	332	4C	F2	3C	32
11	20	0819	333	4C	F3	3C	33
11	21	0820	334	4C	F4	3C	34
11	22	0821	335	4C	F5	3C	35
11	23	0822	336	4C	F6	3C	36
11	24	0823	337	4C	F7	3C	37
11	25	0824	338	4C	F8	3C	38
11	26	0825	339	4C	F9	3C	39
11	27	0826	33A	4C	7A	3C	3A
11	28	0827	33B	4C	7B	3C	23
11	29	0828	33C	4C	7C	3C	40
11	30	0829	33D	4C	7D	3C	27
11	31	0830	33E	4C	7E	3C	3D
11	32	0831	33F	4C	7F	3C	22
11	33	0832	340	4D	40	28	20
11	34	0833	341	4D	C1	28	41
11	35	0834	342	4D	C2	28	42
11	36	0835	343	4D	C3	28	43
11	37	0836	344	4D	C4	28	44
11	38	0837	345	4D	C5	28	45
11	39	0838	346	4D	C6	28	46
11	40	0839	347	4D	C7	28	47
11	41	0840	348	4D	C8	28	48
11	42	0841	349	4D	C9	28	49
11	43	0842	34A	4D	4A	28	5B
11	44	0843	34B	4D	4B	28	2E
11	45	0844	34C	4D	4C	28	3C
11	46	0845	34D	4D	4D	28	28
11	47	0846	34E	4D	4E	28	2B
11	48	0847	34F	4D	4F	28	21
11	49	0848	350	4D	50	28	26
11	50	0849	351	4D	D1	28	4A
11	51	0850	352	4D	D2	28	4B
11	52	0851	353	4D	D3	28	4C
11	53	0852	354	4D	D4	28	4D
11	54	0853	355	4D	D5	28	4E
11	55	0854	356	4D	D6	28	4F
11	56	0855	357	4D	D7	28	50
11	57	0856	358	4D	D8	28	51
11	58	0857	359	4D	D9	28	52
11	59	0858	35A	4D	5A	28	5D
11	60	0859	35B	4D	5B	28	24
11	61	0860	35C	4D	5C	28	2A
11	62	0861	35D	4D	5D	28	29
11	63	0862	35E	4D	5E	28	3B
11	64	0863	35F	4D	5F	28	5E
11	65	0864	360	4D	60	28	2D
11	66	0865	361	4D	61	28	2F
11	67	0866	362	4D	E2	28	53
11	68	0867	363	4D	E3	28	54

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
11	69	0868	364	4D	E4	28	55
11	70	0869	365	4D	E5	28	56
11	71	0870	366	4D	E6	28	57
11	72	0871	367	4D	E7	28	58
11	73	0872	368	4D	E8	28	59
11	74	0873	369	4D	E9	28	5A
11	75	0874	36A	4D	6A	28	7C
11	76	0875	36B	4D	6B	28	2C
11	77	0876	36C	4D	6C	28	25
11	78	0877	36D	4D	6D	28	5F
11	79	0878	36E	4D	6E	28	3E
11	80	0879	36F	4D	6F	28	3F
12	01	0880	370	4D	F0	28	30
12	02	0881	371	4D	F1	28	31
12	03	0882	372	4D	F2	28	32
12	04	0883	373	4D	F3	28	33
12	05	0884	374	4D	F4	28	34
12	06	0885	375	4D	F5	28	35
12	07	0886	376	4D	F6	28	36
12	08	0887	377	4D	F7	28	37
12	09	0888	378	4D	F8	28	38
12	10	0889	379	4D	F9	28	39
12	11	0890	37A	4D	7A	28	3A
12	12	0891	37B	4D	7B	28	23
12	13	0892	37C	4D	7C	28	40
12	14	0893	37D	4D	7D	28	27
12	15	0894	37E	4D	7E	28	3D
12	16	0895	37F	4D	7F	28	22
12	17	0896	380	4E	40	2B	20
12	18	0897	381	4E	C1	2B	41
12	19	0898	382	4E	C2	2B	42
12	20	0899	383	4E	C3	2B	43
12	21	0900	384	4E	C4	2B	44
12	22	0901	385	4E	C5	2B	45
12	23	0902	386	4E	C6	2B	46
12	24	0903	387	4E	C7	2B	47
12	25	0904	388	4E	C8	2B	48
12	26	0905	389	4E	C9	2B	49
12	27	0906	38A	4E	4A	2B	5B
12	28	0907	38B	4E	4B	2B	2E
12	29	0908	38C	4E	4C	2B	3C
12	30	0909	38D	4E	4D	2B	28
12	31	0910	38E	4E	4E	2B	2B
12	32	0911	38F	4E	4F	2B	21
12	33	0912	390	4E	50	2B	26
12	34	0913	391	4E	D1	2B	4A
12	35	0914	392	4E	D2	2B	4B
12	26	0915	393	4E	D3	2B	4C
12	37	0916	394	4E	D4	2B	4D
12	38	0917	395	4E	D5	2B	4E
12	39	0918	396	4E	D6	2B	4F
12	40	0919	397	4E	D7	2B	50
12	41	0920	398	4E	D8	2B	51
12	42	0921	399	4E	D9	2B	52
12	43	0922	39A	4E	5A	2B	5D
12	44	0923	39B	4E	5B	2B	24
12	45	0924	39C	4E	5C	2B	2A
12	46	0925	39D	4E	5D	2B	29
12	47	0926	39E	4E	5E	2B	3B
12	48	0927	39F	4E	5F	2B	5E
12	49	0928	3A0	4E	60	2B	2D
12	50	0929	3A1	4E	61	2B	2F

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
12	51	0930	3A2	4E	E2	2B	53
12	52	0931	3A3	4E	E3	2B	54
12	53	0932	3A4	4E	E4	2B	55
12	54	0933	3A5	4E	E5	2B	56
12	55	0934	3A6	4E	E6	2B	57
12	56	0935	3A7	4E	E7	2B	58
12	57	0936	3A8	4E	E8	2B	59
12	58	0937	3A9	4E	E9	2B	5A
12	59	0938	3AA	4E	6A	2B	7C
12	60	0939	3AB	4E	6B	2B	2C
12	61	0940	3AC	4E	6C	2B	25
12	62	0941	3AD	4E	6D	2B	5F
12	63	0942	3AE	4E	6E	2B	3E
12	64	0943	3AF	4E	6F	2B	3F
12	65	0944	3B0	4E	F0	2B	30
12	66	0945	3B1	4E	F1	2B	31
12	67	0946	3B2	4E	F2	2B	32
12	68	0947	3B3	4E	F3	2B	33
12	69	0948	3B4	4E	F4	2B	34
12	70	0949	3B5	4E	F5	2B	35
12	71	0950	3B6	4E	F6	2B	36
12	72	0951	3B7	4E	F7	2B	37
12	73	0952	3B8	4E	F8	2B	38
12	74	0953	3B9	4E	F9	2B	39
12	75	0954	3BA	4E	7A	2B	3A
12	76	0955	3BB	4E	7B	2B	23
12	77	0956	3BC	4E	7C	2B	40
12	78	0957	3BD	4E	7D	2B	27
12	79	0958	3BE	4E	7E	2B	3D
12	80	0959	3BF	4E	7F	2B	22
13	01	0960	3C0	4F	40	21	20
13	02	0961	3C1	4F	C1	21	41
13	03	0962	3C2	4F	C2	21	42
13	04	0963	3C3	4F	C3	21	43
13	05	0964	3C4	4F	C4	21	44
13	06	0965	3C5	4F	C5	21	45
13	07	0966	3C6	4F	C6	21	46
13	08	0967	3C7	4F	C7	21	47
13	09	0968	3C8	4F	C8	21	48
13	10	0969	3C9	4F	C9	21	49
13	11	0970	3CA	4F	4A	21	5B
13	12	0971	3CB	4F	4B	21	2E
13	13	0972	3CC	4F	4C	21	3C
13	14	0973	3CD	4F	4D	21	28
13	15	0974	3CE	4F	4E	21	2B
13	16	0975	3CF	4F	4F	21	21
13	17	0976	3D0	4F	50	21	26
13	18	0977	3D1	4F	D1	21	4A
13	19	0978	3D2	4F	D2	21	4B
13	20	0979	3D3	4F	D3	21	4C
13	21	0980	3D4	4F	D4	21	4D
13	22	0981	3D5	4F	D5	21	4E
13	23	0982	3D6	4F	D6	21	4F
13	24	0983	3D7	4F	D7	21	50
13	25	0984	3D8	4F	D8	21	51
13	26	0985	3D9	4F	D9	21	52
13	27	0986	3DA	4F	5A	21	5D
13	28	0987	3DB	4F	5B	21	24
13	29	0988	3DC	4F	5C	21	2A
13	30	0989	3DD	4F	5D	21	29
13	31	0990	3DE	4F	5E	21	3B
13	32	0991	3DF	4F	5F	21	5E

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
13	33	0992	3E0	4F	60	21	2D
13	34	0993	3E1	4F	61	21	2F
13	35	0994	3E2	4F	E2	21	53
13	36	0995	3E3	4F	E3	21	54
13	37	0996	3E4	4F	E4	21	55
13	38	0997	3E5	4F	E5	21	56
13	39	0998	3E6	4F	E6	21	57
13	40	0999	3E7	4F	E7	21	58
13	41	1000	3E8	4F	E8	21	59
13	42	1001	3E9	4F	E9	21	5A
13	43	1002	3EA	4F	6A	21	7C
13	44	1003	3EB	4F	6B	21	2C
13	45	1004	3EC	4F	6C	21	25
13	46	1005	3ED	4F	6D	21	5F
13	47	1006	3EE	4F	6E	21	3E
13	48	1007	3EF	4F	6F	21	3F
13	49	1008	3F0	4F	F0	21	30
13	50	1009	3F1	4F	F1	21	31
13	51	1010	3F2	4F	F2	21	32
13	52	1011	3F3	4F	F3	21	33
13	53	1012	3F4	4F	F4	21	34
13	54	1013	3F5	4F	F5	21	35
13	55	1014	3F6	4F	F6	21	36
13	56	1015	3F7	4F	F7	21	37
13	57	1016	3F8	4F	F8	21	38
13	58	1017	3F9	4F	F9	21	39
13	59	1018	3FA	4F	7A	21	3A
13	60	1019	3FB	4F	7B	21	23
13	61	1020	3FC	4F	7C	21	40
13	62	1021	3FD	4F	7D	21	27
13	63	1022	3FE	4F	7E	21	3D
13	64	1023	3FF	4F	7F	21	22
13	65	1024	400	50	40	26	20
13	66	1025	401	50	C1	26	41
13	67	1026	402	50	C2	26	42
13	68	1027	403	50	C3	26	43
13	69	1028	404	50	C4	26	44
13	70	1029	405	50	C5	26	45
13	71	1030	406	50	C6	26	46
13	72	1031	407	50	C7	26	47
13	73	1032	408	50	C8	26	48
13	74	1033	409	50	C9	26	49
13	75	1034	40A	50	4A	26	5B
13	76	1035	40B	50	4B	26	2E
13	77	1036	40C	50	4C	26	3C
13	78	1037	40D	50	4D	26	28
13	79	1038	40E	50	4E	26	2B
13	80	1039	40F	50	4F	26	21
14	01	1040	410	50	50	26	26
14	02	1041	411	50	D1	26	4A
14	03	1042	412	50	D2	26	4B
14	04	1043	413	50	D3	26	4C
14	05	1044	414	50	D4	26	4D
14	06	1045	415	50	D5	26	4E
14	07	1046	416	50	D6	26	4F
14	08	1047	417	50	D7	26	50
14	09	1048	418	50	D8	26	51
14	10	1049	419	50	D9	26	52
14	11	1050	41A	50	5A	26	5D
14	12	1051	41B	50	5B	26	24
14	13	1052	41C	50	5C	26	2A
14	14	1053	41D	50	5D	26	29

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
14	15	1054	41E	50	5E	26	3B
14	16	1055	41F	50	5F	26	5E
14	17	1056	420	50	60	26	2D
14	18	1057	421	50	61	26	2F
14	19	1058	422	50	E2	26	53
14	20	1059	423	50	E3	26	54
14	21	1060	424	50	E4	26	55
14	22	1061	425	50	E5	26	56
14	23	1062	426	50	E6	26	57
14	24	1063	427	50	E7	26	58
14	25	1064	428	50	E8	26	59
14	26	1065	429	50	E9	26	5A
14	27	1066	42A	50	6A	26	7C
14	28	1067	42B	50	6B	26	2C
14	29	1068	42C	60	6C	26	25
14	30	1069	42D	50	6D	26	5F
14	31	1070	42E	50	6E	26	3E
14	32	1071	42F	50	6F	26	3F
14	33	1072	430	50	F0	26	30
14	34	1073	431	50	F1	26	31
14	35	1074	432	50	F2	26	32
14	36	1075	433	50	F3	26	33
14	37	1076	434	50	F4	26	34
14	38	1077	435	50	F5	26	35
14	39	1078	436	50	F6	26	36
14	40	1079	437	50	F7	26	37
14	41	1080	438	50	F8	26	38
14	42	1081	439	50	F9	26	39
14	43	1082	43A	50	7A	26	3A
14	44	1083	43B	50	7B	26	23
14	45	1084	43C	50	7C	26	40
14	46	1085	43D	50	7D	26	27
14	47	1086	43E	50	7E	26	3D
14	48	1087	43F	50	7F	26	22
14	49	1088	440	D1	40	4A	20
14	50	1089	441	D1	C1	4A	41
14	51	1090	442	D1	C2	4A	42
14	52	1091	443	D1	C3	4A	43
14	53	1092	444	D1	C4	4A	44
14	54	1093	445	D1	C5	4A	45
14	55	1094	446	D1	C6	4A	46
14	56	1095	447	D1	C7	4A	47
14	57	1096	448	D1	C8	4A	48
14	58	1097	449	D1	C9	4A	49
14	59	1098	44A	D1	4A	4A	5B
14	60	1099	44B	D1	4B	4A	2E
14	61	1100	44C	D1	4C	4A	3C
14	62	1101	44D	D1	4D	4A	28
14	63	1102	44E	D1	4E	4A	2B
14	64	1103	44F	D1	4F	4A	21
14	65	1104	450	D1	50	4A	26
14	66	1105	451	D1	D1	4A	4A
14	67	1106	452	D1	D2	4A	4B
14	68	1107	453	D1	D3	4A	4C
14	69	1108	454	D1	D4	4A	4D
14	70	1109	455	D1	D5	4A	4E
14	71	1110	456	D1	D6	4A	4F
14	72	1111	457	D1	D7	4A	50
14	73	1112	458	D1	D8	4A	51
14	74	1113	459	D1	D9	4A	52
14	75	1114	45A	D1	5A	4A	5D
14	76	1115	45B	D1	5B	4A	24

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
14	77	1116	45C	D1	5C	4A	2A
14	78	1117	45D	D1	5D	4A	29
14	79	1118	45E	D1	5E	4A	3B
14	80	1119	45F	D1	5F	4A	5E
15	01	1120	460	D1	60	4A	2D
15	02	1121	461	D1	61	4A	2F
15	03	1122	462	D1	E2	4A	53
15	04	1123	463	D1	E3	4A	54
15	05	1124	464	D1	E4	4A	55
15	06	1125	465	D1	E5	4A	56
15	07	1126	466	D1	E6	4A	57
15	08	1127	467	D1	E7	4A	58
15	09	1128	468	D1	E8	4A	59
15	10	1129	469	D1	E9	4A	5A
15	11	1130	46A	D1	6A	4A	7C
15	12	1131	46B	D1	6B	4A	2C
15	13	1132	46C	D1	6C	4A	25
15	14	1133	46D	D1	6D	4A	5F
15	15	1134	46E	D1	6E	4A	3E
15	16	1135	46F	D1	6F	4A	3F
15	17	1136	470	D1	F0	4A	30
15	18	1137	471	D1	F1	4A	31
15	19	1138	472	D1	F2	4A	32
15	20	1139	473	D1	F3	4A	33
15	21	1140	474	D1	F4	4A	34
15	22	1141	475	D1	F5	4A	35
15	23	1142	476	D1	F6	4A	36
15	24	1143	477	D1	F7	4A	37
15	25	1144	478	D1	F8	4A	38
15	26	1145	479	D1	F9	4A	39
15	27	1146	47A	D1	7A	4A	3A
15	28	1147	47B	D1	7B	4A	23
15	29	1148	47C	D1	7C	4A	40
15	30	1149	47D	D1	7D	4A	27
15	31	1150	47E	D1	7E	4A	3D
15	32	1151	47F	D1	7F	4A	22
15	33	1152	480	D2	40	4B	20
15	34	1153	481	D2	C1	4B	41
15	35	1154	482	D2	C2	4B	42
15	36	1155	483	D2	C3	4B	43
15	37	1156	484	D2	C4	4B	44
15	38	1157	485	D2	C5	4B	45
15	39	1158	486	D2	C6	4B	46
15	40	1159	487	D2	C7	4B	47
15	41	1160	488	D2	C8	4B	48
15	42	1161	489	D2	C9	4B	49
15	43	1162	48A	D2	4A	4B	5B
15	44	1163	48B	D2	4B	4B	2E
15	45	1164	48C	D2	4C	4B	3C
15	46	1165	48D	D2	4D	4B	28
15	47	1166	48E	D2	4E	4B	2B
15	48	1167	48F	D2	4F	4B	21
15	49	1168	490	D2	50	4B	26
15	50	1169	491	D2	D1	4B	4A
15	51	1170	492	D2	D2	4B	4B
15	52	1171	493	D2	D3	4B	4C
15	53	1172	494	D2	D4	4B	4D
15	54	1173	495	D2	D5	4B	4E
15	55	1174	496	D2	D6	4B	4F
15	56	1175	497	D2	D7	4B	50
15	57	1176	498	D2	D8	4B	51
15	58	1177	499	D2	D9	4B	52

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
15	59	1178	49A	D2	5A	4B	5D
15	60	1179	49B	D2	5B	4B	24
15	61	1180	49C	D2	5C	4B	2A
15	62	1181	49D	D2	5D	4B	29
15	63	1182	49E	D2	5E	4B	3B
15	64	1183	49F	D2	5F	4B	5E
15	65	1184	4A0	D2	60	4B	2D
15	66	1185	4A1	D2	61	4B	2F
15	67	1186	4A2	D2	E2	4B	53
15	68	1187	4A3	D2	E3	4B	54
15	69	1188	4A4	D2	E4	4B	55
15	70	1189	4A5	D2	E5	4B	56
15	71	1190	4A6	D2	E6	4B	57
15	72	1191	4A7	D2	E7	4B	58
15	73	1192	4A8	D2	E8	4B	59
15	74	1193	4A9	D2	E9	4B	5A
15	75	1194	4AA	D2	6A	4B	7C
15	76	1195	4AB	D2	6B	4B	2C
15	77	1196	4AC	D2	6C	4B	25
15	78	1197	4AD	D2	6D	4B	5F
15	79	1198	4AE	D2	6E	4B	3E
15	80	1199	4AF	D2	6F	4B	3F
16	01	1200	4B0	D2	F0	4B	30
16	02	1201	4B1	D2	F1	4B	31
16	03	1202	4B2	D2	F2	4B	32
16	04	1203	4B3	D2	F3	4B	33
16	05	1204	4B4	D2	F4	4B	34
16	06	1205	4B5	D2	F5	4B	35
16	07	1206	4B6	D2	F6	4B	36
16	08	1207	4B7	D2	F7	4B	37
16	09	1208	4B8	D2	F8	4B	38
16	10	1209	4B9	D2	F9	4B	39
16	11	1210	4BA	D2	7A	4B	3A
16	12	1211	4BB	D2	7B	4B	23
16	13	1212	4BC	D2	7C	4B	40
16	14	1213	4BD	D2	7D	4B	27
16	15	1214	4BE	D2	7E	4B	3D
16	16	1215	4BF	D2	7F	4B	22
16	17	1216	4C0	D3	40	4C	20
16	18	1217	4C1	D3	C1	4C	41
16	19	1218	4C2	D3	C2	4C	42
16	20	1219	4C3	D3	C3	4C	43
16	21	1220	4C4	D3	C4	4C	44
16	22	1221	4C5	D3	C5	4C	45
16	23	1222	4C6	D3	C6	4C	46
16	24	1223	4C7	D3	C7	4C	47
16	25	1224	4C8	D3	C8	4C	48
16	26	1225	4C9	D3	C9	4C	49
16	27	1226	4CA	D3	4A	4C	5B
16	28	1227	4CB	D3	4B	4C	2E
16	29	1228	4CC	D3	4C	4C	3C
16	30	1229	4CD	D3	4D	4C	28
16	31	1230	4CE	D3	4E	4C	2B
16	32	1231	4CF	D3	4F	4C	21
16	33	1232	4D0	D3	50	4C	26
16	34	1233	4D1	D3	D1	4C	4A
16	35	1234	4D2	D3	D2	4C	4B
16	36	1235	4D3	D3	D3	4C	4C
16	37	1236	4D4	D3	D4	4C	4D
16	38	1237	4D5	D3	D5	4C	4E
16	39	1238	4D6	D3	D6	4C	4F
16	40	1239	4D7	D3	D7	4C	50

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
16	41	1240	4D8	D3	D8	4C	51
16	42	1241	4D9	D3	D9	4C	52
16	43	1242	4DA	D3	5A	4C	5D
16	44	1243	4DB	D3	5B	4C	24
16	45	1244	4DC	D3	5C	4C	2A
16	46	1245	4DD	D3	5D	4C	29
16	47	1246	4DE	D3	5E	4C	3B
16	48	1247	4DF	D3	5F	4C	5E
16	49	1248	4E0	D3	60	4C	2D
16	50	1249	4E1	D3	61	4C	2F
16	51	1250	4E2	D3	E2	4C	53
16	52	1251	4E3	D3	E3	4C	54
16	53	1252	4E4	D3	E4	4C	55
16	54	1253	4E5	D3	E5	4C	56
16	55	1254	4E6	D3	E6	4C	57
16	56	1255	4E7	D3	E7	4C	58
16	57	1256	4E8	D3	E8	4C	59
16	58	1257	4E9	D3	E9	4C	5A
16	59	1258	4EA	D3	6A	4C	7C
16	60	1259	4EB	D3	6B	4C	2C
16	61	1260	4EC	D3	6C	4C	25
16	62	1261	4ED	D3	6D	4C	5F
16	63	1262	4EE	D3	6E	4C	3E
16	64	1263	4EF	D3	6F	4C	3F
16	65	1264	4F0	D3	F0	4C	30
16	66	1265	4F1	D3	F1	4C	31
16	67	1266	4F2	D3	F2	4C	32
16	68	1267	4F3	D3	F3	4C	33
16	69	1268	4F4	D3	F4	4C	34
16	70	1269	4F5	D3	F5	4C	35
16	71	1270	4F6	D3	F6	4C	36
16	72	1271	4F7	D3	F7	4C	37
16	73	1272	4F8	D3	F8	4C	38
16	74	1273	4F9	D3	F9	4C	39
16	75	1274	4FA	D3	7A	4C	3A
16	76	1275	4FB	D3	7B	4C	23
16	77	1276	4FC	D3	7C	4C	40
16	78	1277	4FD	D3	7D	4C	27
16	79	1278	4FE	D3	7E	4C	3D
16	80	1279	4FF	D3	7F	4C	22
17	01	1280	500	D4	40	4D	20
17	02	1281	501	D4	C1	4D	41
17	03	1282	502	D4	C2	4D	42
17	04	1283	503	D4	C3	4D	43
17	05	1284	504	D4	C4	4D	44
17	06	1285	505	D4	C5	4D	45
17	07	1286	506	D4	C6	4D	46
17	08	1287	507	D4	C7	4D	47
17	09	1288	508	D4	C8	4D	48
17	10	1289	509	D4	C9	4D	49
17	11	1290	50A	D4	4A	4D	5B
17	12	1291	50B	D4	4B	4D	2E
17	13	1292	50C	D4	4C	4D	3C
17	14	1293	50D	D4	4D	4D	28
17	15	1294	50E	D4	4E	4D	2B
17	16	1295	50F	D4	4F	4D	21
17	17	1296	510	D4	50	4D	26
17	18	1297	511	D4	D1	4D	4A
17	19	1298	512	D4	D2	4D	4B
17	20	1299	513	D4	D3	4D	4C
17	21	1300	514	D4	D4	4D	4D
17	22	1301	515	D4	D5	4D	4E

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
17	23	1302	516	D4	D6	4D	4F
17	24	1303	517	D4	D7	4D	50
17	25	1304	518	D4	D8	4D	51
17	26	1305	519	D4	D9	4D	52
17	27	1306	51A	D4	5A	4D	5D
17	28	1307	51B	D4	5B	4D	24
17	29	1308	51C	D4	5C	4D	2A
17	30	1309	51D	D4	5D	4D	29
17	31	1310	51E	D4	5E	4D	3B
17	32	1311	51F	D4	5F	4D	5E
17	33	1312	520	D4	60	4D	2D
17	34	1313	521	D4	61	4D	2F
17	35	1314	522	D4	E2	4D	53
17	36	1315	523	D4	E3	4D	54
17	37	1316	524	D4	E4	4D	55
17	38	1317	525	D4	E5	4D	56
17	39	1318	526	D4	E6	4D	57
17	40	1319	527	D4	E7	4D	58
17	41	1320	528	D4	E8	4D	59
17	42	1321	529	D4	E9	4D	5A
17	43	1322	52A	D4	6A	4D	7C
17	44	1323	52B	D4	6B	4D	2C
17	45	1324	52C	D4	6C	4D	25
17	46	1325	52D	D4	6D	4D	5F
17	47	1326	52E	D4	6E	4D	3E
17	48	1327	52F	D4	6F	4D	3F
17	49	1328	530	D4	F0	4D	30
17	50	1329	531	D4	F1	4D	31
17	51	1330	532	D4	F2	4D	32
17	52	1331	533	D4	F3	4D	33
17	53	1332	534	D4	F4	4D	34
17	54	1333	535	D4	F5	4D	35
17	55	1334	536	D4	F6	4D	36
17	56	1335	537	D4	F7	4D	37
17	57	1336	538	D4	F8	4D	38
17	58	1337	539	D4	F9	4D	39
17	59	1338	53A	D4	7A	4D	3A
17	60	1339	53B	D4	7B	4D	23
17	61	1340	53C	D4	7C	4D	40
17	62	1341	53D	D4	7D	4D	27
17	63	1342	53E	D4	7E	4D	3D
17	64	1343	53F	D4	7F	4D	22
17	65	1344	540	D5	40	4E	20
17	66	1345	541	D5	C1	4E	41
17	67	1346	542	D5	C2	4E	42
17	68	1347	543	D5	C3	4E	43
17	69	1348	544	D5	C4	4E	44
17	70	1349	545	D5	C5	4E	45
17	71	1350	546	D5	C6	4E	46
17	72	1351	547	D5	C7	4E	47
17	73	1352	548	D5	C8	4E	48
17	74	1353	549	D5	C9	4E	49
17	75	1354	54A	D5	4A	4E	5B
17	76	1355	54B	D5	4B	4E	2E
17	77	1356	54C	D5	4C	4E	3C
17	78	1357	54D	D5	4D	4E	28
17	79	1358	54E	D5	4E	4E	2B
17	80	1359	54F	D5	4F	4E	21
18	01	1360	550	D5	50	4E	26
18	02	1361	551	D5	D1	4E	4A
18	03	1362	552	D5	D2	4E	4B
18	04	1363	553	D5	D3	4E	4C
18	05	1364	554	D5	D4	4E	4D

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
18	06	1365	555	D5	D5	4E	4E
18	07	1366	556	D5	D6	4E	4F
18	08	1367	557	D5	D7	4E	50
18	09	1368	558	D5	D8	4E	51
18	10	1369	559	D5	D9	4E	52
18	11	1370	55A	D5	5A	4E	5D
18	12	1371	55B	D5	5B	4E	24
18	13	1372	55C	D5	5C	4E	2A
18	14	1373	55D	D5	5D	4E	29
18	15	1374	55E	D5	5E	4E	3B
18	16	1375	55F	D5	5F	4E	5E
18	17	1376	560	D5	60	4E	2D
18	18	1377	561	D5	61	4E	2F
18	19	1378	562	D5	E2	4E	53
18	20	1379	563	D5	E3	4E	54
18	21	1380	564	D5	E4	4E	55
18	22	1381	565	D5	E5	4E	56
18	23	1382	566	D5	F6	4E	57
18	24	1383	567	D5	E7	4E	58
18	25	1384	568	D5	E8	4E	59
18	26	1385	569	D5	E9	4E	5A
18	27	1386	56A	D5	6A	4E	7C
18	28	1387	56B	D5	6B	4E	2C
18	29	1388	56C	D5	6C	4E	25
18	30	1389	56D	D5	6D	4E	5F
18	31	1390	56E	D5	6E	4E	3E
18	32	1391	56F	D5	6F	4E	3F
18	33	1392	570	D5	F0	4E	30
18	34	1393	571	D5	F1	4E	31
18	35	1394	572	D5	F2	4E	32
18	36	1395	573	D5	F3	4E	33
18	37	1396	574	D5	F4	4E	34
18	38	1397	575	D5	F5	4E	35
18	39	1398	576	D5	F6	4E	36
18	40	1399	577	D5	F7	4E	37
18	41	1400	578	D5	F8	4E	38
18	42	1401	579	D5	F9	4E	39
18	43	1402	57A	D5	7A	4E	3A
18	44	1403	57B	D5	7B	4E	23
18	45	1404	57C	D5	7C	4E	40
18	46	1405	57D	D5	7D	4E	27
18	47	1406	57E	D5	7E	4E	3D
18	48	1407	57F	D5	7F	4E	22
18	49	1408	580	D6	40	4F	20
18	50	1409	581	D6	C1	4F	41
18	51	1410	582	D6	C2	4F	42
18	52	1411	583	D6	C3	4F	43
18	53	1412	584	D6	C4	4F	44
18	54	1413	585	D6	C5	4F	45
18	55	1414	586	D6	C6	4F	46
18	56	1415	587	D6	C7	4F	47
18	57	1416	588	D6	C8	4F	48
18	58	1417	589	D6	C9	4F	49
18	59	1418	58A	D6	4A	4F	5B
18	60	1419	58B	D6	4B	4F	2E
18	61	1420	58C	D6	4C	4F	3C
18	62	1421	58D	D6	4D	4F	28
18	63	1422	58E	D6	4E	4F	2B
18	64	1423	58F	D6	4F	4F	21
18	65	1424	590	D6	50	4F	26
18	66	1425	591	D6	D1	4F	4A

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
18	67	1426	592	D6	D2	4F	4B
18	68	1427	593	D6	D3	4F	4C
18	69	1428	594	D6	D4	4F	4D
18	70	1429	595	D6	D5	4F	4E
18	71	1430	596	D6	D6	4F	4F
18	72	1431	597	D6	D7	4F	50
18	73	1432	598	D6	D8	4F	51
18	74	1433	599	D6	D9	4F	52
18	75	1434	59A	D6	5A	4F	5D
18	76	1435	59B	D6	5B	4F	24
18	77	1436	59C	D6	5C	4F	2A
18	78	1437	59D	D6	5D	4F	29
18	79	1438	59E	D6	5E	4F	3B
18	80	1439	59F	D6	5F	4F	5E
19	01	1440	5A0	D6	60	4F	2D
19	02	1441	5A1	D6	61	4F	2F
19	03	1442	5A2	D6	E2	4F	53
19	04	1443	5A3	D6	E3	4F	54
19	05	1444	5A4	D6	E4	4F	55
19	06	1445	5A5	D6	E5	4F	56
19	07	1446	5A6	D6	E6	4F	57
19	08	1447	5A7	D6	E7	4F	58
19	09	1448	5A8	D6	E8	4F	59
19	10	1449	5A9	D6	E9	4F	5A
19	11	1450	5AA	D6	6A	4F	7C
19	12	1451	5AB	D6	6B	4F	2C
19	13	1452	5AC	D6	6C	4F	25
19	14	1453	5AD	D6	6D	4F	5F
19	15	1454	5AE	D6	6E	4F	3E
19	16	1455	5AF	D6	6F	4F	3F
19	17	1456	5B0	D6	F0	4F	30
19	18	1457	5B1	D6	F1	4F	31
19	19	1458	5B2	D6	F2	4F	32
19	20	1459	5B3	D6	F3	4F	33
19	21	1460	5B4	D6	F4	4F	34
19	22	1461	5B5	D6	F5	4F	35
19	23	1462	5B6	D6	F6	4F	36
19	24	1463	5B7	D6	F7	4F	37
19	25	1464	5B8	D6	F8	4F	38
19	26	1465	5B9	D6	F9	4F	39
19	27	1466	5BA	D6	7A	4F	3A
19	28	1467	5BB	D6	7B	4F	23
19	29	1468	5BC	D6	7C	4F	40
19	30	1469	5BD	D6	7D	4F	27
19	31	1470	5BE	D6	7E	4F	3D
19	32	1471	5BF	D6	7F	4F	22
19	33	1472	5C0	D7	40	50	20
19	34	1473	5C1	D7	C1	50	41
19	35	1474	5C2	D7	C2	50	42
19	36	1475	5C3	D7	C3	50	43
19	37	1476	5C4	D7	C4	50	44
19	38	1477	5C5	D7	C5	50	45
19	39	1478	5C6	D7	C6	50	46
19	40	1479	5C7	D7	C7	50	47
19	41	1480	5C8	D7	C8	50	48
19	42	1481	5C9	D7	C9	50	49
19	43	1482	5CA	D7	4A	50	5B
19	44	1483	5CB	D7	4B	50	2E
19	45	1484	5CC	D7	4C	50	3C
19	46	1485	5CD	D7	4D	50	28
19	47	1486	5CE	D7	4E	50	2B
19	48	1487	5CF	D7	4F	50	21

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
19	49	1488	5D0	D7	50	50	26
19	50	1489	5D1	D7	D1	50	4A
19	51	1490	5D2	D7	D2	50	4B
19	52	1491	5D3	D7	D3	50	4C
19	53	1492	5D4	D7	D4	50	4D
19	54	1493	5D5	D7	D5	50	4E
19	55	1494	5D6	D7	D6	50	4F
19	56	1495	5D7	D7	D7	50	50
19	57	1496	5D8	D7	D8	50	51
19	58	1497	5D9	D7	D9	50	52
19	59	1498	5DA	D7	5A	50	5D
19	60	1499	5DB	D7	5B	50	24
19	61	1500	5DC	D7	5C	50	2A
19	62	1501	5DD	D7	5D	50	29
19	63	1502	5DE	D7	5E	50	3B
19	64	1503	5DF	D7	5F	50	5E
19	65	1504	5E0	D7	60	50	2D
19	66	1505	5E1	D7	61	50	2F
19	67	1506	5E2	D7	E2	50	53
19	68	1507	5E3	D7	E3	50	54
19	69	1508	5E4	D7	E4	50	55
19	70	1509	5E5	D7	E5	50	56
19	71	1510	5E6	D7	E6	50	57
19	72	1511	5E7	D7	E7	50	58
19	73	1512	5E8	D7	E8	50	59
19	74	1513	5E9	D7	E9	50	5A
19	75	1514	5EA	D7	6A	50	7C
19	76	1515	5EB	D7	6B	50	2C
19	77	1516	5EC	D7	6C	50	25
19	78	1517	5ED	D7	6D	50	5F
19	79	1518	5EE	D7	6E	50	3E
19	80	1519	5EF	D7	6F	50	3F
20	01	1520	5F0	D7	F0	50	30
20	02	1521	5F1	D7	F1	50	31
20	03	1522	5F2	D7	F2	50	32
20	04	1523	5F3	D7	F3	50	33
20	05	1524	5F4	D7	F4	50	34
20	06	1525	5F5	D7	F5	50	35
20	07	1526	5F6	D7	F6	50	36
20	08	1527	5F7	D7	F7	50	37
20	09	1528	5F8	D7	F8	50	38
20	10	1529	5F9	D7	F9	50	39
20	11	1530	5FA	D7	7A	50	3A
20	12	1531	5FB	D7	7B	50	23
20	13	1532	5FC	D7	7C	50	40
20	14	1533	5FD	D7	7D	50	27
20	15	1534	5FE	D7	7E	50	3D
20	16	1535	5FF	D7	7F	50	22
20	17	1536	600	D8	40	51	20
20	18	1537	601	D8	C1	51	41
20	19	1538	602	D8	C2	51	42
20	20	1539	603	D8	C3	51	43
20	21	1540	604	D8	C4	51	44
20	22	1541	605	D8	C5	51	45
20	23	1542	606	D8	C6	51	46
20	24	1543	607	D8	C7	51	47
20	25	1544	608	D8	C8	51	48
20	26	1545	609	D8	C9	51	49
20	27	1546	60A	D8	4A	51	5B
20	28	1547	60B	D8	4B	51	2E
20	29	1548	60C	D8	4C	51	3C
20	30	1549	60D	D8	4D	51	28

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
20	31	1550	60E	D8	4E	51	2B
20	32	1551	60F	D8	4F	51	21
20	33	1552	610	D8	50	51	26
20	34	1553	611	D8	D1	51	4A
20	35	1554	612	D8	D2	51	4B
20	36	1555	613	D8	D3	51	4C
20	37	1556	614	D8	D4	51	4D
20	38	1557	615	D8	D5	51	4E
20	39	1558	616	D8	D6	51	4F
20	40	1559	617	D8	D7	51	50
20	41	1560	618	D8	D8	51	51
20	42	1561	619	D8	D9	51	52
20	43	1562	61A	D8	5A	51	5D
20	44	1563	61B	D8	5B	51	24
20	45	1564	61C	D8	5C	51	2A
20	46	1565	61D	D8	5D	51	29
20	47	1566	61E	D8	5E	51	3B
20	48	1567	61F	D8	5F	51	5E
20	49	1568	620	D8	60	51	2D
20	50	1569	621	D8	61	51	2F
20	51	1570	622	D8	E2	51	53
20	52	1571	623	D8	E3	51	54
20	53	1572	624	D8	E4	51	55
20	54	1573	625	D8	E5	51	56
20	55	1574	626	D8	E6	51	57
20	56	1575	627	D8	E7	51	58
20	57	1576	628	D8	E8	51	59
20	58	1577	629	D8	E9	51	5A
20	59	1578	62A	D8	6A	51	7C
20	60	1579	62B	D8	6B	51	2C
20	61	1580	62C	D8	6C	51	25
20	62	1581	62D	D8	6D	51	5F
20	63	1582	62E	D8	6E	51	3E
20	64	1583	62F	D8	6F	51	3F
20	65	1584	630	D8	F0	51	30
20	66	1585	631	D8	F1	51	31
20	67	1586	632	D8	F2	51	32
20	68	1587	633	D8	F3	51	33
20	69	1588	634	D8	F4	51	34
20	70	1589	635	D8	F5	51	35
20	71	1590	636	D8	F6	51	36
20	72	1591	637	D8	F7	51	37
20	73	1592	638	D8	F8	51	38
20	74	1593	639	D8	F9	51	39
20	75	1594	63A	D8	7A	51	3A
20	76	1595	63B	D8	7B	51	23
20	77	1596	63C	D8	7C	51	40
20	78	1597	63D	D8	7D	51	27
20	79	1598	63E	D8	7E	51	3D
20	80	1599	63F	D8	7F	51	22
21	01	1600	640	D9	40	52	20
21	02	1601	641	D9	C1	52	41
21	03	1602	642	D9	C2	52	42
21	04	1603	643	D9	C3	52	43
21	05	1604	644	D9	C4	52	44
21	06	1605	645	D9	C5	52	45
21	07	1606	646	D9	C6	52	46
21	08	1607	647	D9	C7	52	47
21	09	1608	648	D9	C8	52	48
21	10	1609	649	D9	C9	52	49
21	11	1610	64A	D9	4A	52	5B
21	12	1611	64B	D9	4B	52	2E

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
21	13	1612	64C	D9	4C	52	3C
21	14	1613	64D	D9	4D	52	28
21	15	1614	64E	D9	4E	52	2B
21	16	1615	64F	D9	4F	52	21
21	17	1616	650	D9	50	52	26
21	18	1617	651	D9	D1	52	4A
21	19	1618	652	D9	D2	52	4B
21	20	1619	653	D9	D3	52	4C
21	21	1620	654	D9	D4	52	4D
21	22	1621	655	D9	D5	52	4E
21	23	1622	656	D9	D6	52	4F
21	24	1623	657	D9	D7	52	50
21	25	1624	658	D9	D8	52	51
21	26	1625	659	D9	D9	52	52
21	27	1626	65A	D9	5A	52	5D
21	28	1627	65B	D9	5B	52	24
21	29	1628	65C	D9	5C	52	2A
21	30	1629	65D	D9	5D	52	29
21	31	1630	65E	D9	5E	52	3B
21	32	1631	65F	D9	5F	52	5E
21	33	1632	660	D9	60	52	2D
21	34	1633	661	D9	61	52	2F
21	35	1634	662	D9	E2	52	53
21	36	1635	663	D9	E3	52	54
21	37	1636	664	D9	E4	52	55
21	38	1637	665	D9	E5	52	56
21	39	1638	666	D9	E6	52	57
21	40	1639	667	D9	E7	52	58
21	41	1640	668	D9	E8	52	59
21	42	1641	669	D9	E9	52	5A
21	43	1642	66A	D9	6A	52	7C
21	44	1643	66B	D9	6B	52	2C
21	45	1644	66C	D9	6C	52	25
21	46	1645	66D	D9	6D	52	5F
21	47	1646	66E	D9	6E	52	3E
21	48	1647	66F	D9	6F	52	3F
21	49	1648	670	D9	F0	52	30
21	50	1649	671	D9	F1	52	31
21	51	1650	672	D9	F2	52	32
21	52	1651	673	D9	F3	52	33
21	53	1652	674	D9	F4	52	34
21	54	1653	675	D9	F5	52	35
21	55	1654	676	D9	F6	52	36
21	56	1655	677	D9	F7	52	37
21	57	1656	678	D9	F8	52	38
21	58	1657	679	D9	F9	52	39
21	59	1658	67A	D9	7A	52	3A
21	60	1659	67B	D9	7B	52	23
21	61	1660	67C	D9	7C	52	40
21	62	1661	67D	D9	7D	52	27
21	63	1662	67E	D9	7E	52	3D
21	64	1663	67F	D9	7F	52	22
21	65	1664	680	5A	40	5D	20
21	66	1665	681	5A	C1	5D	41
21	67	1666	682	5A	C2	5D	42
21	68	1667	683	5A	C3	5D	43
21	69	1668	684	5A	C4	5D	44
21	70	1669	685	5A	C5	5D	45
21	71	1670	686	5A	C6	5D	46
21	72	1671	687	5A	C7	5D	47
21	73	1672	688	5A	C8	5D	48
21	74	1673	689	5A	C9	5D	49

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
21	75	1674	68A	5A	4A	5D	5B
21	76	1675	68B	5A	4B	5D	2E
21	77	1676	68C	5A	4C	5D	3C
21	78	1677	68D	5A	4D	5D	28
21	79	1678	68E	5A	4E	5D	2B
21	80	1679	68F	5A	4F	5D	21
22	01	1680	690	5A	50	5D	26
22	02	1681	691	5A	D1	5D	4A
22	03	1682	692	5A	D2	5D	4B
22	04	1683	693	5A	D3	5D	4C
22	05	1684	694	5A	D4	5D	4D
22	06	1685	695	5A	D5	5D	4E
22	07	1686	696	5A	D6	5D	4F
22	08	1687	697	5A	D7	5D	50
22	09	1688	698	5A	D8	5D	51
22	10	1689	699	5A	D9	5D	52
22	11	1690	69A	5A	5A	5D	5D
22	12	1691	69B	5A	5B	5D	24
22	13	1692	69C	5A	5C	5D	2A
22	14	1693	69D	5A	5D	5D	29
22	15	1694	69E	5A	5E	5D	3B
22	16	1695	69F	5A	5F	5D	5E
22	17	1696	6A0	5A	60	5D	2D
22	18	1697	6A1	5A	61	5D	2F
22	19	1698	6A2	5A	E2	5D	53
22	20	1699	6A3	5A	E3	5D	54
22	21	1700	6A4	5A	E4	5D	55
22	22	1701	6A5	5A	E5	5D	56
22	23	1702	6A6	5A	E6	5D	57
22	24	1703	6A7	5A	E7	5D	58
22	25	1704	6A8	5A	E8	5D	59
22	26	1705	6A9	5A	E9	5D	5A
22	27	1706	6AA	5A	6A	5D	7C
22	28	1707	6AB	5A	6B	5D	2C
22	29	1708	6AC	5A	6C	5D	25
22	30	1709	6AD	5A	6D	5D	5F
22	31	1710	6AE	5A	6E	5D	3E
22	32	1711	6AF	5A	6F	5D	3F
22	33	1712	6B0	5A	F0	5D	30
22	34	1713	6B1	5A	F1	5D	31
22	35	1714	6B2	5A	F2	5D	32
22	36	1715	6B3	5A	F3	5D	33
22	37	1716	6B4	5A	F4	5D	34
22	38	1717	6B5	5A	F5	5D	35
22	39	1718	6B6	5A	F6	5D	36
22	40	1719	6B7	5A	F7	5D	37
22	41	1720	6B8	5A	F8	5D	38
22	42	1721	6B9	5A	F9	5D	39
22	43	1722	6BA	5A	7A	5D	3A
22	44	1723	6BB	5A	7B	5D	23
22	45	1724	6BC	5A	7C	5D	40
22	46	1725	6BD	5A	7D	5D	27
22	47	1726	6BE	5A	7E	5D	3D
22	48	1727	6BF	5A	7F	5D	22
22	49	1728	6C0	5B	40	24	20
22	50	1729	6C1	5B	C1	24	41
22	51	1730	6C2	5B	C2	24	42
22	52	1731	6C3	5B	C3	24	43
22	53	1732	6C4	5B	C4	24	44
22	54	1733	6C5	5B	C5	24	45
22	55	1734	6C6	5B	C6	24	46
22	56	1735	6C7	5B	C7	24	47

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
22	57	1736	6C8	5B	C8	24	48
22	58	1737	6C9	5B	C9	24	49
22	59	1738	6CA	5B	4A	24	5B
22	60	1739	6CB	5B	4B	24	2E
22	61	1740	6CC	5B	4C	24	3C
22	62	1741	6CD	5B	4D	24	28
22	63	1742	6CE	5B	4E	24	2B
22	64	1743	6CF	5B	4F	24	21
22	65	1744	6D0	5B	50	24	26
22	66	1745	6D1	5B	D1	24	4A
22	67	1746	6D2	5B	D2	24	4B
22	68	1747	6D3	5B	D3	24	4C
22	69	1748	6D4	5B	D4	24	4D
22	70	1749	6D5	5B	D5	24	4E
22	71	1750	6D6	5B	D6	24	4F
22	72	1751	6D7	5B	D7	24	50
22	73	1752	6D8	5B	D8	24	51
22	74	1753	6D9	5B	D9	24	52
22	75	1754	6DA	5B	5A	24	5D
22	76	1755	6DB	5B	5B	24	24
22	77	1756	6DC	5B	5C	24	2A
22	78	1757	6DD	5B	5D	24	29
22	79	1758	6DE	5B	5E	24	3B
22	80	1759	6DF	5B	5F	24	5E
23	01	1760	6E0	5B	60	24	2D
23	02	1761	6E1	5B	61	24	2F
23	03	1762	6E2	5B	E2	24	53
23	04	1763	6E3	5B	E3	24	54
23	05	1764	6E4	5B	E4	24	55
23	06	1765	6E5	5B	E5	24	56
23	07	1766	6E6	5B	E6	24	57
23	08	1767	6E7	5B	E7	24	58
23	09	1768	6E8	5B	E8	24	59
23	10	1769	6E9	5B	E9	24	5A
23	11	1770	6EA	5B	6A	24	7C
23	12	1771	6EB	5B	6B	24	2C
23	13	1772	6EC	5B	6C	24	25
23	14	1773	6ED	5B	6D	24	5F
23	15	1774	6EE	5B	6E	24	3E
23	16	1775	6EF	5B	6F	24	3F
23	17	1776	6F0	5B	F0	24	30
23	18	1777	6F1	5B	F1	24	31
23	19	1778	6F2	5B	F2	24	32
23	20	1779	6F3	5B	F3	24	33
23	21	1780	6F4	5B	F4	24	34
23	22	1781	6F5	5B	F5	24	35
23	23	1782	6F6	5B	F6	24	36
23	24	1783	6F7	5B	F7	24	37
23	25	1784	6F8	5B	F8	24	38
23	26	1785	6F9	5B	F9	24	39
23	27	1786	6FA	5B	7A	24	3A
23	28	1787	6FB	5B	7B	24	23
23	29	1788	6FC	5B	7C	24	40
23	30	1789	6FD	5B	7D	24	27
23	31	1790	6FE	5B	7E	24	3D
23	32	1791	6FF	5B	7F	24	22
23	33	1792	700	5C	40	2A	20
23	34	1793	701	5C	C1	2A	41
23	35	1794	702	5C	C2	2A	42
23	36	1795	703	5C	C3	2A	43
23	37	1796	704	5C	C4	2A	44
23	38	1797	705	5C	C5	2A	45

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
23	39	1798	706	5C	C6	2A	46
23	40	1799	707	5C	C7	2A	47
23	41	1800	708	5C	C8	2A	48
23	42	1801	709	5C	C9	2A	49
23	43	1802	70A	5C	4A	2A	5B
23	44	1803	70B	5C	4B	2A	2E
23	45	1804	70C	5C	4C	2A	3C
23	46	1805	70D	5C	4D	2A	28
23	47	1806	70E	5C	4E	2A	2B
23	48	1807	70F	5C	4F	2A	21
23	49	1808	710	5C	50	2A	26
23	50	1809	711	5C	D1	2A	4A
23	51	1810	712	5C	D2	2A	4B
23	52	1811	713	5C	D3	2A	4C
23	53	1812	714	5C	D4	2A	4D
23	54	1813	715	5C	D5	2A	4E
23	55	1814	716	5C	D6	2A	4F
23	56	1815	717	5C	D7	2A	50
23	57	1816	718	5C	D8	2A	51
23	58	1817	719	5C	D9	2A	52
23	59	1818	71A	5C	5A	2A	5D
23	60	1819	71B	5C	5B	2A	24
23	61	1820	71C	5C	5C	2A	2A
23	62	1821	71D	5C	5D	2A	29
23	63	1822	71E	5C	5E	2A	3B
23	64	1823	71F	5C	5F	2A	5E
23	65	1824	720	5C	60	2A	2D
23	66	1825	721	5C	61	2A	2F
23	67	1826	722	5C	E2	2A	53
23	68	1827	723	5C	E3	2A	54
23	69	1828	724	5C	E4	2A	55
23	70	1829	725	5C	E5	2A	56
23	71	1830	726	5C	E6	2A	57
23	72	1831	727	5C	E7	2A	58
23	73	1832	728	5C	E8	2A	59
23	74	1833	729	5C	E9	2A	5A
23	75	1834	72A	5C	6A	2A	7C
23	76	1835	72B	5C	6B	2A	2C
23	77	1836	72C	5C	6C	2A	25
23	78	1837	72D	5C	6D	2A	5F
23	79	1838	72E	5C	6E	2A	3E
23	80	1839	72F	5C	6F	2A	3F
24	01	1840	730	5C	F0	2A	30
24	02	1841	731	5C	F1	2A	31
24	03	1842	732	5C	F2	2A	32
24	04	1843	733	5C	F3	2A	33
24	05	1844	734	5C	F4	2A	34
24	06	1845	735	5C	F5	2A	35
24	07	1846	736	5C	F6	2A	36
24	08	1847	737	5C	F7	2A	37
24	09	1848	738	5C	F8	2A	38
24	10	1849	739	5C	F9	2A	39
24	11	1850	73A	5C	7A	2A	3A
24	12	1851	73B	5C	7B	2A	23
24	13	1852	73C	5C	7C	2A	40
24	14	1853	73D	5C	7D	2A	27
24	15	1854	73E	5C	7E	2A	3D
24	16	1855	73F	5C	7F	2A	22
24	17	1856	740	5D	40	29	20
24	18	1857	741	5D	C1	29	41
24	19	1858	742	5D	C2	29	42
24	20	1859	743	5D	C3	29	43

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
24	21	1860	744	5D	C4	29	44
24	22	1861	745	5D	C5	29	45
24	23	1862	746	5D	C6	29	46
24	24	1863	747	5D	C7	29	47
24	25	1864	748	5D	C8	29	48
24	26	1865	749	5D	C9	29	49
24	27	1866	74A	5D	4A	29	5B
24	28	1867	74B	5D	4B	29	2E
24	29	1868	74C	5D	4C	29	3C
24	30	1869	74D	5D	4D	29	28
24	31	1870	74E	5D	4E	29	2B
24	32	1871	74F	5D	4F	29	21
24	33	1872	750	5D	50	29	26
24	34	1873	751	5D	D1	29	4A
24	35	1874	752	5D	D2	29	4B
24	36	1875	753	5D	D3	29	4C
24	37	1876	754	5D	D4	29	4D
24	38	1877	755	5D	D5	29	4E
24	39	1878	756	5D	D6	29	4F
24	40	1879	757	5D	D7	29	50
24	41	1880	758	5D	D8	29	51
24	42	1881	759	5D	D9	29	52
24	43	1882	75A	5D	5A	29	5D
24	44	1883	75B	5D	5B	29	24
24	45	1884	75C	5D	5C	29	2A
24	46	1885	75D	5D	5D	29	29
24	47	1886	75E	5D	5E	29	3B
24	48	1887	75F	5D	5F	29	5E
24	49	1888	760	5D	60	29	2D
24	50	1889	761	5D	61	29	2F
24	51	1890	762	5D	E2	29	53
24	52	1891	763	5D	E3	29	54
24	53	1892	764	5D	E4	29	55
24	54	1893	765	5D	E5	29	56
24	55	1894	766	5D	E6	29	57
24	56	1895	767	5D	E7	29	58
24	57	1896	768	5D	E8	29	59
24	58	1897	769	5D	E9	29	5A
24	59	1898	76A	5D	6A	29	7C
24	60	1899	76B	5D	6B	29	2C
24	61	1900	76C	5D	6C	29	25
24	62	1901	76D	5D	6D	29	5F
24	63	1902	76E	5D	6E	29	3E
24	64	1903	76F	5D	6F	29	3F
24	65	1904	770	5D	F0	29	30
24	66	1905	771	5D	F1	29	31
24	67	1906	772	5D	F2	29	32
24	68	1907	773	5D	F3	29	33
24	69	1908	774	5D	F4	29	34
24	70	1909	775	5D	F5	29	35
24	71	1910	776	5D	F6	29	36
24	72	1911	777	5D	F7	29	37
24	73	1912	778	5D	F8	29	38
24	74	1913	779	5D	F9	29	39
24	75	1914	77A	5D	7A	29	3A
24	76	1915	77B	5D	7B	29	23
24	77	1916	77C	5D	7C	29	40
24	78	1917	77D	5D	7D	29	27
24	79	1918	77E	5D	7E	29	3D
24	80	1919	77F	5D	7F	29	22
25	01	1920	780	5E	40	3B	20
25	02	1921	781	5E	C1	3B	41

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
25	03	1922	782	5E	C2	3B	42
25	04	1923	783	5E	C3	3B	43
25	05	1924	784	5E	C4	3B	44
25	06	1925	785	5E	C5	3B	45
25	07	1926	786	5E	C6	3B	46
25	08	1927	787	5E	C7	3B	47
25	09	1928	788	5E	C8	3B	48
25	10	1929	789	5E	C9	3B	49
25	11	1930	78A	5E	4A	3B	5B
25	12	1931	78B	5E	4B	3B	2E
25	13	1932	78C	5E	4C	3B	3C
25	14	1933	78D	5E	4D	3B	28
25	15	1934	78E	5E	4E	3B	2B
25	16	1935	78F	5E	4F	3B	21
25	17	1936	790	5E	50	3B	26
25	18	1937	791	5E	D1	3B	4A
25	19	1938	792	5E	D2	3B	4B
25	20	1939	793	5E	D3	3B	4C
25	21	1940	794	5E	D4	3B	4D
25	22	1941	795	5E	D5	3B	4E
25	23	1942	796	5E	D6	3B	4F
25	24	1943	797	5E	D7	3B	50
25	25	1944	798	5E	D8	3B	51
25	26	1945	799	5E	D9	3B	52
25	27	1946	79A	5E	5A	3B	5D
25	28	1947	79B	5E	5B	3B	24
25	29	1948	79C	5E	5C	3B	2A
25	30	1949	79D	5E	5D	3B	29
25	31	1950	79E	5E	5E	3B	3B
25	32	1951	79F	5E	5F	3B	5E
25	33	1952	7A0	5E	60	3B	2D
25	34	1953	7A1	5E	61	3B	2F
25	35	1954	7A2	5E	E2	3B	53
25	36	1955	7A3	5E	E3	3B	54
25	37	1956	7A4	5E	E4	3B	55
25	38	1957	7A5	5E	E5	3B	56
25	39	1958	7A6	5E	E6	3B	57
25	40	1959	7A7	5E	E7	3B	58
25	41	1960	7A8	5E	E8	3B	59
25	42	1961	7A9	5E	E9	3B	5A
25	43	1962	7AA	5E	6A	3B	7C
25	44	1963	7AB	5E	6B	3B	2C
25	45	1964	7AC	5E	6C	3B	25
25	46	1965	7AD	5E	6D	3B	5F
25	47	1966	7AE	5E	6E	3B	3E
25	48	1967	7AF	5E	6F	3B	3F
25	49	1968	7B0	5E	F0	3B	30
25	50	1969	7B1	5E	F1	3B	31
25	51	1970	7B2	5E	F2	3B	32
25	52	1971	7B3	5E	F3	3B	33
25	53	1972	7B4	5E	F4	3B	34
25	54	1973	7B5	5E	F5	3B	35
25	55	1974	7B6	5E	F6	3B	36
25	56	1975	7B7	5E	F7	3B	37
25	57	1976	7B8	5E	F8	3B	38
25	58	1977	7B9	5E	F9	3B	39
25	59	1978	7BA	5E	7A	3B	3A
25	60	1979	7BB	5E	7B	3B	23
25	61	1980	7BC	5E	7C	3B	40
25	62	1981	7BD	5E	7D	3B	27
25	63	1982	7BE	5E	7E	3B	3D
25	64	1983	7BF	5E	7F	3B	22

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
25	65	1984	7C0	5F	40	5E	20
25	66	1985	7C1	5F	C1	5E	41
25	67	1986	7C2	5F	C2	5E	42
25	68	1987	7C3	5F	C3	5E	43
25	69	1988	7C4	5F	C4	5E	44
25	70	1989	7C5	5F	C5	5E	45
25	71	1990	7C6	5F	C6	5E	46
25	72	1991	7C7	5F	C7	5E	47
25	73	1992	7C8	5F	C8	5E	48
25	74	1993	7C9	5F	C9	5E	49
25	75	1994	7CA	5F	4A	5E	5B
25	76	1995	7CB	5F	4B	5E	2E
25	77	1996	7CC	5F	4C	5E	3C
25	78	1997	7CD	5F	4D	5E	28
25	79	1998	7CE	5F	4E	5E	2B
25	80	1999	7CF	5F	4F	5E	21
26	01	2000	7D0	5F	50	5E	26
26	02	2001	7D1	5F	D1	5E	4A
26	03	2002	7D2	5F	D2	5E	4B
26	04	2003	7D3	5F	D3	5E	4C
26	05	2004	7D4	5F	D4	5E	4D
26	06	2005	7D5	5F	D5	5E	4E
26	07	2006	7D6	5F	D6	5E	4F
26	08	2007	7D7	5F	D7	5E	50
26	09	2008	7D8	5F	D8	5E	51
26	10	2009	7D9	5F	D9	5E	52
26	11	2010	7DA	5F	5A	5E	5D
26	12	2011	7DB	5F	5B	5E	24
26	13	2012	7DC	5F	5C	5E	2A
26	14	2013	7DD	5F	5D	5E	29
26	15	2014	7DE	5F	5E	5E	3B
26	16	2015	7DF	5F	5F	5E	5E
26	17	2016	7E0	5F	60	5E	2D
26	18	2017	7E1	5F	61	5E	2F
26	19	2018	7E2	5F	E2	5E	53
26	20	2019	7E3	5F	E3	5E	54
26	21	2020	7E4	5F	E4	5E	55
26	22	2021	7E5	5F	E5	5E	56
26	23	2022	7E6	5F	E6	5E	57
26	24	2023	7E7	5F	E7	5E	58
26	25	2024	7E8	5F	E8	5E	59
26	26	2025	7E9	5F	E9	5E	5A
26	27	2026	7EA	5F	6A	5E	7C
26	28	2027	7EB	5F	6B	5E	2C
26	29	2028	7EC	5F	6C	5E	25
26	30	2029	7ED	5F	6D	5E	5F
26	31	2030	7EE	5F	6E	5E	3E
26	32	2031	7EF	5F	6F	5E	3F
26	33	2032	7F0	5F	F0	5E	30
26	34	2033	7F1	5F	F1	5E	31
26	35	2034	7F2	5F	F2	5E	32
26	36	2035	7F3	5F	F3	5E	33
26	37	2036	7F4	5F	F4	5E	34
26	38	2037	7F5	5F	F5	5E	35
26	39	2038	7F6	5F	F6	5E	36
26	40	2039	7F7	5F	F7	5E	37
26	41	2040	7F8	5F	F8	5E	38
26	42	2041	7F9	5F	F9	5E	39
26	43	2042	7FA	5F	7A	5E	3A
26	44	2043	7FB	5F	7B	5E	23
26	45	2044	7FC	5F	7C	5E	40
26	46	2045	7FD	5F	7D	5E	27

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
26	47	2046	7FE	5F	7E	5E	3D
26	48	2047	7FF	5F	7F	5E	22
26	49	2048	800	60	40	2D	20
26	50	2049	801	60	C1	2D	41
26	51	2050	802	60	C2	2D	42
26	52	2051	803	60	C3	2D	43
26	53	2052	804	60	C4	2D	44
26	54	2053	805	60	C5	2D	45
26	55	2054	806	60	C6	2D	46
26	56	2055	807	60	C7	2D	47
26	57	2056	808	60	C8	2D	48
26	58	2057	809	60	C9	2D	49
26	59	2058	80A	60	4A	2D	5B
26	60	2059	80B	60	4B	2D	2E
26	61	2060	80C	60	4C	2D	3C
26	62	2061	80D	60	4D	2D	28
26	63	2062	80E	60	4E	2D	2B
26	64	2063	80F	60	4F	2D	21
26	65	2064	810	60	50	2D	26
26	66	2065	811	60	D1	2D	4A
26	67	2066	812	60	D2	2D	4B
26	68	2067	813	60	D3	2D	4C
26	69	2068	814	60	D4	2D	4D
26	70	2069	815	60	D5	2D	4E
26	71	2070	816	60	D6	2D	4F
26	72	2071	817	60	D7	2D	50
26	73	2072	818	60	D8	2D	51
26	74	2073	819	60	D9	2D	52
26	75	2074	81A	60	5A	2D	5D
26	76	2075	81B	60	5B	2D	24
26	77	2076	81C	60	5C	2D	2A
26	78	2077	81D	60	5D	2D	29
26	79	2078	81E	60	5E	2D	3B
26	80	2079	81F	60	5F	2D	5E
27	01	2080	820	60	60	2D	2D
27	02	2081	821	60	61	2D	2F
27	03	2082	822	60	E2	2D	53
27	04	2083	823	60	E3	2D	54
27	05	2084	824	60	E4	2D	55
27	06	2085	825	60	E5	2D	56
27	07	2086	826	60	E6	2D	57
27	08	2087	827	60	E7	2D	58
27	09	2088	828	60	E8	2D	59
27	10	2089	829	60	E9	2D	5A
27	11	2090	82A	60	6A	2D	7C
27	12	2091	82B	60	6B	2D	2C
27	13	2092	82C	60	6C	2D	25
27	14	2093	82D	60	6D	2D	5F
27	15	2094	82E	60	6E	2D	3E
27	16	2095	82F	60	6F	2D	3F
27	17	2096	830	60	F0	2D	30
27	18	2097	831	60	F1	2D	31
27	19	2098	832	60	F2	2D	32
27	20	2099	833	60	F3	2D	33
27	21	2100	834	60	F4	2D	34
27	22	2101	835	60	F5	2D	35
27	23	2102	836	60	F6	2D	36
27	24	2103	837	60	F7	2D	37
27	25	2104	838	60	F8	2D	38
27	26	2105	839	60	F9	2D	39
27	27	2106	83A	60	7A	2D	3A
27	28	2107	83B	60	7B	2D	23

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
27	29	2108	83C	60	7C	2D	40
27	30	2109	83D	60	7D	2D	27
27	31	2110	83E	60	7E	2D	3D
27	32	2111	83F	60	7F	2D	22
27	33	2112	840	61	40	2F	20
27	34	2113	841	61	C1	2F	41
27	35	2114	842	61	C2	2F	42
27	36	2115	843	61	C3	2F	43
27	37	2116	844	61	C4	2F	44
27	38	2117	845	61	C5	2F	45
27	39	2118	846	61	C6	2F	46
27	40	2119	847	61	C7	2F	47
27	41	2120	848	61	C8	2F	48
27	42	2121	849	61	C9	2F	49
27	43	2122	84A	61	4A	2F	5B
27	44	2123	84B	61	4B	2F	2E
27	45	2124	84C	61	4C	2F	3C
27	46	2125	84D	61	4D	2F	28
27	47	2126	84E	61	4E	2F	2B
27	48	2127	84F	61	4F	2F	21
27	49	2128	850	61	50	2F	26
27	50	2129	851	61	D1	2F	4A
27	51	2130	852	61	D2	2F	4B
27	52	2131	853	61	D3	2F	4C
27	53	2132	854	61	D4	2F	4D
27	54	2133	855	61	D5	2F	4E
27	55	2134	856	61	D6	2F	4F
27	56	2135	857	61	D7	2F	50
27	57	2136	858	61	D8	2F	51
27	58	2137	859	61	D9	2F	52
27	59	2138	85A	61	5A	2F	5D
27	60	2139	85B	61	5B	2F	24
27	61	2140	85C	61	5C	2F	2A
27	62	2141	85D	61	5D	2F	29
27	63	2142	85E	61	5E	2F	3B
27	64	2143	85F	61	5F	2F	5E
27	65	2144	860	61	60	2F	2D
27	66	2145	861	61	61	2F	2F
27	67	2146	862	61	E2	2F	53
27	68	2147	863	61	E3	2F	54
27	69	2148	864	61	E4	2F	55
27	70	2149	865	61	E5	2F	56
27	71	2150	866	61	E6	2F	57
27	72	2151	867	61	E7	2F	58
27	73	2152	868	61	E8	2F	59
27	74	2153	869	61	E9	2F	5A
27	75	2154	86A	61	6A	2F	7C
27	76	2155	86B	61	6B	2F	2C
27	77	2156	86C	61	6C	2F	25
27	78	2157	86D	61	6D	2F	5F
27	79	2158	86E	61	6E	2F	3E
27	80	2159	86F	61	6F	2F	3F
28	01	2160	870	61	F0	2F	30
28	02	2161	871	61	F1	2F	31
28	03	2162	872	61	F2	2F	32
28	04	2163	873	61	F3	2F	33
28	05	2164	874	61	F4	2F	34
28	06	2165	875	61	F5	2F	35
28	07	2166	876	61	F6	2F	36
28	08	2167	877	61	F7	2F	37
28	09	2168	878	61	F8	2F	38
28	10	2169	879	61	F9	2F	39

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
28	11	2170	87A	61	7A	2F	3A
28	12	2171	87B	61	7B	2F	23
28	13	2172	87C	61	7C	2F	40
28	14	2173	87D	61	7D	2F	27
28	15	2174	87E	61	7E	2F	3D
28	16	2175	87F	61	7F	2F	22
28	17	2176	880	E2	40	53	20
28	18	2177	881	E2	C1	53	41
28	19	2178	882	E2	C2	53	42
28	20	2179	883	E2	C3	53	43
28	21	2180	884	E2	C4	53	44
28	22	2181	885	E2	C5	53	45
28	23	2182	886	E2	C6	53	46
28	24	2183	887	E2	C7	53	47
28	25	2184	888	E2	C8	53	48
28	26	2185	889	E2	C9	53	49
28	27	2186	88A	E2	4A	53	5B
28	28	2187	88B	E2	4B	53	2E
28	29	2188	88C	E2	4C	53	3C
28	30	2189	88D	E2	4D	53	28
28	31	2190	88E	E2	4E	53	2B
28	32	2191	88F	E2	4F	53	21
28	33	2192	890	E2	50	53	26
28	34	2193	891	E2	D1	53	4A
28	35	2194	892	E2	D2	53	4B
28	36	2195	893	E2	D3	53	4C
28	37	2196	894	E2	D4	53	4D
28	38	2197	895	E2	D5	53	4F
28	39	2198	896	E2	D6	53	4F
28	40	2199	897	E2	D7	53	50
28	41	2200	898	E2	D8	53	51
28	42	2201	899	E2	D9	53	52
28	43	2202	89A	E2	5A	53	5D
28	44	2203	89B	E2	5B	53	24
28	45	2204	89C	E2	5C	53	2A
28	46	2205	89D	E2	5D	53	29
28	47	2206	89E	E2	5E	53	3B
28	48	2207	89F	E2	5F	53	5E
28	49	2208	8A0	E2	60	53	2D
28	50	2209	8A1	E2	61	53	2F
28	51	2210	8A2	E2	E2	53	53
28	52	2211	8A3	E2	E3	53	54
28	53	2212	8A4	E2	E4	53	55
28	54	2213	8A5	E2	E5	53	56
28	55	2214	8A6	E2	E6	53	57
28	56	2215	8A7	E2	E7	53	58
28	57	2216	8A8	E2	E8	53	59
28	58	2217	8A9	E2	E9	53	5A
28	59	2218	8AA	E2	6A	53	7C
28	60	2219	8AB	E2	6B	53	2C
28	61	2220	8AC	E2	6C	53	25
28	62	2221	8AD	E2	6D	53	5F
28	63	2222	8AE	E2	6E	53	3E
28	64	2223	8AF	E2	6F	53	3F
28	65	2224	8B0	E2	F0	53	30
28	66	2225	8B1	E2	F1	53	31
28	67	2226	8B2	E2	F2	53	32
28	68	2227	8B3	E2	F3	53	33
28	69	2228	8B4	E2	F4	53	34
28	70	2229	8B5	E2	F5	53	35
28	71	2230	8B6	E2	F6	53	36
28	72	2231	8B7	E2	F7	53	37
28	73	2232	8B8	E2	F8	53	38

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
28	74	2233	8B9	E2	F9	53	39
28	75	2234	8BA	E2	7A	53	3A
28	76	2235	8BB	E2	7B	53	23
28	77	2236	8BC	E2	7C	53	40
28	78	2237	8BD	E2	7D	53	27
28	79	2238	8BE	E2	7E	53	3D
28	80	2239	8BF	E2	7F	53	22
29	01	2240	8C0	E3	40	54	20
29	02	2241	8C1	E3	C1	54	41
29	03	2242	8C2	E3	C2	54	42
29	04	2243	8C3	E3	C3	54	43
29	05	2244	8C4	E3	C4	54	44
29	06	2245	8C5	E3	C5	54	45
29	07	2246	8C6	E3	C6	54	46
29	08	2247	8C7	E3	C7	54	47
29	09	2248	8C8	E3	C8	54	48
29	10	2249	8C9	E3	C9	54	49
29	11	2250	8CA	E3	4A	54	5B
29	12	2251	8CB	E3	4B	54	2E
29	13	2252	8CC	E3	4C	54	3C
29	14	2253	8CD	E3	4D	54	28
29	15	2254	8CE	E3	4E	54	2B
29	16	2255	8CF	E3	4F	54	21
29	17	2256	8D0	E3	50	54	26
29	18	2257	8D1	E3	D1	54	4A
29	19	2258	8D2	E3	D2	54	4B
29	20	2259	8D3	E3	D3	54	4C
29	21	2260	8D4	E3	D4	54	4D
29	22	2261	8D5	E3	D5	54	4E
29	23	2262	8D6	E3	D6	54	4F
29	24	2263	8D7	E3	D7	54	50
29	25	2264	8D8	E3	D8	54	51
29	26	2265	8D9	E3	D9	54	52
29	27	2266	8DA	E3	5A	54	5D
29	28	2267	8DB	E3	5B	54	24
29	29	2268	8DC	E3	5C	54	2A
29	30	2269	8DD	E3	5D	54	29
29	31	2270	8DE	E3	5E	54	3B
29	32	2271	8DF	E3	5F	54	5E
29	33	2272	8E0	E3	60	54	2D
29	34	2273	8E1	E3	61	54	2F
29	35	2274	8E2	E3	E2	54	53
29	36	2275	8E3	E3	E3	54	54
29	37	2276	8E4	E3	E4	54	55
29	38	2277	8E5	E3	E5	54	56
29	39	2278	8E6	E3	E6	54	57
29	40	2279	8E7	E3	E7	54	58
29	41	2280	8E8	E3	E8	54	59
29	42	2281	8E9	E3	E9	54	5A
29	43	2282	8EA	E3	6A	54	7C
29	44	2283	8EB	E3	6B	54	2C
29	45	2284	8EC	E3	6C	54	25
29	46	2285	8ED	E3	6D	54	5F
29	47	2286	8EE	E3	6E	54	3E
29	48	2287	8EF	E3	6F	54	3F
29	49	2288	8F0	E3	F0	54	30
29	50	2289	8F1	E3	F1	54	31
29	51	2290	8F2	E3	F2	54	32
29	52	2291	8F3	E3	F3	54	33
29	53	2292	8F4	E3	F4	54	34
29	54	2293	8F5	E3	F5	54	35
29	55	2294	8F6	E3	F6	54	36

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
29	56	2295	8F7	E3	F7	54	37
29	57	2296	8F8	E3	F8	54	38
29	58	2297	8F9	E3	F9	54	39
29	59	2298	8FA	E3	7A	54	3A
29	60	2299	8FB	E3	7B	54	23
29	61	2300	8FC	E3	7C	54	40
29	62	2301	8FD	E3	7D	54	27
29	63	2302	8FE	E3	7E	54	3D
29	64	2303	8FF	E3	7F	54	22
29	65	2304	900	E4	40	55	20
29	66	2305	901	E4	C1	55	41
29	67	2306	902	E4	C2	55	42
29	68	2307	903	E4	C3	55	43
29	69	2308	904	E4	C4	55	44
29	70	2309	905	E4	C5	55	45
29	71	2310	906	E4	C6	55	46
29	72	2311	907	E4	C7	55	47
29	73	2312	908	E4	C8	55	48
29	74	2313	909	E4	C9	55	49
29	75	2314	90A	E4	4A	55	5B
29	76	2315	90B	E4	4B	55	2E
29	77	2316	90C	E4	4C	55	3C
29	78	2317	90D	E4	4D	55	28
29	79	2318	90E	E4	4E	55	2B
29	80	2319	90F	E4	4F	55	21
30	01	2320	910	E4	50	55	26
30	02	2321	911	E4	D1	55	4A
30	03	2322	912	E4	D2	55	4B
30	04	2323	913	E4	D3	55	4C
30	05	2324	914	E4	D4	55	4D
30	06	2325	915	E4	D5	55	4E
30	07	2326	916	E4	D6	55	4F
30	08	2327	917	E4	D7	55	50
30	09	2328	918	E4	D8	55	51
30	10	2329	919	E4	D9	55	52
30	11	2330	91A	E4	5A	55	5D
30	12	2331	91B	E4	5B	55	24
30	13	2332	91C	E4	5C	55	2A
30	14	2333	91D	E4	5D	55	29
30	15	2334	91E	E4	5E	55	3B
30	16	2335	91F	E4	5F	55	5E
30	17	2336	920	E4	60	55	2D
30	18	2337	921	E4	61	55	2F
30	19	2338	922	E4	E2	55	53
30	20	2339	923	E4	E3	55	54
30	21	2340	924	E4	E4	55	55
30	22	2341	925	E4	E5	55	56
30	23	2342	926	E4	E6	55	57
30	24	2343	927	E4	E7	55	58
30	25	2344	928	E4	E8	55	59
30	26	2345	929	E4	E9	55	5A
30	27	2346	92A	E4	6A	55	7C
30	28	2347	92B	E4	6B	55	2C
30	29	2348	92C	E4	6C	55	25
30	30	2349	92D	E4	6D	55	5F
30	31	2350	92E	E4	6E	55	3E
30	32	2351	92F	E4	6F	55	3F
30	33	2352	930	E4	F0	55	30
30	34	2353	931	E4	F1	55	31
30	35	2354	932	E4	F2	55	32
30	36	2355	933	E4	F3	55	33
30	37	2356	934	E4	F4	55	34

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
30	38	2357	935	E4	F5	55	35
30	39	2358	936	E4	F6	55	36
30	40	2359	937	E4	F7	55	37
30	41	2360	938	E4	F8	55	38
30	42	2361	939	E4	F9	55	39
30	43	2362	93A	E4	7A	55	3A
30	44	2363	93B	E4	7B	55	23
30	45	2364	93C	E4	7C	55	40
30	46	2365	93D	E4	7D	55	27
30	47	2366	93E	E4	7E	55	3D
30	48	2367	93F	E4	7F	55	22
30	49	2368	940	E5	40	56	20
30	50	2369	941	E5	C1	56	41
30	51	2370	942	E5	C2	56	42
30	52	2371	943	E5	C3	56	43
30	53	2372	944	E5	C4	56	44
30	54	2373	945	E5	C5	56	45
30	55	2374	946	E5	C6	56	46
30	56	2375	947	E5	C7	56	47
30	57	2376	948	E5	C8	56	48
30	58	2377	949	E5	C9	56	49
30	59	2378	94A	E5	4A	56	5B
30	60	2379	94B	E5	4B	56	2E
30	61	2380	94C	E5	4C	56	3C
30	62	2381	94D	E5	4D	56	28
30	63	2382	94E	E5	4E	56	2B
30	64	2383	94F	E5	4F	56	21
30	65	2384	950	E5	50	56	26
30	66	2385	951	E5	D1	56	4A
30	67	2386	952	E5	D2	56	4B
30	68	2387	953	E5	D3	56	4C
30	69	2388	954	E5	D4	56	4D
30	70	2389	955	E5	D5	56	4E
30	71	2390	956	E5	D6	56	4F
30	72	2391	957	E5	D7	56	50
30	73	2392	958	E5	D8	56	51
30	74	2393	959	E5	D9	56	52
30	75	2394	95A	E5	5A	56	5D
30	76	2395	95B	E5	5B	56	24
30	77	2396	95C	E5	5C	56	2A
30	78	2397	95D	E5	5D	56	29
30	79	2398	95E	E5	5E	56	3B
30	80	2399	95F	E5	5F	56	5E
31	01	2400	960	F5	60	56	2D
31	02	2401	961	E5	61	56	2F
31	03	2402	962	F5	E2	56	53
31	04	2403	963	E5	E3	56	54
31	05	2404	964	E5	E4	56	55
31	06	2405	965	E5	E5	56	56
31	07	2406	966	E5	E6	56	57
31	08	2407	967	E5	E7	56	58
31	09	2408	968	E5	E8	56	59
31	10	2409	969	E5	E9	56	5A
31	11	2410	96A	E5	6A	56	7C
31	12	2411	96B	E5	6B	56	2C
31	13	2412	96C	E5	6C	56	25
31	14	2413	96D	E5	6D	56	5F
31	15	2414	96E	E5	6E	56	3E
31	16	2415	96F	E5	6F	56	3F
31	17	2416	970	E5	F0	56	30
31	18	2417	971	E5	F1	56	31
31	19	2418	972	E5	F2	56	32

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
31	20	2419	973	E5	F3	56	33
31	21	2420	974	E5	F4	56	34
31	22	2421	975	E5	F5	56	35
31	23	2422	976	E5	F6	56	36
31	24	2423	977	E5	F7	56	37
31	25	2424	978	E5	F8	56	38
31	26	2425	979	E5	F9	56	39
31	27	2426	97A	E5	7A	56	3A
31	28	2427	97B	E5	7B	56	23
31	29	2428	97C	E5	7C	56	40
31	30	2429	97D	E5	7D	56	27
31	31	2430	97E	E5	7E	56	3D
31	32	2431	97F	E5	7F	56	22
31	33	2432	980	E6	40	57	20
31	34	2433	981	E6	C1	57	41
31	35	2434	982	E6	C2	57	42
31	36	2435	983	E6	C3	57	43
31	37	2436	984	E6	C4	57	44
31	38	2437	985	E6	C5	57	45
31	39	2438	986	E6	C6	57	46
31	40	2439	987	E6	C7	57	47
31	41	2440	988	E6	C8	57	48
31	42	2441	989	E6	C9	57	49
31	43	2442	98A	E6	4A	57	5B
31	44	2443	98B	E6	4B	57	2E
31	45	2444	98C	E6	4C	57	3C
31	46	2445	98D	E6	4D	57	28
31	47	2446	98E	E6	4E	57	2B
31	48	2447	98F	E6	4F	57	21
31	49	2448	990	E6	50	57	26
31	50	2449	991	E6	D1	57	4A
31	51	2450	992	E6	D2	57	4B
31	52	2451	993	E6	D3	57	4C
31	53	2452	994	E6	D4	57	4D
31	54	2453	995	E6	D5	57	4E
31	55	2454	996	E6	D6	57	4F
31	56	2455	997	E6	D7	57	50
31	57	2456	998	E6	D8	57	51
31	58	2457	999	E6	D9	57	52
31	59	2458	99A	E6	5A	57	5D
31	60	2459	99B	E6	5B	57	24
31	61	2460	99C	E6	5C	57	2A
31	62	2461	99D	E6	5D	57	29
31	63	2462	99E	E6	5E	57	3B
31	64	2463	99F	E6	5F	57	5E
31	65	2464	9A0	E6	60	57	2D
31	66	2465	9A1	E6	61	57	2F
31	67	2466	9A2	E6	E2	57	53
31	68	2467	9A3	E6	E3	57	54
31	69	2468	9A4	E6	E4	57	55
31	70	2469	9A5	E6	E5	57	56
31	71	2470	9A6	E6	E6	57	57
31	72	2471	9A7	E6	E7	57	58
31	73	2472	9A8	E6	E8	57	59
31	74	2473	9A9	E6	E9	57	5A
31	75	2474	9AA	E6	6A	57	7C
31	76	2475	9AB	E6	6B	57	2C
31	77	2476	9AC	E6	6C	57	25
31	78	2477	9AD	E6	6D	57	5F
31	79	2478	9AE	E6	6E	57	3E
31	80	2479	9AF	E6	6F	57	3F

80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		Buffer Address (Hex) ASCII	
32	01	2480	9B0	E6	F0	57	30
32	02	2481	9B1	E6	F1	57	31
32	03	2482	9B2	E6	F2	57	32
32	04	2483	9B3	E6	F3	57	33
32	05	2484	9B4	E6	F4	57	34
32	06	2485	9B5	E6	F5	57	35
32	07	2486	9B6	E6	F6	57	36
32	08	2487	9B7	E6	F7	57	37
32	09	2488	9B8	E6	F8	57	38
32	10	2489	9B9	E6	F9	57	39
32	11	2490	9BA	E6	7A	57	3A
32	12	2491	9BB	E6	7B	57	23
32	13	2492	9BC	E6	7C	57	40
32	14	2493	9BD	E6	7D	57	27
32	15	2494	9BE	E6	7E	57	3D
32	16	2495	9BF	E6	7F	57	22
32	17	2496	9C0	E7	40	58	20
32	18	2497	9C1	E7	C1	58	41
32	19	2498	9C2	E7	C2	58	42
32	20	2499	9C3	E7	C3	58	43
32	21	2500	9C4	E7	C4	58	44
32	22	2501	9C5	E7	C5	58	45
32	23	2502	9C6	E7	C6	58	46
32	24	2503	9C7	E7	C7	58	47
32	25	2504	9C8	E7	C8	58	48
32	26	2505	9C9	E7	C9	58	49
32	27	2506	9CA	E7	4A	58	5B
32	28	2507	9CB	E7	4B	58	2E
32	29	2508	9CC	E7	4C	5B	3C
32	30	2509	9CD	E7	4D	58	28
32	31	2510	9CE	E7	4E	58	2B
32	32	2511	9CF	E7	4F	58	21
32	33	2512	9D0	E7	50	58	26
32	34	2513	9D1	E7	D1	58	4A
32	35	2514	9D2	E7	D2	58	4B
32	36	2515	9D3	E7	D3	58	4C
32	37	2516	9D4	E7	D4	58	4D
32	38	2517	9D5	E7	D5	58	4E
32	39	2518	9D6	E7	D6	58	4F
32	40	2519	9D7	E7	D7	58	50
32	41	2520	9D8	E7	D8	58	51
32	42	2521	9D9	E7	D9	58	52
32	43	2522	9DA	E7	5A	58	5D
32	44	2523	9DB	E7	5B	58	24
32	45	2524	9DC	E7	5C	58	2A
32	46	2525	9DD	E7	5D	58	29
32	47	2526	9DE	E7	5E	58	3B
32	48	2527	9DF	E7	5F	58	5E
32	49	2528	9E0	E7	60	58	2D
32	50	2529	9E1	E7	61	58	2F
32	51	2530	9E2	E7	E2	58	53
32	52	2531	9E3	E7	E3	58	54
32	53	2532	9E4	E7	E4	58	55
32	54	2533	9E5	E7	E5	58	56
32	55	2534	9E6	E7	E6	58	57
32	56	2535	9E7	E7	E7	58	58
32	57	2536	9E8	E7	E8	58	59
32	58	2537	9E9	E7	E9	58	5A
32	59	2538	9EA	E7	6A	58	7C
32	60	2539	9EB	E7	6B	58	2C
32	61	2540	9EC	E7	6C	58	25
32	62	2541	9ED	E7	6D	58	5F

80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		ASCII	
32	63	2542	9EE	E7	6E	58	3E
32	64	2543	9EF	E7	6F	58	3F
32	65	2544	9F0	E7	F0	58	30
32	66	2545	9F1	E7	F1	58	31
32	67	2546	9F2	E7	F2	58	32
32	68	2547	9F3	E7	F3	58	33
32	69	2548	9F4	E7	F4	58	34
32	70	2549	9F5	E7	F5	58	35
32	71	2550	9F6	E7	F6	58	36
32	72	2551	9F7	E7	F7	58	37
32	73	2552	9F8	E7	F8	58	38
32	74	2553	9F9	E7	F9	58	39
32	75	2554	9FA	E7	7A	58	3A
32	76	2555	9FB	E7	7B	58	23
32	77	2556	9FC	E7	7C	58	40
32	78	2557	9FD	E7	7D	58	27
32	79	2558	9FE	E7	7E	58	3D
32	80	2559	9FF	E7	7F	58	22
33	01	2560	A00	E8	40	59	20
33	02	2561	A01	E8	C1	59	41
33	03	2562	A02	E8	C2	59	42
33	04	2563	A03	E8	C3	59	43
33	05	2564	A04	E8	C4	59	44
33	06	2565	A05	E8	C5	59	45
33	07	2566	A06	E8	C6	59	46
33	08	2567	A07	E8	C7	59	47
33	09	2568	A08	E8	C8	59	48
33	10	2569	A09	E8	C9	59	49
33	11	2570	A0A	E8	4A	59	5B
33	12	2571	A0B	E8	4B	59	2E
33	13	2572	A0C	E8	4C	59	3C
33	14	2573	A0D	E8	4D	59	28
33	15	2574	A0E	E8	4E	59	2B
33	16	2575	A0F	E8	4F	59	21
33	17	2576	A10	E8	50	59	26
33	18	2577	A11	E8	D1	59	4A
33	19	2578	A12	E8	D2	59	4B
33	20	2579	A13	E8	D3	59	4C
33	21	2580	A14	E8	D4	59	4D
33	22	2581	A15	E8	D5	59	4E
33	23	2582	A16	E8	D6	59	4F
33	24	2583	A17	E8	D7	59	50
33	25	2584	A18	E8	D8	59	51
33	26	2585	A19	E8	D9	59	52
33	27	2586	A1A	E8	5A	59	5D
33	28	2587	A1B	E8	5B	59	24
33	29	2588	A1C	E8	5C	59	2A
33	30	2589	A1D	E8	5D	59	29
33	31	2590	A1E	E8	5E	59	3B
33	32	2591	A1F	E8	5F	59	5E
33	33	2592	A20	E8	60	59	2D
33	34	2593	A21	E8	61	59	2F
33	35	2594	A22	E8	E2	59	53
33	36	2595	A23	E8	E3	59	54
33	37	2596	A24	E8	E4	59	55
33	38	2597	A25	E8	E5	59	56
33	39	2598	A26	E8	E6	59	57
33	40	2599	A27	E8	E7	59	58
33	41	2600	A28	E8	E8	59	59
33	42	2601	A29	E8	E9	59	5A
33	43	2602	A2A	E8	6A	59	7C
33	44	2603	A2B	E8	6B	59	2C

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
33	45	2604	A2C	E8	6C	59	25
33	46	2605	A2D	E8	6D	59	5F
33	47	2606	A2E	E8	6E	59	3E
33	48	2607	A2F	E8	6F	59	3F
33	49	2608	A30	E8	F0	59	30
33	50	2609	A31	E8	F1	59	31
33	51	2610	A32	E8	F2	59	32
33	52	2611	A33	E8	F3	59	33
33	53	2612	A34	E8	F4	59	34
33	54	2613	A35	E8	F5	59	35
33	55	2614	A36	E8	F6	59	36
33	56	2615	A37	E8	F7	59	37
33	57	2616	A38	E8	F8	59	38
33	58	2617	A39	E8	F9	59	39
33	59	2618	A3A	E8	7A	59	3A
33	60	2619	A3B	E8	7B	59	23
33	61	2620	A3C	E8	7C	59	40
33	62	2621	A3D	E8	7D	59	27
33	63	2622	A3E	E8	7E	59	3D
33	64	2623	A3F	E8	7F	59	22
33	65	2624	A40	E9	40	5A	20
33	66	2625	A41	E9	C1	5A	41
33	67	2626	A42	E9	C2	5A	42
33	68	2627	A43	E9	C3	5A	43
33	69	2628	A44	E9	C4	5A	44
33	70	2629	A45	E9	C5	5A	45
33	71	2630	A46	E9	C6	5A	46
33	72	2631	A47	E9	C7	5A	47
33	73	2632	A48	E9	C8	5A	48
33	74	2633	A49	E9	C9	5A	49
33	75	2634	A4A	E9	4A	5A	5B
33	76	2635	A4B	E9	4B	5A	2E
33	77	2636	A4C	E9	4C	5A	3C
33	78	2637	A4D	E9	4D	5A	28
33	79	2638	A4E	E9	4E	5A	2B
33	80	2639	A4F	E9	4F	5A	21
34	01	2640	A50	E9	50	5A	26
34	02	2641	A51	F9	D1	5A	4A
34	03	2642	A52	E9	D2	5A	4B
34	04	2643	A53	E9	D3	5A	4C
34	05	2644	A54	E9	D4	5A	4D
34	06	2645	A55	E9	D5	5A	4E
34	07	2646	A56	E9	D6	5A	4F
34	08	2647	A57	E9	D7	5A	50
34	09	2648	A58	E9	D8	5A	51
34	10	2649	A59	E9	D9	5A	52
34	11	2650	A5A	E9	5A	5A	5D
34	12	2651	A5B	E9	5B	5A	24
34	13	2652	A5C	E9	5C	5A	2A
34	14	2653	A5D	E9	5D	5A	29
34	15	2654	A5E	E9	5E	5A	3B
34	16	2655	A5F	E9	5F	5A	5E
34	17	2656	A60	E9	60	5A	2D
34	18	2657	A61	E9	61	5A	2F
34	19	2658	A62	E9	E2	5A	53
34	20	2659	A63	E9	E3	5A	54
34	21	2660	A64	E9	E4	5A	55
34	22	2661	A65	E9	E5	5A	56
34	23	2662	A66	E9	E6	5A	57
34	24	2663	A67	E9	E7	5A	58
34	25	2664	A68	E9	E8	5A	59
34	26	2665	A69	E9	E9	5A	5A

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
34	27	2666	A6A	E9	6A	5A	7C
34	28	2667	A6B	E9	6B	5A	2C
34	29	2668	A6C	E9	6C	5A	25
34	30	2669	A6D	E9	6D	5A	5F
34	31	2670	A6E	E9	6E	5A	3E
34	32	2671	A6F	E9	6F	5A	3F
34	33	2672	A70	E9	F0	5A	30
34	34	2673	A71	E9	F1	5A	31
34	35	2674	A72	E9	F2	5A	32
34	36	2675	A73	E9	F3	5A	33
34	37	2676	A74	E9	F4	5A	34
34	38	2677	A75	E9	F5	5A	35
34	39	2678	A76	E9	F6	5A	36
34	40	2679	A77	E9	F7	5A	37
34	41	2680	A78	E9	F8	5A	38
34	42	2681	A79	E9	F9	5A	39
34	43	2682	A7A	E9	7A	5A	3A
34	44	2683	A7B	E9	7B	5A	23
34	45	2684	A7C	E9	7C	5A	40
34	46	2685	A7D	E9	7D	5A	27
34	47	2686	A7E	E9	7E	5A	3D
34	48	2687	A7F	E9	7F	5A	22
34	49	2688	A80	6A	40	7C	20
34	50	2689	A81	6A	C1	7C	41
34	51	2690	A82	6A	C2	7C	42
34	52	2691	A83	6A	C3	7C	43
34	53	2692	A84	6A	C4	7C	44
34	54	2693	A85	6A	C5	7C	45
34	55	2694	A86	6A	C6	7C	46
34	56	2695	A87	6A	C7	7C	47
34	57	2696	A88	6A	C8	7C	48
34	58	2697	A89	6A	C9	7C	49
34	59	2698	A8A	6A	4A	7C	5B
34	60	2699	A8B	6A	4B	7C	2E
34	61	2700	A8C	6A	4C	7C	3C
34	62	2701	A8D	6A	4D	7C	28
34	63	2702	A8E	6A	4E	7C	2B
34	64	2703	A8F	6A	4F	7C	21
34	65	2704	A90	6A	50	7C	26
34	66	2705	A91	6A	D1	7C	4A
34	67	2706	A92	6A	D2	7C	4B
34	68	2707	A93	6A	D3	7C	4C
34	69	2708	A94	6A	D4	7C	4D
34	70	2709	A95	6A	D5	7C	4E
34	71	2710	A96	6A	D6	7C	4F
34	72	2711	A97	6A	D7	7C	50
34	73	2712	A98	6A	D8	7C	51
34	74	2713	A99	6A	D9	7C	52
34	75	2714	A9A	6A	5A	7C	5D
34	76	2715	A9B	6A	5B	7C	24
34	77	2716	A9C	6A	5C	7C	2A
34	78	2717	A9D	6A	5D	7C	29
34	79	2718	A9E	6A	5E	7C	3B
34	80	2719	A9F	6A	5F	7C	5E
35	01	2720	AA0	6A	60	7C	2D
35	02	2721	AA1	6A	61	7C	2F
35	03	2722	AA2	6A	E2	7C	53
35	04	2723	AA3	6A	E3	7C	54
35	05	2724	AA4	6A	E4	7C	55
35	06	2725	AA5	6A	E5	7C	56
35	07	2726	AA6	6A	E6	7C	57
35	08	2727	AA7	6A	E7	7C	58

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
35	09	2728	AA8	6A	E8	7C	59
35	10	2729	AA9	6A	E9	7C	5A
35	11	2730	AAA	6A	6A	7C	7C
35	12	2731	AAB	6A	6B	7C	2C
35	13	2732	AAC	6A	6C	7C	25
35	14	2733	AAD	6A	6D	7C	5F
35	15	2734	AAE	6A	6E	7C	3E
35	16	2735	AAF	6A	6F	7C	3F
35	17	2736	AB0	6A	F0	7C	30
35	18	2737	AB1	6A	F1	7C	31
35	19	2738	AB2	6A	F2	7C	32
35	20	2739	AB3	6A	F3	7C	33
35	21	2740	AB4	6A	F4	7C	34
35	22	2741	AB5	6A	F5	7C	35
35	23	2742	AB6	6A	F6	7C	36
35	24	2743	AB7	6A	F7	7C	37
35	25	2744	AB8	6A	F8	7C	38
35	26	2745	AB9	6A	F9	7C	39
35	27	2746	ABA	6A	7A	7C	3A
35	28	2747	ABB	6A	7B	7C	23
35	29	2748	ABC	6A	7C	7C	40
35	30	2749	ABD	6A	7D	7C	27
35	31	2750	ABE	6A	7E	7C	3D
35	32	2751	ABF	6A	7F	7C	22
35	33	2752	AC0	6B	40	2C	20
35	34	2753	AC1	6B	C1	2C	41
35	35	2754	AC2	6B	C2	2C	42
35	36	2755	AC3	6B	C3	2C	43
35	37	2756	AC4	6B	C4	2C	44
35	38	2757	AC5	6B	C5	2C	45
35	39	2758	AC6	6B	C6	2C	46
35	40	2759	AC7	6B	C7	2C	47
35	41	2760	AC8	6B	C8	2C	48
35	42	2761	AC9	6B	C9	2C	49
35	43	2762	ACA	6B	4A	2C	5B
35	44	2763	ACB	6B	4B	2C	2E
35	45	2764	ACC	6B	4C	2C	3C
35	46	2765	ACD	6B	4D	2C	28
35	47	2766	ACE	6B	4E	2C	2B
35	48	2767	ACF	6B	4F	2C	21
35	49	2768	AD0	6B	50	2C	26
35	50	2769	AD1	6B	D1	2C	4A
35	51	2770	AD2	6B	D2	2C	4B
35	52	2771	AD3	6B	D3	2C	4C
35	53	2772	AD4	6B	D4	2C	4D
35	54	2773	AD5	6B	D5	2C	4E
35	55	2774	AD6	6B	D6	2C	4F
35	56	2775	AD7	6B	D7	2C	50
35	57	2776	AD8	6B	D8	2C	51
35	58	2777	AD9	6B	D9	2C	52
35	59	2778	ADA	6B	5A	2C	5D
35	60	2779	ADB	6B	5B	2C	24
35	61	2780	ADC	6B	5C	2C	2A
35	62	2781	ADD	6B	5D	2C	29
35	63	2782	ADE	6B	5E	2C	3B
35	64	2783	ADF	6B	5F	2C	5E
35	65	2784	AE0	6B	60	2C	2D
35	66	2785	AE1	6B	61	2C	2F
35	67	2786	AE2	6B	E2	2C	53
35	68	2787	AE3	6B	E3	2C	54
35	69	2788	AE4	6B	E4	2C	55
35	70	2789	AE5	6B	E5	2C	56

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
35	71	2790	AE6	6B	E6	2C	57
35	72	2791	AE7	6B	E7	2C	58
35	73	2792	AE8	6B	E8	2C	59
35	74	2793	AE9	6B	E9	2C	5A
35	75	2794	AEA	6B	6A	2C	7C
35	76	2795	AEB	6B	6B	2C	2C
35	77	2796	AEC	6B	6C	2C	25
35	78	2797	AED	6B	6D	2C	5F
35	79	2798	AEE	6B	6E	2C	3E
35	80	2799	AEF	6B	6F	2C	3F
36	01	2800	AF0	6B	F0	2C	30
36	02	2801	AF1	6B	F1	2C	31
36	03	2802	AF2	6B	F2	2C	32
36	04	2803	AF3	6B	F3	2C	33
36	05	2804	AF4	6B	F4	2C	34
36	06	2805	AF5	6B	F5	2C	35
36	07	2806	AF6	6B	F6	2C	36
36	08	2807	AF7	6B	F7	2C	37
36	09	2808	AF8	6B	F8	2C	38
36	10	2809	AF9	6B	F9	2C	39
36	11	2810	AFA	6B	7A	2C	3A
36	12	2811	AFB	6B	7B	2C	23
36	13	2812	AFC	6B	7C	2C	40
36	14	2813	AFD	6B	7D	2C	27
36	15	2814	AFE	6B	7E	2C	3D
36	16	2815	AFF	6B	7F	2C	22
36	17	2816	B00	6C	40	25	20
36	18	2817	B01	6C	C1	25	41
36	19	2818	B02	6C	C2	25	42
36	20	2819	B03	6C	C3	25	43
36	21	2820	B04	6C	C4	25	44
36	22	2821	B05	6C	C5	25	45
36	23	2822	B06	6C	C6	25	46
36	24	2823	B07	6C	C7	25	47
36	25	2824	B08	6C	C8	25	48
36	26	2825	B09	6C	C9	25	49
36	27	2826	B0A	6C	4A	25	5B
36	28	2827	B0B	6C	4B	25	2E
36	29	2828	B0C	6C	4C	25	3C
36	30	2829	B0D	6C	4D	25	28
36	31	2830	B0E	6C	4E	25	2B
36	32	2831	B0F	6C	4F	25	21
36	33	2832	B10	6C	50	25	26
36	34	2833	B11	6C	D1	25	4A
36	35	2834	B12	6C	D2	25	4B
36	36	2835	B13	6C	D3	25	4C
36	37	2836	B14	6C	D4	25	4D
36	38	2837	B15	6C	D5	25	4E
36	39	2838	B16	6C	D6	25	4F
36	40	2839	B17	6C	D7	25	50
36	41	2840	B18	6C	D8	25	51
36	42	2841	B19	6C	D9	25	52
36	43	2842	B1A	6C	5A	25	5D
36	44	2843	B1B	6C	5B	25	24
36	45	2844	B1C	6C	5C	25	2A
36	46	2845	B1D	6C	5D	25	29
36	47	2846	B1E	6C	5E	25	3B
36	48	2847	B1F	6C	5F	25	5E
36	49	2848	B20	6C	60	25	2D
36	50	2849	B21	6C	61	25	2F
36	51	2850	B22	6C	E2	25	53

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
36	52	2851	B23	6C	E3	25	54
36	53	2852	B24	6C	E4	25	55
36	54	2853	B25	6C	E5	25	56
36	55	2854	B26	6C	E6	25	57
36	56	2855	B27	6C	E7	25	58
36	57	2856	B28	6C	E8	25	59
36	58	2857	B29	6C	E9	25	5A
36	59	2858	B2A	6C	6A	25	7C
36	60	2859	B2B	6C	6B	25	2C
36	61	2860	B2C	6C	6C	25	25
36	62	2861	B2D	6C	6D	25	5F
36	63	2862	B2E	6C	6E	25	3E
36	64	2863	B2F	6C	6F	25	3F
36	65	2864	B30	6C	F0	25	30
36	66	2865	B31	6C	F1	25	31
36	67	2866	B32	6C	F2	25	32
36	68	2867	B33	6C	F3	25	33
36	69	2868	B34	6C	F4	25	34
36	70	2869	B35	6C	F5	25	35
36	71	2870	B36	6C	F6	25	36
36	72	2871	B37	6C	F7	25	37
36	73	2872	B38	6C	F8	25	38
36	74	2873	B39	6C	F9	25	39
36	75	2874	B3A	6C	7A	25	3A
36	76	2875	B3B	6C	7B	25	23
36	77	2876	B3C	6C	7C	25	40
36	78	2877	B3D	6C	7D	25	27
36	79	2878	B3E	6C	7E	25	3D
36	80	2879	B3F	6C	7F	25	22
37	01	2880	B40	6D	40	5F	20
37	02	2881	B41	6D	C1	5F	41
37	03	2882	B42	6D	C2	5F	42
37	04	2883	B43	6D	C3	5F	43
37	05	2884	B44	6D	C4	5F	44
37	06	2885	B45	6D	C5	5F	45
37	07	2886	B46	6D	C6	5F	46
37	08	2887	B47	6D	C7	5F	47
37	09	2888	B48	6D	C8	5F	48
37	10	2889	B49	6D	C9	5F	49
37	11	2890	B4A	6D	4A	5F	5B
37	12	2891	B4B	6D	4B	5F	2E
37	13	2892	B4C	6D	4C	5F	3C
37	14	2893	B4D	6D	4D	5F	28
37	15	2894	B4E	6D	4E	5F	2B
37	16	2895	B4F	6D	4F	5F	21
37	17	2896	B50	6D	50	5F	26
37	18	2897	B51	6D	D1	5F	4A
37	19	2898	B52	6D	D2	5F	4B
37	20	2899	B53	6D	D3	5F	4C
37	21	2900	B54	6D	D4	5F	4D
37	22	2901	B55	6D	D5	5F	4E
37	23	2902	B56	6D	D6	5F	4F
37	24	2903	B57	6D	D7	5F	50
37	25	2904	B58	6D	D8	5F	51
37	26	2905	B59	6D	D9	5F	52
37	27	2906	B5A	6D	5A	5F	5D
37	28	2907	B5B	6D	5B	5F	24
37	29	2908	B5C	6D	5C	5F	2A
37	30	2909	B5D	6D	5D	5F	29
37	31	2910	B5E	6D	5E	5F	3B
37	32	2911	B5F	6D	5F	5F	5E
37	33	2912	B60	6D	60	5F	2D

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
37	34	2913	B61	6D	61	5F	2F
37	35	2914	B62	6D	E2	5F	53
37	36	2915	B63	6D	E3	5F	54
37	37	2916	B64	6D	E4	5F	55
37	38	2917	B65	6D	E5	5F	56
37	39	2918	B66	6D	E6	5F	57
37	40	2919	B67	6D	E7	5F	58
37	41	2920	B68	6D	E8	5F	59
37	42	2921	B69	6D	E9	5F	5A
37	43	2922	B6A	6D	6A	5F	7C
37	44	2923	B6B	6D	6B	5F	2C
37	45	2924	B6C	6D	6C	5F	25
37	46	2925	B6D	6D	6D	5F	5F
37	47	2926	B6E	6D	6E	5F	3E
37	48	2927	B6F	6D	6F	5F	3F
37	49	2928	B70	6D	F0	5F	30
37	50	2929	B71	6D	F1	5F	31
37	51	2930	B72	6D	F2	5F	32
37	52	2931	B73	6D	F3	5F	33
37	53	2932	B74	6D	F4	5F	34
37	54	2933	B75	6D	F5	5F	35
37	55	2934	B76	6D	F6	5F	36
37	56	2935	B77	6D	F7	5F	37
37	57	2936	B78	6D	F8	5F	38
37	58	2937	B79	6D	F9	5F	39
37	59	2938	B7A	6D	7A	5F	3A
37	60	2939	B7B	6D	7B	5F	23
37	61	2940	B7C	6D	7C	5F	40
37	62	2941	B7D	6D	7D	5F	27
37	63	2942	B7E	6D	7E	5F	3D
37	64	2943	B7F	6D	7F	5F	22
37	65	2944	B80	6E	40	3E	20
37	66	2945	B81	6E	C1	3E	41
37	67	2946	B82	6E	C2	3E	42
37	68	2947	B83	6E	C3	3E	43
37	69	2948	B84	6E	C4	3E	44
37	70	2949	B85	6E	C5	3E	45
37	71	2950	B86	6E	C6	3E	46
37	72	2951	B87	6E	C7	3E	47
37	73	2952	B88	6E	C8	3E	48
37	74	2953	B89	6E	C9	3E	49
37	75	2954	B8A	6E	4A	3E	5B
37	76	2955	B8B	6E	4B	3E	2E
37	77	2956	B8C	6E	4C	3E	3C
37	78	2957	B8D	6E	4D	3E	28
37	79	2958	B8E	6E	4E	3E	2B
37	80	2959	B8F	6E	4F	3E	21
38	01	2960	B90	6E	50	3E	26
38	02	2961	B91	6E	D1	3E	4A
38	03	2962	B92	6E	D2	3E	4B
38	04	2963	B93	6E	D3	3E	4C
38	05	2964	B94	6E	D4	3E	4D
38	06	2965	B95	6E	D5	3E	4E
38	07	2966	B96	6E	D6	3E	4F
38	08	2967	B97	6E	D7	3E	50
38	09	2968	B98	6E	D8	3E	51
38	10	2969	B99	6E	D9	3E	52
38	11	2970	B9A	6E	5A	3E	5D
38	12	2971	B9B	6E	5B	3E	24
38	13	2972	B9C	6E	5C	3E	2A
38	14	2973	B9D	6E	5D	3E	29
38	15	2974	B9E	6E	5E	3E	3B

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
38	16	2975	B9F	6E	5F	3E	5E
38	17	2976	BA0	6E	60	3E	2D
38	18	2977	BA1	6E	61	3E	2F
38	19	2978	BA2	6E	E2	3E	53
38	20	2979	BA3	6E	E3	3E	54
38	21	2980	BA4	6E	E4	3E	55
38	22	2981	BA5	6E	E5	3E	56
38	23	2982	BA6	6E	E6	3E	57
38	24	2983	BA7	6E	E7	3E	58
38	25	2984	BA8	6E	E8	3E	59
38	26	2985	BA9	6E	E9	3E	5A
38	27	2986	BAA	6E	6A	3E	7C
38	28	2987	BAB	6E	6B	3E	2C
38	29	2988	BAC	6E	6C	3E	25
38	30	2989	BAD	6E	6D	3E	5F
38	31	2990	BAE	6E	6E	3E	3E
38	32	2991	BAF	6E	6F	3E	3F
38	33	2992	BB0	6E	F0	3E	30
38	34	2993	BB1	6E	F1	3E	31
38	35	2994	BB2	6E	F2	3E	32
38	36	2995	BB3	6E	F3	3E	33
38	37	2996	BB4	6E	F4	3E	34
38	38	2997	BB5	6E	F5	3E	35
38	39	2998	BB6	6E	F6	3E	36
38	40	2999	BB7	6E	F7	3E	37
38	41	3000	BB8	6E	F8	3E	38
38	42	3001	BB9	6E	F9	3E	39
38	43	3002	BBA	6E	7A	3E	3A
38	44	3003	BBB	6E	7B	3E	23
38	45	3004	BBC	6E	7C	3E	40
38	46	3005	BBD	6E	7D	3E	27
38	47	3006	BBE	6E	7E	3E	3D
38	48	3007	BBF	6E	7F	3E	22
38	49	3008	BC0	6F	40	3F	20
38	50	3009	BC1	6F	C1	3F	41
38	51	3010	BC2	6F	C2	3F	42
38	52	3011	BC3	6F	C3	3F	43
38	53	3012	BC4	6F	C4	3F	44
38	54	3013	BC5	6F	C5	3F	45
38	55	3014	BC6	6F	C6	3F	46
38	56	3015	BC7	6F	C7	3F	47
38	57	3016	BC8	6F	C8	3F	48
38	58	3017	BC9	6F	C9	3F	49
38	59	3018	BCA	6F	4A	3F	5B
38	60	3019	BCB	6F	4B	3F	2E
38	61	3020	BCC	6F	4C	3F	3C
38	62	3021	BCD	6F	4D	3F	28
38	63	3022	BCE	6F	4E	3F	2B
38	64	3023	BCF	6F	4F	3F	21
38	65	3024	BD0	6F	50	3F	26
38	66	3025	BD1	6F	D1	3F	4A
38	67	3026	BD2	6F	D2	3F	4B
38	68	3027	BD3	6F	D3	3F	4C
38	69	3028	BD4	6F	D4	3F	4D
38	70	3029	BD5	6F	D5	3F	4E
38	71	3030	BD6	6F	D6	3F	4F
38	72	3031	BD7	6F	D7	3F	50
38	73	3032	BD8	6F	D8	3F	51
38	74	3033	BD9	6F	D9	3F	52
38	75	3034	BDA	6F	5A	3F	5D
38	76	3035	BDB	6F	5B	3F	24
38	77	3036	BDC	6F	5C	3F	2A

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
38	78	3037	BDD	6F	5D	3F	29
38	79	3038	BDE	6F	5E	3F	3B
38	80	3039	BDF	6F	5F	3F	5E
39	01	3040	BE0	6F	60	3F	2D
39	02	3041	BE1	6F	61	3F	2F
39	03	3042	BE2	6F	E2	3F	53
39	04	3043	BE3	6F	E3	3F	54
39	05	3044	BE4	6F	E4	3F	55
39	06	3045	BE5	6F	E5	3F	56
39	07	3046	BE6	6F	E6	3F	57
39	08	3047	BE7	6F	E7	3F	58
39	09	3048	BE8	6F	E8	3F	59
39	10	3049	BE9	6F	E9	3F	5A
39	11	3050	BEA	6F	6A	3F	7C
39	12	3051	BEB	6F	6B	3F	2C
39	13	3052	BEC	6F	6C	3F	25
39	14	3053	BED	6F	6D	3F	5F
39	15	3054	BEE	6F	6E	3F	3E
39	16	3055	BEF	6F	6F	3F	3F
39	17	3056	BF0	6F	F0	3F	30
39	18	3057	BF1	6F	F1	3F	31
39	19	3058	BF2	6F	F2	3F	32
39	20	3059	BF3	6F	F3	3F	33
39	21	3060	BF4	6F	F4	3F	34
39	22	3061	BF5	6F	F5	3F	35
39	23	3062	BF6	6F	F6	3F	36
39	24	3063	BF7	6F	F7	3F	37
39	25	3064	BF8	6F	F8	3F	38
39	26	3065	BF9	6F	F9	3F	39
39	27	3066	BFA	6F	7A	3F	3A
39	28	3067	BFB	6F	7B	3F	23
39	29	3068	BFC	6F	7C	3F	40
39	30	3069	BFD	6F	7D	3F	27
39	31	3070	BFE	6F	7E	3F	3D
39	32	3071	BFF	6F	7F	3F	22
39	33	3072	C00	F0	40	30	20
39	34	3073	C01	F0	C1	30	41
39	35	3074	C02	F0	C2	30	42
39	36	3075	C03	F0	C3	30	43
39	37	3076	C04	F0	C4	30	44
39	38	3077	C05	F0	C5	30	45
39	39	3078	C06	F0	C6	30	46
39	40	3079	C07	F0	C7	30	47
39	41	3080	C08	F0	C8	30	48
39	42	3081	C09	F0	C9	30	49
39	43	3082	C0A	F0	4A	30	5B
39	44	3083	C0B	F0	4B	30	2E
39	45	3084	C0C	F0	4C	30	3C
39	46	3085	C0D	F0	4D	30	28
39	47	3086	C0E	F0	4E	30	2B
39	48	3087	C0F	F0	4F	30	21
39	49	3088	C10	F0	50	30	26
39	50	3089	C11	F0	D1	30	4A
39	51	3090	C12	F0	D2	30	4B
39	52	3091	C13	F0	D3	30	4C
39	53	3092	C14	F0	D4	30	4D
39	54	3093	C15	F0	D5	30	4E
39	55	3094	C16	F0	D6	30	4F
39	56	3095	C17	F0	D7	30	50
39	57	3096	C18	F0	D8	30	51
39	58	3097	C19	F0	D9	30	52
39	59	3098	C1A	F0	5A	30	5D

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
39	60	3099	C1B	F0	5B	30	24
39	61	3100	C1C	F0	5C	30	2A
39	62	3101	C1D	F0	5D	30	29
39	63	3102	C1E	F0	5E	30	3B
39	64	3103	C1F	F0	5F	30	5E
39	65	3104	C20	F0	60	30	2D
39	66	3105	C21	F0	61	30	2F
39	67	3106	C22	F0	E2	30	53
39	68	3107	C23	F0	E3	30	54
39	69	3108	C24	F0	E4	30	55
39	70	3109	C25	F0	E5	30	56
39	71	3110	C26	F0	E6	30	57
39	72	3111	C27	F0	E7	30	58
39	73	3112	C28	F0	E8	30	59
39	74	3113	C29	F0	E9	30	5A
39	75	3114	C2A	F0	6A	30	7C
39	76	3115	C2B	F0	6B	30	2C
39	77	3116	C2C	F0	6C	30	25
39	78	3117	C2D	F0	6D	30	5F
39	79	3118	C2E	F0	6E	30	3E
39	80	3119	C2F	F0	6F	30	3F
40	01	3120	C30	F0	F0	30	30
40	02	3121	C31	F0	F1	30	31
40	03	3122	C32	F0	F2	30	32
40	04	3123	C33	F0	F3	30	33
40	05	3124	C34	F0	F4	30	34
40	06	3125	C35	F0	F5	30	35
40	07	3126	C36	F0	F6	30	36
40	08	3127	C37	F0	F7	30	37
40	09	3128	C38	F0	F8	30	38
40	10	3129	C39	F0	F9	30	39
40	11	3130	C3A	F0	7A	30	3A
40	12	3131	C3B	F0	7B	30	23
40	13	3132	C3C	F0	7C	30	40
40	14	3133	C3D	F0	7D	30	27
40	15	3134	C3E	F0	7E	30	3D
40	16	3135	C3F	F0	7F	30	22
40	17	3136	C40	F1	40	31	20
40	18	3137	C41	F1	C1	31	41
40	19	3138	C42	F1	C2	31	42
40	20	3139	C43	F1	C3	31	43
40	21	3140	C44	F1	C4	31	44
40	22	3141	C45	F1	C5	31	45
40	23	3142	C46	F1	C6	31	46
40	24	3143	C47	F1	C7	31	47
40	25	3144	C48	F1	C8	31	48
40	26	3145	C49	F1	C9	31	49
40	27	3146	C4A	F1	4A	31	5B
40	28	3147	C4B	F1	4B	31	2E
40	29	3148	C4C	F1	4C	31	3C
40	30	3149	C4D	F1	4D	31	28
40	31	3150	C4E	F1	4E	31	2B
40	32	3151	C4F	F1	4F	31	21
40	33	3152	C50	F1	50	31	26
40	34	3153	C51	F1	D1	31	4A
40	35	3154	C52	F1	D2	31	4B
40	36	3155	C53	F1	D3	31	4C
40	37	3156	C54	F1	D4	31	4D
40	38	3157	C55	F1	D5	31	4E
40	39	3158	C56	F1	D6	31	4F
40	40	3159	C57	F1	D7	31	50
40	41	3160	C58	F1	D8	31	51

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
40	42	3161	C59	F1	D9	31	52
40	43	3162	C5A	F1	5A	31	5D
40	44	3163	C5B	F1	5B	31	24
40	45	3164	C5C	F1	5C	31	2A
40	46	3165	C5D	F1	5D	31	29
40	47	3166	C5E	F1	5E	31	3B
40	48	3167	C5F	F1	5F	31	5E
40	49	3168	C60	F1	60	31	2D
40	50	3169	C61	F1	61	31	2F
40	51	3170	C62	F1	E2	31	53
40	52	3171	C63	F1	E3	31	54
40	53	3172	C64	F1	E4	31	55
40	54	3173	C65	F1	E5	31	56
40	55	3174	C66	F1	E6	31	57
40	56	3175	C67	F1	E7	31	58
40	57	3176	C68	F1	E8	31	59
40	58	3177	C69	F1	E9	31	5A
40	59	3178	C6A	F1	6A	31	7C
40	60	3179	C6B	F1	6B	31	2C
40	61	3180	C6C	F1	6C	31	25
40	62	3181	C6D	F1	6D	31	5F
40	63	3182	C6E	F1	6E	31	3E
40	64	3183	C6F	F1	6F	31	3F
40	65	3184	C70	F1	F0	31	30
40	66	3185	C71	F1	F1	31	31
40	67	3186	C72	F1	F2	31	32
40	68	3187	C73	F1	F3	31	33
40	69	3188	C74	F1	F4	31	34
40	70	3189	C75	F1	F5	31	35
40	71	3190	C76	F1	F6	31	36
40	72	3191	C77	F1	F7	31	37
40	73	3192	C78	F1	F8	31	38
40	74	3193	C79	F1	F9	31	39
40	75	3194	C7A	F1	7A	31	3A
40	76	3195	C7B	F1	7B	31	23
40	77	3196	C7C	F1	7C	31	40
40	78	3197	C7D	F1	7D	31	27
40	79	3198	C7E	F1	7E	31	3D
40	80	3199	C7F	F1	7F	31	22
41	01	3200	C80	F2	40	32	20
41	02	3201	C81	F2	C1	32	41
41	03	3202	C82	F2	C2	32	42
41	04	3203	C83	F2	C3	32	43
41	05	3204	C84	F2	C4	32	44
41	06	3205	C85	F2	C5	32	45
41	07	3206	C86	F2	C6	32	46
41	08	3207	C87	F2	C7	32	47
41	09	3208	C88	F2	C8	32	48
41	10	3209	C89	F2	C9	32	49
41	11	3210	C8A	F2	4A	32	5B
41	12	3211	C8B	F2	4B	32	2E
41	13	3212	C8C	F2	4C	32	3C
41	14	3213	C8D	F2	4D	32	28
41	15	3214	C8E	F2	4E	32	2B
41	16	3215	C8F	F2	4F	32	21
41	17	3216	C90	F2	50	32	26
41	18	3217	C91	F2	D1	32	4A
41	19	3218	C92	F2	D2	32	4B
41	20	3219	C93	F2	D3	32	4C
41	21	3220	C94	F2	D4	32	4D
41	22	3221	C95	F2	D5	32	4E

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
41	23	3222	C96	F2	D6	32	4F
41	24	3223	C97	F2	D7	32	50
41	25	3224	C98	F2	D8	32	51
41	26	3225	C99	F2	D9	32	52
41	27	3226	C9A	F2	5A	32	5D
41	28	3227	C9B	F2	5B	32	24
41	29	3228	C9C	F2	5C	32	2A
41	30	3229	C9D	F2	5D	32	29
41	31	3230	C9E	F2	5E	32	3B
41	32	3231	C9F	F2	5F	32	5E
41	33	3232	CA0	F2	60	32	2D
41	34	3233	CA1	F2	61	32	2F
41	35	3234	CA2	F2	E2	32	53
41	36	3235	CA3	F2	E3	32	54
41	37	3236	CA4	F2	E4	32	55
41	38	3237	CA5	F2	E5	32	56
41	39	3238	CA6	F2	E6	32	57
41	40	3239	CA7	F2	E7	32	58
41	41	3240	CA8	F2	E8	32	59
41	42	3241	CA9	F2	E9	32	5A
41	43	3242	CAA	F2	6A	32	7C
41	44	3243	CAB	F2	6B	32	2C
41	45	3244	CAC	F2	6C	32	25
41	46	3245	CAD	F2	6D	32	5F
41	47	3246	CAE	F2	6E	32	3E
41	48	3247	CAF	F2	6F	32	3F
41	49	3248	CB0	F2	F0	32	30
41	50	3249	CB1	F2	F1	32	31
41	51	3250	CB2	F2	F2	32	32
41	52	3251	CB3	F2	F3	32	33
41	53	3252	CB4	F2	F4	32	34
41	54	3253	CB5	F2	F5	32	35
41	55	3254	CB6	F2	F6	32	36
41	56	3255	CB7	F2	F7	32	37
41	57	3256	CB8	F2	F8	32	38
41	58	3257	CB9	F2	F9	32	39
41	59	3258	CBA	F2	7A	32	3A
41	60	3259	CBB	F2	7B	32	23
41	61	3260	CBC	F2	7C	32	40
41	62	3261	CBD	F2	7D	32	27
41	63	3262	CBE	F2	7E	32	3D
41	64	3263	CBF	F2	7F	32	22
41	65	3264	CC0	F3	40	33	20
41	66	3265	CC1	F3	C1	33	41
41	67	3266	CC2	F3	C2	33	42
41	68	3267	CC3	F3	C3	33	43
41	69	3268	CC4	F3	C4	33	44
41	70	3269	CC5	F3	C5	33	45
41	71	3270	CC6	F3	C6	33	46
41	72	3271	CC7	F3	C7	33	47
41	73	3272	CC8	F3	C8	33	48
41	74	3273	CC9	F3	C9	33	49
41	75	3274	CCA	F3	4A	33	5B
41	76	3275	CCB	F3	4B	33	2E
41	77	3276	CCC	F3	4C	33	3C
41	78	3277	CCD	F3	4D	33	28
41	79	3278	CCE	F3	4E	33	2B
41	80	3279	CCF	F3	4F	33	21
42	01	3280	CD0	F3	50	33	26
42	02	3281	CD1	F3	D1	33	4A
42	03	3282	CD2	F3	D2	33	4B
42	04	3283	CD3	F3	D3	33	4C

80 Col		Position		Buffer Address (Hex)			
R	C	Dec	Hex	EBCDIC		ASCII	
42	05	3284	CD4	F3	D4	33	4D
42	06	3285	CD5	F3	D5	33	4E
42	07	3286	CD6	F3	D6	33	4F
42	08	3287	CD7	F3	D7	33	50
42	09	3288	CD8	F3	D8	33	51
42	10	3289	CD9	F3	D9	33	52
42	11	3290	CDA	F3	5A	33	5D
42	12	3291	CDB	F3	5B	33	24
42	13	3292	CDC	F3	5C	33	2A
42	14	3293	CDD	F3	5D	33	29
42	15	3294	CDE	F3	5E	33	3B
42	16	3295	CDF	F3	5F	33	5E
42	17	3296	CE0	F3	60	33	2D
42	18	3297	CE1	F3	61	33	2F
42	19	3298	CE2	F3	E2	33	53
42	20	3299	CE3	F3	E3	33	54
42	21	3300	CE4	F3	E4	33	55
42	22	3301	CE5	F3	E5	33	56
42	23	3302	CE6	F3	E6	33	57
42	24	3303	CE7	F3	E7	33	58
42	25	3304	CE8	F3	E8	33	59
42	26	3305	CE9	F3	E9	33	5A
42	27	3306	CEA	F3	6A	33	7C
42	28	3307	CEB	F3	6B	33	2C
42	29	3308	CEC	F3	6C	33	25
42	30	3309	CED	F3	6D	33	5F
42	31	3310	CEE	F3	6E	33	3E
42	32	3311	CEF	F3	6F	33	3F
42	33	3312	CF0	F3	F0	33	30
42	34	3313	CF1	F3	F1	33	31
42	35	3314	CF2	F3	F2	33	32
42	36	3315	CF3	F3	F3	33	33
42	37	3316	CF4	F3	F4	33	34
42	38	3317	CF5	F3	F5	33	35
42	39	3318	CF6	F3	F6	33	36
42	40	3319	CF7	F3	F7	33	37
42	41	3320	CF8	F3	F8	33	38
42	42	3321	CF9	F3	F9	33	39
42	43	3322	CFA	F3	7A	33	3A
42	44	3323	CFB	F3	7B	33	23
42	45	3224	CFC	F3	7C	33	40
42	46	3325	CFD	F3	7D	33	27
42	47	3326	CFE	F3	7E	33	3D
42	48	3327	CFF	F3	7F	33	22
42	49	3328	D00	F4	40	34	20
42	50	3329	D01	F4	C1	34	41
42	51	3330	D02	F4	C2	34	42
42	52	3331	D03	F4	C3	34	43
42	53	3332	D04	F4	C4	34	44
42	54	3333	D05	F4	C5	34	45
42	55	3334	D06	F4	C6	34	46
42	56	3335	D07	F4	C7	34	47
42	57	3336	D08	F4	C8	34	48
42	58	3337	D09	F4	C9	34	49
42	59	3338	D0A	F4	4A	34	5B
42	60	3339	D0B	F4	4B	34	2E
42	61	3340	D0C	F4	4C	34	3C
42	62	3341	D0D	F4	4D	34	28
42	63	3342	D0E	F4	4E	34	2B
42	64	3343	D0F	F4	4F	34	21
42	65	3344	D10	F4	50	34	26

80 Col R	C	Position Dec	Hex	Buffer Address (Hex) EBCDIC		ASCII	
42	66	3345	D11	F4	C1	34	4A
42	67	3346	D12	F4	D2	34	4B
42	68	3347	D13	F4	D3	34	4C
42	69	3348	D14	F4	D4	34	4D
42	70	3349	D15	F4	D5	34	4E
42	71	3350	D16	F4	D6	34	4F
42	72	3351	D17	F4	D7	34	50
42	73	3352	D18	F4	D8	34	51
42	74	3353	D19	F4	D9	34	52
42	75	3354	D1A	F4	5A	34	5D
42	76	3355	D1B	F4	5B	34	24
42	77	3356	D1C	F4	5C	34	2A
42	78	3357	D1D	F4	5D	34	29
42	79	3358	D1E	F4	5E	34	3B
42	80	3359	D1F	F4	5F	34	5E
43	01	3360	D20	F4	60	34	2D
43	02	3361	D21	F4	61	34	2F
43	03	3362	D22	F4	E2	34	53
43	04	3363	D23	F4	E3	34	54
43	05	3364	D24	F4	E4	34	55
43	06	3365	D25	F4	E5	34	56
43	07	3366	D26	F4	E6	34	57
43	08	3367	D27	F4	E7	34	58
43	09	3368	D28	F4	E8	34	59
43	10	3369	D29	F4	E9	34	5A
43	11	3370	D2A	F4	6A	34	7C
43	12	3371	D2B	F4	6B	34	2C
43	13	3372	D2C	F4	6C	34	25
43	14	3373	D2D	F4	6D	34	5F
43	15	3374	D2E	F4	6E	34	3E
43	16	3375	D2F	F4	6F	34	3F
43	17	3376	D30	F4	F0	34	30
43	18	3377	D31	F4	F1	34	31
43	19	3378	D32	F4	F2	34	32
43	20	3379	D33	F4	F3	34	33
43	21	3380	D34	F4	F4	34	34
43	22	3381	D35	F4	F5	34	35
43	23	3382	D36	F4	F6	34	36
43	24	3383	D37	F4	F7	34	37
43	25	3384	D38	F4	F8	34	38
43	26	3385	D39	F4	F9	34	39
43	27	3386	D3A	F4	7A	34	3A
43	28	3387	D3B	F4	7B	34	23
43	29	3388	D3C	F4	7C	34	40
43	30	3389	D3D	F4	7D	34	27
43	31	3390	D3E	F4	7E	34	3D
43	32	3391	D3F	F4	7F	34	22
43	33	3392	D40	F5	40	35	20
43	34	3393	D41	F5	C1	35	41
43	35	3394	D42	F5	C2	35	42
43	36	3395	D43	F5	C3	35	43
43	37	3396	D44	F5	C4	35	44
43	38	3397	D45	F5	C5	35	45
43	39	3398	D46	F5	C6	35	46
43	40	3399	D47	F5	C7	35	47
43	41	3400	D48	F5	C8	35	48
43	42	3401	D49	F5	C9	35	49
43	43	3402	D4A	F5	4A	35	5B
43	44	3403	D4B	F5	4B	35	2E
43	45	3404	D4C	F5	4C	35	3C
43	46	3405	D4D	F5	4D	35	28
43	47	3406	D4E	F5	4E	35	2B

80 Col R	80 Col C	Position Dec	Position Hex	Buffer Address (Hex) EBCDIC		ASCII	
43	48	3407	D4F	F5	4F	35	21
43	49	3408	D50	F5	50	35	26
43	50	3409	D51	F5	D1	35	4A
43	51	3410	D52	F5	D2	35	4B
43	52	3411	D53	F5	D3	35	4C
43	53	3412	D54	F5	D4	35	4D
43	54	3413	D55	F5	D5	35	4E
43	55	3414	D56	F5	D6	35	4F
43	56	3415	D57	F5	D7	35	50
43	57	3416	D58	F5	D8	35	51
43	58	3417	D59	F5	D9	35	52
43	59	3418	D5A	F5	5A	35	5D
43	60	3419	D5B	F5	5B	35	24
43	61	3420	D5C	F5	5C	35	2A
43	62	3421	D5D	F5	5D	35	29
43	63	3422	D5E	F5	5E	35	3B
43	64	3423	D5F	F5	5F	35	5E
43	65	3424	D60	F5	60	35	2D
43	66	3425	D61	F5	61	35	2F
43	67	3426	D62	F5	E2	35	53
43	68	3427	D63	F5	E3	35	54
43	69	3428	D64	F5	E4	35	55
43	70	3429	D65	F5	E5	35	56
43	71	3430	D66	F5	E6	35	57
43	72	3431	D67	F5	E7	35	58
43	73	3432	D68	F5	E8	35	59
43	74	3433	D69	F5	E9	35	5A
43	75	3434	D6A	F5	6A	35	7C
43	76	3435	D6B	F5	6B	35	2C
43	77	3436	D6C	F5	6C	35	25
43	78	3437	D6D	F5	6D	35	5F
43	79	3438	D6E	F5	6E	35	3E
43	80	3439	D6F	F5	6F	35	3F
		3440	D70	F5	F0	35	30
		3441	D71	F5	F1	35	31
		3442	D72	F5	F2	35	32
		3443	D73	F5	F3	35	33
		3444	D74	F5	F4	35	34
		3445	D75	F5	F5	35	35
		3446	D76	F5	F6	35	36
		3447	D77	F5	F7	35	37
		3448	D78	F5	F8	35	38
		3449	D79	F5	F9	35	39
		3450	D7A	F5	7A	35	3A
		3451	D7B	F5	7B	35	23
		3452	D7C	F5	7C	35	40
		3453	D7D	F5	7D	35	27
		3454	D7E	F5	7E	35	3D
		3455	D7F	F5	7F	35	22
		3456	D80	F6	40	36	20
		3457	D81	F6	C1	36	41
		3458	D82	F6	C2	36	42
		3459	D83	F6	C3	36	43
		3460	D84	F6	C4	36	44
		3461	D85	F6	C5	36	45
		3462	D86	F6	C6	36	46
		3463	D87	F6	C7	36	47
		3464	D88	F6	C8	36	48
		3465	D89	F6	C9	36	49
		3466	D8A	F6	4A	36	5B
		3467	D8B	F6	4B	36	2E
		3468	D8C	F6	4C	36	3C

Appendix C. Status Indicator Code

This appendix lists the error status indications of the 3276, possible causes of errors, the handling of each error by the 3276, and the recommended recovery technique.

The symbol that appears in the Operator Information Area for each error is shown in parentheses in the "Indicator" column and is described in Appendix A.

For Test Subsystem switch (see Figure A-2) operation, when operated in the external modem configuration, the Test Operate switch of the DCE side of the modem cable connector should be set to Test.

An indication consists of a symbol and a numeric code, as shown in the "Indicator" column of Figure C-1, and is described as follows:

X PROG nn — Program Check.
This symbol is displayed when a programming error is detected in the data received by the control unit.

X -z- nn — Communication Check.
A communication reminder symbol (-z-nn) is displayed when a communication link error is detected; it indicates that data cannot be sent. The reminder is automatically cleared when the error condition is removed. If the operator attempts to communicate with the host when a communication error reminder is displayed, a communication check (X-z-nn) condition occurs and is not cleared until the operator presses the RESET key.

X ⚒ nn — Machine Check.
This symbol is displayed when the problem is located in the display station.

The numeric codes consist of two digits, if the display unit is attached to the 3276. (When the display unit is attached to the 3276, the 6 symbol is displayed in the Readiness location of the Operator Information Area.) These codes and their meaning are subject to change.

Error Code	Indicator	Probable Cause	Effect	Recovery
11 (SDLC)	Sys Chk Light Program Chk: (X PROG 11)	3276 received a negative response from host.	Display error condition at affected display station.	System Check light is turned off when 3276 receives any I-frame, valid PIU, or an SNRM. Press RESET to reset Program Check symbol. Wait for host error recovery if X is indicated. If problem persists and 3276 is in Encrypt/Decrypt session, log off and then log on.
12 (BSC)	Sys Chk Light Program Chk: (X PROG 12)	Invalid command received; host programming problem in write data stream.	Display error indication at affected display station. Set BSC Sense: CR. Send EOT. Go to Control mode.	Receipt of poll or selection with 3276 address resets System Check light. Press RESET to reset Program Check symbol. Call host-support programmer if problem persists.
13 (BSC)	Sys Chk Light Program Chk: (X PROG 13)	Invalid buffer address received or incomplete order sequence in Write, Erase/Write, or Erase/Write Alternate command received.	Display error indication at affected display station. Set BSC Sense: OC. Send EOT. Got to control mode.	
14 (BSC)	Sys Chk Light Program Chk: (X PROG 14)	Invalid Copy command received.		
15 (BSC)	Sys Chk Light Program Chk: (X PROG 15)	Invalid command sequence.		
16 (BSC)	Sys Chk Light Program Chk: (X PROG 16)	Line buffer overflow.	Display error indication at affected display station. Set BSC Sense: OC. Send EOT. Got to control mode.	
20 (BSC)	Sys Chk Light Comm Reminder: (↯Z 20)	3276 has sent a NAK because: ● Block-character-checking error was detected, or ● Three seconds elapsed during a read operation without receiving Syn, ETX, or ETB.	Display error indication at affected display station. Replace display image with image displayed before receive operation began.	Host recovery (Host should retransmit the last transmission). Receipt of poll, selection, or data resets System Check light and Communication Reminder symbol. If switched network, redial; if SNBU is installed, use it.
22 (BSC)	Sys Chk Light Comm Reminder: (↯Z 22)	No SYN characters received for 3 seconds and this occurred 7 times while monitoring selection or polling.	Display error indication at all display stations. Continue to monitor the line.	Verify the operational status of the communication network. Host recovery. Receipt of poll or selection with 3276 address resets System Check light and Communication Reminder symbol.
22 (SDLC)	Sys Chk Light Comm Reminder: (↯Z 22)	No flags received for about 24 to 32 seconds, and the host communication adapter has not been in Sync during this period.	Display error indication at all display stations.	Verify the operational status of the communication network. Host recovery. Receipt of valid frame resets System Check light and Communication Reminder symbol.
23 (BSC)	Sys Chk Light Comm Reminder: (↯Z 23)	Fifteen 3-second timeouts occurred when the host expected the 3276 to send a text block as a response to a read-type command.	Display error indication at all display stations. Go to control mode.	Host recovery. Receipt of poll or selection with 3276 address resets System Check light and Communication Reminder symbol. If problem persists, press Test Subsystem.
24 (BSC)	Sys Chk Light Comm Reminder: (↯Z 24)	Fifteen 3-second timeouts occurred when PAD, SYN, and data were not received after sending ACK or RVI.		

Error Code	Indicator	Probable Cause	Effect	Recovery
25 (SDLC)	Sys Chk Light Comm Reminder: (+Z 25)	Something in the link is preventing establishment or reestablishment of communication.	Display error indication at all display stations.	Verify the operational status of the communication network. Host recovery. System Check light and Communication Reminder symbol are reset when an SNRM or a DISC is received or when write operation is completed. If problem persists, press Test Subsystem.
26 (BSC)	Sys Chk Light Comm Reminder: (+Z 26)	Fifteen continuous ACK0 received, instead of ACK1—or vice versa (Wrong ACK - ENQ exchange). Wrong ACK receptions or 3-second timeouts occurred 15 times in a row	Display error indication at affected display station. Go to control mode.	Host recovery Receipt of poll or selection with 3276 address resets System Check light and Communication Reminder symbol. If problem persists, call host operator.
27 (BSC)	Sys Chk Light Comm Reminder: (+Z 27)	Fifteen continuous NAKs received for transmitted/retransmitted text.		
29 (SDLC)	Sys Chk Light Comm Reminder: (+Z 29)	Command reject caused by: a. Detection of an NR sequence error, or b. Receipt of a command that has no data field defined, or c. Receipt of an invalid command.	Display error indication at all display stations.	Host recovery. Receipt of valid SNRM or DISC command from host resets System Check light and Communication Reminder symbol. If problem persists, call host-support programmer.
31 (SDLC)	Sys Chk Light Comm Reminder: (+Z 31)	Loop Adapter did not receive RLSD for more than 4 seconds.	Display error indication at all display stations.	Host recovery. Call host operator.
33 (BSC and SDLC)	Sys Chk Light Comm Reminder: (+Z 33)	Data Set Ready (DSR) signal from modem has dropped.	Display error indication at all display stations. BSC: Go to control mode. SDLC: Go to line-monitor mode.	Check modem. Host recovery. BSC: Receipt of poll or selection resets System Check light and Communication Reminder symbol. SDLC: Receipt of valid SDLC frame resets System Check light and Communication Reminder symbol. If problem persists, press Test Subsystem.
34 (BSC and SDLC)	Sys Chk Light Comm Reminder: (+Z 34)	Write timeout caused by: a. Modem clocking missing, or b. Dropping of CTS.		
35 (BSC)	Sys Chk Light Comm Reminder: (+Z 35)	Fifteen 3-second timeouts occurred with no response received for the transmitted text to the host. 3276 component or host facility problem, or host is busy.	Display error indication at all display stations. Continue operation.	Receipt of poll or selection with 3276 address resets System Check light and Communication Reminder symbol. If problem persists, call host operator.
36 (BSC)	Sys Chk Light Comm Reminder: (+Z 36)	Fifteen continuous ACK0s received instead of ACK1s, or vice versa.	Display error indication at all display stations. Continue operation.	Receipt of poll or selection with 3276 address resets System Check light and Communication Reminder symbol. If problem persists, call host operator.

Error Code	Indicator	Probable Cause	Effect	Recovery
41 (Keyboard)	Mach Chk: (X ⚡ 41)	Internal malfunction.	Display error indication at affected display station.	Press RESET. Retry operation.
42 (Keyboard)	Retry: (X?+42)	Keystroke lost because of temporary system overload. Keying was attempted when device was busy or not functioning. Conflicting operations were attempted simultaneously, for example, the CLEAR key was pressed during selector-light-pen operation.		(If ALT or Alpha was struck just prior to error, restrike to remove keyboard from ALT or Alpha shift status before pressing RESET.) Press RESET, and retry the operation.
43 (Feature)	Mach Chk: (X ⚡ 43)	Internal malfunction.		Press RESET. Retry operation.
44 (Feature)	Mach Chk: (X ⚡ 44)			
45 (Feature)	Retry: (X?+45)	No response/receive parity error from MSR or MHS read command.		
55 (Feature)	Mach Chk: (X ⚡ 55)	Battery in the 3279 is discharged, or internal malfunction.	Display error indication at affected 3279 display station.	Set Normal/Test switch from Normal to Test, then back to Normal. If no indication displayed, check the battery, and replace it if necessary. If indication displayed, call service representative.
56 (Feature)	Mach Chk: (X ⚡ 56)	Internal malfunction.	Display error indication at affected 3279 display station.	Press RESET. Retry operation. If operation cannot be continued, call service representative.
59	Mach Chk Light Mach Chk: (X ⚡ 59)	Bad parity in master key of Encrypt/Decrypt feature.	Display error indication at affected display station. Disable Encrypt/Decrypt function if RESET is pressed.	Check the battery, and replace it if necessary.
60 (Feature)	Mach Chk: (X ⚡ 60)	Internal malfunction.	Display error indication at affected display station. Disable MSR/MHS function.	Press RESET. Retry operation.
61 (Feature)	Mach Chk: (X ⚡ 61)		Disable Selector Light-Pen feature. Display error indication at affected display station.	Set Normal/Test switch from Normal to Test, then back to Normal.
63	Mach Chk Light Mach Chk: (X ⚡ 63)	Error in Encrypt/Decrypt function.	Display error indication at affected display station. Disable Encrypt/Decrypt function if RESET is pressed.	Press and release Test Subsystem.
65 (Feature)	Mach Chk: (X ⚡ 65)	Internal malfunction.	Display error indication at affected display station. Disable display. Set sense: BSC : DC/US SNA: 081C Issue hardware poll and accept only POR from station.	At 3276/3278, set Normal/Test from Normal to Test and back again (or switch power off, then on). At 3230/3268, press Test switch, or switch printer power off, then on. At 3287: 1. Press and hold Test switch. 2. Press and release Reset switch. 3. Release Test switch. Or switch 3287 power off, then on. At 5210, press START. Or switch 5210 power off, wait 10 seconds, and switch 5210 power on.
66 (Feature)	Mach Chk: (X ⚡ 66)	Internal malfunction.	Display error indication at affected display station. Disable display. Set sense: BSC : DC/US - IR SNA: 082B - 081C Issue hardware poll and accept only POR from station.	Set Normal/Test switch from Normal to Test, then back to Normal (or switch power off, then on).

Error Code	Indicator	Probable Cause	Effect	Recovery
69 (Display or Printer)	Mach Chk: (X ⚡ 69)	Internal malfunction.	Display error indication at affected display station. Disable display. Set sense: BSC : DC/US SNA: 081C Issue hardware poll and accept only POR from station.	At 3276/3278, set Normal/Test from Normal to Test and back again (or switch power off, then on). At 3230/3262/3268, press Test switch, or switch printer power off, then on. At 3287: 1. Press and hold Test switch. 2. Press and release Reset switch. 3. Release Test switch. Or switch 3287 power off, then on. At 3289, press Reset. At 5210, press START. Or switch 5210 power off, wait 10 seconds, and switch 5210 power on.
70 (Display) or Printer)	Mach Chk: (X ⚡ 70)		Display error indication at affected display station (display may not be successful because of display failure). Disable display. Set sense: BSC : IR SNA: 081C Issue hardware poll and accept only POR from station.	
71 (Display) or Printer)	Mach Chk: (X ⚡ 71)		Display error indication at affected display station. Disable display. Set sense: BSC : DC/US SNA: 081C Issue hardware poll and accept only POR from station.	
72 (Display or Printer)	Mach Chk: (X ⚡ 72)			
73 (Display or Printer)	Mach Chk: (X ⚡ 73)	Wrong configuration: 3276 has a 3278 Model 5 in subsystem.		Disconnect 3278 Model 5
74 (Feature)	Mach Chk: (X ⚡ 74)	Internal malfunction.	Display error indication at affected display station. Disable display. Set sense: BSC : DC/US or IR SNA: 081C Issue hardware poll and accept only POR from station.	At 3276/3278, set Normal/Test from Normal to Test and back again (or switch power off, then on). At 3230/3262/3268, press Test switch, or switch printer power off, then on. At 3287: 1. Press and hold Test switch. 2. Press and release Reset switch. 3. Release Test switch. Or switch 3287 power off, then on. At 3289, press Reset. At 5210, press START. Or switch 5210 power off, wait 10 seconds, and switch 5210 power on.
75 (MC)	Mach Chk Light Mach Chk: (X ⚡ 75)		Display error indication at affected display station. Disable terminal. Set sense: BSC: DC/US or IR SNA: 081C Poll is not issued and POR from station cannot be received.	Press and release Test Subsystem.
76 (MC)	Mach Chk Light Mach Chk: (X ⚡ 76)		Display error indication at affected display station. Disable terminal. Set sense: BSC: DC/US or IR SNA: 081C Poll is not issued, and power on reset (POR) from terminals cannot be received.	

Error Code	Indicator	Probable Cause	Effect	Recovery
77 (Display)	Mach Chk: (X 77)		Clear display. Display error indication at affected display station. Set sense: BSC: DC/US or IR SNA: 082B or 081C Disable display; set sense: BSC: DC/US SNA: 081C	Set Normal/Test switch from Normal to Test, then back to Normal (or switch power off, then on).
78 (BSC or SDLC)	Mach Chk Light Mach Chk: (X 78)	Internal malfunction.	Display error indication at affected display station. Disable terminal. Set sense: BSC: DC/US or IR SNA: 081C	Press and release Text/ Subsystem.
79 (BSC or SDLC)	Mach Chk Light Mach Chk: (X 79)			
81 (SDLC)	Mach Chk Light Mach Chk: (X 81)			
82 (MC)	Mach Chk Light Mach Chk: (X 82)	Error in Encrypt/Decrypt function.	Display error indication at affected display station. Disable Encrypt/Decrypt function if RESET is pressed.	
83 (MC)	Mach Chk Light Mach Chk: (X 83)			
85 (BSC or SDLC)	Mach Chk Light Mach Chk: (X 85)	Internal malfunction.	Display error indication at all display stations. Turn off Line Ready (OK). Stop machine.	Press and release Test/ Subsystem.
86 (SDLC) Loop	Mach Chk Light Mach Chk: (X 86)		Display error indication at all display stations. Turn off Line Ready (OK). Stop machine	
87 (BSC and SDLC)	Mach Chk Light Mach Chk: (X 87)			Press and release Test Subsystem. Perform host recovery if required.
88 (BSC and SDLC)	Mach Chk Light Mach Chk: (X 88)		Display error indication at all display stations. Turn off Line Ready (OK) and other Unit Operable lights. Stop machine.	
89 (MC)	Mach Chk Light Mach Chk: (X 89)			
90 (MC)	Mach Chk Light Mach Chk: (X 90)			
91 (MC)	Mach Chk Mach Chk: (X 91)			
92-98 (MC)	Mach Chk Light Mach Chk: (X 92-98)			
99 (MC)	Mach Chk Light Mach Chk: (X 99)			

Error Code	Indicator	Probable Cause	Effect	Recovery
21–87 (SDLC)	Sys Chk Light Program Chk: (X PROG 21–87)	21: EXR from upstream node. 22: Invalid OAF for PU (sense bits 800F). 23: PU Not Active (sense bits 8008). 24: Unrecognized DAF (sense bits 8004). 25: Segmenting Error. 26: LU is not active (sense bits 8009). 27: No LU-LU session (sense bits 8005). 28: Invalid ACTPU parameter (0821). 29: REQMS error (sense bits 080C, 0815, 1003, 1007). 30: Data Traffic Reset state (sense bits 2005). 31: Sequence number error (sense bits 2001). 32: FM data chaining error (sense bits 2002). 33: Normal flow DFC in INC state (sense bits 2002). 34: BB is not found on FM data request (sense bits 2003). 35: DFC carries EB in BETB (sense bits 2003). 40: Invalid 3270 command (sense bits 1003). 41: Data follows READ type command (sense bits 1003). 42: Nonsupported SNA command (sense bits 1003). 43: Control Function carries Null RU (sense bits 1003). 44: Invalid Signal request code (sense bits 1003). 50: ORDER with invalid buffer address (sense bits 1005). 51: Incomplete order sequence (sense bits 1005). 59: FI bit in RH0 is not supported (sense bits 400F). 60: CD in RH2 is required (sense bits 0829). 61: Exception Mode is not allowed for copy (sense bits 0843). 68: Invalid ACTLU parameter (sense bits 0821). 69: Second BIND is received from current PLU (sense bits 0815). 70: Session limit exceeded (sense bits 0805). 71: Bind RU is incomplete (sense bits 0821). 72: Invalid support level (RU1-3) (sense bits 0821). 73: Invalid PLU protocol (RU4) (sense bits 0821). 74: Invalid SLU protocol (RU5) (sense bits 0821). 75: Invalid common protocol (RU6, 7) (sense bits 0821). 76: Too small RU length (RU10) (sense bits 0821). 77: Too large buffer size (RU9, 11) (sense bits 0821). 78: Invalid LU type (RU14) (sense bits 0821). 79: Invalid screen size (RU20-24) (sense bits 0821). 80: Encrypt/Decrypt not supported; BIND parameter or error (RU26) (sense bits 0821). 82: Encrypt/Decrypt session (BIND) parameter error (sense bits 0821). 85: Encrypt/Decrypt state error (sense bits 2009). 86: Encrypt/Decrypt CRV failure (sense bits 0821). 87: Encrypt/Decrypt RU data error (sense bits 1001).	Display error indication at affected display station; if it cannot be displayed there, display it at all other display stations. Set sense bits XXXX (as indicated in adjacent column).	Press RESET. Await recovery from host. 43: If 3276 is in Encrypt/ Decrypt session, log off and then log on. 86: Verify the master key value. 87: Log off and then log on.

Note: Program checks 22, 23, 24, 26, 27, or 59 will not be displayed by some 3276s.

Appendix D. APL/Text Feature

The APL/Text processing capabilities of the IBM 3270 Information Display System are available on the devices shown in Figure D-1 when attached to a 3276 Control Unit Display Station. These devices must be equipped with the appropriate APL/Text and Extended Character Set Adapter or Text Print features, and must be attached to an APL/Text-featured 3276 Control Unit Display Station.

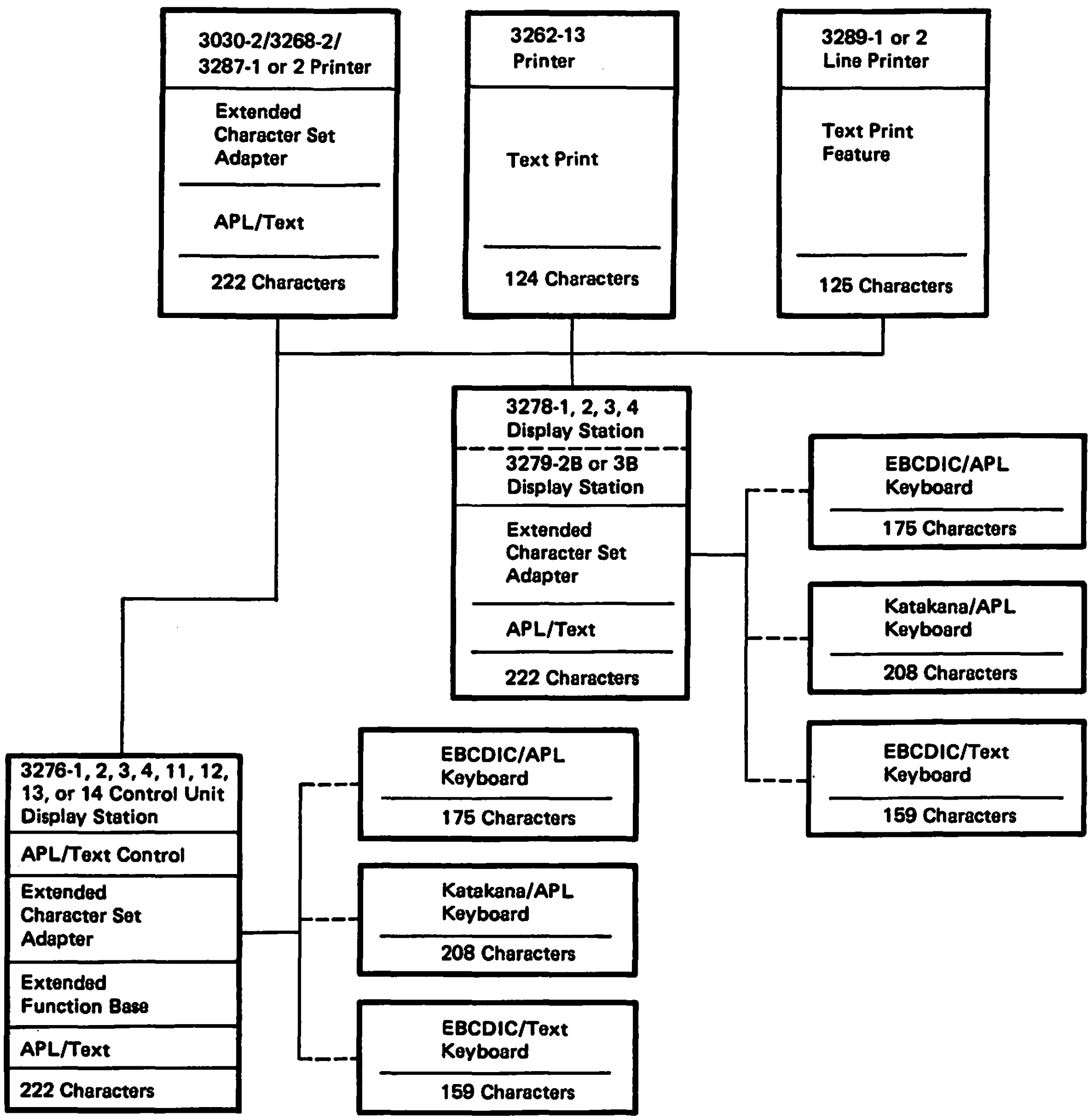


Figure D-1. Diagram of APL/Text Devices

APL/Text and Text Printer Data Streams

The I/O interface codes used by the APL/Text-featured 3276, the 3230, 3268, 3278, and 3287 with APL/Text and Extended Character Set Adapter features, and the 3279 Models 2B and 3B are shown in Figures D-2 and D-3. I/O interface codes with National Use differences are shown in Figure D-4. Codes used with Katakana/APL and Extended Character Set Adapter features are shown in Figures D-5 and D-6. The I/O interface codes used by the APL/Text-featured 3276 and 3262/3289 with the Text-Print feature are shown in Figure D-7. The 3230/3268/3278/3279/3287 APL/Text and the 3262/3289 Text print I/O interface codes do not affect the operation of any 3276 data stream commands, orders, or control characters. All 3230/3268/3278/3279/3287 APL-specific and Text-specific characters are specified by 2-byte sequences; each 2-byte sequence consists of a Graphic Escape (GE) (hex '08') control character followed by a character code.

The 3276 APL/Text data streams:

- Contain 94 EBCDIC characters (plus space).
- Specify all APL- and Text-specific characters by using a 2-byte sequence consisting of a hex 08 control character followed by a character code.
- Contain 10 graphic plot characters.

The 3276 Text print data streams:

- Contain 93 U.S. English set characters (plus space).

3276 APL/Text

The 3276 APL/Text Control special feature, the Extended Function Base special feature (prerequisite for the APL/Text Control feature), the APL/Text special feature, and the Extended Character Set Adapter special feature (prerequisite for the APL/Text feature) enable the 3276 to control 3230s, 3268s, 3278s, 3279s, and 3287s that have APL/Text capability and 3262s and 3289s that have text-print capability.

Attachment of the appropriate APL or Text keyboard to an APL/Text-featured 3276 enables the 3276 operator to interact with either APL or text applications as well as existing applications.

		00				01				10				11				← Bits 0,1
	Hex 1	00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11	← 2,3
Bits 4567	↓	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	← Hex 0
0000	0					SP	&	–						$\boxed{12}$	$\boxed{13}$	$\boxed{14}$	0	
0001	1							/	É	a	j	$\boxed{11}$		A	J		1	
0010	2					â	ê	Â	Ê	b	k	s		B	K	S	2	
0011	3						ë		Ë	c	l	t		C	L	T	3	
0100	4							À	È	d	m	u		D	M	U	4	
0101	5									e	n	v		E	N	V	5	
0110	6						î		Î	f	o	w		F	O	W	6	
0111	7						ï		Ï	g	p	x		G	P	X	7	
1000	8					ç		Ç		h	q	y		H	Q	Y	8	
1001	9								$\boxed{7}$	i	r	z		I	R	Z	9	
1010	A					$\boxed{1}$	$\boxed{3}$	$\boxed{6}$	:									
1011	B					.	$\boxed{4}$	,	$\boxed{8}$					ô	û	Ô	Û	
1100	C					<	*	%	$\boxed{9}$						ü		Ü	
1101	D					(	)	_	'								Ù	
1110	E					+	;	>	=									
1111	F					$\boxed{2}$	$\boxed{5}$	?	$\boxed{10}$									

Notes:

$\boxed{1}$ *through* $\boxed{14}$ *are the National use differences. They are shown in Figure D-4.*

[box with dark corner] = Canadian French characters.

1. *No control characters are shown in this chart.*
2. *All codes can be entered from the keyboard.*
3. *Character code assignments other than those shown within all outlined areas of this chart are undefined. If an undefined character code is programmed, the character that will be displayed or printed is a hyphen (hex 60) (3287 prints a blank); also, a hex 60 will be returned on a subsequent read operation. (For control units with Configuration Support C installed, undefined control codes (X'00' to X'3F') cause a negative response (SNA) or an Op Chk (BSC). The character displayed or printed for an undefined character code is unpredictable.) The character displayed or printed for a given undefined character code may be different for other devices. IBM reserves the right to change at any time the character displayed or printed for any undefined character code.*
4. *NL (hex 15), EM (hex 19), FF (hex 0C), and NUL (hex 00) are not displayed or printed. The DUP (hex 1C) and FM (hex 1E) control characters on dual case terminals are displayed as * and ; respectively, and are printed as * and;.*
5. *DUP (hex 1C) and FM (hex 1E) control characters on mono case terminals are displayed as * and ; respectively, and are printed as * and;.*

Figure D-2. APL/Text Feature, 1-Byte I/O Interface Codes (3230/3268/3276/3278/3279/3287)

<table>
<tr><td></td><td></td><td colspan="4">00</td><td colspan="4">01</td><td colspan="4">10</td><td colspan="4">11</td><td>Bits 0,1</td></tr>
<tr><td></td><td></td><td>00</td><td>01</td><td>10</td><td>11</td><td>00</td><td>01</td><td>10</td><td>11</td><td>00</td><td>01</td><td>10</td><td>11</td><td>00</td><td>01</td><td>10</td><td>11</td><td>2,3</td></tr>
<tr><td>Bits 4567</td><td>Hex 1</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>Hex 0</td></tr>
<tr><td>0000</td><td>0</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>~</td><td>⎕</td><td>¯</td><td>α</td><td>{</td><td>}</td><td></td><td>⊙</td><td></td></tr>
<tr><td>0001</td><td>1</td><td></td><td></td><td></td><td></td><td>A̲</td><td>J̲</td><td></td><td>∧</td><td>| |</td><td>█</td><td>°</td><td>∊</td><td>(</td><td>)</td><td>1</td><td>1</td><td></td></tr>
<tr><td>0010</td><td>2</td><td></td><td></td><td></td><td></td><td>B̲</td><td>K̲</td><td>S̲</td><td>¨</td><td>═</td><td>█</td><td>–</td><td>\</td><td>+</td><td>-</td><td>2</td><td>2</td><td></td></tr>
<tr><td>0011</td><td>3</td><td></td><td></td><td></td><td></td><td>C̲</td><td>L̲</td><td>T̲</td><td></td><td>|</td><td>▬</td><td>●</td><td>ρ</td><td>■</td><td>+</td><td>3</td><td>3</td><td></td></tr>
<tr><td>0100</td><td>4</td><td></td><td></td><td></td><td></td><td>D̲</td><td>M̲</td><td>U̲</td><td></td><td>|</td><td>▬</td><td>n</td><td>ω</td><td>└</td><td>┘</td><td></td><td>4</td><td></td></tr>
<tr><td>0101</td><td>5</td><td></td><td></td><td></td><td></td><td>E̲</td><td>N̲</td><td>V̲</td><td></td><td>|</td><td>█</td><td></td><td></td><td>┌</td><td>┐</td><td></td><td>5</td><td></td></tr>
<tr><td>0110</td><td>6</td><td></td><td></td><td></td><td></td><td>F̲</td><td>O̲</td><td>W̲</td><td></td><td></td><td></td><td></td><td>×</td><td>⊢</td><td>⊣</td><td></td><td>6</td><td></td></tr>
<tr><td>0111</td><td>7</td><td></td><td></td><td></td><td></td><td>G̲</td><td>P̲</td><td>X̲</td><td></td><td></td><td></td><td></td><td>\</td><td>⊥</td><td>⊤</td><td></td><td>7</td><td></td></tr>
<tr><td>1000</td><td>8</td><td></td><td></td><td></td><td></td><td>H̲</td><td>Q̲</td><td>Y̲</td><td>∨</td><td></td><td></td><td></td><td>÷</td><td>§</td><td>¶</td><td></td><td>8</td><td></td></tr>
<tr><td>1001</td><td>9</td><td></td><td></td><td></td><td></td><td>I̲</td><td>R̲</td><td>Z̲</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>9</td><td></td></tr>
<tr><td>1010</td><td>A</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>↑</td><td>⊃</td><td>∩</td><td>∇</td><td>⍲</td><td>⌶</td><td>⌿</td><td></td><td></td></tr>
<tr><td>1011</td><td>B</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>↓</td><td>⊂</td><td>∪</td><td>∆</td><td>⍱</td><td>!</td><td>⍀</td><td>⍫</td><td></td></tr>
<tr><td>1100</td><td>C</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>≤</td><td>⌑</td><td>⊥</td><td>⊤</td><td></td><td>⍒</td><td></td><td>⍙</td><td></td></tr>
<tr><td>1101</td><td>D</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>⌈</td><td>○</td><td>[</td><td>]</td><td>⌽</td><td>⍋</td><td>⊖</td><td>⍟</td><td></td></tr>
<tr><td>1110</td><td>E</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>⌊</td><td>±</td><td>≥</td><td>≠</td><td></td><td>⍞</td><td>⌹</td><td>⍎</td><td></td></tr>
<tr><td>1111</td><td>F</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>→</td><td>←</td><td>.</td><td>|</td><td>⍉</td><td>A</td><td>⍕</td><td></td><td></td></tr>
</table>

Notes:

Subscripts

Superscripts

1. *These codes, preceded by a hex 08 control character, transmit the graphics shown.*
2. *No control characters are shown in this chart.*
3. *All codes within the solid outlined areas of this chart can be entered from the keyboard; the 10 graphic plot characters within the dashed outlined area cannot be entered from the keyboard.*
4. *Character code assignments other than those shown within all outlined areas of this chart are undefined. If an undefined character code is programmed, the character that will be displayed or printed is a hyphen (hex 60) (3287 prints a blank); also, a hex 60 will be returned on a subsequent read operation. For control units with Configuration Support C installed, undefined control codes from X'00' to X'3F' cause a negative response (SNA) or an Op Chk (BSC). The character displayed or printed for a given undefined character code may be different for other devices. IBM reserves the right to change at any time the character displayed or printed for any undefined character code.*

Figure D-3. APL/Text Feature, 2-Byte I/O Interface Codes (3230/3268/3276/3278/3279/3287)

<table>
<tr><th rowspan="2">Character Set</th><th>Code Key (Note 1)</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th><th>13</th><th>14</th></tr>
<tr><th>EBCDIC</th><th>4A</th><th>4F</th><th>5A</th><th>5B</th><th>5F</th><th>6A</th><th>79</th><th>7B</th><th>7C</th><th>7F</th><th>A1</th><th>C0</th><th>D0</th><th>E0</th></tr>
<tr><td colspan="2">English (US)</td><td>¢</td><td>|</td><td>!</td><td>$</td><td>¬</td><td>¦</td><td>`</td><td>#</td><td>@</td><td>"</td><td>~</td><td>{</td><td>}</td><td>\</td></tr>
<tr><td colspan="2">Austrian/German</td><td>Ä</td><td>!</td><td>Ü</td><td>$</td><td>^</td><td>ö</td><td>`</td><td>#</td><td>§</td><td>"</td><td>β</td><td>ä</td><td>ü</td><td>Ö</td></tr>
<tr><td colspan="2">Austrian/German (Alternate)</td><td>ö</td><td>|</td><td>ü</td><td>Ü</td><td>¬</td><td>β</td><td></td><td>Ä</td><td>Ö</td><td>ä</td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="2">Danish/Norwegian</td><td>#</td><td>!</td><td>¤</td><td>Å</td><td>^</td><td>ϕ</td><td>`</td><td>Æ</td><td>ø</td><td>"</td><td>ü</td><td>æ</td><td>å</td><td>\</td></tr>
<tr><td colspan="2">Danish/Norwegian (Alternate)</td><td>ϕ</td><td>|</td><td>å</td><td>Å</td><td>¬</td><td>¦</td><td></td><td>Æ</td><td>ø</td><td>æ</td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="2">Finnish/Swedish</td><td>§</td><td>!</td><td>¤</td><td>Å</td><td>^</td><td>ö</td><td>é</td><td>Ä</td><td>Ö</td><td>"</td><td>ü</td><td>ä</td><td>å</td><td>É</td></tr>
<tr><td colspan="2">Finnish/Swedish (Alternate)</td><td>ö</td><td>|</td><td>å</td><td>Å</td><td>¬</td><td>¦</td><td></td><td>Ä</td><td>Ö</td><td>ä</td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="2">French</td><td>°</td><td>!</td><td>§</td><td>$</td><td>^</td><td>ù</td><td>`</td><td>£</td><td>à</td><td>"</td><td>¨</td><td>e</td><td>è</td><td>ç</td></tr>
<tr><td colspan="2">Italian</td><td>°</td><td>!</td><td>é</td><td>$</td><td>^</td><td>ò</td><td>ù</td><td>£</td><td>§</td><td>"</td><td>ì</td><td>à</td><td>è</td><td>ç</td></tr>
<tr><td colspan="2">Portuguese (Note 2)</td><td>[</td><td>!</td><td>]</td><td>$</td><td>^</td><td>õ</td><td>`</td><td>Ã</td><td>Õ</td><td>"</td><td>ç</td><td>ã</td><td>´</td><td>Ç</td></tr>
<tr><td colspan="2">Spanish</td><td>[</td><td>|</td><td>]</td><td>Pts</td><td>¬</td><td>ñ</td><td>`</td><td>Ñ</td><td>@</td><td>"</td><td>¨</td><td>{</td><td>}</td><td>\</td></tr>
<tr><td colspan="2">Spanish (Alternate)</td><td>¢</td><td>|</td><td>!</td><td>Pts</td><td>¬</td><td>¦</td><td></td><td>Ñ</td><td>@</td><td>ñ</td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="2">English (UK)</td><td>$</td><td>|</td><td>!</td><td>£</td><td>¬</td><td>¦</td><td>`</td><td>#</td><td>@</td><td>"</td><td>‾</td><td>{</td><td>}</td><td>\</td></tr>
<tr><td colspan="2">Belgian</td><td>[</td><td>!</td><td>]</td><td>$</td><td>^</td><td>ù</td><td>`</td><td>#</td><td>à</td><td>"</td><td>¨</td><td>é</td><td>è</td><td>ç</td></tr>
<tr><td colspan="2">Brazilian/Portuguese</td><td>É</td><td>!</td><td>$</td><td>Ç</td><td>^</td><td>ç</td><td>ã</td><td>Õ</td><td>Ã</td><td>"</td><td>˜</td><td>õ</td><td>é</td><td>\</td></tr>
<tr><td colspan="2">Japanese (English)</td><td>£</td><td>|</td><td>!</td><td>¥</td><td>¬</td><td>¦</td><td>`</td><td>#</td><td>@</td><td>"</td><td>‾</td><td>{</td><td>}</td><td>$</td></tr>
<tr><td colspan="2">Spanish Speaking</td><td>[</td><td>|</td><td>]</td><td>$</td><td>¬</td><td>ñ</td><td>`</td><td>Ñ</td><td>@</td><td>"</td><td>¨</td><td>{</td><td>}</td><td>\</td></tr>
<tr><td colspan="2">Canadian (French)</td><td>à</td><td>!</td><td>´</td><td>$</td><td>^</td><td>ù</td><td>`</td><td>#</td><td>@</td><td>"</td><td>¨</td><td>é</td><td>è</td><td>¸</td></tr>
<tr><td colspan="2">International</td><td>[</td><td>!</td><td>]</td><td>$</td><td>^</td><td>¦</td><td>`</td><td>#</td><td>@</td><td>"</td><td>~</td><td>{</td><td>}</td><td>\</td></tr>
</table>

Notes:

1. *See Figure D-2 for code points.*
2. *Portugal*
   a. *Host system to control unit -4C or EO is Ç*
   b. *Control unit to host system -EO is Ç*
   c. *Control unit to host system -4C (<) is removed.*

Figure D-4. National Use Differences I/O Interface Code (3230/3268/3276/3278/3279/3287)

Bits 4567	Hex 1 ↓	00 00 0	00 01 1	00 10 2	00 11 3	01 00 4	01 01 5	01 10 6	01 11 7	10 00 8	10 01 9	10 10 A	10 11 B	11 00 C	11 01 D	11 10 E	11 11 F
0000	0					SP	&	-			ソ					$	0
0001	1					。	ェ	/		ア	タ	—		A	J		1
0010	2					「	ォ			イ	チ	ヘ		B	K	S	2
0011	3					」	ャ			ウ	ツ	ホ		C	L	T	3
0100	4					、	ュ			エ	テ	マ		D	M	U	4
0101	5					•	ョ			オ	ト	ミ		E	N	V	5
0110	6					ヲ	ッ			カ	ナ	ム		F	O	W	6
0111	7					ァ				キ	ニ	メ		G	P	X	7
1000	8					ィ	ー			ク	ヌ	モ		H	Q	Y	8
1001	9					ゥ				ケ	ネ	ヤ		I	R	Z	9
1010	A					£	!		:	コ	ノ	ユ	レ				
1011	B						¥	,	#				ロ				
1100	C					<	*	%	@	サ		ヨ	ワ				
1101	D					(	)	_	'	シ	ハ	ラ	ン				
1110	E					+	;	>	=	ス	ヒ	リ	•				
1111	F					|	¬	?	"	セ	フ	ル	°				

Notes:

1. *No control characters are shown in this chart.*
2. *All codes can be entered from the keyboard.*
3. *Character code assignments other than those shown within all outlined areas of this chart are undefined. If an undefined character code is programmed, the character that will be displayed or printed is a hyphen (hex 60) (3287 prints a blank); also, a hex 60 will be returned on a subsequent read operation. The character displayed or printed for a given undefined character code may be different for other devices. IBM reserves the right to change at any time the character displayed or printed for any undefined character code.*
4. *NL (hex 15), EM (hex 19), FF (hex 0C), and NUL (hex 00) are not displayed or printed. The DUP (hex 1C) and FM (hex 1E) control characters on dual case terminals are displayed as * and ; respectively, and are printed as * and;.*
5. *DUP (hex 1C) and FM (hex 1E) control characters on mono case terminals are displayed as * and ; respectively, and are printed as * and;.*

Figure D-5. Katakana/APL 1-Byte I/O Interface Codes (3230/3268/3276/3278/3279/3287)

Bits 4567	Hex 1	00				01				10				11				← Bits 0,1
		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11	← 2,3
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	← Hex 0
0000	0									~	⎕	—	⍺	{	}		$^{⊙}$	
0001	1					$\underline{A}$	$\underline{J}$		∧	▏▕	▌	°	∊	$^{(}$	$^{)}$	$_{1}$	^{1}	
0010	2					$\underline{B}$	$\underline{K}$	$\underline{S}$	¨	═	▐	−	⍀	$^{+}$	$^{-}$	$_{2}$	^{2}	
0011	3					$\underline{C}$	$\underline{L}$	$\underline{T}$		▏	▀	●	⍴	■	+	$_{3}$	^{3}	
0100	4					$\underline{D}$	$\underline{M}$	$\underline{U}$		▕	▄	n	⍵	└	┘		^{4}	
0101	5					$\underline{E}$	$\underline{N}$	$\underline{V}$		|	█			┌	┐		^{5}	
0110	6					$\underline{F}$	$\underline{O}$	$\underline{W}$					×	⊢	⊣		^{6}	
0111	7					$\underline{G}$	$\underline{P}$	$\underline{X}$					\	⊥	⊤		^{7}	
1000	8					$\underline{H}$	$\underline{Q}$	$\underline{Y}$	∨				÷	§	¶		^{8}	
1001	9					$\underline{I}$	$\underline{R}$	$\underline{Z}$									^{9}	
1010	A									↑	⊃	∩	∇	⍲	⌶	⌿		
1011	B									↓	⊂	∪	Δ	⍱	!	⍀	⍫	
1100	C									≤	¤	⊥	⊤		⍒		⍙	
1101	D									⌈	∘	[	]	⌽	⍋	⊖	⍟	
1110	E									⌊	±	≥	≠		⍞	⌹	⍎	
1111	F									→	←	.	|	⍉	A	⍕		

Notes:

Subscripts

Superscripts

1. *These codes, preceded by a hex 08 control character, transmit the graphics shown.*
2. *No control characters are shown in this chart.*
3. *All codes within the solid outlined areas of this chart can be entered from the keyboard; the 10 graphic plot characters within the dashed outlined area cannot be entered from the keyboard.*
4. *Character code assignments other than those shown within all outlined areas of this chart are undefined. If an undefined character code is programmed, the character that will be displayed or printed is a hyphen (hex 60) (3287 prints a blank); also, a hex 60 will be returned on a subsequent read operation. For control units with Configuration Support C installed, undefined control codes from X'00' to X'3F' cause a negative response (SNA) or an Op Chk (BSC). The character displayed or printed for a given undefined character code may be different for other devices. IBM reserves the right to change at any time the character displayed or printed for any undefined character code.*

Figure D-6. Katakana/APL 2-Byte I/O Interface Codes (3230/3268/3276/3278/3279/3287)

Bits 0,1		00				01				10				11			
Bits 2,3		00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
Bits 4567	Hex 1 / Hex 0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0					SP	&	–				–	0	{	}	\	0
0001	1							/		a	j	° ~	1	A	J		1
0010	2									b	k	s	2	B	K	S	2
0011	3									c	l	t	3	C	L	T	3
0100	4									d	m	u	4	D	M	U	4
0101	5									e	n	v	5	E	N	V	5
0110	6									f	o	w	6	F	O	W	6
0111	7									g	p	x	7	G	P	X	7
1000	8									h	q	y	8	H	Q	Y	8
1001	9								`	i	r	z	9	I	R	Z	9
1010	A					¢	!	¦	:								
1011	B					.	$	,	#	{	}	└	┘				
1100	C					<	*	%	@	≤	¤	┌	┐				
1101	D	CR				(	)	_	'	(	)	[	]				
1110	E					+	;	>	=	+	±	≥	≠				
1111	F					|	¬	?	"	+	■	●	_				

Notes:

Superscripts

1. *No control characters except CR (hex 0D) are shown in this chart. The CR control character provides the capability to inhibit line advance after a line of characters is printed.*
2. *Character code hex A1 causes a ° (degree) character to print when the 3289 text print belt is installed and a ~ (tilde) character to print when a U.S. English 3289 print belt is installed.*
3. *Character code assignments other than those shown within the outlined areas of this chart are undefined. If an undefined character code is programmed, the character that will be printed is a hyphen (hex 60); also, a hex 60 will be returned on a subsequent read operation. IBM reserves the right to change at any time the character printed for an undefined character code.*
4. *NL (hex 15), EM (hex 19), FF (hex 0C), and NUL (hex 00) are not printed. The DUP (hex 1C) and FM (hex 1E) control characters are printed as * and ; respectively.*

Figure D-7. Text Print I/O Interface Codes (3262/3289)

3278-1, -2, -3, and -4 or 3279-2B and -3B APL/Text

The APL/Text special feature, the Extended Character Set Adapter special feature (prerequisite for the APL/Text feature), and the appropriate APL or Text keyboard enable a 3278 or 3279 operator to interact with either APL or text applications as well as existing applications.

APL Keyboards

The 3276 and 3278/3279 APL keyboards are typewriter-like keyboards with keys that contain both APL and the featured-language characters. The APL characters are colored orange (on white keys). The PF1 through PF12 keys on the APL keyboards are located on the right side of the keyboard instead of on the front of the top row of keys as on non-APL keyboards; PF13 through PF24 keys are not available on APL keyboards. The Numeric Lock feature is available for all APL keyboards.

87- and 88-Key Typewriter/APL Keyboards

The 87-key typewriter/APL (U.S. English) keyboard is shown in Figure D-8 (the Japanese English typewriter/APL keyboard has 88 keys). This keyboard is available in all 3276 and 3278/3279 keyboard languages.

The typewriter/APL keyboard enables a 3276 or 3278/3279 operator to enter the 81 APL-specific characters as well as the 94-character-plus-space EBCDIC dual-case character set. The following characters can be entered:

With APL "off"	–	94	EBCDIC characters plus space
With APL "on"	–	81	APL-specific characters plus:
		10	numerics (0 through 9)
		26	uppercase alphabet characters
		16	invariant symbols (excluding & and %)

When the display station is first turned on, the typewriter/APL keyboard operates similarly to the 75-key typewriter keyboard without APL, with the exception of the PF1 through PF12 keys. Pressing the APL ON/OFF key (with the ALT key held down) causes the keyboard to enter APL mode (the letters APL display in the Operator Information Area); in this mode the APL characters on the right half of the keys may be entered (the Shift, Lock, and ALT keys are used to select the desired character on a key). The keyboard is returned to normal (non-APL) mode by pressing the APL ON/OFF key again.

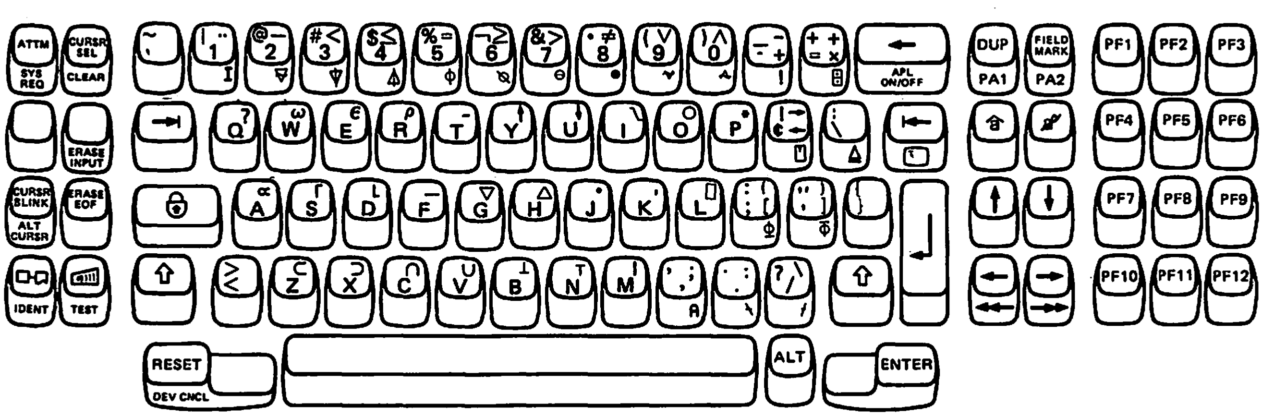


Figure D-8. 87-Key Typewriter/APL Keyboard

88-Key Katakana Typewriter/APL Keyboard

The 88-key Katakana typewriter/APL keyboard (available for IBM World Trade Americas/Far East only) is shown in Figure D-9.

The Katakana typewriter/APL keyboard enables a 3276 or 3278/3279 operator to enter the 81 APL-specific characters as well as the 127-plus-space Japanese Katakana character set. The following characters can be entered:

With APL "off"	–	127-character Japanese Katakana set plus space	
With APL "on"	–	81	APL-specific characters plus:
		10	numerics (0 through 9)
		26	uppercase alphabet characters
		16	invariant symbols (excluding & and %)

When the display station is first turned on, the typewriter/APL keyboard operates like the 88-key Katakana typewriter keyboard without APL, with the exception of the PF1 through PF12 keys. Momentarily pressing the APL ON/OFF key (with the ALT key held down) places the keyboard in APL downshift mode (the letters APL display in the Operator Information Area). APL upshift characters can be entered either by pressing and holding either ⇧ (upshift) key or by pressing the ⊖ (Lock) key; when the keyboard is locked in APL upshift mode, pressing either ⇧ key returns the keyboard to APL downshift mode. The APL characters on the right front of keys can be entered by pressing and holding the ALT key. The keyboard is returned to non-APL mode (ALPHA downshift) by pressing the APL ON/OFF key again.

APL Keyboard World Trade Considerations

The APL programming support does not support certain Canadian-French and Katakana characters on the Canadian-French and Katakana typewriter/APL keyboards. The unsupported Canadian-French characters are all those enterable by a dead key sequence except à,è,e, and u. The unsupported Katakana characters are those with I/O interface codes that are not included in the 94-character-plus-space EBCDIC character set. However, the 3276 control unit does not block these unsupported codes when they are sent inbound to the host system.

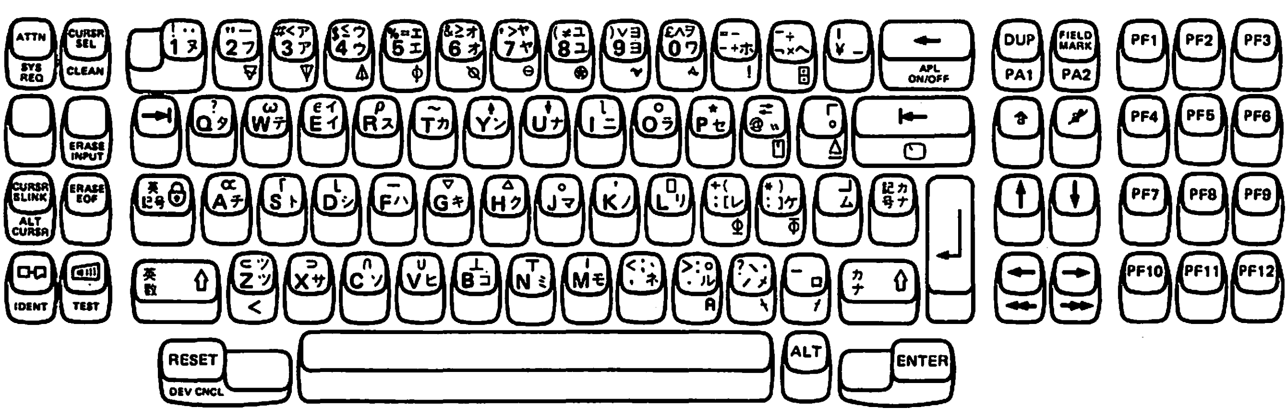


Figure D-9. 88-Key Katakana Typewriter/APL Keyboard

87-Key Typewriter/Text Keyboard

The 87-key typewriter/Text keyboard (shown in Figure D-10) is a typewriter-like keyboard with keys that contain both U.S. English and Text-specific characters. This keyboard is available for U.S. English only (the Text keyboard is not available in IBM Europe/Middle East/Africa countries).

The Text-specific characters are colored green (on white keys). The PF1 through PF12 keys on the typewriter/Text keyboard are located on the right side of the keyboard instead of on the front of the top row of keys as on non-Text keyboards: PF13 through PF24 are not available on the typewriter/Text keyboard.

The 3276 or 3278/3279 operator can use the typewriter/Text keyboard to enter the 65 Text-specific characters as well as the 94-character-plus-space U.S. English character set. The following characters can be entered:

With Text "off"	–	94	U.S. English characters plus space
With Text "on"	–	65	Text-specific characters plus:
		10	numerics (0 through 9)
		26	uppercase alphabet characters
		26	lowercase alphabet characters
		9	symbols (. < ; , > ? : !)

When the display station is first turned on, the typewriter/Text keyboard operates like the 75-key typewriter keyboard without Text, with the exception of the PF1 through PF12 keys. Pressing the TEXT ON/OFF key causes the keyboard to enter Text mode (the letters TEXT display in the Operator Information Area); in this mode the text characters on the right half of the keys may be entered (the Shift, Lock, and ALT keys are used to select the desired character on a key). The keyboard is returned to normal (non-Text) mode by pressing the TEXT ON/OFF key again.

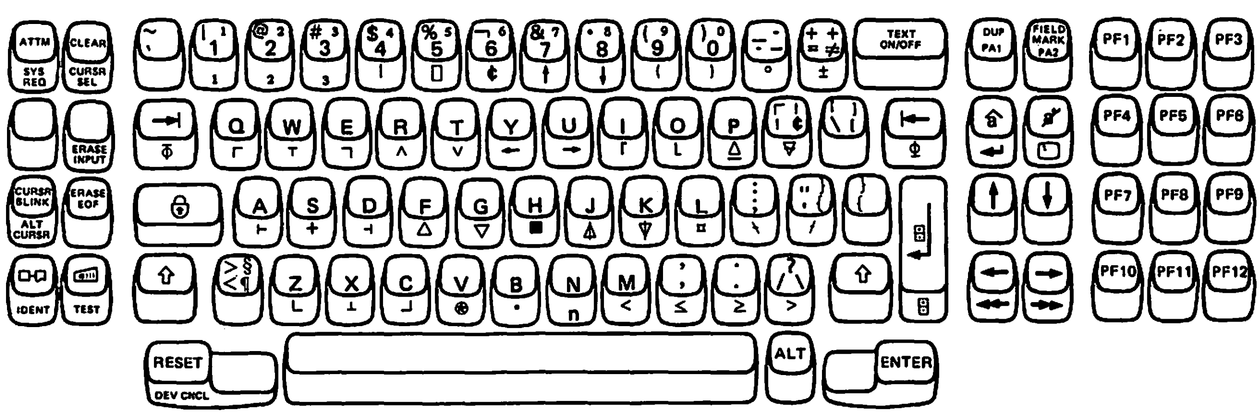


Figure D-10. 87-Key Typewriter/Text Keyboard

3230-2, 3268-2, and 3287-1 and -2 with APL/Text

The APL/Text feature for 3230, 3268, and 3287 (standard feature for 3230 and 3268, special feature for 3287), and its prerequisite Extended Character Set Adapter enable the 3230, 3268, and 3287 to print the following characters:

- 94 EBCDIC characters plus space
- 81 APL-specific characters
- 37 Text-unique characters
- 10 graphic plot characters

3262-13, and 3289-1 and -2 with Text Print

The 3289 Text Print special feature (not available in IBM Europe/Middle East/ Africa countries) and the 3262 Text Print standard feature print the following characters:

3262-13	3289-1 and 2
• 94 U.S. English characters • 30 TN characters	• 93 U.S. English characters plus space • 32 TN characters

Note: *The 93-character U.S. English set for 3289 is identical with the normal 94-character U.S. English set except the tilde (~) symbol is not included.*

The printing speed in lines per minute (1pm) varies with the size of the character set as follows:

Print Bands Characters per Set	Nominal Speed (1pm)		
	3262-13	3289-1	3289-2
48	325	155	400
64	230	120	300
94		80	230
96	180		
125		40	160
128	125		

Note: *Actual printer throughput depends upon operational and system characteristics. The print speed may be affected by such factors as communication line speed, control unit load, character set, and application program.*

Local or host-initiated copy operations from a 3278/3279 to a 3262/3289, with or without the Text Print feature installed, are limited to the normal 94-character U.S. English set.

BSC Copy Command

For control units operating under BSC, if APL-or TEXT-specific characters reside in the device buffer, a copy operation initiated by the BSC Copy command will be allowed only to another ECSA-featured device. If the "to" device is not equipped with an ECSA feature, an operation check will be returned to the host.

Local Copy

A local copy from an ECSA featured display with APL/Text characters on the screen will print correctly on an ECSA-featured 3287 printer with APL ROS installed. Local copy from an ECSA-featured display with APL/Text characters on the screen will be allowed to print on a non-ECSA-featured 3287 printer. The standard EBCDIC character set will print correctly, but APL/Text-specific characters will print as EBCDIC characters or hyphens.

Appendix E. Katakana Feature

This appendix contains Katakana unique information interface codes and the keyboard shift operations.

Interface Codes

Figure E-1 shows the Japanese Katakana EBCDIC interface codes for several control unit/device combinations.

Katakana Keyboards Shift Operations (3178, 3276, 3278, and 3279)

The Katakana keyboards shift operations are different from the EBCDIC keyboards described in Chapter 2. The following paragraphs discuss the unique keys and operations.

Katakana Shift Keys (3178, 3278, and 3279)

Four shifts [upper and lower left (UL and LL) and upper and lower right (UR and LR)] on the Katakana keyboards are used with the 3178, 3276, 3278, and 3279 displays:

Shift	Typewriter Keyboard		Data Entry Keyboard		Operator Message
UL	英記号	Alpha Symbol	数字	Alpha Symbol Numeric	ALPHA ⇧
LL	英数	Alphameric	英字	Alpha	ALPHA
UR	カナ記号	KANA Symbol	カナ記号	KANA Symbol	カナ ⇧
LR	カナ	Katakana	カナ	Katakana	カナ

The characters associated with each shift level are shown in the corresponding position of the key tops. In normal operation, the appropriate shift key is pressed and released to enter the required shift level; the keyboard remains in that shift level until another is selected. However, in a programmed numeric field (program attribute), the keyboard is automatically set to the upper left (UL) shift, and all characters for that shift are valid unless a keyboard with the Numeric Lock feature is being used. The Numeric Lock feature limits the entries to 0-9, minus (-), decimal sign, and DUP. This automatic UL shift may be overridden by pressing and holding the desired shift key; releasing the shift key returns the keyboard to the UL shift.

Holding a shift key when leaving the programmed numeric field causes the keyboard to enter and remain in that shift level until another shift key is pressed.

On a data entry of data-entry keypunch layout keyboard, the Numeric Lock feature is disabled while the Alpha, Numeric, Latin Shift, Lock, or upper left shift (3178, 3276, or 3278) key is operated.

	Bits 0,1 →	00				01				10				11			
	Bits 2,3 →	00	01	10	11	00	01	10	11	00	01	10	11	00	01	10	11
Bits 4567	Hex 1 ↓ / Hex 0 →	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL				SP	&	–			ソ					$	0
0001	1		SBA			。	ェ	/		ア	タ	ー		A	J		1
0010	2		EUA			「	ォ			イ	チ	ヘ		B	K	S	2
0011	3		IC			」	ャ			ウ	ツ	ホ		C	L	T	3
0100	4					、	ュ			エ	テ	マ		D	M	U	4
0101	5	PT	NL			・	ョ			オ	ト	ミ		E	N	V	5
0110	6					ヲ	ッ			カ	ナ	ム		F	O	W	6
0111	7					ァ				キ	ニ	メ		G	P	X	7
1000	8	GE				ィ	–			ク	ヌ	モ		H	Q	Y	8
1001	9		EM			ゥ				ケ	ネ	ヤ		I	R	Z	9
1010	A					£	!		:	コ	ノ	ユ	レ				
1011	B					.	¥	,	#				ロ				
1100	C	FF	DUP		RA	<	*	%	@	サ		ヨ	ワ				
1101	D	CR	SF			(	)	_	'	シ	ハ	ラ	ン				
1110	E		FM			+	;	>	=	ス	ヒ	リ	·				
1111	F					|	¬	?	"	セ	フ	ル	°				

Notes:

1. *Character code assignments other than those shown within all outlined areas of this chart are undefined. If an undefined character code is programmed, the character that will be displayed or printed is a hyphen (–); hex code 60 will be returned on a subsequent read operation. IBM reserves the right to change at any time the character displayed for an undefined character code.*
2. *CR, NL, EM, and FF control characters are displayed or printed as blank characters. The DUP and FM control characters are displayed as * and ; respectively.*
3. *Hex code 6A is used for CU addressing, device addressing, buffer addressing, and control purposes (for example, WCC and CCC), but has no associated graphic character.*
4. *For AID, attribute, write control (WCC), copy control (CCC), CU and device address, buffer address, sense, and status characters, bits 0 and 1 are assigned so that each character can be represented by a graphic character in Figure 1-4.*
5. *For BSC data-link control characters, see Chapter 3. For the SCS control codes associated with the SNA Character String feature on 3230, 3262, 3268, 3287 (with the 3274/3276 Attachment feature) and 3289 printers, see Chapter 2.*

Figure E-1. Japanese Katakana EBCDIC I/O Interface Code for 3276 Units with 3178, 3230, 3262, 3268, 3278, 3279, 3287 (with 3276 Attachment Feature), 3289, and 5210 Terminals Attached

Appendix F. Encrypt/Decrypt Feature

Encrypt/Decrypt Products

The IBM Cryptographic Subsystem is a combination hardware and programming implementation of cryptography for data security. It consists of the following separate products:

- IBM Programmed Cryptographic Facility Program Product (OS/VS1 and OS/VS2 MVS only).
- ACF/VTAM (Level 3.0 or higher) Encrypt/Decrypt feature.
- 3276 Encrypt/Decrypt feature.

The first two products reside at the host processor; the third resides in the control unit.

IBM Programmed Cryptographic Facility Program Product

This product contains the following functions: encrypt/decrypt, key generation, and key management. The encrypt/decrypt function is an IBM programmed implementation of the Federal Data Encryptions Standard (DES) algorithm as published by the National Bureau of Standards in January 1977 and adopted as the United States Federal Information Processing Standard (FIPS 46) in July 1977.

The other functions of the IBM Programmed Cryptographic Facility generate new keys upon request and in general manage all the keys used throughout the network. Under the IBM key management concept, since the enciphering algorithm is published, protection is derived from keeping the keys secret.

ACF/VTAM Encrypt/Decrypt Feature

This feature provides cryptographic support in ACF/VTAM by:

- Allowing the specification of a physical cryptographic feature on a Logical Unit (LU) basis.
- Being an interface with the Programmed Cryptographic Facility Program Product for enciphering and deciphering messages and key management.
- Supporting cryptographic changes to SNA.

3276 Encrypt/Decrypt Feature

This feature provides hardware implementation of the DES algorithm for encrypting and decrypting data on a TP line. It is applicable to the 3276 Models 11-14 only. When used with the ACF/VTAM Encrypt/Decrypt feature described above, data transmitted via the transmission subsystem can be safeguarded through cryptography from modification, disclosure, or both. Installed in the control unit with SDLC line control, this feature provides encrypt/decrypt services for up to 8 attached terminals. Included in the feature are:

- A single secondary LU key (terminal master key) storage element and logic to perform enciphering and deciphering operations for secondary LUs by block-chaining.
- A security keylock located in the customer access area of the control units.
- A mercury battery, IBM PN 1743456, to sustain the terminal master key when the control unit power is off.

When the Encrypt/Decrypt feature is used in conjunction with other IBM Cryptographic Subsystem products and is operating in an SNA/SDLC environment, data may be transmitted between the control unit and the host computer in a form that precludes accidental or intentional disclosure; neither can the data be modified without detection.

In SNA terminology, communication occurs between network nodes (application programs and terminals), each node being an LU. Data may be transmitted between the host computer (the primary LU) and a terminal attached to the control unit (the secondary LU) once the LUs have established an LU-LU session. When the cryptographic function is not used, the data is transmitted in the clear, that is, not enciphered. When the cryptographic function is used, the data is enciphered, thus permitting the end-users to communicate the data between the LUs in a secure manner.

It is important to note that only the data transmitted via the transmission subsystem between the host computer and the control unit may be protected by cryptography. Data passing between the control unit and its attached terminals (display stations and printers) is not enciphered.

Two types of cryptographic LU-LU sessions may be established: required cryptographic and selective cryptographic sessions. In the first type, all data transmitted between the host computer and the control units is enciphered during the LU-LU session. In the second type, data is enciphered at the option of the application program; thus, enciphering of data can be selected or suppressed by the host LU, but not by the control unit LU.

Establishing Cryptographic Sessions

Before a cryptographic session can be established, the ACF/VTAM Encrypt/Decrypt feature must recognize a request for a cryptographic session and determine the cryptographic capability of the host processor and the control unit. The ACF/VTAM Encrypt/Decrypt feature calls the IBM Programmed Cryptographic Facility Program Product to generate a cryptographic session key in two versions. The first version is enciphered under the host master key and is stored in the host processor. From this first version, the program product produces a second version enciphered under the secondary LU key. The secondary LU key is a key-encrypting key associated with the secondary LU and is used to protect the cryptographic session key during transmission to the secondary LU. The cryptographic session key is used to encipher and decipher data that will be transmitted between the primary and secondary LUs once a cryptographic session has been established.

To establish a cryptographic session, the host processor transmits the enciphered cryptographic session key to the control unit as part of the Bind command. The control unit can decipher the session key, since the secondary LU key is known (having previously been installed in the control unit by a security officer).

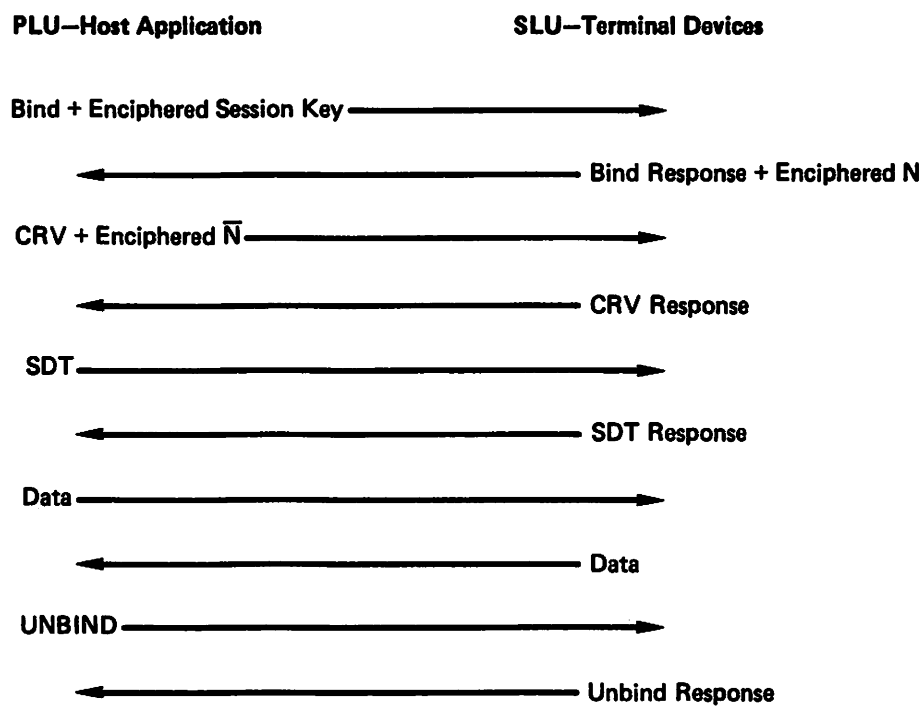
In addition to storing the encrypted session key, the control unit takes part in the following cryptographic protocol:

> A pseudo-random value (N) is encrypted under the just-received session key (KS), and this 8-byte quantity EKS(N) is sent to the host as part of the Bind response.
>
> A valid host will decrypt EKS(N), invert 4 bytes of N, re-encipher the value, and send this 8-byte quantity EKS(N) to the control unit as part of the crypto verification (CRV) command.

The control unit decrypts EKS($\overline{N}$), inverts $\overline{N}$, and compares this value N with the original N. If the values are identical, a positive response is sent to the host, and the conditions of a cryptographic protocol have been met. This cryptographic protocol serves two purposes:

> It verifies that both host and control unit are using the same data-encrypting key (KS).
>
> It validates the host's cryptographic capability, thus preventing an active wire-tapper from using the control unit to decipher captured enciphered data.

The following chart illustrates how the cryptographic protocol fits in with the SNA commands which invoke and terminate a cryptographic session:



Appendix G. Record Formatted Maintenance Statistics (RECFMS) Formats

This appendix describes the formats of the four RECFMS responses the 3276 Control Unit Display Station can send to the host system in response to an REQMS command.

Counters in type 1, 2, and 3 responses do not wrap when they exceed their maximum value; they maintain the maximum value.

The log areas are reset when:

- The 3276 is turned off (types 1, 2, and 3).
- The execution of RECFMS is completed normally as the response to an REQMS with a "RESET" request (types 1, 2, and 3).

REQMS Request Type 1 - Link Test Statistics

Bytes 14, 15 = Number of times the Test command was received.
Bytes 16, 17 = Number of times the Test response was transmitted.

REQMS Request Type 2 - Summary Counters

Byte 14 = Mask bits of the summary counters supported. All supported counters, including those containing zero count, are sent to the host by RECFMS.

- Bit 0 = 1 = Machine Check.
- Bit 1 = 1 = Communication Check.
- Bit 2 = 1 = Program Check.
- Bit 3 - 7 = Reserved.
- Bytes 15, 16 = Reserved.
- Bytes 17, 18 = Machine Check Summary Counter.
- Bytes 19, 20 = Communication Check Summary Counter.
- Bytes 21, 22 = Program Check Summary Counter.

REQMS Request Type 3 - Communication Adapter Data Error Counts

Byte 14 = Adapter Type.
= X'01' = CCA Link Adapter.
= X'02' = (not applicable to the 3276).
= X'03' - X'FF' = Reserved.

Byte 15 = Mask bits of the Communication Adapter Error Counters supported. All supported counters, including those containing zero count, are sent to the host by RECFMS.

- Bit 0 = 1 = Nonproductive Timeout.
- Bit 1 = 1 = Idle Timeout.
- Bit 2 = 1 = Write Retry.
- Bit 3 = 1 = Overrun.
- Bit 4 = 1 = Underrun.
- Bit 5 = 1 = Connection Problem.
- Bit 6 = 1 = FCS Error.
- Bit 7 = 1 = Primary Abort.

Byte 16	=	Mask bits of the Communication Adapter Error Counters supported. All supported counters, including those containing zero count, are sent to the host by RECFMS.
Bit 0 = 1	=	Command Reject.
Bit 1 = 1	=	DCE Error.
Bit 2 = 1	=	Write Timeout.
Bit 3 - 7	=	Reserved.
Byte 17	=	Reserved.
Byte 18	=	Nonproductive Timeout Counter.
Byte 19	=	Idle Timeout Counter.
Byte 20	=	Write Retry Counter.
Byte 21	=	Overrun Counter.
Byte 22	=	Underrun Counter.
Byte 23	=	Connection Problem Counter.
Byte 24	=	FCS Error Counter.
Byte 25	=	Primary Abort Counter.
Byte 26	=	Command Reject Counter.
Byte 27	=	DCE Error Counter.
Byte 28	=	Write Timeout Counter.

REQMS Request Type 5 - 3276 Machine Level Information

Bytes 14–229 = 3276 Machine Level Information.

Bytes 14–205 = ROS chip part number information; each is made up of two 4-byte chip part numbers.

Refer to ROS EC History (SY18-2023) to cross-reference chip PNs and machine EC level.

Byte		Chip PNs
14- 21	=	K2 – Module 1
22- 29	=	K2 – Module 2
30- 37	=	K2 – Module 3
38- 45	=	K2 – Module 4
46- 53	=	J2 – Module 1
54- 61	=	J2 – Module 2
62- 69	=	J2 – Module 3
70- 77	=	J2 – Module 4
78- 85	=	H2 – Module 1
86- 93	=	H2 – Module 2
94-101	=	H2 – Module 3
102-109	=	H2 – Module 4
110-117	=	K2 – Module 5
118-125	=	K2 – Module 6
126-133	=	K2 – Module 7
134-141	=	K2 – Module 8
142-149	=	J2 – Module 5
150-157	=	J2 – Module 6
158-165	=	J2 – Module 7
166-173	=	J2 – Module 8
174-181	=	H2 – Module 5
182-189	=	H2 – Module 6
190-197	=	H2 – Module 7
198-205	=	H2 – Module 8

Note: *Zeros for a chip PN means there is no ROS chip for that position.*

Abbreviations

A

A. Attention
ACK. Positive acknowledge
ACTLU. Activate logical unit
ACTPU. Activate physical unit
ADDCNVRT. Add Converter
ADDR, adr. Address
AID. Attention Identifier
ALPHA. Alphameric
ALT. Alternate
A/N. Alphameric/numeric
APL. A programming language
ASCII. American National Standard Code for Information Interchange
async. Asynchronous
atb, Atb. Attribute
ATT. Attribute
ATTN. Attention

B

B. Busy
BB. Begin bracket
BC. Begin chain
BCC. Block check character
BCEC. Begin chain/end chain
BETB. Between-bracket
BIU. Basic information unit
BOC. Bus-out check
bps. Bits per second
BS. Back space
BSC. Binary synchronous communication

C

C. Column
CA. Character Attribute
CALC. Calculator
CAW. Channel address word
CC. Control check, Chain Command (flag)
CCC. Copy control character
CCW. Channel control word
CD. Change direction
CE. Channel End
char. Character
Chk. Check
cmd. Command
CNCL. Cancel
cnt. Count
COL. Column
CONT. Contention
cps. Characters per second
CPU. Central processing unit
CR. Command Reject, carriage return
CRT. Cathode-ray tube
CRV. Crypto Verification
CSW. Channel status word
ctl. Control
CTS. Clear to Send
CU. Control unit
CUE. Control Unit End
CURSR. Cursor

D

D. Display
DAA. Data access arrangement
DACTLU. Deactivate logical unit
DACTPU. Deactivate physical unit
DAF. Destination address field
DB. Device Busy
DC. Data Check
DE. Device End
Dec. Decimal
DEL. Delete
DES. Data encryption standard
DEV. Device
DFC. Data flow control
DISC. Disconnect
DLE. Data link escape
DM. Disconnect mode
DR. Definite response
DS. Data Set
DSC. Data stream compatibility
DSR. Data Set Ready
DTR. Data Terminal Ready
DUP. Duplicate

E

EAU. Erase All Unprotected
EB. End brackets
EBCDIC. Extended binary-coded decimal interchange code
EC. Equipment Check
ECSA. Extended character set adapter

EFA. Extended Field Attribute

EFI. Expedited flow indicator

EIA. Electronic Industries Association

EM. End of Message

ENP. Enable Presentation

ENQ. Enquiry

EOF. End of Field

EOI. End of Inquiry

EOR. End of Record

EOT. End of Transmission

ERP. Error recovery procedure(s)

ESC. Escape

ETB. End of Transmission Block

ETX. End of Text

EUA. Erase Unprotected to Address

EW, E/W. Erase Write (command)

EWA. Erase Write Alternate (command)

EX (response). Exception

F

FA. Field Attribute

FF. Forms feed, flip-flop

FIC. First in chain

FID. Format identifier

FIE. Function interpret error

FIPS. Federal Information Processing Standard

FIS. First in Segment

FM. Field mark, function management

FMD. Function management data

FRMR. Frame reject

G

GE. Graphic escape

GP. General Poll

H

HDX. Half-duplex

Hex. Hexadecimal

HT. Horizontal tab

Hz. Hertz

I

I (format). Information

IC. Insert Cursor

ID. Identification

ident. Identification

IML. Initial machine load, initial microprogram load

INB. In Bracket

Ind. Indicator

INOP. Inbound Operation Device Characteristic

INP. Inhibit presentation

INS. Insert

IOS. Input/Output Supervisor

IPR. Isolated pacing response

IR. Intervention Required

IRS. Interrecord separator

ITB. End of intermediate transmission block

K

kbd. Keyboard

L

LEN. Length

LF. Line feed

LIC. Last in chain

LIS. Last in segment

LL. Lower left

LM. Left margin

LPI. Lines per inch

LPM. Lines per minute

LR. Lower right

LRC. Longitudinal redundancy check

LU/SSCP. Logical unit/system services control point

LUSTAT. Logical-Unit Status

LUT. Logical Unit Type

M

MAX. Maximum

MAXL. Maximum Length

MC. Machine check

MCL. Multiuse Communication Loop

MDT. Modified data tag

MF. Modify Field

MHS. Magnetic hand scanner

MIC. Middle in chain

MIS. Middle in segment

mm. Millimeter

MOD. Model

MPF. Mapping field

MPL. Maximum Page Length

MPP. Maximum Print Position

MRU. Maximum request/response unit

MSR. Magnetic slot reader

N

NA, N/A. Not applicable

NAK. Negative acknowledge

NC. Not supported

NCCF. Network communication control facility

NCP. Network control program

NDAP. Network determination aid processor

NDM. Normal disconnect mode

neg. Negative

NL. New Line

Nr. Number

NRM. Normal response mode

NS (format). Nonsequenced

NUL. Null

NUM. Numeric

O

OAF. Origin address field

OC. Operation Check

OIC. Only in chain

OIS. Only in segment

OP. Operator, operation

P

P. Printer, protected

PA. Program access, program attention

PC. Pacing count

PEND.BB. Pending Begin Bracket

PF. Program function

PLU. Primary Logical Unit

PN. Part number

POS. Position

PROT. Protect

PS. Programmed Symbols

PSI. Primary to secondary indicator

PT. Program Tab

PU. Physical unit

R

R. Row

RA. Repeat to Address

RB. Read Buffer

RBM. Read Buffer Modified

RCV. Receive

Rd Mod. Read Modified

RECFMS. Record Formatted Maintenance Statistics

REG. Register

Req, REQ. Request

REQMS. Request Maintenance Statistics

RES. Reserved

resp. Response

Rev. Reverse

RH. Request/response header

RM. Read Modified, right margin

RMA. Read Modify All

RNR. Request not ready

RPQ. Read Partition-Query

R/R. Request/response

RR. Request ready

RSP. Response

RTR. Ready to receive

RTS. Request to send

RU. Request/response unit

RVI. Reverse interrupt

S

S (format). Sequenced, supervisory

SA. Selection addressing, Set Attribute

SBA. Set Buffer Address

SC. Session control

SCS. SNA Character String

SDLC. Synchronous data link control

SDT. Start data traffic

SEL. Select

SF. Start Field

SFE. Start Field Extended

SHF. Set Horizontal Format

SI. Suppress Index

SIOF. Start I/O Fast Release

SLD. Set line density

SLU. Secondary logical unit

SM. Status Modifier

SNA. Systems network architecture

SNBU. Switched network backup

SNRM. Set normal response mode

SOH. Start of heading

SOR. Start of record

SP. Space, Specific Poll

SPD. Selector pen detect

S/S. Status and sense

SSCP. System services control point

SSR. Secure string record

STX. Start of text

SUB. Substitute

SVF. Set Vertical Format

sw. Switch

SYN. Synchronous idle

SYS. System

T

TC. Transmission Check

TCU. Transmission control unit

TH. Transmission header

TM. Top margin

TRN. Transparent

TTD. Temporary text delay

U

U. Unprotected

UA. Unnumbered acknowledgment

UC. Unit Check

UE. Unit Exception

UL. Upper left

UNPROT. Unprotect

UR. Upper right

US. Unit Specify

V

V. Volts

VCS. Vertical channel select

VFC. Vertical forms control

VID. Video

VT. Vertical tab

VTAM. Virtual Telecommunications Access Method

W

WACK. Wait before transmit

WCC. Write control character

WRT. Write

WSF. Write Structured Field

X

XID. Exchange Station Identification

XMIT. Transmit

Index

Note: Staples can cause problems with automated mail sorting equipment.
Please use pressure sensitive or other gummed tape to seal this form.

Cut or Fold Along Line

IBM 3270 Information Display System

GA18-2081-1

READER's COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.
IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.
Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name and mailing address:

What is your occupation? ______________________________

Number of latest Technical Newsletter (if any) concerning this publication: ______________

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)

GA18-2081-1

Reader's Comment Form

Fold and tape | **Please Do Not Staple** | Fold and tape

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y. U.S.A.

POSTAGE WILL BE PAID BY ADDRESSEE

INTERNATIONAL BUSINESS MACHINES CORPORATION
DEPARTMENT 825J
1133 WESTCHESTER AVENUE
WHITE PLAINS, NEW YORK 10604 U.S.A.

Fold and tape | **Please Do Not Staple** | Fold and tape

IBM®

Cut or Fold Along Line

IBM 3270 Information Display System (File No. S360/S370/S3/4300/8100-09) Reprinted in Japan GA18-2081-1

IBM 3270 Information Display System

GA18-2081-1

READER's COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name and mailing address:

What is your occupation? ______________________________

Number of latest Technical Newsletter (if any) concerning this publication: ______________________________

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure sensitive or other gummed tape to seal this form.

Cut or Fold Along Line

GA18-2081-1

Cut or Fold Along Line

Reader's Comment Form

Fold and tape

Please Do Not Staple

Fold and tape

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y. U.S.A.

POSTAGE WILL BE PAID BY ADDRESSEE

INTERNATIONAL BUSINESS MACHINES CORPORATION
DEPARTMENT 825J
1133 WESTCHESTER AVENUE
WHITE PLAINS, NEW YORK 10604 U.S.A.

Fold and tape

Please Do Not Staple

Fold and tape

IBM®

IBM 3270 Information Display System (File No. S360/S370/S3/4300/8100-09) Reprinted in Japan GA18-2081-1

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure sensitive or other gummed tape to seal this form.

Cut or Fold Along Line

IBM 3270 Information Display System

GA18-2081-1

READER's COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.
IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.
Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name and mailing address:

What is your occupation? ______________________________

Number of latest Technical Newsletter (if any) concerning this publication: ______________________________

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)

GA18-2081-1

Reader's Comment Form

Fold and tape — **Please Do Not Staple** — Fold and tape

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y. U.S.A.

POSTAGE WILL BE PAID BY ADDRESSEE

INTERNATIONAL BUSINESS MACHINES CORPORATION
DEPARTMENT 825J
1133 WESTCHESTER AVENUE
WHITE PLAINS, NEW YORK 10604 U.S.A.

Fold and tape — **Please Do Not Staple** — Fold and tape

IBM®

Cut or Fold Along Line

IBM 3270 Information Display System (File No. S360/S370/S3/4300/8100-09) Reprinted in Japan GA18-2081-1

IBM 3270 Information Display System

GA18-2081-1

READER's COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. This form may be used to communicate your views about this publication. They will be sent to the author's department for whatever review and action, if any, is deemed appropriate. Comments may be written in your own language; use of English is not required.
IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.
Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Coding Retrieval Legibility

If you wish a reply, give your name and mailing address:

What is your occupation? ______________________________

Number of latest Technical Newsletter (if any) concerning this publication: ______________________________

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments.)

Note: Staples can cause problems with automated mail sorting equipment. Please use pressure sensitive or other gummed tape to seal this form.

Cut or Fold Along Line

Reader's Comment Form

Fold and tape **Please Do Not Staple** Fold and tape

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y. U.S.A.

POSTAGE WILL BE PAID BY ADDRESSEE

INTERNATIONAL BUSINESS MACHINES CORPORATION
DEPARTMENT 825J
1133 WESTCHESTER AVENUE
WHITE PLAINS, NEW YORK 10604 U.S.A.

Fold and tape **Please Do Not Staple** Fold and tape

IBM®

Cut or Fold Along Line

IBM 3270 Information Display System (File No. S360/S370/S3/4300/8100-09) Reprinted in Japan GA18-2081-1

GA18-2081-1

IBM 3270 Information Display System (File No. S360/S370/S3/4300/8100-09) Reprinted in Japan GA18-2081-1