GA23-0060-0
File No. S360/S370/S3/4300-09

# IBM 3270 Information Display System

# 3271 Control Unit
# 3272 Control Unit
# 3275 Display Station
# Description and Programmer's Guide

Systems

IBM

**First Edition (November 1980)**

The material in this publication was formerly contained in the *IBM 3270 Information Display System Component Description,* GA27-2749-10. For the applicable publications for other units of the IBM 3270 Information Display System, see the *IBM 3270 Information Display System Library User's Guide,* GA23-0058.

# Preface

This publication provides management, programmers, and system analysts with detailed reference material relating to the IBM 3270 Information Display System. The 3270 display system comprises the following units:

- IBM 3271 Control Unit Models 1, 2, 11, and 12
- IBM 3272 Control Unit Models 1 and 2
- IBM 3275 Display Station Models 1, 2, 11, and 12
- IBM 3277 Display Station Models 1 and 2
- IBM 3284 Printer Models 1, 2, and 3
- IBM 3286 Printer Models 1 and 2
- IBM 3287 Printer Models 1 and 2
- IBM 3288 Line Printer Model 2

## Organization of This Publication

This publication is divided into eight chapters:

Chapter 1, Introduction, contains a general description of the individual 3270 units and features, and presents local and remote attachment configurations and system concepts.

Chapter 2, Terminal Operations, contains information on data buffering and display image and printout formatting. Display, keyboard, selector pen, printer, and operator identification card reader operations are described in detail.

Chapter 3, Commands and Orders, describes in detail the functions of the commands and orders that can be executed by the 3270.

Chapter 4, Local Operations, outlines the unique operations of locally attached 3270 systems. Described are operations with the channel, selection, command initiation and chaining, status bit definition, and error-recovery procedures.

Chapter 5, Remote Operations–BSC, discusses the unique operations of remotely attached 3270 systems using binary synchronous communication (BSC) line discipline. Described are BSC procedures, the functions and usage of data link control characters, 3270 command, selection, and polling operational sequences (including interaction with the access method and the channel program), remote 3270 command chaining, and error-recovery procedures.

Chapter 6, Remote Operations–SDLC, discusses the operation of remotely attached 3270 systems using synchronous data link control (SDLC) line discipline. This section describes command operation, data transfer, and error-recovery procedures.

Chapter 7, Screen Design, discusses the elements of screen design, field concepts, panel design, data stream coding, and the relationship between data streams.

Chapter 8, Screen Management, discusses the decoding and generating of data streams.

This publication also has six appendixes:

Appendix A, Indicators and Controls

Appendix B, Buffer Address I/O Interface Codes

Appendix C, Katakana Feature

Appendix D, Data Analysis–APL Feature

Appendix E, Abbreviations

Appendix F, Glossary

## Related Publications

This document assumes that the reader has read the following publications, as appropriate:

- *IBM System/360 Principles of Operation,* GA22-6821
- *IBM System/370 Principles of Operation,* GA22-7000
- *General Information-Binary Synchronous Communications,* GA27-3004
- *IBM Synchronous Data Link Control General Information,* GA27-3093
- *IBM 2701 Data Adapter Unit Component Description,* GA22-6824
- *IBM 2703 Transmission Control Component Description,* GA27-2703
- *Introduction to the IBM 3704 and 3705 Communications Controllers,* GA27-3051
- *IBM System/3 Model 10 Components Reference Manual,* GA21-9103
- *IBM System/3 Model 10 Multi-line/Multi-point Binary Synchronous Communications Reference Manual,* GC21-7573
- *IBM Systems Network Architecture General Information,* GA27-3102
- *Virtual Storage Supplement,* GC20-0001, *for IBM System/360 and System/370 Bibliography,* GA22-6822

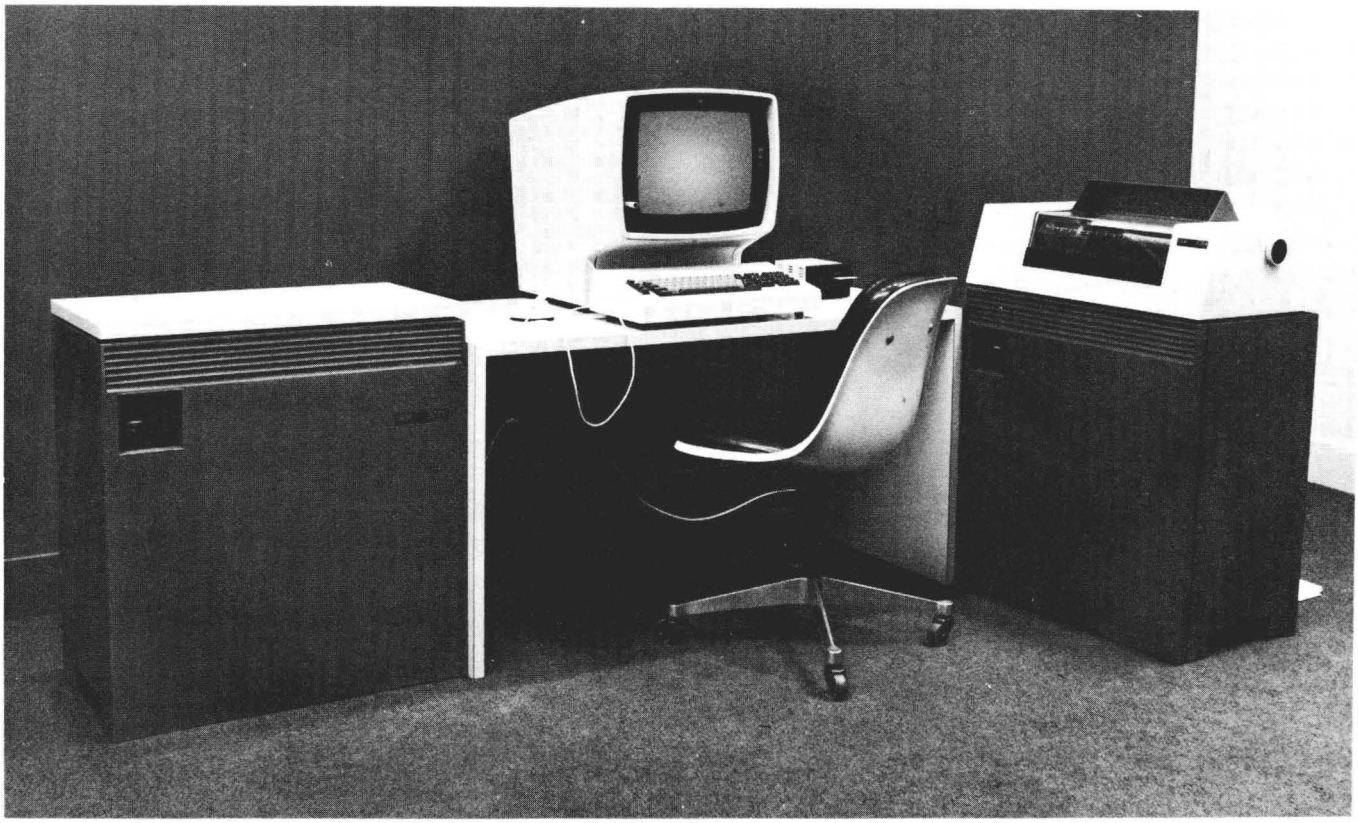
The following publications may also be of interest:

- *An Introduction to the IBM 3270 Information Display System,* GA27-2739
- *Operator's Guide for IBM 3270 Information Display Systems,* GA27-2742
- *IBM 3270 Information Display System Configurator,* GA27-2849
- *IBM 3270 Information Display System: Character Set Reference,* GA27-2837

For a description of all 3270 publications, see the *IBM 3270 Information Display System: Library User's Guide,* GA23-0058

# Contents

## Figures



Frontispiece. IBM 3270 Information Display System: 3271/3272 Control Unit, and Attached 3277 Display Station and 3284/3286 Printer

# Chapter 1. Introduction

An example of an IBM 3270 Information Display System is shown in the frontispiece. The 3270 system offers the user a wide selection of components and configurations. Also available are a large variety of standard and special features that improve performance, provide additional operational capability, and permit expansion of the display system. See the *IBM 3270 Information Display System Configurator,* GA27-2849, for features and configurations.

The 3270 system can attach locally (via cable) or remotely (via common carrier or equivalent facilities) to a host sytem. Remote systems employ binary synchronous communication (BSC) or synchronous data link control (SDLC) line discipline.

## Display System Components

The 3270 Information Display System has three types of components: control unit, display station, and printer.

### *Control Unit*

The control unit provides for the 3270 system's attachment to a data processing system. It directs the operation of up to 32 attached 3270 display stations and printers. Two control units of the 3270 system are described in this publication:

**3271 Control Unit**

- Models 1 and 11 have a 480-character buffer capacity.
- Models 2 and 12 have a 1,920-character buffer capacity.
- Models 1 and 2 attach to a System/360 or System/370 via modems and a BSC data link, and operate with any of the following: an IBM 2701 Data Adapter Unit, an IBM 2703 Transmission Control Unit (TCU), an integrated communication adapter, or an IBM 3705 Communications Controller.
- Models 1 and 2 attach to System/3 Model 10 via a System/3 BSC adapter or to System/3 Model 15 via the local communication adapter.
- Models 11 and 12 attach to a System/370 via modems and an SDLC communication link, and operate in Network Control Program (NCP) mode with an IBM 3704 or 3705 Communications Controller.
- Models 1 and 2 attach to the 4300 Processors via a 2701 Data Adapter Unit or a 3704 or 3705 Communications Controller, and to the 4331 Processor via the communication adapter.

**3272 Control Unit**

- Model 1 has a 480-character buffer capacity.
- Model 2 has a 1,920-character buffer capacity.
- The unit attaches to a System/360 or System/370 via a selector, multiplexer, or block multiplexer channel.
- The unit attaches to the 4300 Processors via a byte multiplexer or block multiplexer channel.

## *Display Station*

The display station provides image display of data transmitted from the data processing unit. A display station with an attached keyboard enables the user to enter, modify, or delete data on the display, and to cause the revised display to be returned to the processing system for storage or additional processing. Two display stations of the 3270 system are described in this publication:

**3275 Display Station**

- Models 1 and 11 have a 480-character buffer capacity.
- Models 2 and 12 have a 1,920-character buffer capacity.
- Models 1 and 2 are stand-alone units that attach to a System/360 or System/370 via modems or data access arrangements (DAAs) and any of the following: an IBM 2701, an IBM 2703, an integrated communication adapter, or an IBM 3705 Communications Controller.
- Models 1 and 2 attach to System/3 Model 10 or 15 via a BSC adapter or the local communication adapter.
- Models 11 and 12 attach to System/370 via modems and an SDLC communication link, and operate in NCP mode with a 3704 or 3705 Communications Controller.

**3277 Display Station**

- Model 1 has a 480-character display image.
- Model 2 has a 1,920-character display image.
- Model 1 attaches to a 3271 Control Unit (all models) or to a 3272 Control Unit Model 1 or 2.
- Model 2 attaches to a 3271 Control Unit Model 2 or 12 or to a 3272 Control Unit Model 2.
- Models 1 and 2 attach to the 3791 Controller (3790 Communication System).

## *Printer*

The printer provides printed copy of data displayed at a display station or of data transmitted from the data processing system. Four printers of the 3270 system are described in this publication:

**3284 Printer**

- Model 1 has a 480-character buffer capacity with a 40-cps print rate.
- Model 1 attaches to a 3271 or a 3272 Control Unit (all models).
- Model 2 has a 1,920-character buffer capacity with a 40-cps print rate.
- Model 2 attaches to a 3271 Model 2 or 12 or to a 3272 Control Unit Model 2.
- Model 3 has no buffer; the print rate is 40 cps.
- Model 3 attaches to a 3275 Display Station (all models).

**3286 Printer**

- Model 1 has a 480-character buffer capacity with a 66-cps print rate.
- Model 1 attaches to a 3271 or a 3272 Control Unit (all models).
- Model 2 has a 1,920-character buffer capacity with a 66-cps print rate.
- Model 2 attaches to a 3271 Model 2 or 12 or to a 3272 Control Unit Model 2.

**3287 Printer**

- Model 1 has a 480- or 1,920-character buffer capacity with an 80-cps maximum print rate and 132 print positions.
- Model 2 has a 480- or 1,920-character buffer capacity with a 120-cps maximum print rate and 132 print positions.
- Both models attach to a 3271 or 3272 Control Unit (all models).

**3288 Line Printer (Model 2 only)**

- Model 2 has a 1,920-character buffer capacity. The average print rate is 120 lines per minute.
- Model 2 attaches to a 3271 Control Unit Model 2 or 12 or a 3272 Control Unit Model 2.

## Display System Configurations

### *Local Attachment*

Locally attached 3270 display systems (Figure 1-1) use a 3272 Control Unit Model 1 or 2. The 3272 Control Unit Model 1 can communicate with up to 32 devices, consisting of Model-1 3277 display stations, Model-1 3284 or 3286 printers, and Model-1 or Model-2 3287 printers. The 3272 Control Unit Model 2 can attach up to 32 devices, consisting of Model-1 or Model-2 3277 display stations, Model-1 or Model-2 3284, 3286, or 3287 printers, and Model-2 3288 Line Printers. At least one display station with a keyboard must be attached to any control unit. The 3272 is attached to a System/360 or System/370 through a block multiplexer, a byte multiplexer, or a selector channel via one of the eight control unit positions on the channel interface. The channel provides the 3272 with data to be displayed and with control information needed to direct the operation of the display station or printer attached to the 3272. Separate buffer storage in the display stations or printers holds digitally coded data for display or printing.

### *Remote Attachment*

Remote attachment differs from local attachment in the medium through which the control unit and the system channel communicate. In a local configuration, the control unit is cabled directly to the system channel. In remote attachment, common-carrier (or equivalent customer) facilities of unlimited length are employed to communicate between the host and the 3270 system.

Two types of remote attachment are available: BSC data link mode and SDLC operating mode. Display data and control information are relayed from the system channel to a control unit by a TCU, an integrated communication adapter, or a communications controller in BSC mode of operation, or by use of a communications controller in SDLC operating mode. Transmission is via modems and common-carrier nonswitched network communication facilities, such as telephone lines, microwave transmission, and satellite, or via switched facilities (BSC mode only). See Figure 1-2.

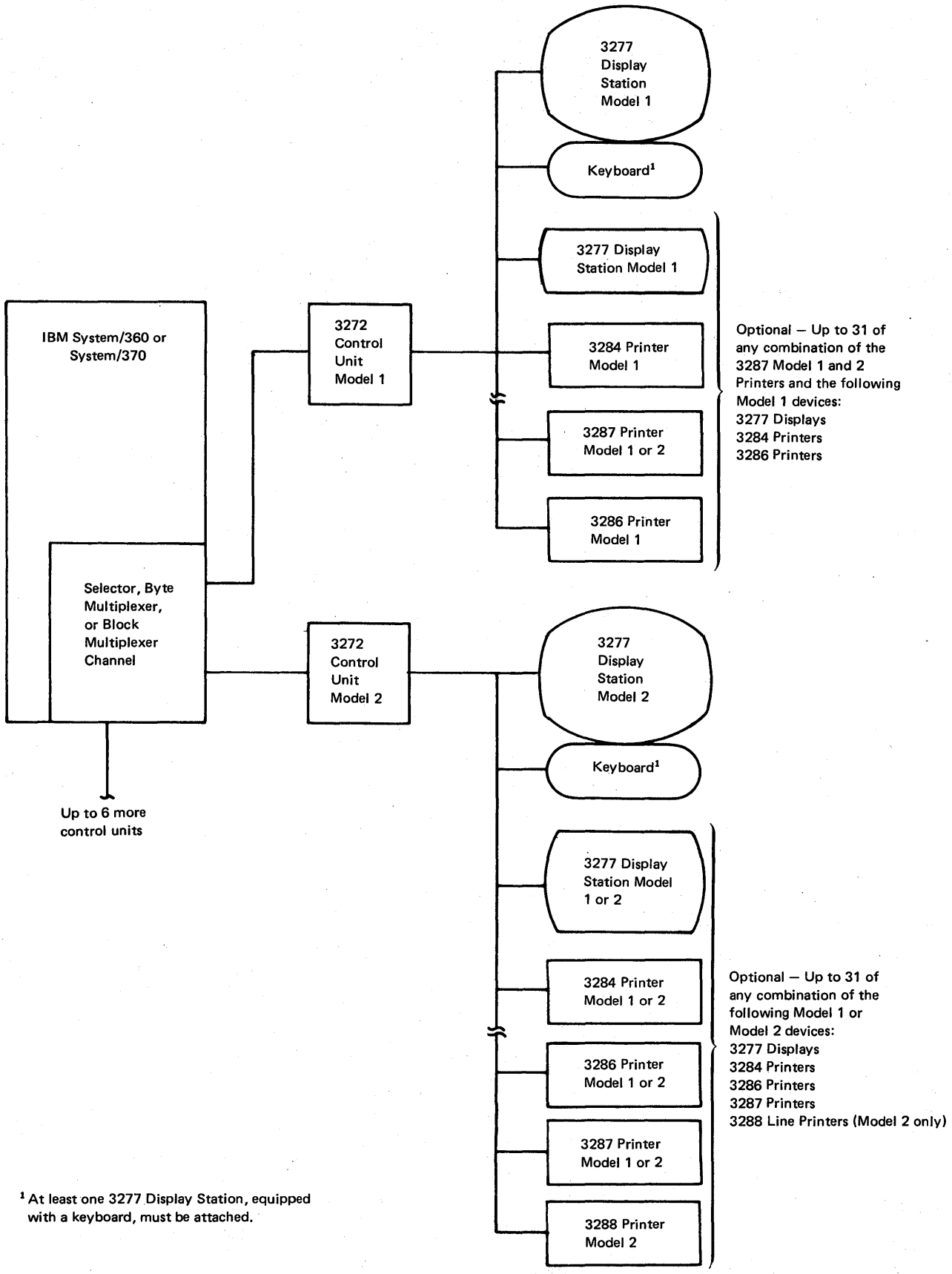
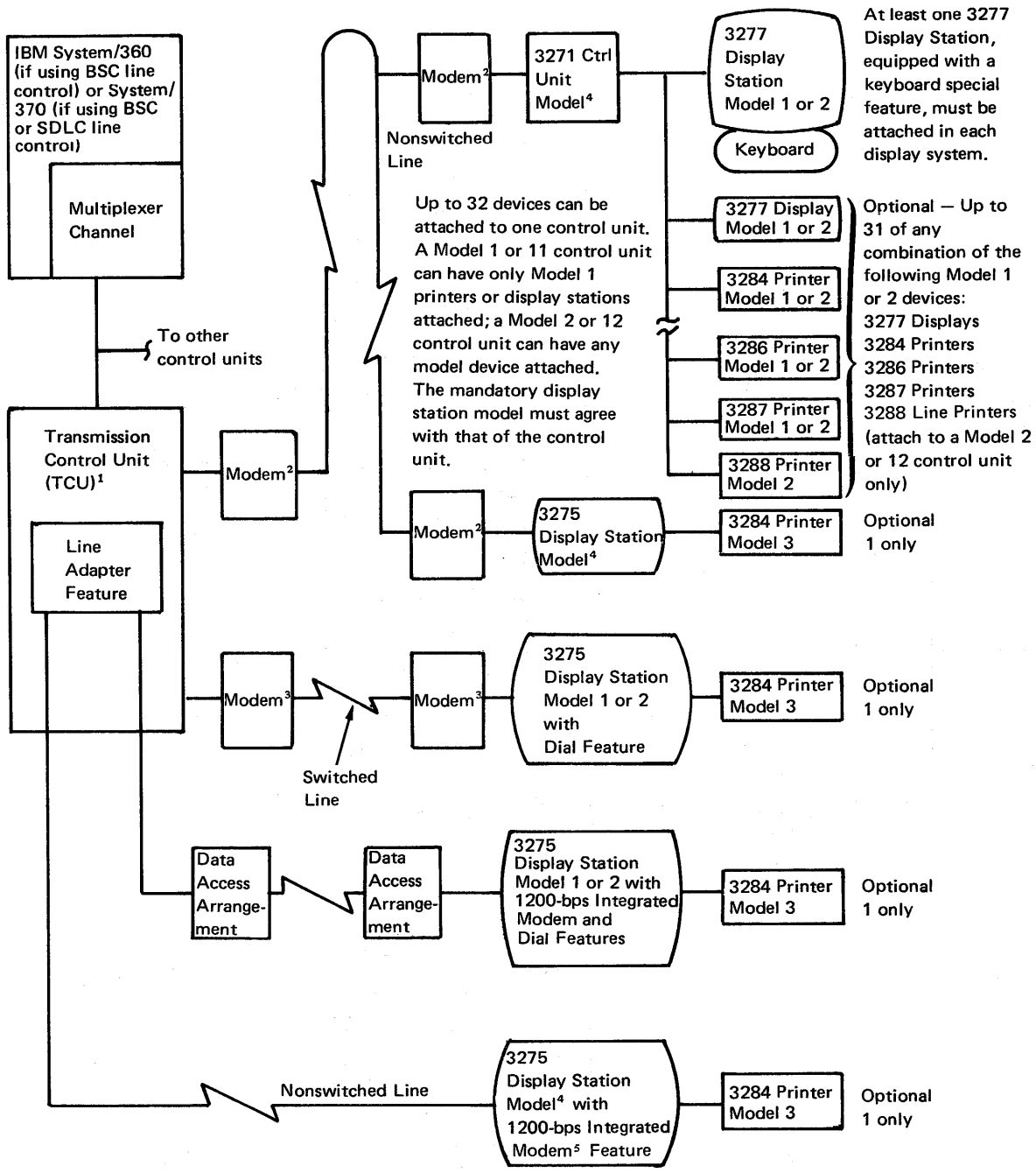



Figure 1-1. Locally Attached 3270 Information Display System




[1] 2701 Data Adapter Unit, 2703 Transmission Control (non-switched with external modem only), 3705 Communications Controller, or equivalent Integrated Communication Adapter. In addition, the 3705 Communications Controller attaches to a selector or block multiplexer channel. In BSC mode, the choice of unit is dependent upon the processing system model, the type of channel, and the communication network selected. SDLC mode of operation requires a 3704 or 3705 Communications Controller.

[2] IBM 3872, 3874, 3875, or 4872 Modems (or equivalent), as required. When switched network backup capability is provided, an IBM 3872, 3874, or 3875 modem is used, with a dial telephone attached, to communicate with the transmission control unit.

[3] 1200-bps operation only.

[4] 3271 Control Unit Model 1 or 2 and 3275 Display Station Model 1 or 2 are required for BSC operation. 3271 Control Unit Model 11 or 12 and 3275 Display Station Model 11 or 12 are required when using the SDLC operating mode.

[5] 1200-bps operation only, on nonswitched line.

**Figure 1-2. Remotely Attached 3270 Information Display System**

### Remotely Attached 3270 Systems Using BSC Operating Mode

A 3271 Control Unit Model 1 or 2 or a 3275 Display Station Model 1 or 2 is used to remotely attach a 3270 system to the teleprocessing network employing BSC operating mode, allowing communication with a host System/360 or System/370. A 2701 Data Adapter Unit, a 2703 Transmission Control Unit, a 3705 Communications Controller, or an equivalent integrated communication adapter, depending upon the host system and channel selected, connects the teleprocessing network to the host system channel.

The 3271 Control Unit Model 1 can attach up to 32 devices, consisting of Model-1 3277 display stations and Model-1 3284 or 3286 printers, or Model-1 or Model-2 3287 printers.

The 3271 Control Unit Model 2 can attach up to 32 devices, consisting of Model-1 or Model-2 3277 display stations, Model-1 or Model-2 3284, 3286, or 3287 printers, and Model-2 3288 line printers. One display station with a keyboard must attach to each control unit. The model number of the display station and that of the control unit must be the same.

The 3275 Display Station Model 1 or 2 provides added convenience for remote locations that require a single display device. The 3275 functions as a control unit and as a display station, and is therefore more economical than a 3271 with a single 3277 attached. The 3275 capabilities can be expanded by attaching a 3284 Printer Model 3 to provide a paper copy of displayed messages. The 3275 can be attached to (multidropped from) the same nonswitched communication line as other 3270 display systems and other IBM products that use the BSC mode of operation, or, with the Dial feature installed, it can be attached by use of a point-to-point common-carrier switched network.

### Remotely Attached 3270 Systems Using SDLC Operating Mode

When employing SDLC line discipline, the 3270 system is remotely attached to a host System/370 via a 3271 Control Unit Model 11 or 12, or a 3275 Display Station Model 11 or 12, over a teleprocessing network. A 3704 or 3705 Communications Controller is required for this configuration. Display data and control information are relayed from the host system channel by the communications controller to the 3271 or 3275 Model 11 or 12 unit, via modems and common-carrier voice-grade lines.

The 3271 Control Unit Model 11 can communicate with up to 32 devices, consisting of Model-1 3277 Display Stations, 3284 or 3286 Model 1 Printers, and 3287 Model 1 or 2 Printers.

The 3271 Control Unit Model 12 can direct the operation of up to 32 Model-1 or Model-2 3277 Display Stations, 3284, 3286, or 3287 Printers, or Model-2 3288 Printers.

At least one display station with a keyboard must attach to a control unit.

The 3275 Display Station Model 11 or 12 does not require a control unit for attachment to a nonswitched line teleprocessing network. When a paper copy of a computer message is desired, a 3284 Model 3 Printer can be attached to the 3275 Display Station. The 3275 Display Station Model 11 or 12 can be attached to the same nonswitched remote communication line as other 3270 systems and other IBM products that use the SDLC mode of operation.

### *Teleprocessing Networks and Modems*

Remotely attached 3270 display systems that use BSC or SDLC line discipline operate in half-duplex transmission mode on half-duplex or full-duplex communication facilities.

The 3271 Model 1 or 2 can attach to a multipoint nonswitched line network. The 3275 Model 1 or 2 can operate in multipoint mode on nonswitched lines or on switched network lines when the Dial feature is installed.

The 3271 and 3275 Models 11 and 12 can attach to multipoint nonswitched line networks. Messages may be simultaneously transmitted and received by the 3704 or 3705 units on full-duplex facilities (duplex-multipoint operation), when two or more SDLC devices are multidropped and attached to the same communications controllers.

IBM modems that can be used in remote systems that employ BSC or SDLC line control (specified in Figure 1-2) are as follows:

- 3872 Model 1 (2,400 bps)
- 3874 Model 1 (4,800 bps)
- 4872 Models 1 and 3 (4,800 bps)
- 3875 Model 1 (7,200 bps)

Switched network backup is a method of replacing a failing nonswitched line with a switched communication system. This capability is available when the IBM 3872 and 3875 Modems are being used. The 3875 operates on nonswitched lines at transmission speeds of 7,200 and 3,600 bps, and on switched lines at speeds of 3,600 and 1,800 bps. The 3872 operates at transmission speeds of 2,400 and 1,200 bps on both nonswitched and switched lines. If an excessively high error rate occurs during operation on a nonswitched line at the maximum transmission speed (7,200 or 2,400 bps), the speed is reduced by one-half at both modems used in the system, and a check is made for a continued high error rate. If the error rate is still high, the display-terminal operator establishes a switched-line connection by dialing the 2701 (or equivalent unit). If the 3872 or 3875 modem was operating at half-speed when the error condition began, the operator establishes the switched-line connection without first changing the transmission speed. The lower line speeds available for dial operation (1,800 or 1,200 bps) may be used if too many errors occur at the higher line speeds.

## Features

No attempt has been made in this publication to catalog all the features available for the 3270 system, although some features are discussed. For details on the availability of various 3270 features, see the *IBM 3270 Information Display System Configurator,* GA27-2849, or discuss the matter with your IBM sales representative.

## System Concepts

The 3271 and 3272 Control Units and the 3275 Display Station control the operations of, and the transfer of data to or from, their attached terminals. See Chapter 2, Terminal Operations, for details.

The 3271, 3272, and 3275 handle all communications with the host system, using the 3270 data stream and the appropriate interface codes.

### *Data Stream*

The 3270 data stream consists of user-provided data, commands, and orders transmitted between the control unit and the host system (Figure 2-1). 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 3270 device buffer. Orders can be included in write data streams, either alone or intermixed with display or print data.

Two types of orders are available. One type is executed as it is received by the control unit. This type is used to position, define, and format data being written into the buffer, to erase selected unprotected data in the buffer, and to reposition the cursor. The second type or order specifies printer format. These orders are initially stored in the buffer as data and are executed only during a print operation.

See Chapter 3 for a detailed description of the 3270 commands and orders associated with the 3270 units described in this publication.

### *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 Display System: Character Set Reference,* GA27-2837, for details. (Chapter 2 contains the U.S. codes, Appendix C contains the Katakana codes, and Appendix D contains the Data Analysis–APL codes.)

### *Local and Remote Operations*

See Chapter 4 for local operations, Chapter 5 for remote BSC operations, and Chapter 6 for remote SDLC operations.

# Chapter 2. Terminal Operations

## Buffer Concepts

Each unit in the 3270 Information Display System (except the 3284 Printer Model 3) has its own buffer for storing data (Figure 2-1).

Buffers are checked to determine whether all characters in the buffers have correct parity. A parity check error occurs when circuitry detects one or more characters with bad parity.

The 3275, as a stand-alone display station, contains its own control unit and executes commands in the same way as the 3271 with one device attached. The 3275 contains one buffer, which it uses both for preparing and for displaying data. When a printout is required at an attached 3284 Printer Model 3 (which has no buffer), the 3275 buffer is used to format and store the printer data.

When not executing a command operation, the 3271 and 3272 control unit hardware continually performs an internal poll of all attached devices. Internal polling is performed to determine what the device status is and whether the device has an input/output (I/O) pending condition.

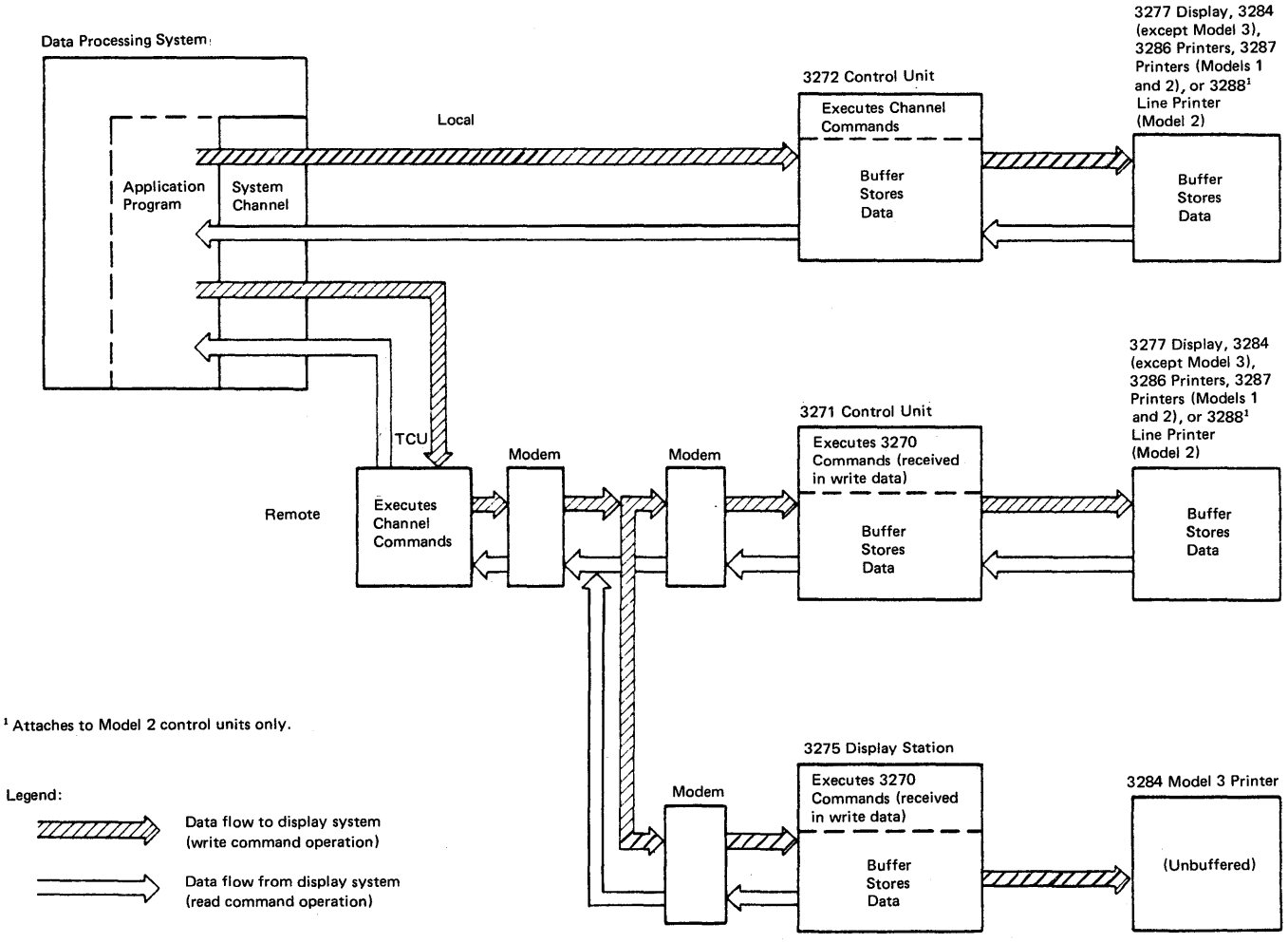



**Figure 2-1. Data Flow between Data Processing System and 3270 Information Display System**

The current status of each device indicates to the control unit whether the device is available, ready, or busy. This information is recorded in the associated device adapter in the control unit.

When an I/O pending condition is detected at a device, polling stops and the control unit communicates solely with that device. When communication is ended, the control unit commences polling at the next sequential device.

In addition, when the program addresses a specific device, the control unit stops the sequential polling and polls the addressed device to obtain its latest status. If conditions permit, the control unit communicates solely with that device until the operation is completed. At that time, sequential polling is resumed.

The 3270 terminal operations are divided into display operations and printer operations.

## Display Operations

This section provides information on the functions and operation of display stations and their associated special features. No distinction is made between the 3277 and 3275 Display Stations, since the units have the same display capabilities. In addition, no distinction is made between various keyboard special features unless they are pertinent to the topic being discussed.

### *Display Images*

Display data that is stored in a display station buffer is presented to the operator on a cathode-ray tube (CRT) screen in the form of alphameric characters and symbols.

When a keyboard is attached, input messages can be generated at the keyboard and displayed on the screen as they are composed.

The image on a 480-character unit is displayed on 12 horizontal rows of 40 characters each (Figure 2-2). The image on a 1,920-character unit is displayed on 24 horizontal rows of 80 characters each.

There is a fixed relationship between each display station buffer storage location and its related character position on the display screen (Figure 2-2). Buffer address locations are referenced from 0 (the first displayable character location in the upper-left corner of the screen) to 479 or 1,919 (the last displayable character location in the bottom-right corner of the screen). Figure 2-3 shows the layouts of these address locations for display buffers of both sizes. By using these address locations under appropriate commands, a program can load a display station buffer with many combinations of control and data characters to present to the operator a display image that exactly fits the application. A total of 93 character codes may be transferred from the sytem processor and stored in the display station buffer. These include the uppercase and lowercase alphameric characters and special characters shown in Figures 2-5 and 2-6. They include printer control characters EM, NL, and FF (which is displayed as < for the 3288 Line Printer) and two selector-pen-detectable control characters (? and >), also shown in Figures 2-5 and 2-6. They also include attribute characters described below under "Display Fields."

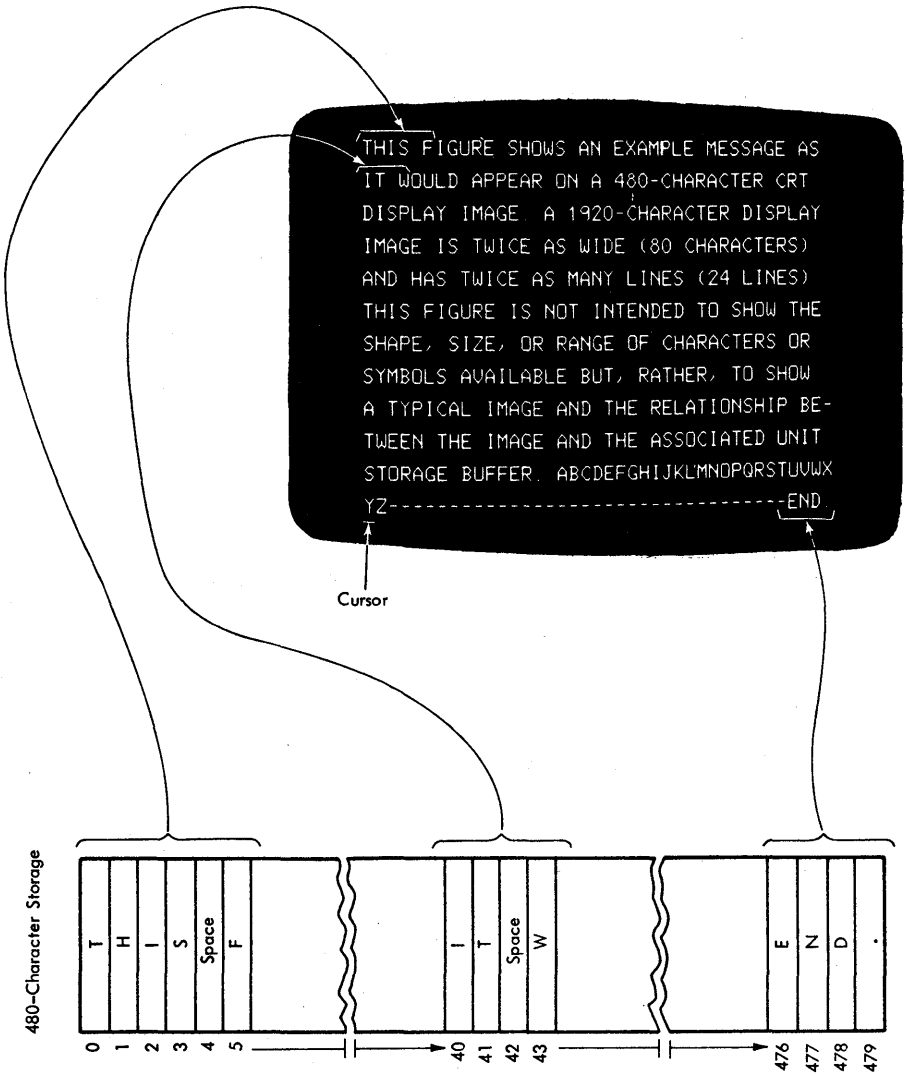
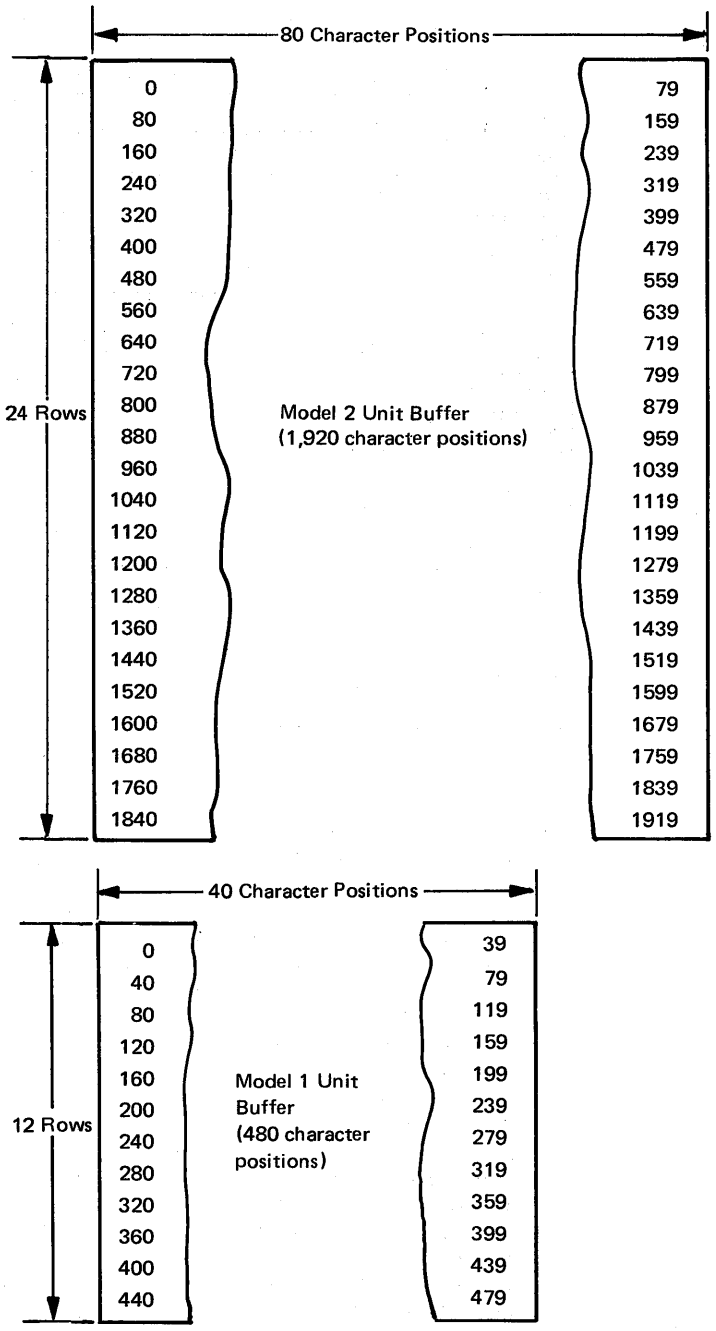



**Figure 2-2. Relationship between Display Station Buffer and Character Position on Display Image Screen (Unformatted Display)**

## Unformatted and Formatted Display Images

An application program can communicate with a display operator by one of two basic methods. In one method, the display screen is left unformatted and the display operator uses the screen in a free-form manner. In the second method, the display image is completely or partially formatted (organized or arranged) by the application program.

The display image shown in Figure 2-4 illustrates the flexibility available with 3270 display image formatting. In this example, the visible characters represent displayed data stored in the display buffer. Character positions indicated by dotted squares represent buffer locations where control characters are stored. Dotted characters represent display data that is defined by the program as not displayable, that is , not visible to the operator. In all display images, control characters stored in a display unit buffer are not displayed; data characters may or may not be displayed, depending upon program definition.




**Note:** *See Appendix B for hexadecimal equivalents.*

**Figure 2-3. Buffer Addressing Layouts for Model 1 and Model 2 Devices**

NAME : JOHN B DOE
SALARY 12525
JOB TITLE : WRITER
PHONE #: 383-7628

**Figure 2-4. Examples of Display Image Fields (Formatted Display)**

| 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 | 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 | | | | | | | | | h | q | y | | H | Q | Y | 8 | |
| 1001 | 9 | | EM | | | | | | | i | r | z | | I | R | Z | 9 | |
| 1010 | A | | | | | ¢ | ! | ¦ | : | | | | | | | | | |
| 1011 | B | | | | | . | $ | , | # | | | | | | | | | |
| 1100 | C | FF | DUP | | RA | < | * | % | @ | | | | | | | | | |
| 1101 | D | | SF | | | ( | ) | _ | ' | | | | | | | | | |
| 1110 | E | | FM | | | + | ; | > | = | | | | | | | | | |
| 1111 | F | | | | SUB | \| | ¬ | ? | " | | | | | | | | | |

**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 not specified. The character displayed by the 3277 or 3275 for a given undefined character code may be different for other devices. IBM reserves the right to change at any time the character displayed for an undefined character code.*
2. *Lowercase alphabetic characters (shown within the dotted outlined area) are converted to uppercase by the display station or printer and displayed or printed as uppercase characters.*
3. *NL, EM, FF, DUP, and FM control characters are displayed or printed as 5 9 < * and ; characters, respectively, except by printers under format control, in which case NL and EM do not result in the printing of a character, and by printers successfully executing FF, in which case < is not printed.*
4. *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 2-7.*
5. *This table also applies for Belgian, French, and Italian mono-case I/O interface codes and graphics.*
6. *The ¦ character (hex 6A) is not displayed and is printed by the 3287 and 3288 only.*
7. *For BSC data-link control characters, see Chapter 5.*
8. *See Appendix C for Katakana EBCDIC codes.*
9. *See Appendix D for Data Analysis – APL codes.*
10. *See* IBM 3270 Information Display System: Character Set Reference, *GA27-2837, for all interface codes for the 3270 system.*

**Figure 2-5. United States I/O Interface Code – EBCDIC**

| Bits 4321 | Hex 1 | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |
|---|---|---|---|---|---|---|---|---|---|
| | | 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 | SUB | * | : | J | Z | j | z |
| 1011 | B | | | + | ; | K | [ | k | |
| 1100 | C | FF | DUP | , | < | L | \ | l | |
| 1101 | D | | SF | – | = | M | ] | m | |
| 1110 | E | | FM | . | > | N | ^ | n | |
| 1111 | F | | | / | ? | O | _ | o | |

Bits 7, 6, 5 ← (column header)

Hex 0 ← (row header)

**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 and the I/O interface code returned on a subsequent read operation are not specified. The character displayed or printed by these terminals for a given undefined character code may be different for other terminals. 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. *Lowercase alphabetic characters (shown within the dotted outlined area) are converted to uppercase by the display station or printer and displayed or printed as uppercase characters.*
3. *NL, EM, FF, DUP, and FM control characters are displayed or printed as 5 9 < * and ; characters, respectively, except by printers under format control, in which case NL and EM do not result in the printing of a character, and by printers successfully executing FF, in which case < is not printed.*
4. *AID, attribute, write control (WCC), copy control (CCC), CU and device address, buffer address, sense, and status characters are assigned as specified in Figure 2-7 so that each character can be represented by a graphic character within the solid outlined portion of this chart.*
5. *ASCII A option displays and prints | and ¬ for interface codes 21 and 5E (hex), respectively. ASCII B option displays and prints ! and ∧ for codes 21 and 5E (hex), respectively.*
6. *The FF control character (0C) is returned to the host during a subsequent read operation as 46.*
7. *For BSC data-link control characters, see Chapter 5.*

Figure 2-6. United States I/O Interface Code – ASCII

## Display Fields

The control characters (dotted squares) shown in Figure 2-4 are constructed by the program. They define the characteristics or attributes of the data that follow them and are called *attribute characters*. Each attribute character plus all the data following it up to the next attribute character is called a *field*. When a field "wraps" the screen, the field continues from the last character location in the buffer to the first location in the buffer until it is terminated by an attribute character. Figure 2-4 shows eight fields. Figures 2-5, 2-6, and 2-7 show the United States I/O interface codes used.

Organizing the display data into fields facilitates display operations for the program and for the operator. Fields are also used in most 3270 programming operations: functions that involve the storage, display, printing, or transmission of data are primarily field-oriented. Some operations performed on fields that wrap the screen are terminated by the last buffer address rather than by the field terminating attribute. This effect is noted in the descriptions of the specific operations.

Attribute characters, in addition to defining the start of a field, define the following field characteristics for all character locations contained in that field:

- Protected (from modification by a display operator) or unprotected (available for the operator to modify or enter data). The unprotected definition classifies a field as an input field.
- Alphameric (an input field in which an operator can enter alphabetic, numeric, or symbol characters) or numeric (has special meaning for protected fields, data entry keyboards, and the Numeric Lock special feature).
- Character display (nondisplay, display, intensified display).
- Detectable or nondetectable (by use of the selector pen).
- Tab stop positions (first character position of unprotected fields).

Each attribute character occupies one of the 480- or 1,920-character locations in the buffer, but it cannot be displayed or printed. During a display or a printout, its character location appears as a space. Figure 2-8 shows the bit definition for an attribute character.

Attribute characters are treated as characters that are protected from operator intervention; that is, they cannot be replaced by alphameric characters entered from the keyboard or modified by use of the selector pen. However, the modified data tab (MDT) bit (7) of the attribute character can be changed by an operator, as described in Figure 2-8. Also, attribute characters are not protected from being overwritten by alphameric data that is included in the data stream of a Write or Erase Write command. When the operator uses the CLEAR key, attribute characters and all characters in a formatted buffer are erased. See Chapter 7 for details of screen design. See Chapter 8 for screen management.

**Programming Note:** *Refer to "Selector Pen Operations" for use of intensified field attributes when formatting selector-pen-detectable fields.*

| 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. 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. See Note 5 of Figure 2-6 for ASCII A and B graphic character difference for ASCII codes 21 and 5E (hex).*

*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 2-7. Control Character I/O Codes

Attribute-Character Bit Assignments

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

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

**Note:** *Bits 0 and 1 are not decoded when received by the 3270. When characters are transferred to the CPU, bit 1 is a 1 and bit 0 is set (as shown in Figure 2-7), depending upon the character being transferred. All attribute characters are part of the defined character set. The default option (bits 2 through 7 all set to 0) results in an unprotected, alphameric, displayed, nondetectable field.*

**Figure 2-8. Attribute-Character Bit Definition**

## *Keyboard Operations*

Keyboards, which can be attached to a 3277 or 3275, 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 displayed on the subsequent display regeneration cycle.

When the operator completes an operation and presses the ENTER key, an I/O pending interruption occurs. In local operations, this causes an interruption to inform the program; the program may then read the modified data fields from the display buffer. In remote operations, an interruption cannot be generated; instead, the modified data fields are read automatically in response to a Poll sequence.

## Cursor

A special symbol (that resembles an underscore), called a *cursor*, is displayed beneath a character or character position on the display screen to indicate where the next character entered from the keyboard will be stored (Figure 2-2). For example, when the cursor is displayed under one character in a line of characters, that character can be changed or deleted by keyboard action. Also, if the cursor is displayed under a position without a display character, a character can be inserted in that position by keyboard action. All these operations, when performed on a formatted display, cause the MDT bit (7) of the attribute character for the field to be set to 1. However, when the cursor appears beneath a character in a protected field or an attribute character, that position cannot be modified by keyboard action, and the MDT bit is not set.

One, and only one, cursor is always displayed on the display. 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 Key Lock special feature; it is displayed even when positioned in a nondisplay/nonprint field and when the Key Lock special feature (if installed) is turned off.

## Keyboards

Four types of keyboards are available for the 3277 and 3275 Display Stations: typewriter, data entry, data entry–keypunch layout, and operator console keyboard. All keyboards have special symbol keys and control keys for entering data. The type of keyboard determines the characters and symbols that can be key-entered from the display station, but does not determine which type of characters and symbols can be transmitted from the system for the display image.

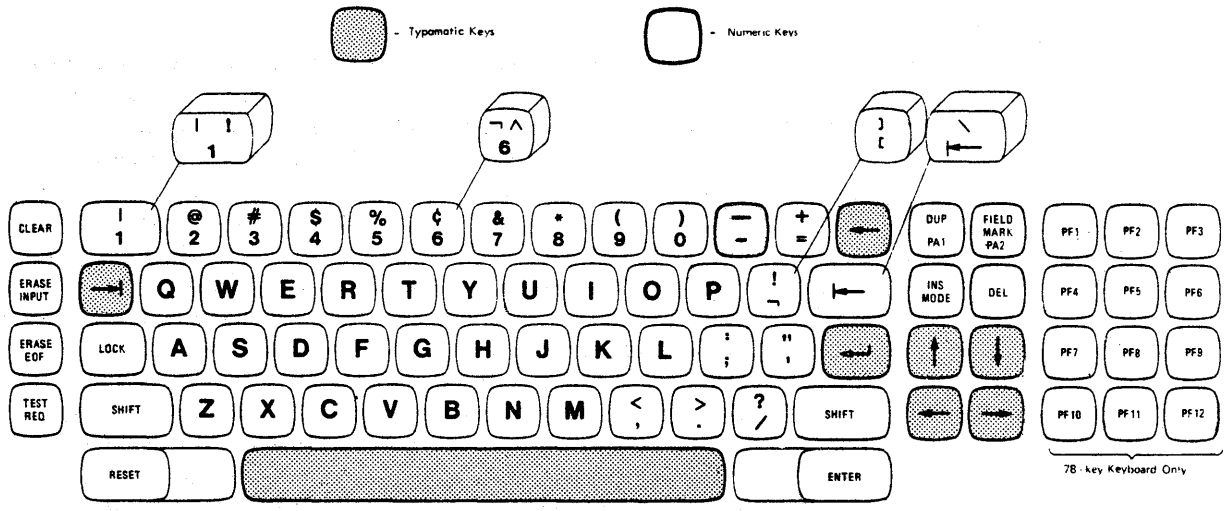
Variations between keyboards include 66-key and 78-key versions. The 66-key keyboard provides all the basic operator keys. The 78-key keyboard provides expanded operator-to-program message flexibility with 12 additional keys that may be defined to fit the requirements of the application program. The four basic types of keyboards, shown in Figure 2-9, are defined below. Refer to the *IBM 3270 Information Display System: Character Set Reference,* GA27-2837, for World Trade (WT) keyboard key layouts and nomenclature.

**Typewriter Keyboard:** This keyboard provides the basic typewriter key layout. Alphameric keys are encoded with both lowercase and uppercase codes. The typewriter keyboard is available with program-function keys PF1 through PF12 (78-key version) or without (66-key version).

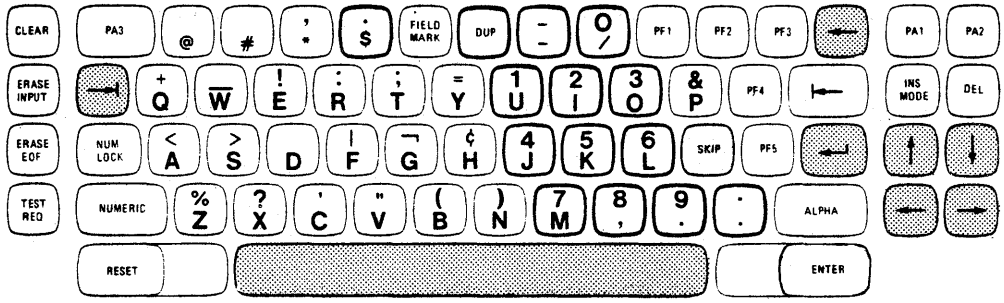
**Data Entry Keyboard:** This keyboard provides the basic data-entry type of key layout. When characters are entered in a numeric field, the keyboard is automatically upshifted to take advantage of the grouped numeric keys (bold-outlined in Figure 2-9). The data entry keyboard contains 66 keys, including program-function keys PF1 through PF5.

**Data Entry Keyboard–Keyboard Layout:** This keyboard has the same keys and features as the data entry keyboard. The key layout of this keyboard more closely resembles the layout of the 29 Card Punch and 129 Card Data Recorder. In many cases the layout is identical with that of the keypunch units except for function-key designations. This keyboard is recommended for data entry applications.

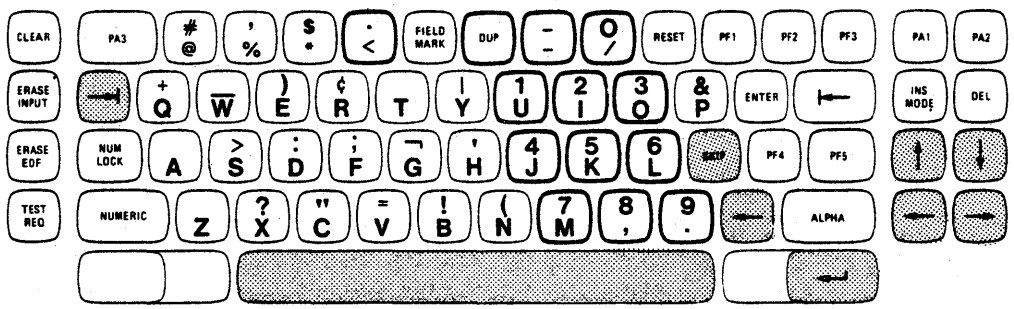
**Operator Console Keyboard:** This keyboard provides an IBM 1052 Model 7 type of key layout. It has 78 keys, which include program-function keys PF1 through PF12.



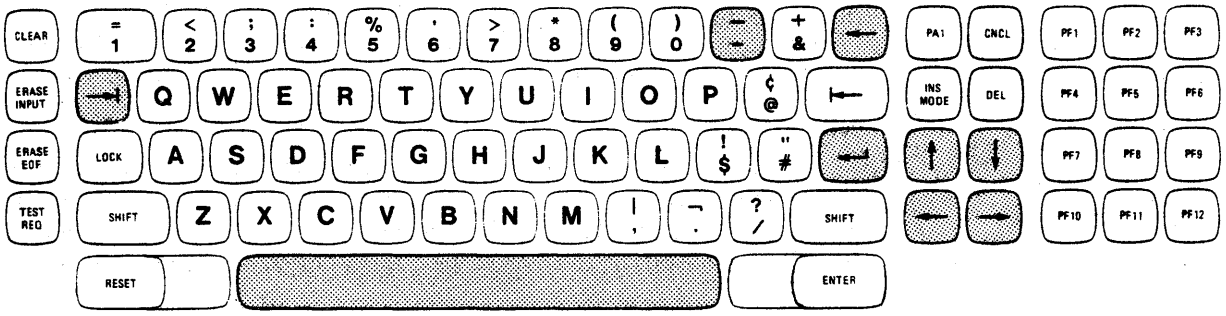

**Typewriter Keyboard (EBCDIC) — The ASCII typewriter keyboard, which accommodates both ASCII-A and ASCII-B character set options, has four different keys, shown above keyboard.**




**Data Entry**




**Data Entry — Keypunch Layout**




**Operator Console**

Figure 2-9. Basic Keyboards for 3277 and 3275 Display Stations

Alphameric character keys encompass the complete 63-character EBCDIC and 64-character ASCII character sets (as shown within the bold outline in Figures 2-5 and 2-6, respectively), including Space.

Alphabetic characters can be entered into the display buffer in either uppercase or lowercase code, depending upon the position of the SHIFT key, from the typewriter or operator console keyboard. Only uppercase alphabetic codes can be entered from the data entry keyboards. All alphabetic characters in the buffer (uppercase or lowercase codes) are displayed as uppercase characters.

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.

**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 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 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: ↑ (up), ↓ (down), →(right), and ←(left). A fifth key, the backspace key, occupies its normal position on the main keyboard. It is also designated by ← and 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 attribute character locations, through the use of these keys. Operation of these keys does not affect the MDT bit.

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. 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.

⇤ ***(Backtab) Key:*** When the cursor is located in the attribute character 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 Backtab key has no 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.

***SKIP Key (Data Entry Keyboards Only):*** Performs the same functions as the Tab key.

**ERASE EOF (Erase to End of Field):** 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 in that field, to nulls. The operation can wrap from the end of the last line on the display to the beginning of the top line. 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 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.

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 and the cursor is repositioned to location 0.

**INS (Insert) MODE Key:** This key lights the INSERT MODE indicator and places the keyboard controls in an insert mode of operation, regardless of the cursor location.

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) are 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 disables the keyboard. Attribute characters remain in their fixed character locations and are not shifted as part of the insert operation.

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.

Operating an alphameric key in insert mode when the cursor is located in an 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.

Operation of the RESET key returns the keyboard to normal mode.

**DEL (Delete) Key:** If the cursor is located in an alphameric character in an unprotected field, operation of the DEL key deletes the character from the character location occupied by the cursor and sets the MDT bit to 1 (if it has not previously been 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. 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.

Operating this key when the cursor is located in an 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 a *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.

**DUP (Duplicate) Key:** Operation of this key causes a unique character code to be entered into the display buffer, a standard Tab key operation to be performed, and the MDT bit to be set to 1. The DUP key is provided only on the typewriter, data entry, and data entry–keypunch layout keyboards. 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 2-5 and 2-6) when the data is read from the display to the program. No duplicate operation is performed at the 3270 control unit. The DUP character, when stored in a device buffer, is displayed or printed as an asterisk (*).

Operation of this key when the cursor is located in an 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.

**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 (Figures 2-5 and 2-6) when the data is read from the display to the program. The field mark character, when stored in a device buffer, is displayed or printed as a semicolon (;). The field mark is not provided on operator console type keyboards.

Operating this key when the cursor is located in an 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.

**Program Attention Keys:** These keys solicit program action by causing an I/O pending to occur at the device. The program is notified of the interruption by an Attention status indication in locally attached systems and by response to a poll in remotely attached systems. In remotely attached systems that are using a 3275 Display Station, the display screen momentarily goes blank while the program accepts and responds to the attention signal. An attention identification (AID) character is generated at the time of the interruption to identify which key caused the interruption, but the MDT bit is not affected. The program-attention keys are CLEAR, ENTER, CNCL (cancel), TEST REQ, all program-function (PF) keys, and the program-access (PA) keys. Operation of the CLEAR key also clears the entire display buffer to nulls and positions the cursor to character location 0.

Operation of any program-attention key disables the keyboard, lights the INPUT INHIBITED indicator, and extinguishes the SYSTEM AVAILABLE indicator.

**Note:** *Not all program-attention keys are available on each type of keyboard. See Figure 2-9.*

### Numeric Lock Special Feature Operation

When the Numeric Lock special feature is installed, the operator may enter the characters 0-9, the period (.), the minus sign (-), and DUP in a field identified in the attribute byte as numeric and unprotected. Operation of any other key that can enter a displayable character lights the INPUT INHIBITED indicator and disables all keys except the RESET key. Operation of the RESET key enables the keyboard (if disabled) and extinguishes the INPUT INHIBITED light. The nondisplay/nonprint attribute bits 4 and 5 and MDT bit 7 operate normally.

On a data entry or data entry–keypunch layout keyboard (Figure 2-9), the Numeric Lock special feature is disabled while the ALPHA or NUMERIC key is operated.

On a typewriter or operator console keyboard, the characters that can be entered in the field identified in the attribute byte as numeric and unprotected are 0-9, the period (.), and the minus sign (-); in addition, on typewriter keyboards when the SHIFT or the LOCK key is operated, the DUP character may be entered by the operator. It is not possible to disable the Numeric Lock special feature for entry of other displayable characters.

**Note:** *On Austrian/German, Belgian, Danish, French, Italian, Norwegian, Portuguese, and Spanish keyboards with the Numeric Lock feature installed, the comma (,) replaces the period (.) as a valid numeric character.*

### Keyboard Disabled (INPUT INHIBITED Indicator On)

When INPUT INHIBITED is on, the keyboard and other input devices are disabled. In cases caused by operator key action, the input-inhibited condition can be cleared by use of the RESET key unless one of the following conditions coexists:

- A command is being executed for a device to which the keyboard is attached.
- A card is being read at the operator identification card reader.
- The 3284 Printer Model 3 is printing.
- A parity error or cursor check is detected in a device buffer. (The INPUT INHIBITED indicator remains off as long as the RESET key is pressed, but turns on when the RESET key is released.)
- The security keylock is in the off position. (Turning on the security keylock clears this condition.)

The conditions that can be cleared by use of the RESET key are as follows:

- A program-attention key operation preceded initiation of a command for a device with an attached keyboard.
- A selector-pen attention operation preceded initiation of a command for a device with an attached keyboard.
- The operator initiated the input-inhibited condition by pressing an alphameric key not included in the numeric key grouping when the Numeric Lock special feature was installed.
- The operator tried to change the data displayed in a protected display field. (The CLEAR key can also be used in this case, which places nulls in all buffer positions and turns on the INPUT INHIBITED indicator. INPUT INHIBITED can then be turned off by pressing the RESET key prior to initiation of a command for a device with an attached keyboard.)

### *Indicators and Controls*

See Appendix A for the functions of indicators and controls.

### *Selector-Pen Operations*

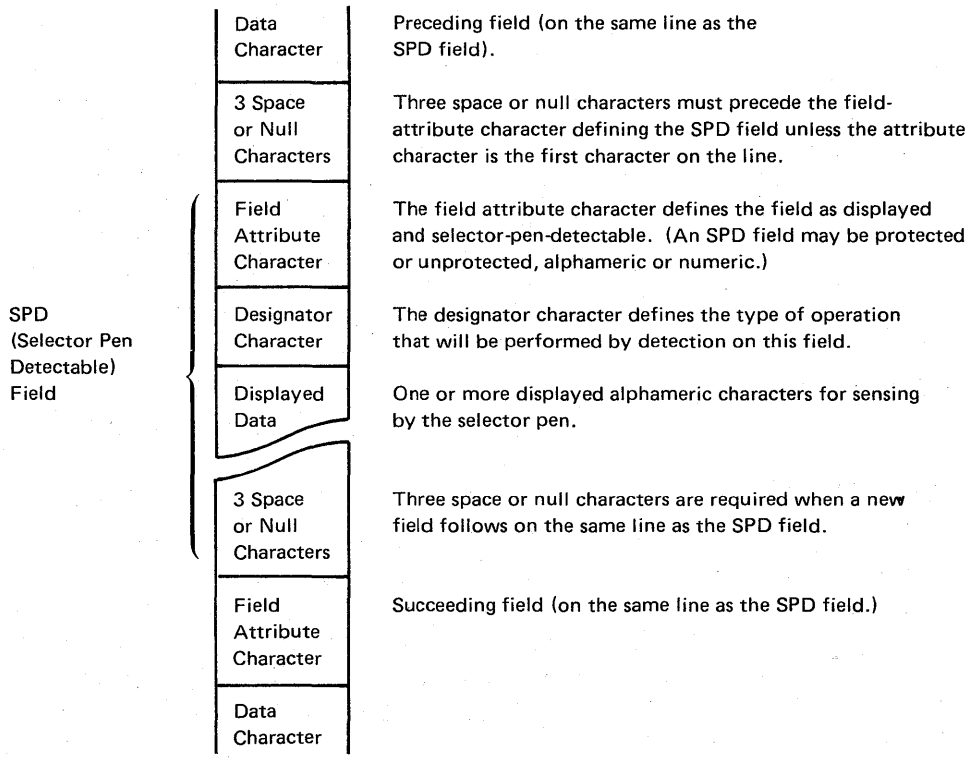
The selector pen, shown in Figure 2-10, is a light-sensitive pen that can detect the light emitted from characters displayed on the 3275 or 3277. With the selector pen, the operator can select from a list or table of displayed items and can then cause those selections to be identified to the application program.

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


Figure 2-10. Selector Pen

### Selector-Pen Field Format

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




**Note:** *On 3275 displays, selector-pen operation resets the same input-inhibited condition as the RESET key. After reset occurs, the functions are executed.*

The attribute character, the designator character, and displayed alphameric characters must be on the same line. If the field extends beyond one line, the selector pen can detect only those characters on the same line as the attribute character. A maximum of 6 detectable fields in the 3277 or 3275 Model 1, or 12 detectable fields in the 3277 or 3275 Model 2, may precede the last detectable field on any given line. When detectable and nondetectable fields are mixed, a maximum of 14 mixed fields on both Model 1 and Model 2 3277 and 3275 units may precede the last detectable field on any given line. In this situation, therefore, a nondetectable field could be one character long.

### Designator Characters

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

The selection field is defined by a question mark (?) designator character. When the selector pen detects on a selection field, the MDT bit in the attribute character for that field is set (1) in the display buffer. In addition, the designator character is automatically changed on the screen to a greater than (>) sign to indicate 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 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-pen operation has been completed. The program may then issue a Read Modified command to obtain the address of each field that the operator selected or modified.

**Programming Notes:**

1. *The application programmer should be aware that high-intensity/unprotected fields can be modified by the display station operator to become selector-pen-detectable fields.*
2. *Use of the selector pen feature is expected 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-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-pen-detect input with keyboard input must use the keyboard to generate the I/O pending. Use of the selector pen 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 2-11 shows a sample display with fields defined for selector-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 will occur and the program will read the locations of the four selected fields.

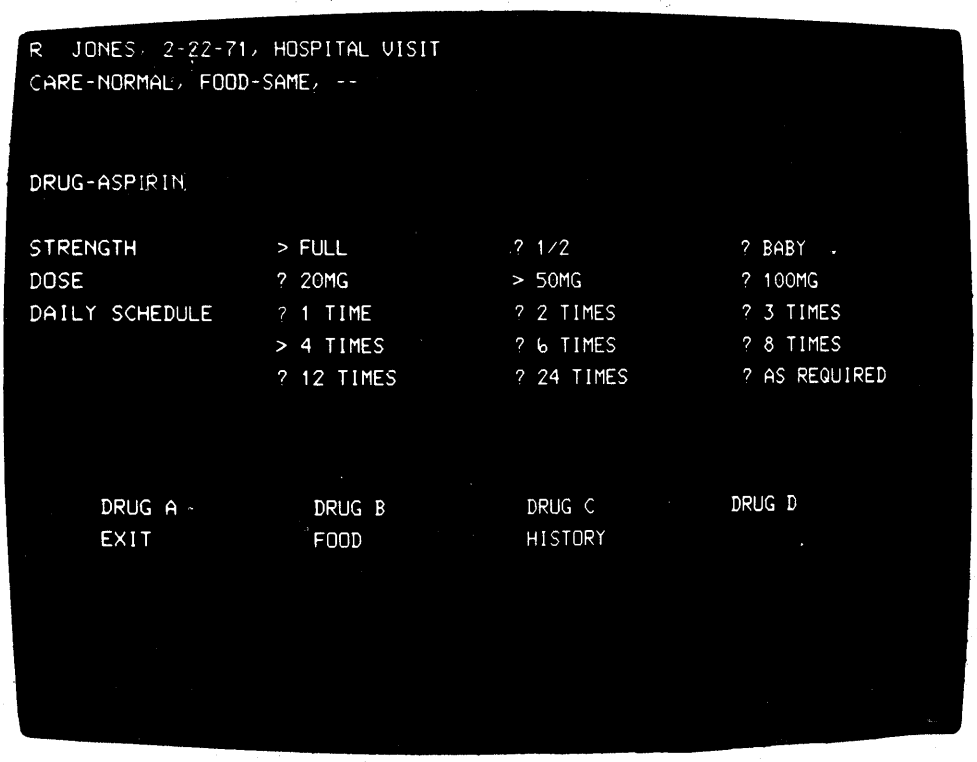



**Figure 2-11. Sample Display Screen for Selector-Pen Operations**

### *Security Keylock*

The security keylock is a security-enhancement special feature that provides a key-controlled lock for 3275 and 3277 displays. When the key is in the off position or is removed from the display station, the message buffer is locked, preventing the 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-pen operations.

Programmed attempts to access display stations that have the key turned off or removed from the lock cause the 3270 devices to return responses to the central processing unit (CPU). 3270 responses are device- and operation-dependent. They are summarized in the following table:

| Device Attachment | Operation | Response |
|---|---|---|
| 3272 Models 1 and 2 | All | UC, IR Status and Sense |
| 3271 (all models) | Specific Poll | IR Status and Sense |
| | General Poll | EOT |
| | Selection Addressing Sequence | RVI |
| 3275 (all models) | Specific Poll<br>General Poll | No response (timeout)<br>EOT |
| | Selection Addressing Sequence | No response (timeout) |

**Programming Note:** *When no response is received from a 3275 after a specific poll or selection-addressing sequence, a general poll should be issued. An end of transmission (EOT) response to the general poll indicates that the 3275 buffer is locked.*

### *Operator Identification Card Reader*

The operator identification card reader (OICR), which is attached by a cable to a 3275 or 3277 (Figure 2-12), reads data, such as a unique operator ID number, encoded on a magnetic-striped card. As the card is inserted into the reader, the ID number is read from the magnetic stripe and written into the display buffer, in nondisplay mode and at the location specified by the cursor. The ID number, therefore, is not displayed on the screen. An I/O pending is generated at the display to inform the program that the ID number can be retrieved and transferred to main storage.

With the 10-character set, plus control characters (Figure 2-13), the maximum number of characters that can be read is 40 characters at 3 bits per millimeter (75 bits per inch). This number includes the start of record (SOR), longitudinal redundancy check (LRC), and either end of record (EOR) or end of inquiry (EOI) characters.

With the 10-character set shown in Figure 2-13, 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 systems network architecture (SNA) mode (LU-LU session only, *not SSCP-LU session*) or in a non-SNA mode.

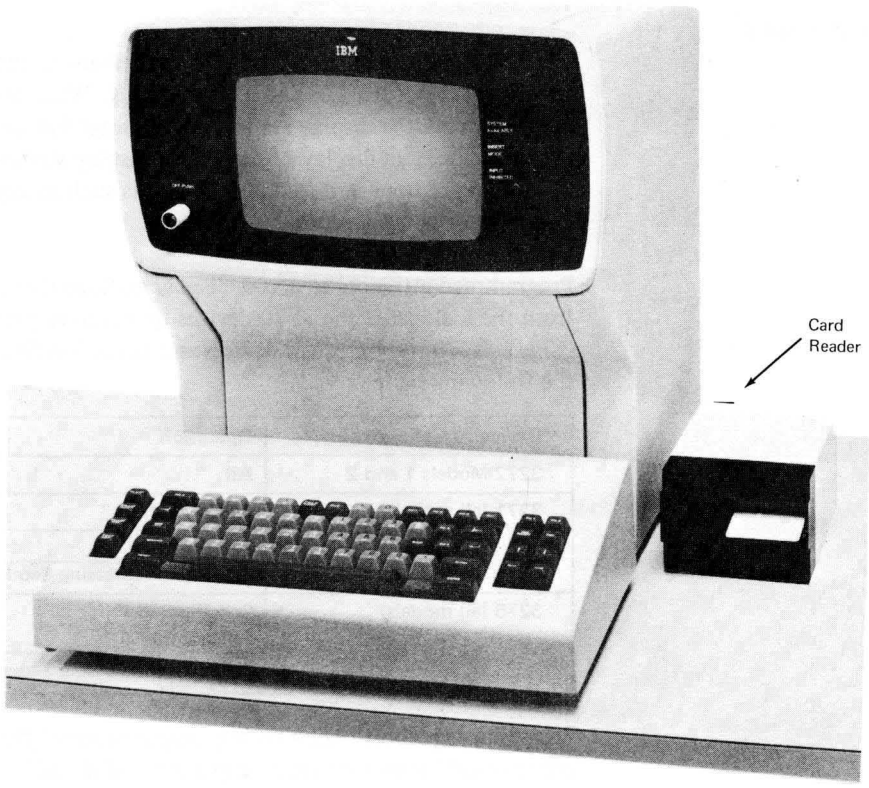



Figure 2-12. Operator Identification Card Reader

| Character | | Bit Pattern $2^0$ | $2^1$ | $2^2$ | $2^3$ | P | Hex Code | I/O Interface Code (Note 5) 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 |
| | SOR (Note 2) | 1 | 1 | 0 | 1 | 0 | B | 7B | 23 |
| | EOI (Note 3) | 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 |
| | EOR (Note 4) | 1 | 1 | 1 | 1 | 1 | F | 7F | 22 |

**Notes:**

1. *This character is reserved for operator identification only and must be located in the first data character position.*
2. *Start of Record*
3. *End of Inquiry. May also be used as a termination character on the operator identification card reader.*
4. *End of Record*
5. *Programmers use only the four least-significant bits of the hex codes.*

Figure 2-13. 10-Character Set Used with Operator Identification Card Reader

### 10-Character Set

The 10-character set shown in Figure 2-13 comprises 10 numeric characters, 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. An 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 2-14.

When the SOR 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 uprotected 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.

### Operational Differences Due to Screen Format

When the 10-character set is being used with the OICR, 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 or fields) is being used, the operation of the display results in an inbound data stream, as shown in Figure 2-15.

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.

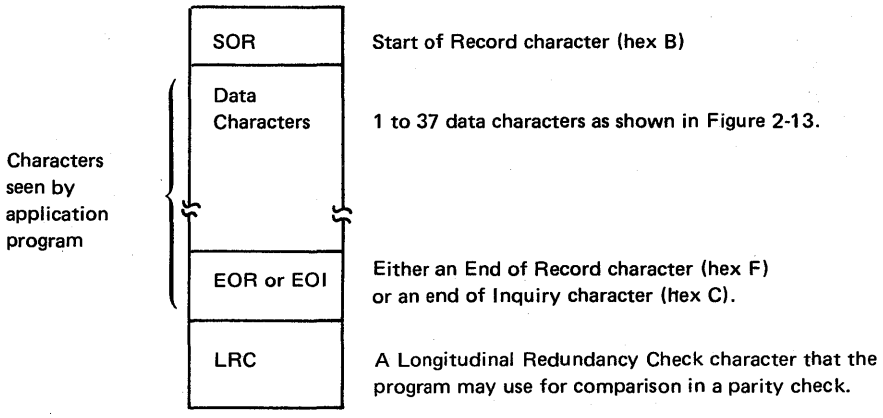
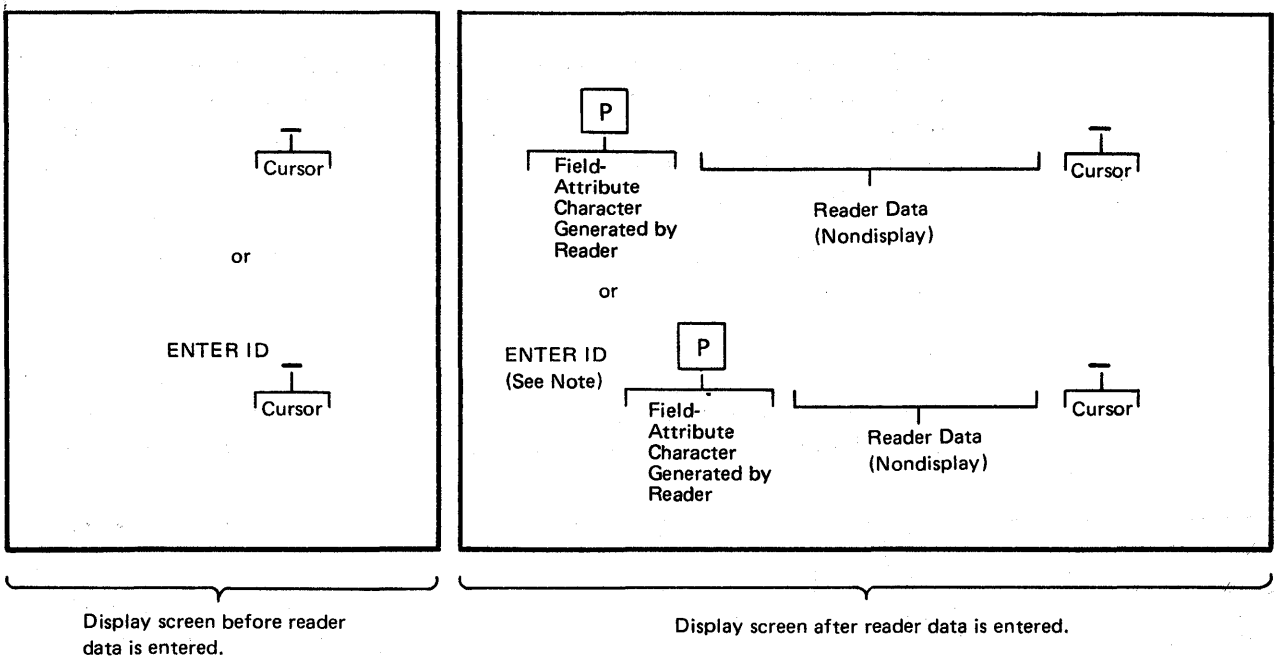



**Figure 2-14. Magnetic-Stripe Format (10-Character Set)**




**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

| | |
|---|---|
| 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.

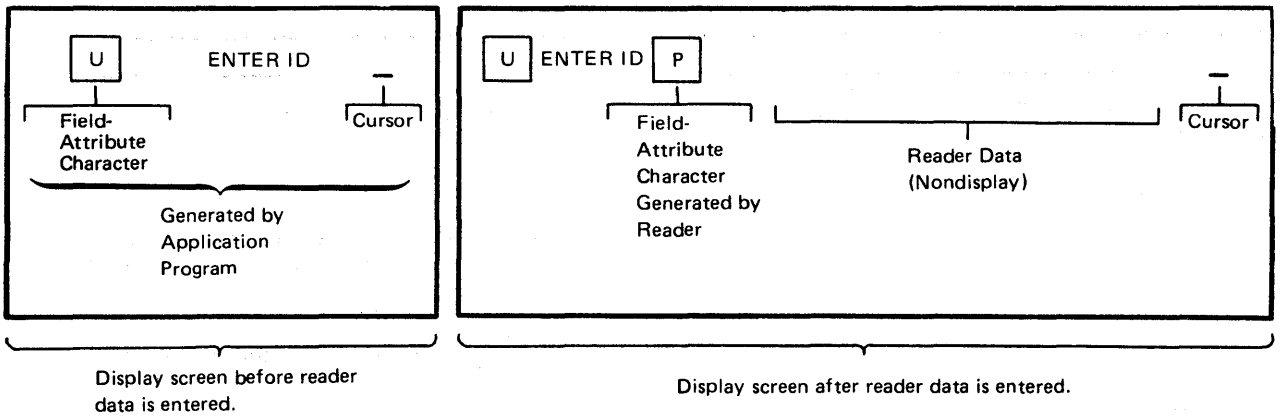
**Figure 2-15. Operation of the Display with an Unformatted Screen (Using 10-Character Set)**

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 the information has been written into the display buffer. In addition, 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 OICR 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 2-16.




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). |

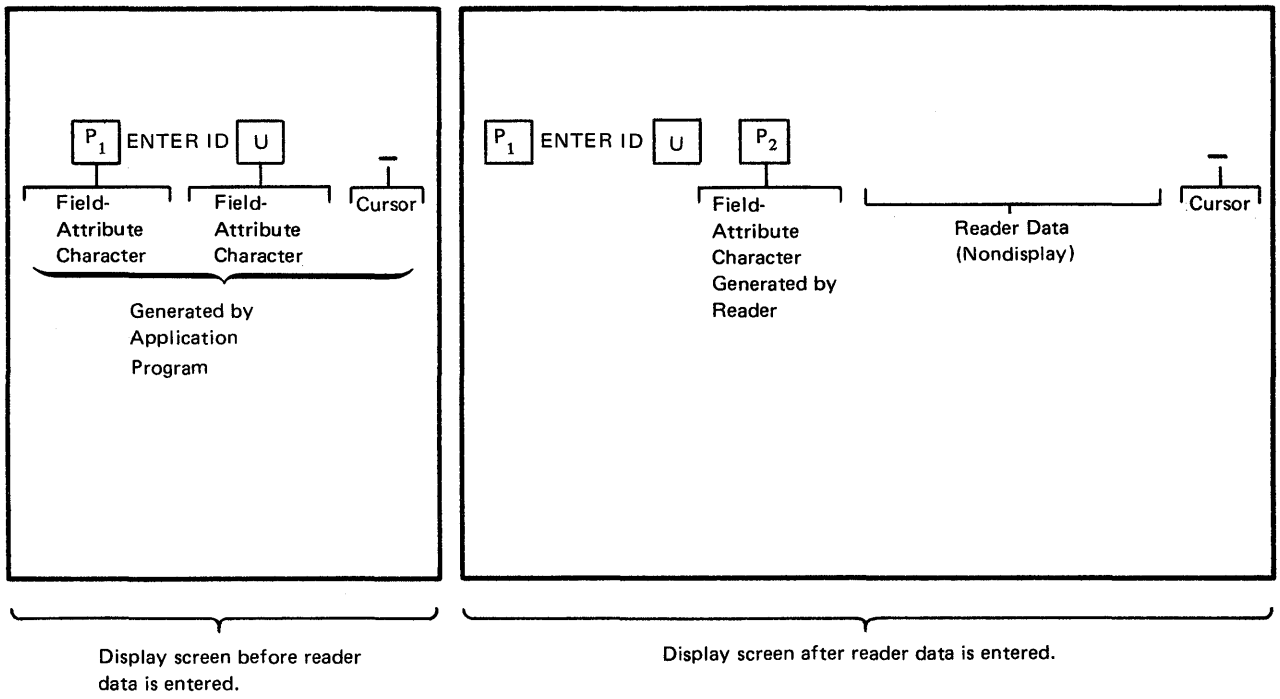
U = Unprotected field-attribute character

P = Protected field-attribute character

**Figure 2-16. Operation of the Display with a Formatted Screen (Using 10-Character-Set), Example 1**

### Example 2

When the OICR 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 2-17.




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 | Address 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 2-17. Operation of the Display with a Formatted Screen (Using 10-Character Set), Example 2**

**Error Conditions**

If any of the following error conditions exists when the magnetic stripe is read, OICR data will not be written into the display buffer:

- The SOR 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.

**Programming Notes:** *The proper use of the OICR 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 display, 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 EOI/EOR character. Absence of this character means that the reader data has not been transferred successfully. This condition can occur under the following error conditions:*
   a. *Normal data flow from the reader has been interrupted.*
   b. *The cursor has been moved to a field-attribute character location. This means that the field defined for reader input is too small or that 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 EOI/EOR 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 four bits are of concern. The application program should set up a 1-byte field initialized to X'0B'. This is the SOR 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 EOR/EOI 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 EOR/EOI, 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. In addition, 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 is delivered with each OICR and is available for system validation. The test card data (in 4-bit code) written into the display buffer is as follows:*

   BB123456789012345678901234567895 5ABDEF7

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

## Printer Operations

Printers for the 3270 display system are used to provide a printed copy (for future reference) of information that is displayed at a 3277 or 3275, or of information written from the program. Printed data appears in the same alphameric characters and symbols that appear on a display image, and printouts can be formatted in the same manner as a display image. Cursor information is ignored by the printer.

Two types of printers are available, a buffered printer and an unbuffered printer. The buffered printer, with its own buffer and a unique device address, can be attached to a 3271 or 3272 and operates in the same manner as a 3277. The buffered printer includes the 3284, 3286, and 3287 Models 1 and 2 and the 3288 Model 2 Line Printer.

The unbuffered printer is the 3284 Model 3, which is provided for attachment to the 3275. The relationship between the 3284, 3286, 3287, or 3288 printer buffer or the 3275 Display Station buffer and a printout is shown in Figure 2-18.

### *Print Line Formatting*

Printout operations are specified by a Write command or a Copy command (3271 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 simply define the print line length: 40, 64, or 80 character positions per line. If a format is not specified, the print line length is determined by platen length on 3284 and 3286 printers, while the line length is 132 character positions on the 3287 and 3288 printers.

**Programming Note:** *To duplicate the copy function when operating with the 3272 local CU, the display buffer must be read and then written to the printer.*

### *NL, EM, and FF Printer Orders*

New Line (NL), End of Message (EM), and Forms Feed (FF) printer orders are transferred as part of the data stream from the application program. They are stored in the buffer as data.

The NL order is executed only when encountered in a print field during 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 carriage length), the printer automatically performs a line feed and continues printing. NL orders are not executed when located in a nondisplay/nonprint field; instead, they are treated as alphameric characters and printed as spaces. In addition, they are not executed when they are encountered in a print field during a printout that uses a line-length format; instead, they are printed as the graphic 5.

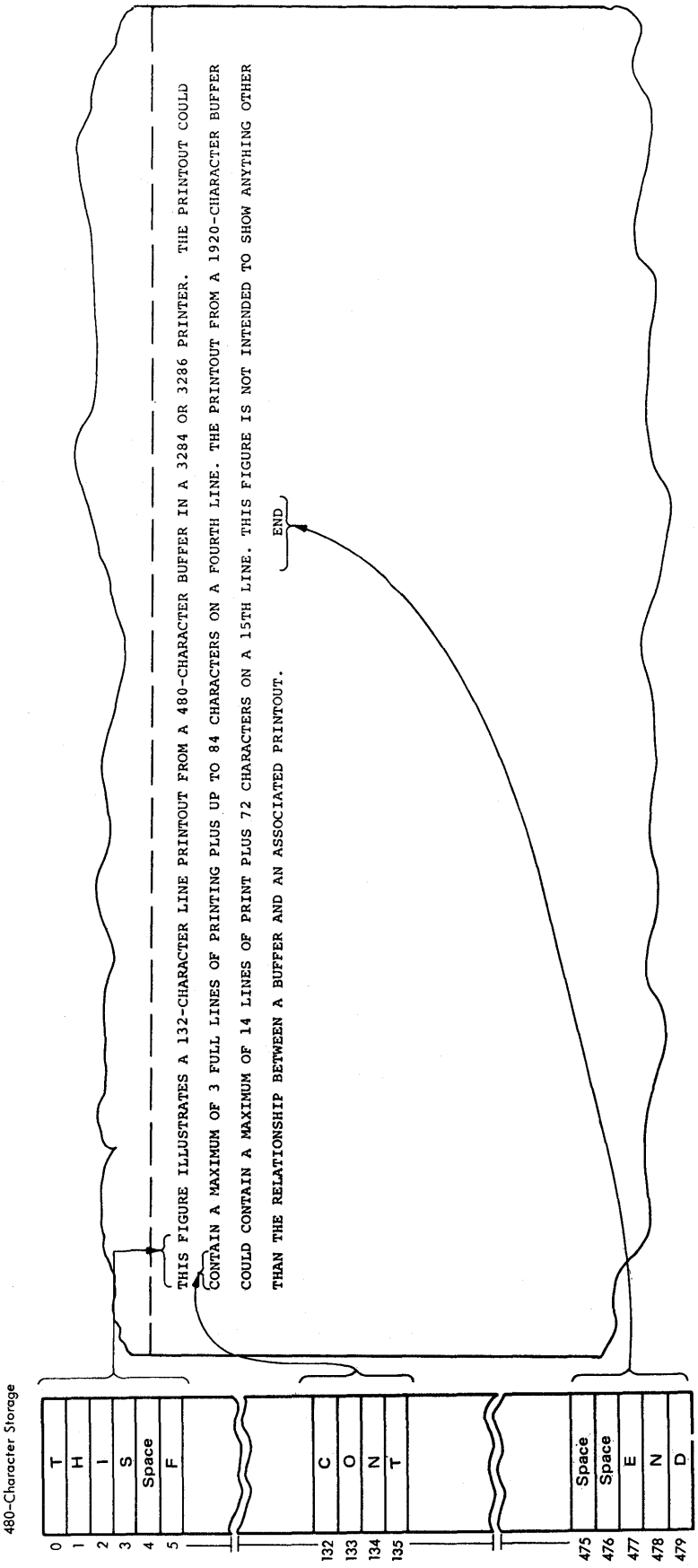



**Figure 2-18. Relationship between Buffer Data and Printer Data**

For buffered printer operation (described under the heading "Buffered Printer Operation"), EM orders are executed only when they are encountered in a print field during a printout that does not have line-length format specified. EM orders are not executed when they are located in a nondisplay/nonprint field. They are treated as alphameric characters and printed as spaces. When encountered in a print field of a printout that uses line-length format, they are not executed; instead, they are printed as a graphic 9. For unbuffered printer operation (described under the heading "Unbuffered Printer Operation"), EM orders are executed when encountered, whether or not line-length format is specified. When an EM is encountered, the printing operation is terminated. None of the data following the EM order in the buffer is printed.

Valid FF orders are executed by the 3287 and 3288 printers during printouts, both with and without a line-length format specified. (The FF order is described under the heading "Page Length Control/VFC Operations.") 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.

### *Buffered Printer Operations*

When a command specifying a printout is received from the system, the contents of the addressed printer are transferred to the 3271 or 3272 buffer, where they are modified and transferred back to the printer. The printout starts after the transfer from the control unit to the printer buffer is completed.

During the print operation, if line format is specified, 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 the 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.

When line-length format is not specified, 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.

### *Unbuffered Printer Operations*

Attachment of an unbuffered printer to a 3275 does not affect operations between the 3275 and the system. However, when a printout is being executed, the 3275 will be busy to all other request command operations.

When a command specifying a printout is received from the system, the 3275 transfers its printer data to the printer. As characters are transferred to the printer, display regeneration continues and the cursor advances on the display screen by one position with each character transferred.

Data is not scanned before printout. Attribute characters, null characters, and alphameric characters in nondisplay/nonprint fields are transferred as spaces. When these characters constitute an entire line, that line will be printed as spaces and a blank line will appear in the printout. The print operation is terminated with the printing of the last buffer position, unless an EM order is encountered first.

The NL order is executed only when the line-length format is not specified. Whether or not line-length format is specified, EM orders are executed when encountered.

At the end of each printout, a final line feed is executed so that the printer is ready to start the next printout, except when the print terminating EM order appears in the first print position of the print line, in which case a final line feed is not executed, because the printer is already positioned at the left margin for the next printout.

### *Page Length Control/VFC 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 3287 Printer and by the vertical forms control (VFC) specify feature for the 3288. Special inks and preprinted forms containing index marks are not required to make this feature operational.

When a valid 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 write control character (WCC) in a write, erase/write, or erase/write alternate data stream to the printer or (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 "<" character, except when the FF order is located in a nonprint field. The "<" character prints during either formatted or unformatted printouts. When an FF order is sent to a 3288 that does not have the VFC feature installed, or if the skip operation is not executed, the FF character is printed as a "<" character. A valid FF order prints as a space character.

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 3288, the skip operation is executed, and a blank form results. The 3287 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 printers will stop printing and skip to line 2 of the next form.*

Before beginning page-length-control/VFC 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 page-length-control/VFC circuitry synchronizes with the skip stop line on the form as the cover is closed and the printer goes from not ready to ready. If the cover must be raised or if a not-ready condition occurs, the form must be checked to ensure that the skip stop line is in the proper position before the cover is reclosed.

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/VFC 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 lines per inch for the 3288; 6 or 8 lines per inch for the 3287 printers. (For example, when 11-inch forms are installed on the 3288, the switches should be set at 66.)

**Programming Notes:**

1. *If an NL order and an FF order appear on the last line of a 3288 printout and VFC is installed, FF is suppressed and the printer will not skip a full form. If this condition occurs on a 3287 printer, subsequent printing will begin on a new form.*
2. *The page-length-control function on the 3287 printer is synchronized when power is applied or when the FF switch is pressed.*
3. *If a 3287 or 3288 buffer containing FF characters (hex'0C') is read back by the program, the FF characters are returned to the program as '8C' (EBCDIC hex) or '46' (ASCII hex). This is a hardware function of the control units and should not be mistaken for a printer error.*

## *Error Conditions*

Four error conditions may be encountered at both the buffered and unbuffered printers. In each of the following cases, when an error is detected, the program is notified. (Power should *never* be removed from unbuffered printers during a printout; the error conditions that may be returned to the program are unpredictable if this is done.)

### 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). When a 3284 or 3286 printer mechanism experiences a "printer hang" condition (see Appendix F, Glossary) during a printout, the printer stays busy with an equipment check (EC) present. For 15 seconds, the mechanism automatically attempts to recover. If the recovery attempt is successful, the printer returns to the ready condition. If the recovery attempt is not successful after 15 seconds, the printer becomes not ready, as indicated by Intervention Required (IR) status.

If a printer 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 again each time a printout is initiated.

When the 3287 detects other than parity errors, the Check indicator lights, and the associated error code is displayed in the 2-digit Status indicator. The operator may be able to correct the error and continue operation.

**Character Generator or Sync Check Errors**

The characters printed by a buffered or unbuffered printer are a function of the character generator or character belt installed. When an incorrectly formed character is printed during a printout (not the 3287), no attempt is made to replace or alter the character. When the printout operation is completed, a new line function is executed and an X is printed (feature-dependent). A sync check error occurs when a character belt hammer is out of sync.

**Parity Error**

If a parity error is detected on a character about to be printed, the graphic X (3284, 3286, 3288) or an error graphic (*prx10T,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 and 3288) serves to indicate the detection of the parity error.

**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.

### *Unit and Model-Dependent Differences*

The following differences between 3270 units affect printer operations.

**Buffer Size**

The buffer size of the 3284, 3286, and 3288 is model-dependent. Model 1 units contain 480 characters, and Model 2 units contain 1,920 characters.

The 3287 buffer size is specified as 480 or 1,920 bytes.

During an erase/write operation to a 3284, 3286, or 3288, the full 480- or 1,920-character buffer is erased. When an Erase/Write command is sent to the 3287, the buffer is erased up to the specified size (480 or 1,920 characters).

A data or attribute wrap operation to buffer position 0 occurs when data characters are addressed beyond the effective end of the buffer. The last effective position in the buffer is the default buffer size when operating in SNA/SDLC protocol.

**Uppercase and Lowercase Printouts**

The 3284, 3286, and 3288 print uppercase alphabetic characters unless the Extended Character Set feature is installed (which provides additional characters, including lowercase).

Printouts in either uppercase or lowercase characters may be obtained from the 3287 printer, depending upon the setting of the 3287 Change Case switch, and the command or print operation in process. During execution of an Erase/Write or Erase/Write Alternate command, the printer switch setting determines the character case, and the previous request is erased. During a Copy command or local print operation, the character case is determined by the setting of the Mono/Dual switch on the *from* display.

During a Write command or buffer reprint operation, the previous print case request is honored. Charge Case switch settings are ineffective during transmission of the SCS data stream to a 3287 (SCS is always dual case). The proper character code points must be used to ensure the correct printout.

**Split Vertical Bar (¦) Character**

The split vertical bar (¦) character, hex 6A, is available on the 3287 and 3288.

# Chapter 3. Commands and Orders

Program control of 3270 operations is accomplished with a flexible set of commands and orders. Commands are issued by the channel program to initiate such operations as the total or partial writing, reading, and erasing of data in a selected 3270 device buffer. Orders can be included in write data streams, either alone or intermixed with display or print data.

Two types of orders are available. One type is executed as it is received by the 3271, 3272, or 3275. This type is used to position, define, and format data being written into the buffer, to erase selected unprotected data in the buffer, and to reposition the cursor. The second type of orders specifies printer format. These orders are initially stored in the buffer as data and are executed only during a print operation.

## Commands

Four basic types of commands are executed by the 3270 system:

1. Write commands, which transfer data and orders from main storage to the 3270 system.
2. Read commands, which transfer 3270 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.
4. Sense command (local configurations only), which transfers to main storage a byte of sense data that reflects certain control or check conditions existing in the device or control unit (CU) to which the command was addressed.

Figure 3-1 lists the commands, and associated codes, that can be executed by the 3270 system.

### *Timing Considerations*

The rate at which data is transferred between main storage of the data processing system and a device attached to the 3270 display system depends on the information-transfer capability of the channel, on whether data or command codes are transferred, and on whether a local or remote 3270 system is attached.

In a local configuration, the 3272 CU provides information to, and accepts it from, the channel at a byte rate established by the channel or by the CU, whichever is the slower rate. The maximum data-transfer rate for a Write command operation is 650,000 bytes per second. For a read operation, the maximum data-transfer rate is 400,000 bytes per second.

| | Local | Remote | | |
|---|---|---|---|---|
| Command | EBCDIC (Hex) | EBCDIC (Hex) | ASCII (Hex) | Graphic |
| Write | 01 | F1 | 31 | 1 |
| Erase/Write | 05 | F5 | 35 | 5 |
| Read Buffer | 02 | F2 | 32 | 2 |
| Read Modified | 06 | F6 | 36 | 6 |
| Copy | NA | F7 | 37 | 7 |
| Select | 0B | NA | NA | NA |
| Erase All Unprotected | 0F | 6F | 3F | ? |
| No Operation | 03 | NA | NA | NA |
| Sense | 04 | NA | NA | NA |

**Figure 3-1. Local and Remote Command Codes**

When a remotely attached 3270 display system is in operation, the rate at which data is transferred between the data processing system's main storage and the 3271 CU depends on the type of transmission control unit (TCU) and on the modems and communication facilities used. The 3270 system accepts data from, and provides data to, the TCU or communication facility at the byte rate established by the TCU or communication facility.

All command operations that direct movement of data to and from the 3270 system result in transfer of data between the CU and a device buffer. When commands are not being performed, the CU and the device buffer cycle asynchronously, and the last image displayed by a previous command is continuously regenerated at a visible rate.

The CU contains the timing controls required to move data between the CU and the device buffers. To accomplish a data transfer to a CU buffer from a device buffer, as, for example, during a Read command, the device buffer must first shift to the buffer position where data will initially be sent. Because buffers are loaded one position at a time, a 480-character device buffer can be filled faster than a 1,920-character buffer. During a read or write type command, the average time required to transfer data from a 1,920-character device buffer to the CU is approximately 50 milliseconds (ms). During execution of a Write command with a 1,920-character position buffer, approximately 80 ms (average time) is needed for buffer transfers, since the contents of the device buffer must first be brought from the device to the CU, where the contents are updated and then returned to the device.

An average time of approximately 30 ms is required to transfer data from a 480-character device buffer to a 480-character CU buffer during a read operation, and approximately 40 ms is needed for 480-character-buffer transfers during execution of a Write command. To obtain the total command execution time, the time needed to transfer information between the channel and the CU must be determined and added to the buffer transfer times given here.

During the short periods when information is transferred between buffers, the display buffer regeneration cycle is suspended, causing the display image to blink momentarily.

### *Read Commands*

Two read-type commands are executed by the 3270: Read Buffer and Read Modified. Read Buffer, which is provided primarily for diagnostic purposes, causes the entire contents of the selected 3275, 3277, 3284 (Model 1 or 2), 3286 (Model 1 or 2), or 3288 (Model 2) buffer to be read into main storage. The operation initiated by Read Modified is determined by 3275 or 3277 operator actions. The information read during execution of Read Modified could consist of fields of data modified by keyboard, data entered by the card reader, buffer addresses or data of selector-pen fields, or the code of a program-function or program-access key.

In remote configurations, reading is normally accomplished by a General or Specific Poll sequence (described later under "Remote Operations"). In local configurations, an operator action that requires program interaction causes an attention interruption; the program would respond to this attention interruption with a read command. In remote, the 3271 or 3275 cannot generate attention interruptions. Instead, the program should issue poll sequences periodically. Upon receipt of a poll sequence, the 3271 or 3275 hardware 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 hardware-generated Read Modified command operation is performed by the 3271 or 3275.

3. If no operator action has occurred and status and sense information is not pending, the 3271 or 3275 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, 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 ends as follows: (1) in local, Unit Check is sent in the ending status byte; (2) in remote, the transmission is terminated with ENQ in place of ETX or ETB.

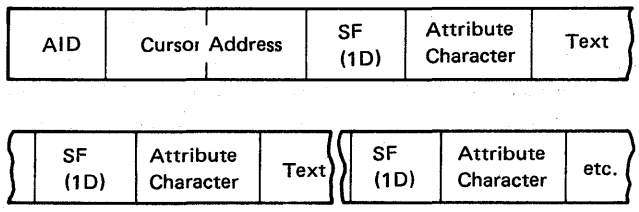
### Read Buffer Command

Execution of the Read Buffer command causes all data in the addressed terminal 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, or if it is chained from a Sense, Select, No Operation, or Copy command.
2. 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 terminates when the last character location in the buffer has been transferred, or before the last character location has been transferred as follows: (1) in local configurations, when the channel byte count reaches 0 or (2) in remote configurations, when the last character of a text block has been transferred (described under "Remote Operations").

The transferred data stream begins with a 3-character read heading consisting of the attention identification (AID) character followed by a 2-character cursor address. The contents of all buffer locations are transferred, including nulls. The 3270 inserts Start Field (SF) order codes before each attribute character to identify the beginning of each field. An example of the read data stream follows:




The possible cursor address byte configurations are shown in Appendix B. The possible AID byte configurations are shown in Figure 3-2. 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: (1) pressing a program-function or program-access key, (2) entering a card into the card reader, or (3) with the selector pen, detecting on an attention field. The attribute character is shown in Figure 2-7.

| 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 | 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 | 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 | |
| Operator Identification Card Reader | E6 | 57 | W | Rd Mod | |
| Selector Pen Attention | 7E | 3D | = | Rd Mod | AID code, cursor address, and field address 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 key | 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 I/O interface code is used, refer to the applicable figure in the* IBM 3270 Information Display System: Character Set Reference, *GA27-2837, for possible graphic character differences.*

**Figure 3-2. Attention ID (AID) Configurations**

## Read Modified Command

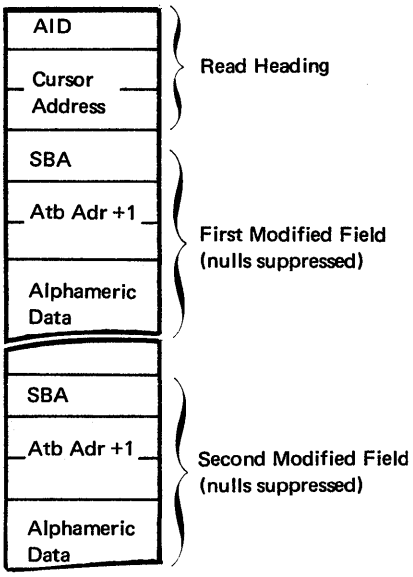
Read Modified initiates one of three operations, as determined by operator actions at the display station: (1) Read Modified, (2) Short Read, or (3) Test Request Read. Figure 3-2 lists the operator actions and the resulting Read Modified command operation initiated by each action. Read Modified commands are not normally used for remote configurations, since polling initiates a hardware-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. When operations start at a device, the device buffer is cleared to all nulls (1) when the operator turns power on or presses the CLEAR key, or (2) when the erase portion of an Erase/ Write command is executed with that device selected. In addition, the Erase All Unprotected command and certain orders can clear selected portions of a buffer to nulls. During Read Modified command operations, all modified characters are sent to main storage; null codes are not sent.

**Read Modified Operation.** During a Read Modified command, if an AID other than Selector Pen Attention, a PA key, or CLEAR key is generated, all fields that have been modified by keyboard, selector-pen, or operator identification card reader (OICR) activity 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 three bytes of the read data stream are always the AID code (Figure 3-2) and the 2-byte cursor address; these bytes are called the *read heading.*

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 hardware-generated 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 selector-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-pen and/or keyboard activity), only 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-pen-attention operation, a resulting Read Modified operation will read only the address of the modified fields, not the modified data.

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

- Buffer address 0 if the Read Modified command is unchained or is chained from a Copy, Select, Sense, or No Operation command.
- The current address if the Read Modified command is chained from a Write, Erase/Write, Read Modified, or Read Buffer command.

The search for modified-field attribute bytes ends when the last buffer location is checked or, during 3272 operations, when the channel byte count reaches zero.

The transfer of read data is terminated as follows:

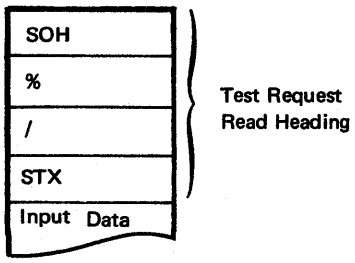
1. If the last modified field is wrapped from the last buffer location (479 or 1919) to the first location, the operation is terminated after all data in the field is transferred (nulls are suppressed). The buffer address at the end of the operation is the address of the next attribute byte in the buffer. 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.
3. During 3272 operations, if the channel byte count reaches zero before all modified data is transferred, read operations are terminated and the remaining modified data is not transferred. The buffer address after termination is undefined.

If the buffer is formatted (contains fields) but none of the fields has 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 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's 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.** The Read Modifed command causes a Test Request Read operation if the TEST REQ key has been pressed at the selected device. The Test Request Read data stream sent to main storage is as follows:



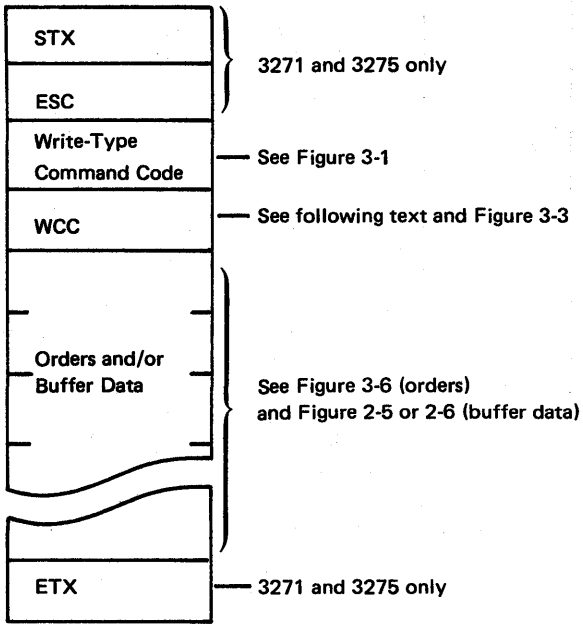

The Test Request Read heading is generated by hardware. The remainder of the data stream is the same as described previously for Read Modified operations, excluding the 3-byte read heading (AID and cursor address). If the buffer is unformatted, all alphameric data in the buffer is included in the data 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 TEST REQ key.

### *Write Commands*

Two write-type commands, Write and Erase/Write, are used by the channel program 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 selected 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 3271 and the 3272 initiate a device-to-CU buffer transfer before Write command operations, but not before Erase/Write command operations.

The bytes received by the 3271, 3272, or 3275 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. The 3271 or 3275 also receives appropriate framing (data-link control) characters. The sequence of bytes is as follows:




The minimum Write command data stream to the 3272 consists of one byte, a WCC. [This is assured since the byte count field of the write channel control word (CCW) must be set to a minimum of 1 or else the command code is not sent to the 3272.] The minimum Write command data stream to the 3271 or 3275 consists of framing characters (STX, ESC, and ETX) and the command code. To be meaningful, a WCC byte should follow the command code; if ETX follows the command code, hardware generates an all-zero WCC byte and command execution is ended normally. An order or display/print data byte that immediately follows the command code is interpreted by hardware as a WCC.

The WCC byte format is as follows:

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

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

Figure 3-3 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 not performed and status or sense information is not generated. 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 2–7 as shown in Figure 2-7. |
| 1 | Always a 1. |
| 2, 3 | Define the printout format, as follows:<br>= 00 - The NL order in the data stream determines the print-line length.<br>= 01 - Specifies 40-character print line.<br>= 10 - Specifies 64-character print line.<br>= 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 the INPUT INHIBITED indicator. 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. |

**Figure 3-3. Write Control Character (WCC)**

Orders and buffer data can follow the WCC character. (Orders are described later in this chapter, following commands.) 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 that 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:

- The starting location may be specified by a Set Buffer Address order that follows the WCC. (This order is described under "Orders.")
- 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 control or Sense command.
- 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 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:*
   a. *The 3272 aborts the Write or Erase/Write command that specifies Start Print.*
   b. *The 3271 or 3275 performs the print operation and aborts the next command.*
2. *The Printout Format bits are honored only if the Start Print bit is set in the same WCC.*
3. *In 3271 operations, if a Write command that includes data is chained from a previous Write command, a Set Buffer Address (SBA) order shall 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.*

4. *Every text message to a 3275 must have an SBA order immediately following the WCC to enable recovery from a line error.*

**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. For its erase operation, this command clears the entire device buffer to nulls (all zero characters), 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.

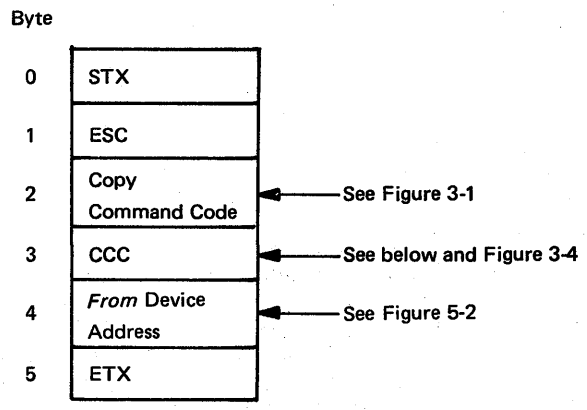
## *Control Commands*

Control commands initiate certain CU and/or device operations not involved with the transfer of data (other than status). Four control-type commands are executed by the 3270: Copy, Select, Erase All Unprotected, and No Operation. Copy is valid for the 3271 only, Select and No Operation are valid for the 3272 only, and Erase All Unprotected is valid for the 3271, 3272, and 3275.

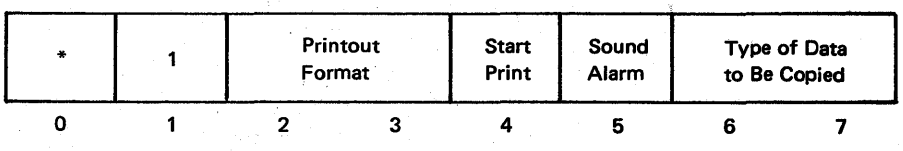
### Copy Command

This command is executed by a 3271 only, and is invalid for the 3272 and 3275. Copy is used to transfer buffer data from one device to another device attached to the same 3271. 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 two 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 sound the audible alarm.

The Copy data stream is as follows:




The CCC-byte format is as follows:




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

Figure 3-4 describes the function of each CCC bit. A CCC and an address byte must always follow the command code; if they do not, the 3271 aborts the command and generates error status.

Copy command operations are similar to Write command operations. After the 3271 accepts the Copy data stream, it initiates the transfer of all 480 or 1,920 bytes from the *from* device buffer to the 3271 buffer. Upon completion of this transfer, the 3271 inserts nulls in all character locations that do *not* contain the type of data specified by CCC bits 6 and 7. The updated CU buffer contents (480 or 1,920 bytes) 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.

**Programming Note:** *Although not essential for locking, it is recommended that a null be written in address 1 of the buffer to facilitate possible future use of the address 1 position.*

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 CU, nulls are inserted as determined by the CCC, and the resulting buffer contents are transferred back to the same device buffer.

**Programming Notes:**

1. *Copy should not be chained* from *a Write or Erase/Write command, since it will destroy the data already written for the selected device.*
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 3271 aborts the subsequent command.*

| Bit | Explanation |
|---|---|
| 0 | Determined by the contents of bits 2–7 as shown in Figure 2-7. |
| 1 | Always a 1. |
| 2, 3 | Define the printout format as follows:<br>= 00 - The NL order in the data stream determines print line length.<br>= 01 - Specifies a 40-character print line.<br>= 10 - Specifies a 64-character print line.<br>= 11 - Specifies an 80-character print line. |
| 4 | The Start Printer 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:<br>= 00 - Only attribute characters are copied.<br>= 01 - Attribute characters and unprotected alphameric fields (including nulls) are copied. Nulls are transferred for the alphameric characters not copied from the protected fields.<br>= 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.<br>= 11 - The entire contents of the storage buffer (including nulls) are copied. |

**Figure 3-4. Copy Control Character (CCC)**

### Select Command

Select is an immediate command executed only by the 3272; it is invalid for the 3271 and 3275. The 3272 executes a Select command by performing a device-to-3272 buffer transfer. If not preceded by a Select command, this same buffer transfer operation is performed as part of an initial (unchained) Write, Read Modified, or Read Buffer command.

The advantages of Select command usage are realized when the 3272 is attached to a block multiplexer channel or to a byte multiplexer channel operating in forced burst mode for the complete data transfer. Upon receipt of Select, the 3272 sends Channel End as initial status to the channel. This frees a block multiplexer channel to perform other operations. Upon successful completion of the buffer transfer, the 3272 sends Device End status asynchronously to the channel. Upon receipt of this status by the channel, a chain operation to the desired command (Write, Read Modified, or Read Buffer) must be initiated for effective use of the Select command. Note that device-to-3272-buffer-transfer time is not part of the execution time for this command.

At the conclusion of the command following the Select command, the 3272 again issues Device End status. At this point, the channel may chain to another command of the same type or it may disconnect. If a chaining operation is performed, another Select command is unnecessary since the addressed-device buffer contents are already in the 3272 buffer.

Thus, the Select command is used to separate the device-to-3272 buffer transfer operation portion of a Write, Read Modified, or Read Buffer command from the actual execution of the command. By doing so, the channel can use the buffer transfer time for other operations.

### 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.
4. Resets the AID byte.
5. Repositions the cursor to the first character location in the first unprotected field of the buffer.

If the entire buffer is protected, buffer data is not cleared and MDT bits are not reset. However, the keyboard is unlocked, AID is reset, and the cursor is repositioned to buffer address 0.

If the first unprotected field wraps the buffer, the cursor is positioned to buffer location 0.

In local configurations, Erase All Unprotected is an immediate-type command. Upon acceptance of this command, the 3272 goes *busy* and sends Channel End initial status to the channel. Upon successful completion of this command, the 3272 sends Device End status asynchronously to the channel and then goes *not busy*.

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

### No Operation Command

This command is valid for the 3272 only. It performs no functional operation in the 3272, but may be used to retrieve pending status. No Operation is an immediate command, and therefore Channel End and Device End normally will be presented as initial status unless pending status or a busy condition exists.

### *Sense Command*

Sense is valid for the 3272 only. It should be issued in response to Unit Check status for further definition of the Unit Check condition. The 3272 responds to a Sense command by sending one byte of sense data to the channel and resets the sense register when the channel accepts the Device End (DE) for the command. All commands, except No Operation or Test I/O (command code X'00'), reset the sense register immediately when the command is issued, including a Sense command to a different address for which the sense data is pending. For this reason, the 3272 in a multiple-program environment may, at times, respond with zero sense data even after a unit check.

The sense byte configuration is as follows:

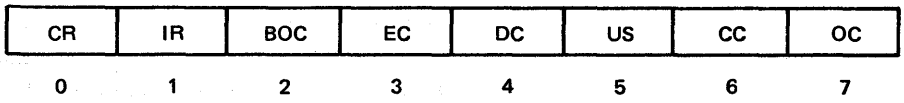



Figure 3-5 summarizes the significance of each sense bit. The various sense and status bit combinations are described in Figures 4-5, 4-6, and 4-7.

| Bit | Name | Significance |
|---|---|---|
| 0 | Command Reject | Set if the 3272 has received an invalid command; the valid commands are listed in Figure 3-1. |
| 1 | Intervention Required (IR) | Set if a command other than Sense was addressed to a device that is unavailable or *not ready*. |
| 2 | Bus Out Check (BOC) | Set if the 3272 has detected bad parity on any command or data byte received from the channel. |
| 3 | Equipment Check (EC) | Set if (1) the 3272 has asynchronously detected a parity check on data received from a device in response to an internal poll for attention status (the internal poll is tried twice before EC is set), or (2) a printer error occurs. If this is a device-detected condition, Unit Specify is also set. |
| 4 | Data Check (DC) | Set if (1) the 3272 or a device has detected bad parity on data transferred internally or between the 3272 and a device during command operations, (2) a 3277 has detected a cursor check, or (3) a device has detected a buffer check. If this is a device-detected condition, Unit Specify is also set. |
| 5 | Unit Specify (US) | Set if the sense bits resulted from a device-detected error. |
| 6 | Control Check (CC) | Set when the 3272 has detected a timeout condition. (The addressed device fails to perform a specified operation or respond to the 3272 within a specified period of time.) |
| 7 | Operation Check (OC) | Set when the 3272 has received a valid command or order that it cannot execute, as follows:<br>1. SBA, RA, or EUA order specifies an illegal buffer address.<br>2. Write data stream ends before all required bytes of SBA, RA, EUA, or SF order sequence are received.<br>3. Write, or Erase/Write with Start Print bit set in WCC, is chained to the next command; the print operation is suppressed. |

**Figure 3-5. Sense Bit Description**

## Orders

Orders can be included in Write or Erase/Write 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. (These orders are described under "Printer Operations" in Chapter 2.)

The following paragraphs describe buffer control orders, which are executed as they are received in the write data stream by the 3271, 3272, or 3275; these orders are not stored in the buffer. Six buffer control orders (see Figure 3-6) are provided (1) to position, define, and format data being written into the buffer, (2) to erase selected unprotected data in the buffer, and (3) to reposition the cursor.

### *Start Field (SF) Order*

This order identifies to the CU that the next byte in the write data stream is an attribute character. (The attribute character is described in Figure 2-8.) The CU always stores the next byte as the attribute character at the current buffer address. As the attribute character is stored, the CU 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.

During execution of a Read Buffer command, the CU 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 Sequence / Order | Byte 1 (Order Code) EBCDIC (Hex) | Byte 1 (Order Code) ASCII (Hex) | Byte 2 | Byte 3 | Byte 4 |
|---|---|---|---|---|---|
| Start Field (SF) | 1D | 1D | Attribute Character[1] | | |
| Set Buffer Address (SBA) (Starting Address) | 11 | 11 | 1st Address Byte[2] | 2nd Address Byte[2] | |
| Insert Cursor (IC) | 13 | 13 | | | |
| Program Tab (PT) | 05 | 09 | | | |
| Repeat to Address (RA) (Stop Address) | 3C | 14 | 1st Address Byte[2] | 2nd Address Byte[2] | Character to Be Repeated[3] |
| Erase Unprotected to Address (EUA) (Stop Address) | 12 | 12 | 1st Address Byte[2] | 2nd Address Byte[2] | |

[1] Figure 2-8 shows the attribute byte, and Figure 2-7 shows the coding of this byte.
[2] Appendix B lists the 2-byte code for each possible address. To be valid, an address must not exceed 479 (if issued to a Model 1) or 1,919 (if issued to a Model 2).
[3] Figures 2-5 and 2-6 show the coding of this byte.

**Figure 3-6. 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 (1) to specify the starting address for a PT, RA, or EUA order, (2) to specify the address at which an attribute byte is to be stored by an SF order, or (3) to specify the address at which the cursor is to be repositioned by an IC order.

**Programming Note:** *Every text message to a 3275 must have an SBA order immediately following the WCC to enable recovery from a line error.*

If the SBA order specifies an invalid address (greater than 479 if Model 1 or 1919 if Model 2), 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 CU of fields that are modified. When a Read Modified command is executed in a remote 3270 unit, this 3-byte sequence is always sent in the same text block. Remote 3270 units do 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 removed 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. When the PT order follows a control command, order, or order sequence, the buffer is not modified.

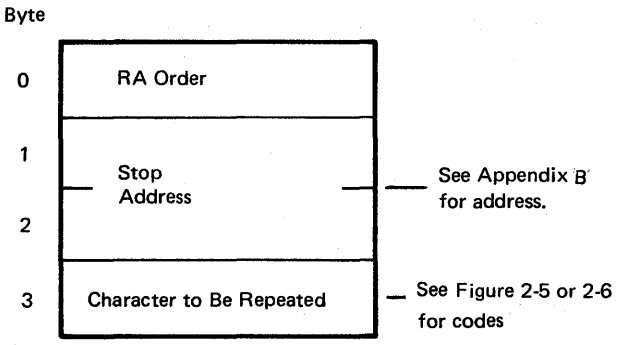
The PT order stops its search at the last location in the buffer. If an attribute character for an unprotected field is not found by this point, the buffer address is set to location 0. (If the PT order finds an attribute character for an unprotected field in the last buffer location, the buffer address 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 buffer location, the new PT order will continue to insert nulls from buffer location 0 to the end of the current field.

**Programming Restriction (for Remote Operations):** *Successive PT orders, without intervening characters or other orders (not including the Insert Cursor order), should not be issued to a 3271 Model 2 Control Unit when the buffer (1) contains one unprotected field or (2) is unformatted. To do so may cause the Write command to be aborted and error status to be generated.*

### *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 three bytes immediately following the RA order in the write data stream, as follows:




The third character following the RA order is always interpreted as the character that will be repeated. If an invalid stop address (greater than 479 if a Model 1, or 1919 if a Model 2) is specified, the write operation is terminated at this point without storing the character, and error status is generated.

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 can 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 feature), only X'ID' is repeated.*

**Programming Restriction (for 3271 and 3275 Only):** *If the RA order specifies storing a character in more than 480 locations, the write operation may be aborted and error status generated.*

### *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 two address bytes, which immediately follow the EUA order in the write data stream. If an invalid address (greater than 479 if a Model 1, or 1919 if a Model 2) is specified, the write operation is terminated at this point, no erasure (insertion of nulls) occurs, and error status is generated.

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 4. Local Operations (3272 Models 1 and 2)

## Introduction

The 3272 Control Unit Models 1 and 2 can attach to a selector channel, a byte multiplexer channel, or a block multiplexer channel, each through the I/O interface (Figure 4-1). When it is attached to a byte multiplexer channel, operations can be in forced-burst mode or in single-byte-multiplex mode. The channel, in turn, is attached to main storage and to the central processing unit (CPU).

The channel program controls all control unit operations by transmitting information across the I/O interface. This information consists of (1) an address byte, which selects one 3272 control unit and one device (display or printer) attached to the control unit, (2) command bytes, which specify the type of operation to be performed by the control unit for that device, (3) data bytes, which either are stored in the control unit buffer for ultimate use by the selected device as display or printout data or are decoded as orders and used by the control unit for formatting the buffer, and (4) various control signals. Status bytes, which are automatically generated by the control unit, inform the channel program (1) of the general condition of the control unit and selected device at various stages of command operations and (2) of unique conditions of the control unit and any attached device when command operations are not in progress.

## Interface Operations

Local interface operations are summarized in the following paragraphs and are described in detail in the *IBM System/370 Principles of Operation* manual, GA22-7000. The CPU program initiates control unit operations with a Start I/O instruction. This instruction identifies the I/O control unit and device (in this case, the control unit and a display or printer) and causes the channel to fetch a channel address word (CAW) from a fixed location in main storage. The CAW designates the storage protection key and the location in main storage from which the channel subsequently fetches the first channel command word (CCW). The CCW specifies the command to be executed and the number and address, in main storage, of any bytes to be transmitted.

### *Selection*

The channel attempts to select the control unit and an attached device by sending a unique address byte to the control unit (and to all other control units attached to the same channel or subchannel). When a control unit has 16 or fewer devices attached, the first four bits of the address byte specify the control unit address and the last four bits of the address byte specify the device address (Figure 4-2). Up to 32 devices can attach to control units that have even-numbered addresses; these addresses are coded as shown in Figure 4-3. Note that no more than 16 devices can be attached to a control unit that has an odd-numbered address. Device address must always be assigned sequentially, starting with address 0. However, no priority is given to any particular device address.

When a control unit recognizes both addresses, it logically connects to the channel and responds to the selection by returning the address byte to the channel.

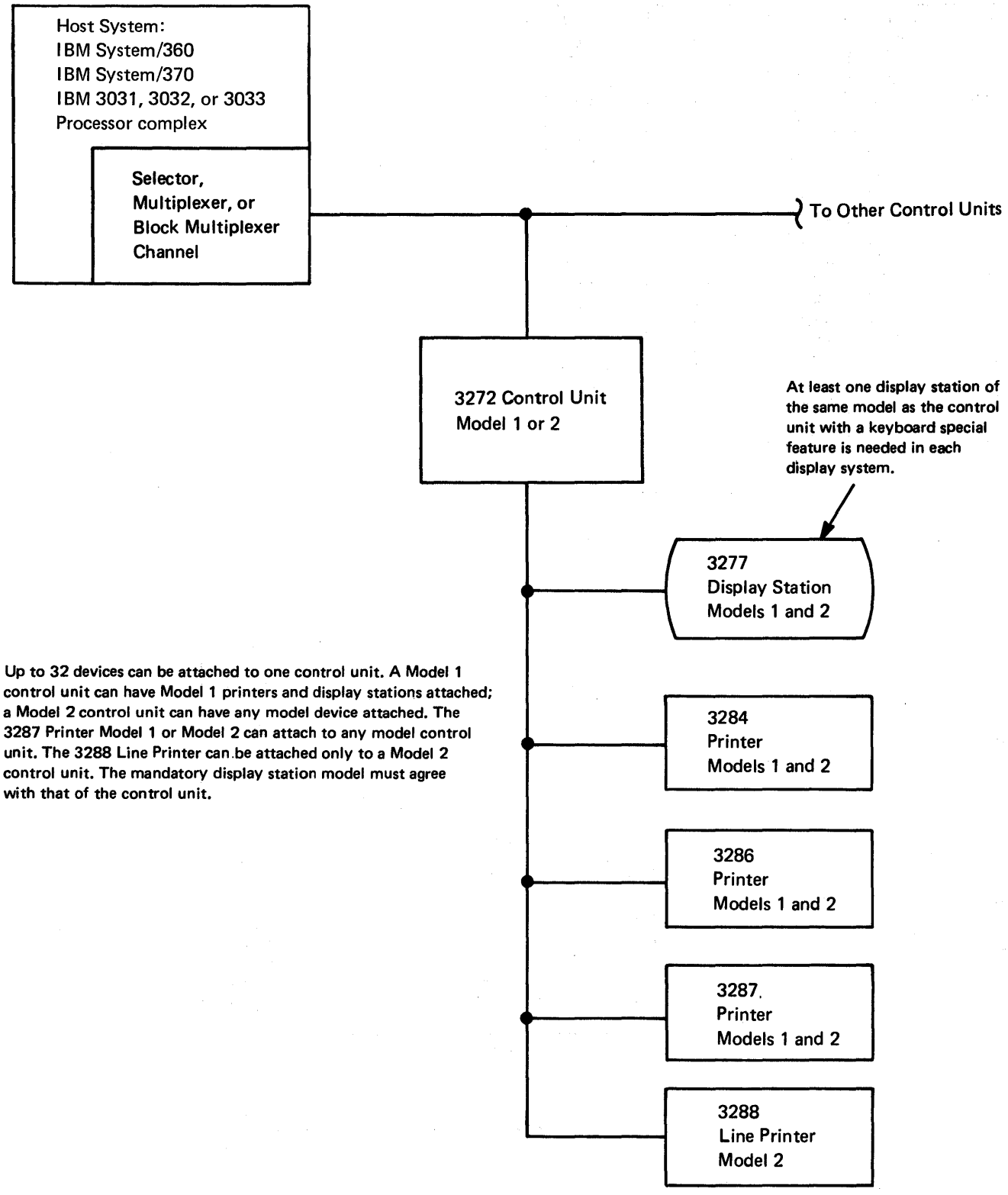
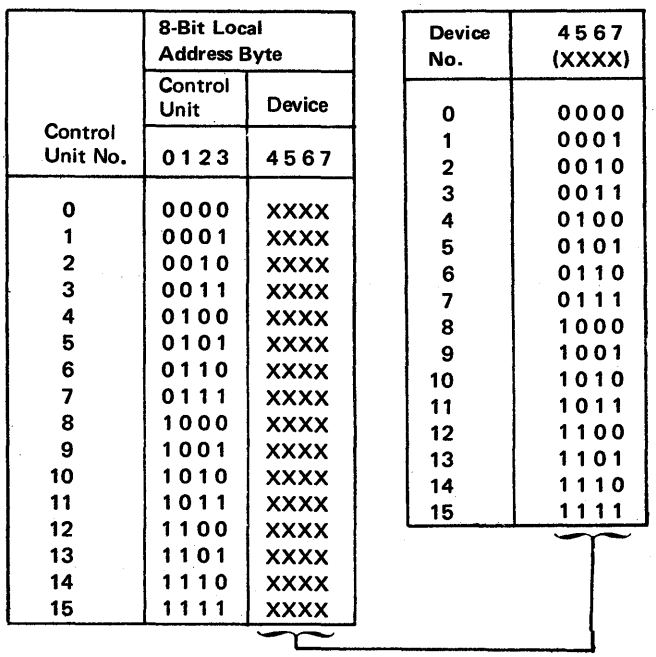


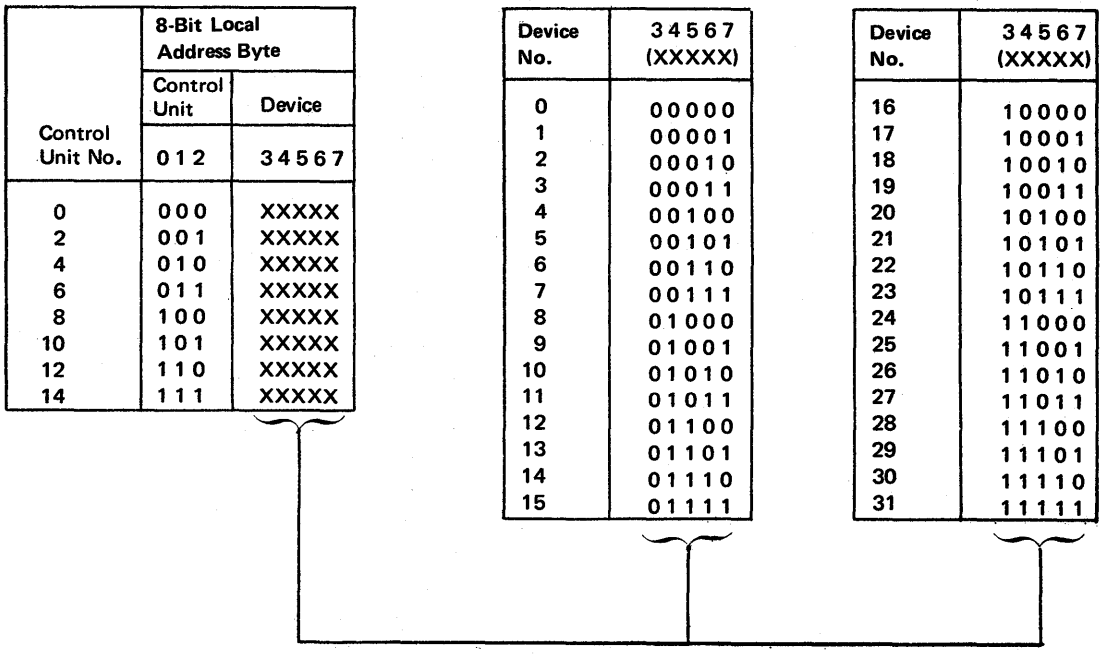

Figure 4-1. Locally Attached 3270 Information Display System



| Control Unit No. | 8-Bit Local Address Byte: Control Unit 0123 | Device 4567 |
|---|---|---|
| 0 | 0000 | XXXX |
| 1 | 0001 | XXXX |
| 2 | 0010 | XXXX |
| 3 | 0011 | XXXX |
| 4 | 0100 | XXXX |
| 5 | 0101 | XXXX |
| 6 | 0110 | XXXX |
| 7 | 0111 | XXXX |
| 8 | 1000 | XXXX |
| 9 | 1001 | XXXX |
| 10 | 1010 | XXXX |
| 11 | 1011 | XXXX |
| 12 | 1100 | XXXX |
| 13 | 1101 | XXXX |
| 14 | 1110 | XXXX |
| 15 | 1111 | XXXX |

| Device No. | 4567 (XXXX) |
|---|---|
| 0 | 0000 |
| 1 | 0001 |
| 2 | 0010 |
| 3 | 0011 |
| 4 | 0100 |
| 5 | 0101 |
| 6 | 0110 |
| 7 | 0111 |
| 8 | 1000 |
| 9 | 1001 |
| 10 | 1010 |
| 11 | 1011 |
| 12 | 1100 |
| 13 | 1101 |
| 14 | 1110 |
| 15 | 1111 |

Figure 4-2. 3272 and Device Addressing – 16 or Fewer Devices per Control Unit



| Control Unit No. | 8-Bit Local Address Byte: Control Unit 012 | Device 34567 |
|---|---|---|
| 0 | 000 | XXXXX |
| 2 | 001 | XXXXX |
| 4 | 010 | XXXXX |
| 6 | 011 | XXXXX |
| 8 | 100 | XXXXX |
| 10 | 101 | XXXXX |
| 12 | 110 | XXXXX |
| 14 | 111 | XXXXX |

| Device No. | 34567 (XXXXX) |
|---|---|
| 0 | 00000 |
| 1 | 00001 |
| 2 | 00010 |
| 3 | 00011 |
| 4 | 00100 |
| 5 | 00101 |
| 6 | 00110 |
| 7 | 00111 |
| 8 | 01000 |
| 9 | 01001 |
| 10 | 01010 |
| 11 | 01011 |
| 12 | 01100 |
| 13 | 01101 |
| 14 | 01110 |
| 15 | 01111 |

| Device No. | 34567 (XXXXX) |
|---|---|
| 16 | 10000 |
| 17 | 10001 |
| 18 | 10010 |
| 19 | 10011 |
| 20 | 10100 |
| 21 | 10101 |
| 22 | 10110 |
| 23 | 10111 |
| 24 | 11000 |
| 25 | 11001 |
| 26 | 11010 |
| 27 | 11011 |
| 28 | 11100 |
| 29 | 11101 |
| 30 | 11110 |
| 31 | 11111 |

**Note:** *Control Unit Nos. 1, 3, 5, 7, 9, 11, 13, and 15 cannot be assigned when attached devices are assigned Device No. 16 or greater.*

Figure 4-3. 3272 and Device Addressing – 17 or More Devices per Control Unit

### *Command Initiation*

Command operations by the control unit start when the control unit and a device are successfully selected. When a command is to be executed by the control unit (not by the channel alone), the channel sends the command code (CCW bits 0–7) to the control unit.

When execution of the command involves a transfer of data (such as Write or Read Modified), the control unit responds to the command with a status byte (called *initial* status) indicating whether it can execute the command. If the command can be executed, the channel is set up to respond automatically to service requests from the control unit, and the control unit assumes further control of the operation. Command operation can be terminated by the control unit or when the channel byte count reaches 0. At this time, the control unit sends the channel a second status byte (called *ending* status), which indicates whether the command operation was successfully performed.

When the function of the 3270 command does not involve the transfer of data (such as EAU), it is called an *immediate* command. The resulting control unit operation depends on the particular command, as follows. If the command is No Operation, ending status and initial status are combined to indicate to the channel that the control unit has completed execution of the command. If the command is Select or Erase All Unprotected, which initiate certain control unit and device operations, the initial status from the control unit is such that block and byte multiplexer channels are released to perform other operations (selector channels remain logically connected to the control unit). When command execution is completed by the control unit and selected device (and regains selection if attached to a block or byte multiplexer channel), the control unit sends ending status to the channel, indicating whether the command was successfully performed.

### *Chaining*

When the channel has completed the operations specified by a CCW, it can continue the activity initiated by the Start I/O by fetching a new CCW, thereby starting execution of another command. The fetching of this new CCW is called *command chaining,* and the CCWs belonging to such a sequence are said to be chained. All CCWs in a chain apply to the control unit and device specified by the Start I/O instruction.

Either of two types of chaining can be specified by the current CCW (bits 32 and 33): data chaining or command chaining. During data chaining (current CCW bit 32=1), the new CCW fetched by the channel defines a new main storage area (data address) for the current command. During command chaining (current CCW bit 33=1), the new CCW specifies a new command and a data address for that new command.

Thus, when command chaining is used, the control unit is selected following the Start I/O instruction when the channel receives the first CCW in the chain that involves operations with the control unit. The control unit is dedicated to one CCW string until final channel-end time or until operations are abnormally terminated. Programming restrictions that must be observed when command chaining is used are described in Chapter 3, "Commands and Orders."

### *Status*

The control unit generates a status byte to inform the channel of certain control unit and device conditions. This status byte can be generated synchronously (while the control unit is selected and performing a command operation with the channel) or asynchronously (while the control unit is not selected).

Synchronous status is passed to the channel as both *initial* and *ending* status to a command. Initial status reflects the condition of the selected device and/or control unit upon receipt of a command and indicates to the channel whether the command can be executed. Ending status reflects the condition of the control unit and selected device after all channel/3270 interface operations of a nonimmediate command are completed. Asynchronous status reflects (1) ending status for an immediate command other than No Operation, (2) a second ending status for a Write, Erase/Write, or Erase/Write Alternate command, indicating that the control-unit-to-device-buffer transfer is completed, or (3) an equipment condition or operator action not associated with command execution (an attention).

Figure 4-4 describes each bit of the status byte. Status is reset by the control unit once it has been accepted by the channel.

| Bit | Name | Condition |
|---|---|---|
| 0 | Attention (A) | Indicates a request for services from a 3277 attached to a 3272. Set by certain keyboard, selector-pen, or card-reader activity at a 3277 (Figure 3-2). The program should respond by issuing a Read Modified command (chained from a Select command if block or byte multiplexer channel) to the 3277 requesting attention. The attention bit is also set with the Unit Check bit as a result of an asynchronously detected equipment malfunction; in this case, the program should respond by issuing a Sense command. |
| 1 | Status Modifier (SM) | Is set, with Busy bit, in initial status byte to indicate that there is pending status for a device other than the one selected. |
| 2 | Control Unit End (CUE) | Is set following a busy condition, after pending status is cleared or when control unit is no longer busy, to indicate that control unit is now not busy and is free to accept a new command. |
| 3 | Busy (B) | Is set alone in initial status byte when addressed device is busy because it is performing a print operation or an Erase All Unprotected command. Set with SM when addressed control unit is busy. When the channel addresses a device other than the one that is busy and control unit is not busy, addressed device becomes selected and the command is honored. Busy bit is also set with pending status if addressed device has such status; if pending status is for a device other than the one addressed, Status Modifier bit is also set. |
| 4 | Channel End (CE) | Indicates channel data transfer operations are completed. Is set alone (1) in initial status for Select or Erase All Unprotected command, or (2) as ending status for Write, Erase/Write, or Erase/Write Alternate command; in all cases, Device End status is sent asynchronously when device operations (command execution or control-unit-to-device-buffer transfer) are completed.<br><br>Is set with Device End, to indicate that control unit and device operations (except printing) are completed (1) in initial status for No Operation command, (2) in ending status for Read Buffer, Read Modified, or Sense command, or (3) asynchronously if only Channel End status was pending and the device operation is completed before the channel accepts status.<br><br>Is set with Device End and Unit Exception in initial status for Read or Write command if addressed device is busy executing another command. |
| 5 | Device End (DE) | Indicates that control unit and device have completed all command operations and are free to execute another command. Is set (1) in initial status for No Operation command, (2) in ending status for Read Buffer, Read Modified, or Sense command, and (3) in asynchronous status for Write, Erase/Write, Erase/Write Alternate, Select, or Erase All Unprotected command. |
| 6 | Unit Check (UC) | Is set when an irregular program or equipment condition is detected by control unit or the device. Program should always respond to Unit Check status by issuing a Sense command for further definition of condition. |
| 7 | Unit Exception (UE) | Is set in ending status (synchronous or asynchronous) when control unit has attempted to execute a command but has found, after initial status was returned, that addressed device was busy. |

**Figure 4-4. Status-Byte Bit Assignments for 3272**

## Initial Status

Initial status is generated by the control unit in response to initial selection, by the channel, of the control unit and an attached device. During the initial selection sequence, the status byte is sent to the channel after the control unit receives a command.

Figure 4-5 shows the possible initial status bit configurations. An all-zero status byte is sent when a nonimmediate command is accepted for execution by the control unit; it is also sent in response to Test I/O if other status is not pending. The unit-check bit is set if the command is not accepted by the control unit because of a program or equipment error.

Initial status to immediate commands is as follows. For No Operation, channel end and device end are both set to indicate completion of the command. For Select and Erase All Unprotected, which do not involve data transfer between the channel and the control unit, channel end is set. This frees a block or byte multiplexer channel for other operations while the command is being executed. When command execution is completed, ending status is presented asynchronously.

If a Start I/O Fast Release (SIOF) is executed by the channel, then unchained initial status becomes ending status. (See *System/370 Principles of Operation*, GA22-7000.)

| Status[1] (Hex) | Sense (Hex) | Display | Printer | Error-Recovery Procedure | Condition |
|---|---|---|---|---|---|
| All Zeros (00) | | X | X | | Normal status for any command other than No Operation, Select, or Erase All Unprotected. |
| CE (08) | | X | X | | Normal status for a Select or Erase All Unprotected command. |
| CE, DE (0C) | | X | X | | Normal status for a No Operation command. |
| UC (02) | BOC (20) | X | X | 1 | A parity check was detected on the command byte. |
| UC (02) | IR (40) | X | X | 2 | A command other than Sense was addressed to a device that the control unit has recorded as *unavailable* or *not ready*. |
| UC (02) | CR (80) | X | X | 3 | An invalid command was issued to control unit. |
| UC (02) | None (00) | X | X | 1 | 3272 sense data was reset by a command to another device on the control unit. |
| B (10) | | X | X | | Response to a command addressed to a device that is being serviced by the control unit or that is completing a command previously issued. |
| B, SM (50) | | X | X | | Response to a command addressed to a device other than a device whose status is pending or a device being serviced by the control unit. |

[1] If an SIOF is executed by the channel, unchained initial status becomes ending status.

Figure 4-5. Initial Status and Sense Conditions for 3272

When status is pending (a previous status byte is awaiting transfer to the channel), the pending status byte, with the busy bit set, is sent to the channel in response to any command (not to a Test I/O instruction), and that command is not accepted by the control unit. For Test I/O, the pending status byte is presented without the Busy bit set. If the pending status is for a device other than the one selected during the initial command sequence, only busy, status modifier (B, SM) is presented to the channel and the pending status is retained at the control unit.

### Ending Status

When the control unit completes channel operations for a nonimmediate command, it sends an ending status byte to the channel, freeing the channel for other operations. This status byte always relates to the command operation that has been executed. The normal ending status byte for a Read Buffer, Read Modified, or Sense command will have only the channel-end and device-end bits set, indicating that the command has been executed. Normal ending status for a Write, Erase/Write, or Erase/Write Alternate command is channel end alone. When the control-unit-to-device-buffer transfer is completed, ending the command operation, device-end status is sent to the channel as asynchronous status. Any error condition associated with the operation just executed will cause additional status bits to be set. Figure 4-6 shows the possible ending status bit configurations. Ending status causes an I/O interruption unless chaining is specified.

When the control unit has pending status, it attempts to gain selection of the channel asynchronously to pass this status. It is passed to the channel either when selection is accomplished or as initial status for the next command (with the busy bit set), whichever occurs first.

### Asynchronous Status

Asynchronous status reflects (1) the ending status of an "immediate" command other than No Operation, (2) the second ending status for a Write, Erase/Write, or Erase/Write Alternate command, indicating that all command-initiated operations are completed, (3) an action by the device operator that requires program intervention (attention status), or (4) a control-unit or attached-device equipment malfunction. Figure 4-7 shows the possible asynchronous status bit configurations.

When an asynchronous status condition occurs, the control unit attempts to gain selection by the channel (this is a hardware function) and passes this status to the channel when selection is accomplished. This status is called *pending* status until selection is accomplished. If the channel issues a command before retrieving this pending status, the pending status is returned, with the Busy bit set, in place of initial status for the command; in this case, the command is not executed, unless it is a Test I/O instruction.

When an asynchronous condition occurs at a device while the control unit is performing command operations with another device, the asynchronous status remains pending until the control unit completes the current command operation, returns ending status to the channel, and becomes not busy. The control unit then retrieves the pending status from the device and attempts to present it to the channel in the same manner as other asynchronous statuses.

Some other conditions of multiple status that can occur are not covered here. These conditions can be caused by multiple error conditions occurring simultaneously.

| Status (Hex) | Sense (Hex) | Display | Printer | Error-Recovery Procedure | Condition |
|---|---|---|---|---|---|
| CE[1] (08) | | X | X | | Sent at end of data stream on Write, Erase/Write, or Erase/Write Alternate command. |
| CE, DE[1,2] (0C) | | X | X | | Sent at end of data stream on a Read Buffer, Read Modified, or Sense command or when channel byte count goes to zero on a Read Modified or Read Buffer command. |
| CE, DE, UC[2] (0E) | BOC (20) | X | X | 10 | The control unit detected a parity error on a character in data stream of a Write, Erase/Write, or Erase/Write Alternate command.[3] |
| CE, DE, UC[1,2] (0E) | DC, US (0C) | X | X | 1 | Addressed device detected a parity or cursor check during a Write, Read Buffer, or Read Modified command. |
| CE, DE, UC[1,2] (0E) | DC (08) | X | X | 1 | The control unit detected a cursor or parity check during receipt of data stream on a Write, Erase/Write Alternate, or Erase/Write command. |
| CE, DE, UC[1,2] (0E) | DC (08) | X | X | 10 | The control unit detected a cursor or parity check during transmission of data stream on a Read Buffer or Read Modified command. |
| CE, DE, UC[1,2] (0E) | CC (02) | X | X | 10 | Addressed device failed to respond in a specified period of time to an Erase/Write, Erase/Write Alternate, unchained Read Buffer, Read Modified, or Write command. |
| CE, DE, UC[2] (0E) | OC (01) | X | X | 3 | The 3272 received an invalid buffer address in data stream of a Write, Erase/Write, or Erase/Write Alternate command, or data stream ended before providing all characters required for an SBA, RA, SF, or EUA order on a Write, Erase/Write, or Erase/Write Alternate command. |
| CE, DE, UC[2] (0E) | None (00) | X | X | 1 | 3272 sense data was reset by a command to another device on the control unit. |
| CE, DE, UE[1,2] (0D) | | X | X | 9 | The control unit attempted to perform Read Buffer, Read Modified, Write, Erase/Write, or Erase/Write Alternate command but found, after returning initial status, that the addressed device was busy. |

[1] Occurs if a Start I/O Fast Release (SIOF) is executed by the channel for Select, Erase All Unprotected, or No Operation.

[2] If this status is stacked by the channel, CUE could be generated and combined with it before the stacked status is accepted by the channel.

[3] The 3272 updates the device buffer after the total data stream has been processed.

**Figure 4-6. Ending Status and Sense Conditions for 3272**

| Status[1] (Hex) | Sense (Hex) | Display | Printer | Error-Recovery Procedure | Condition |
|---|---|---|---|---|---|
| A (80) | | X | | | An attention-generating action (e.g., program access key has been depressed) was performed by the operator. |
| DE (04) | | X | X | | The control unit-to-device buffer transfer is completed on a Write, Erase/Write, or Erase/Write Alternate command which did not start a printer.<br>The device becomes "not busy" after completing an Erase All Unprotected command or the printer becomes "not busy" after completing a printout.<br>The device-to-control unit buffer transfer is completed on a Select command.<br>A device changes from "not available" to "available" or from "not ready" to "ready".<br>A device becomes "not busy" after having previously sent Unit Exception when the control unit attempted to execute a command with the device when it was "busy".<br>The 3272 Online/Offline switch is thrown from Offline to Online. This causes each "available" device to present a Device End to the channel. |
| A, DE (84) | | X | | | The 3272 Online/Offline switch is thrown from Offline to Online and an attention-generating action (e.g., program access key has been depressed) was performed by the operator. |
| A, UC[2] (82) | EC (10) | X | X | 5 | An idle 3272 polled a device twice and detected a "transmit" parity check each time on the data in the device reply. |
| A, UC[2] (82) | DC, US (0C) | X | X | 1 | An idle device detected a parity check or cursor check in its buffer. |
| A, DE, UC[2] (86) | DC, US (0C) | X | X | 4 or 8 | A device on a 3272 changes from "not available" to "available" or from "not ready" to "ready" and has detected a parity check or cursor check in its buffer or a printer detected parity check while printing. |
| A, DE, UC[2] (86) | IR (40) | | X | 6 | The addressed printer became Not Ready (out of paper or cover open) before completion of a print operation. |
| DE, UC[2] (06) | IR (40) | | X | 6 | A command attempting to start a printer found it Not Ready. |
| A, DE, UC[2] (86) | IR, EC, US (54) | | X | 6 | A printer became mechanically disabled during a printout and an automatic recovery was not successful, the printer CARRIAGE MOTOR POWER switch was off, or the switch fuse was blown. |
| DE, UC[2] (06) | IR, EC, US (54) | | X | 6 | A command attempted to start a print operation, but the printer CARRIAGE MOTOR POWER switch is turned off. |
| A, DE, UC[2] (86) | EC, US (14) | | X | 7 | A printer character generator or sync check error occurred or the printer became mechanically disabled during printout, but restored itself. |
| DE, UC[2] (06) | DC (08) | X | X | 10 | During a Select, Erase/Write, or Erase/Write Alternate command the control unit (1) detected a parity or cursor error, or (2) detected a parity check on data received from the addressed device in response to an internal poll during a command. |

Figure 4-7 (Part 1 of 2). Asynchronous Status and Sense Conditions for 3272

| Status[1] (Hex) | Sense (Hex) | Display | Printer | Error-Recovery Procedure | Condition |
|---|---|---|---|---|---|
| DE, UC[2] (06) | DC (08) | X | X | 1 | During a Write command, the control unit (1) detected a parity or cursor error, or (2) detected a parity check on data received from the addressed device in response to an internal poll during a command. |
| DE, UC[2] (06) | DC, US (OC) | X | X | 1 | The addressed device detected a parity or cursor check while executing a Select, Write, Erase/Write, Erase/Write Alternate, or Erase All Unprotected command. |
| DE, UC[2] (06) | OC (01[3]) | X | X | 3 | A Write, Erase/Write, or Erase/Write Alternate command, containing a WCC with a Start Print bit, is chained to a subsequent command |
| DE, UC[2] (06) | CC (02) | X | X | 10 | The addressed device failed to respond in a specified period of time to a Select, Write, Erase/Write, Erase/Write Alternate, or Erase All Unprotected command, a display was in test mode, or a printer was assigned as a local copy device. (UC, IR will be reported on a subsequent operation.) |
| DE, UE (05) | | X | | 9 | The control unit attempted to perform a Select or Erase All Unprotected command, but found, after returning initial status, that the addressed device was busy. |
| CUE (20) | | X | X | | The control unit had been addressed while busy, but is now not busy and is free to accept a new command. |

[1] If this asynchronous status is stacked by the channel, an asynchronous CUE could be generated and combined with it before the stacked status is accepted by the channel.

[2] If the 3272 sense byte is zeros after a unit-check status, it can be assumed to have been reset by an intervening command to another device on the control unit. Use error-recovery procedure 1.

[3] The 3272 does not set OC upon receipt of a WCC=X'88'.

**Figure 4-7 (Part 2 of 2). Asynchronous Status and Sense Conditions for 3272**

## Error-Recovery Procedures

### *3272 Device-Detected Errors*

Error conditions detected by the control unit or by an attached device are indicated to the program by unit-check status. The program must respond to this status by using a Sense command for further definition of the condition. If a Sense command is not performed and the sense conditions still exist, the control unit will not honor any other interruptions from the devices. Subsequent recovery operations are then determined by the combined configurations of unit-check status bits and associated sense bits.

Figures 4-5, 4-6, and 4-7 list the initial, ending, and asynchronous status and sense bit combinations, respectively. The abbreviations used in these figures are as follows:

- Status Bits
  - B – Busy
  - CE – Channel end
  - DE – Device end
  - SM – Status modifier
  - UE – Unit exception
  - UC – Unit check

- Sense Bits
  - BOC – Bus out check
  - CC – Control check
  - CR – Command reject
  - DC – Data check
  - EC – Equipment check
  - IR – Intervention required
  - OC – Operation check
  - US – Unit specify

### *Recommended Procedures*

The procedures referred to in the Error-Recovery Procedure column of Figures 4-5, 4-6, and 4-7 are as follows:

1. Reconstruct the entire buffer image, and retry the failing chain of commands. The sequence of commands used to reconstruct this image should start with an Erase/Write command. If, after two retries, the problem is not corrected, follow procedure 4.
2. The error indicates the device is *unavailable.* Request and wait for operator intervention to *ready* the device; then, upon receipt of DE status, retry the chain of commands.
3. A nonrecoverable program error has occurred. Examine the data stream to locate the problem.
4. Request maintenance for the device that is giving trouble. After the repair, reconstruct the buffer image, starting with an Erase/Write command.
5. Record the error for future reference, and continue with the program. This error occurred while the control unit was *idle* and does not indicate a data error.
6. The error indicates the printer is out of paper, has the cover open, or has a disabled print mechanism. Request operator intervention to ready the printer; then, upon receipt of DE status, retry the print operation by issuing a Write command with the proper WCC and no data stream. (There is no data error; the data is still intact in the device buffer and can be reused.) If this procedure is unsuccessful, follow procedure 1.

7. The error occurred during a printout and indicates either a character generator or sync check error or a disabled print mechanism. There is no buffer data error. The proper error-recovery procedure is application-dependent, because the user may or may not want a new printout. Since the buffer contents are still good, procedure 6 may be followed.
8. A data error occurred at the device during a printout. Follow procedure 1.
9. A device is busy but the control unit was not informed of this in time to respond with busy status in the initial-status byte. A DE status will be generated asynchronously when the device becomes not busy. After the DE is received, retry the chain of commands that was being executed when the unit-exception (UE) status was received.
10. Retry the failing chain of commands. If, after two retries, the problem is not corrected, follow procedure 1.

### *Channel-Detected Errors*

Errors detected by the channel are indicated to the program by the channel status byte in the channel status word (CSW). If the channel status byte indicates a channel control check, an interface control check, or a channel data check, the recommended error-recovery procedure is to retry the chain of commands. If the problem is not corrected after three retries, request maintenance for the channel that is giving trouble.

# Chapter 5. Remote Operations – BSC (3271 and 3275 Models 1 and 2)

## Introduction

When using binary synchronous communications (BSC) operating mode, the 3271 Model 1 and 2 and 3275 Model 1 and 2 units 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.

**Note:** *In the following paragraphs, the term* 3270 CU *is used in statements that apply to the 3271 and 3275 BSC units. If a statement applies to only one 3270 unit, the appropriate unit number is used.*

The 3270 CU uses BSC procedures over duplex or half-duplex facilities (nonswitched or privately owned); these communications use the Multipoint Data Link mode of operation only. A 3275 with the Dial feature uses the BSC point-to-point data link procedure over a switched line.

### *Code Structures*

Each 3270 CU can operate with one of two code structures: extended binary-coded decimal interchange code (EBCDIC) or American National Standard Code for Information Interchange (ASCII). The choice of code depends on the application. For system compatibility, however, the same code must be chosen for all units on a particular communication line.

### *Channel Program Concepts*

In remote configurations, the TCU becomes the intermediary between the 3270 CU and the channel program. As such, the TCU, not the 3270 CU, executes channel commands and initiates I/O interruptions. At the start of each I/O operation involving the TCU, the Start I/O instruction addresses the TCU and a communication 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 3270 commands.

Selection of a 3270 CU and all subsequent command operations are specified by character sequences in TCU Write CCW data streams. Write CCW data to the TCU communication 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 3270 CU, 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 the use of the Write command to issue EOT to the 3270.*

### *Text Blocking*

The 3270 CU performs inbound text blocking. Each block of data can contain a maximum of 256 test 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 3270 CU.*

Block check characters (BCCs) are transmitted as the last characters of a data stream. (See "Redundancy Checking.") A 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 the 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 communication 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 capabilites)
- *Introduction to the IBM 3704 and 3705 Communications Controller,* GA27-3051

## Multipoint (Nonswitched Line) Data Link Control

Each 3270 CU can operate on a nonswitched communication line with multiple stations. Time-sharing of the line is accomplished by interleaving transmissions between the TCU and all units on the line. A 3271 or 3275 (without the Dial feature) operates multi-dropped on the same line with other properly featured units, such as other 3270 units, IBM 2770s, and IBM 2780s. [Differences for a 3275 with the Dial feature are discussed under "Point-to-Point (Switched Line) Data Link Control."]

The TCU is the *control station* of the multipoint, centralized network. All units attached by communication 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 the receiver of every communication.

### *3270 Modes of Operation*

In the multipoint environment, the 3270 CU is always in one of three modes of operation: control mode, text mode, or transparent-monitor mode.

### Control Mode

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

- A valid selection or poll addressing sequence, by which the 3270 CU will become selected for entry into text mode.
- A DLE-STX sequence, placing the 3270 CU in transparent-monitor mode.

### Text Mode

Once a 3270 CU is successfully selected, it enters text mode. In text mode, the 3270 CU 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 3270 CU 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 3270 CU returns to control mode.

The 3270 CU 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

Transparent-monitor mode is provided with EBCDIC 3270 CUs only. It permits the transmission of data in any of the 256 possible EBCDIC bit patterns between the TCU and another unit on the same communication line with the 3270 CU. 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 3270 CU operation.

When an EBCDIC 3270 CU decodes a DLE STX sequence while in control mode, it enters transparent-monitor mode. While in this mode, the 3270 CU disregards *all* data configurations that may appear on the communication 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 3270 CU 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 communication line data:

- 3270 CU command-sequence characters (including the write data of a Write, Erase/Write, or Erase/Write Alternate command).
- Data transmitted to the TCU in response to a read-type command or to a polling sequence.

A BCC is accumulated for each block of data at both the TCU and the 3270 CU. 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 3270 CU is the receiving unit and detects a BCC error, it responds to the transmission by sending EOT (3275) or NAK (3271) to the TCU. When the TCU is the receiving unit, it sets unit check in the ending status for the TCU command being executed when the BCC error was detected; it also sets 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 3270 CU; they are not stored in main storage or in the 3270 CU 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 prevent the acceptance of faulty messages. Received or transmitted data blocks are counted odd-even-odd-even, etc., by both the transmitter and the 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 3270: CU data-link control characters and 3270 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 3270 CU to establish and control all data-link operations in an orderly fashion. The 3270 message data consists of all address, command, order, and display/print characters sent to the 3270 CU and of all buffer data, AID bytes, and status/sense bytes read from the 3270 CU. Data-link control characters are described individually in the following paragraphs and are described with 3270 message data later in this chapter, 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 (2 bytes) | 1070 | 1030 |
| ACK 1 (2 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 (2 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 3270 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 3270 CU hardware to ensure complete transmission or receipt of the first and last significant characters of each transmission.

**SYN (Synchronous Idle)**

Two consecutive SYN characters are generated by TCU or 3270 CU hardware to establish character synchronization. The TCU can also embed SYN characters in text for time-fill to maintain synchronization; the 3270 CU discards these SYN characters (does not store them in the buffer).

**DLE (Data Link Escape)**

DLE is always the 1st byte in the following 2-byte control characters: ACK 0, ACK 1, WACK, and RVI. DLE is also used as the 1st 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:

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

ACK 0 is transmitted by the 3270 CU after a successful selection addressing (not poll) sequence to indicate to the TCU that the 3270 CU is ready to accept transmission. ACK 0 is also transmitted by the 3270 CU or by the TCU upon receipt and validation of an even-numbered (2nd, 4th, etc.) text block.

**ACK 1 (Odd Acknowledge)**

ACK 1 is a 2-byte character:

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

ACK 1 is transmitted by the 3270 CU or TCU upon receipt and validation of an odd-numbered (1st, 3rd, etc.) text block.

**NAK (Negative Acknowledgment)**

NAK is transmitted by the 3270 CU in response to a TCU text transmission that (1) terminates with ENQ, (2) has ENQ embedded in text, (3) has invalid BCC (3271), (4) contains a TTD sequence (STX ENQ), or (5) has ETX missing (3271). (The 3275 responds with EOT to a TCU text transmission that has invalid BCC or missing ETX.)

When NAK is received by the 3270 CU in response to a text transmission, the 3270 CU 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 3270 CU; this NAK causes the 3270 CU to send EOT and retain the status for error recovery.*

**ENQ (Enquiry)**

The 3270 CU 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 3270 CU. (See "Programming Note" above.)

When the 3270 CU receives ENQ in response to a transmission, the last 3270 CU transmission to the TCU is repeated. The 3270 CU 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 3270 CU must receive ENQ as the last character of a polling or selection addressing sequence.

**WACK (Wait before Transmit)**

WACK is a 2-byte character:

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

WACK is generated by the 3270 CU (1) in response to a selection addressing (not poll) sequence when a printer (attached to a 3270 CU) or a 3277 attached to a 3271 is busy, and (2) in response to a Write or Copy (3271) command text transmission when the Start Printer bit is set in the WCC or CCC. The 3270 CU responds with ENQ to a WACK from the TCU.

**RVI (Reverse Interrupt)**

RVI is a 2-byte character:

- EBCDIC: 1070 (hex)
- ASCII: 103C (hex)

RVI is generated by the 3270 CU in response to an attempted selection (not poll) by the TCU when the 3270 CU has a status and sense message to be transmitted. Whenever the 3270 CU accepts RVI from the TCU, the CU responds with EOT and resets all pending status and sense information.

**STX (Start of Text)**

The 3270 CU receives STX as the first character of a command or TTD sequence. The STX causes the 3270 CU 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 3270 CU to the TCU as the 1st 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 1st character in status and test-request messages is SOH, with STX following 2 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 3270 CU 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 3270 CU treats ETB as though it were ETX by checking BCC and then generating the appropriate response; the 3270 CU 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 3270 CU 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 3270 CU. 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 3270 CU. Receipt of ETX by the 3270 CU 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 3270 CU (1) when the 3270 CU is a slave station and is unable to perform an operation requested by the TCU, (2) when the 3270 CU is a master station, as normal termination of a read operation, (3) when the 3271 has completed general-poll operations with each attached device, (4) as an answer to RVI sent by the TCU, (5) when the 3275 in text mode has invalid BCC, or (6) when the 3275 ETX is missing. Line synchronization is dropped, and the 3270 CU is returned to control mode. Note that the program can also issue EOT to the 3270 CU in order to drop line synchronization and return the 3270 CU to control mode. EOT does not reset status and sense in the 3270 CU; therefore, it should not be sent as a response to a status message.

**ITB (End of Intermediate Transmission Block)**

The 3270 CU does not accept conventionally blocked text. However, to coexist on a BSC multipoint line on which ITB may be used, the 3270 CU 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 3270 CU 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 3270 CU, as follows: STX, ESC, CMD, ——. The 3270 CU does not generate ESC.

**TTD (Temporary Text Delay)**

TTD is a 2-character sequence: STX ENQ. The 3270 CU responds to TTD by transmitting NAK to the TCU. The 3270 CU does not generate TTD. TTD may also be used by the master station to terminate an operation (that is, initiate a forward abort). The 3270 CU (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 3270 operating on a nonswitched line. Differences for a 3275 with a Dial feature are discussed under "Operational Sequences (Switched Line)." These sequences are divided into four categories:

- Specific and General Poll
- Selection addressing
- Write and control-type commands
- Read-type commands

The description of each category is associated with a sequence/response diagram, which shows (1) all 3270 CU responses to program-generated transmissions by the TCU and (2) normal program-handling of 3270 CU transmissions. These diagrams show the I/O supervisor/access method as examining each 3270 response to determine which operation to initiate next; for specific applications, however, additional use of command chaining in the channel programs may be desirable.

A selection addressing sequence selects a 3270 CU 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 3270 Commands*

For remote operations, 3270 command codes are included in the data stream of a Write CCW to the TCU. Remote chaining of 3270 commands is defined as the transmission of more than one command sequence to a 3270 CU 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 3270 Write command and text block and read the 3270 response for evaluation, and then (3) write a 3270 Write command followed by a second text block and read the 3270 response for evaluation.

The program may chain 3270 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 3270 CU permits any valid command to be chained following a poll sequence; Read Buffer or Read Modified should not be chained, however, because the BSC rules for limited conversational mode (a maximum of two consecutive data transfers without an intervening ACK) will be violated.

Any 3270 command (except Erase All Unprotected) may be chained from a Write, Erase/Write, Erase/Write Alternate, or Copy command. However, if the Write, Erase/Write, Erase/Write Alternate, or Copy command has started a print operation, the 3270 CU will abort the subsequent chained command (the print operation is completed normally).

### *General and Specific Poll Sequences*

When a General or Specific Poll sequence is issued (Figure 5-1), one of three possible results occurs:

- If status and sense information is pending with or without an AID present, a status and sense message is generated.
- If status and sense information is not pending and an AID is present, a Read Modified command is executed.
- If there is no status or sense information or *no* AID pending, an EOT response is generated.

The conditions under which status and sense messages are transmitted are listed under "Error-Recovery Procedures."

**Note:** *When a program attention key is pressed at a 3275 Display Station, and status is not to be sent, the display station screen will momentarily go blank while the AID character is accepted during the polling cycle and a read or write type command reply is sent.*

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

- General Poll address byte sequence:

  3270 CU Poll Address (See Figure 5-2.)
  3270 CU Poll Address (See Figure 5-2.)
  7F (EBCDIC) or 22 (ASCII) } Used in place of the 2
  7F (EBCDIC) or 22 (ASCII) } device-address bytes

- Specific Poll address byte sequence:

  3270 CU Poll Address (See Figure 5-2.)
  3270 CU Poll Address (See Figure 5-2.)
  Device Address }
  Device Address } For the 3275, this is always the address of device 0.

The selected 3270 CU 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 3270 CU 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 3270 CU transmits EOT; the 3270 CU transmits EOT when the 3270 CU has no pending status or messages or after it receives NAK from the TCU in response to a message ending with ENQ.

Specific Poll addresses the 3270 CU and one device to determine whether status and sense information or a manually entered message is awaiting transfer to the TCU. The 3270 CU automatically transfers the pending status and sense information or message upon receipt of the Specific Poll addressing sequence.

General Poll addresses the 3270 CU and examines each attached device in sequence (starting at a random device address) to determine whether 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. The 3275 responds to a General Poll the same as a 3271 with one device attached.

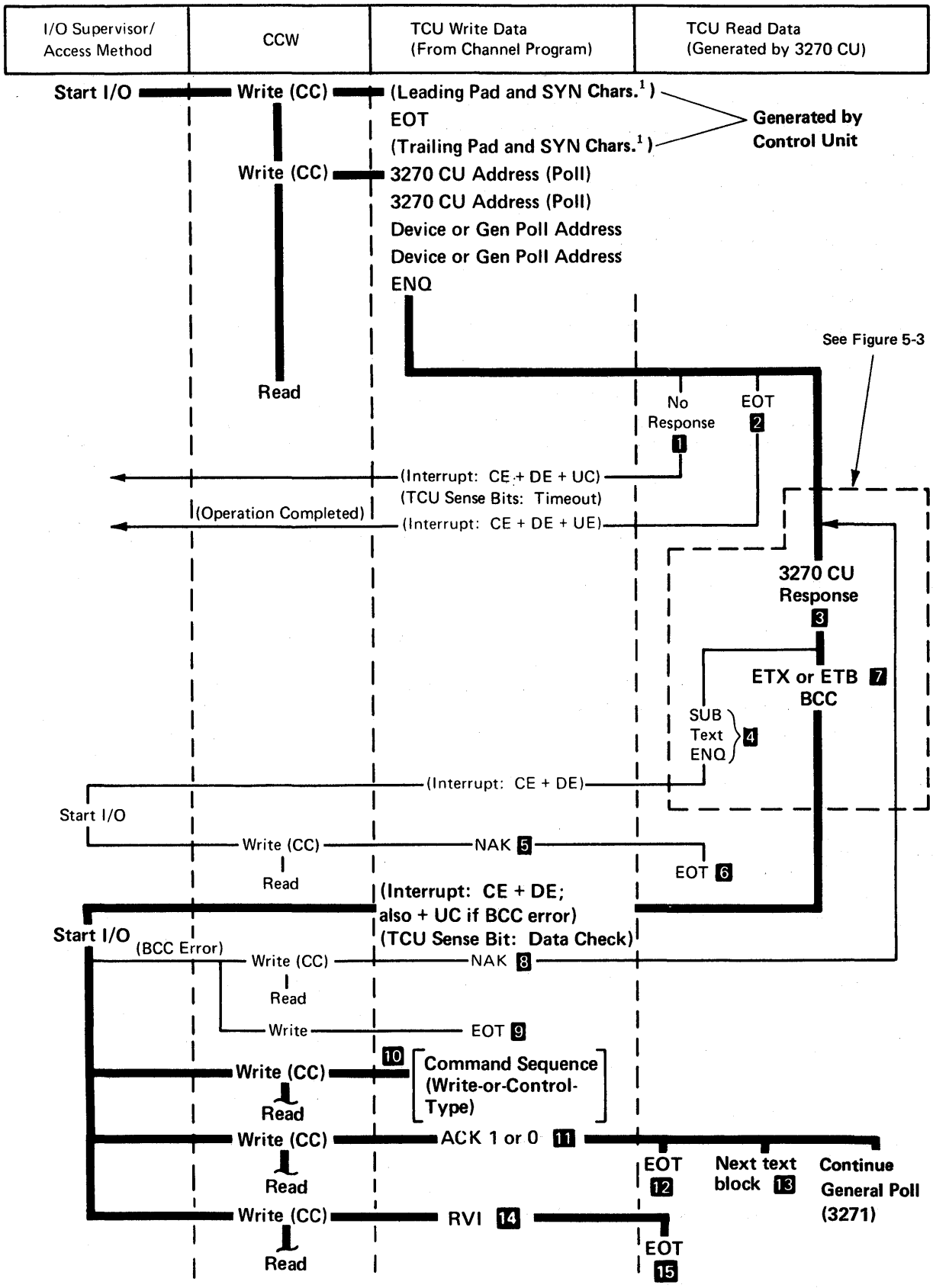
Upon completion of this transfer, an ACK response from the program causes the 3270 CU to continue the General Poll operation, either by transferring another block of a text message or by examining other attached devices for pending messages. The program could issue a command rather than ACK to the device from which the message was just received, only after inbound blocks that end with ETX. Once the 3270 CU 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 3270, forcing an EOT response. A command issued rather than the ACK (after blocks that end with ETX) will also terminate the General Poll.

Figure 5-3 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 3. Note that a device address is not provided in the heading of a Test Request message. The operator must enter an address manually as part of the text, because the operator may specify the address of another device for test operations with the program.

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

### *Selection Addressing Sequence*

The selection addressing sequence (Figure 5-4) specifies a 3270 CU 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




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

**Figure 5-1 (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).
   - The 3275 is *unavailable* to a Specific Poll sequence because the Security Keylock is in the off position.
   - Any character in the polling sequence is invalid.
   - The characters in the polling sequence are out of order.
   - The polling sequence is incomplete (fewer than 7 characters).
   - The 3270 CU address is incorrect in the write data stream.
   - The addressed 3270 CU was left selected from the previous transmission.
2. There is no I/O pending or 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 5-3.)

   3271: 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 3270 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 Note 2 of Figure 5-6 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 3270 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 3270 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 5-5). A read-type command would violate BSC standards on limited conversational mode.

    3271: 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 acknowledgment. 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. This response to a text block terminated in ETX turns on the 3275 SYSTEM AVAILABLE indicator.
12. Normal termination of a Specific Poll.

    3271: Normal termination of a General Poll.

    3275: No additional response is generated by the 3275 at the end of a General Poll.
13. The second and all succeeding 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 interruption (CE = Channel End, DE = Device End, UE = Unit Exception, UC = Unit Check).

1 Reversed numbers refer to notes.

**Figure 5-1 (Part 2 of 2). General Poll and Specific Poll, Sequence/Response Diagram**

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

| CU or Device Number | EBCDIC I/O Char. | EBCDIC (Hex) (Note 1) | ASCII I/O Char. | ASCII (Hex) |
|---|---|---|---|---|
| 0 | SP (Note 2) | 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 | \| 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 1) | 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 3) | 7F | " | 22 |

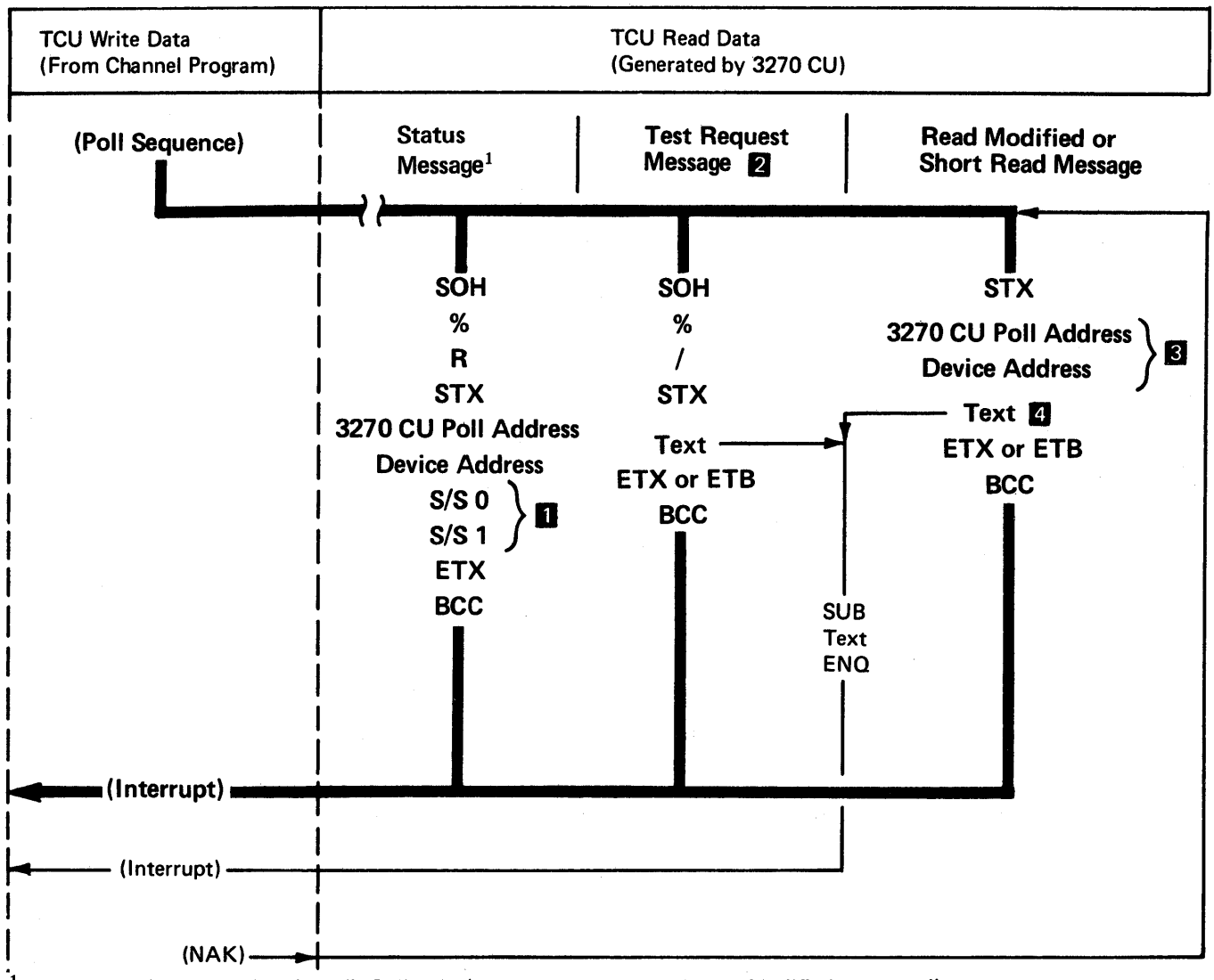
Examples:

| 3271 Addressing | | | | 3275 Addressing | | | |
|---|---|---|---|---|---|---|---|
| | | EBCDIC | ASCII | | | EBCDIC | ASCII |
| General Poll CU5 | CU Address | C5 C5 | 45 45 | General Poll CU5 | CU Address | C5 C5 | 45 45 |
| | Device Address | 7F 7F | 22 22 | | Device Address | 7F 7F | 22 22 |
| Specific Poll Device 4 on CU5 | CU Address | C5 C5 | 45 45 | Specific Poll CU5 | CU Address | C5 C5 | 45 45 |
| | Device Address | C4 C4 | 44 44 | | Device Address | 40 40 | 20 20 |
| Select Device 4 on CU5 | CU Address | E5 E5 | 56 56 | Select CU5 | CU Address | E5 E5 | 56 56 |
| | Device Address | C4 C4 | 44 44 | | Device Address | 40 40 | 20 20 |

**Notes:**

1. *Graphic characters 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.*
2. *I/O character address (SP) is always used as the device address to select a 3275.*
3. *I/O character address (") is used as the device address to specify a General Poll operation.*

**Figure 5-2. Remote Control Unit and Device Addressing – BSC**




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

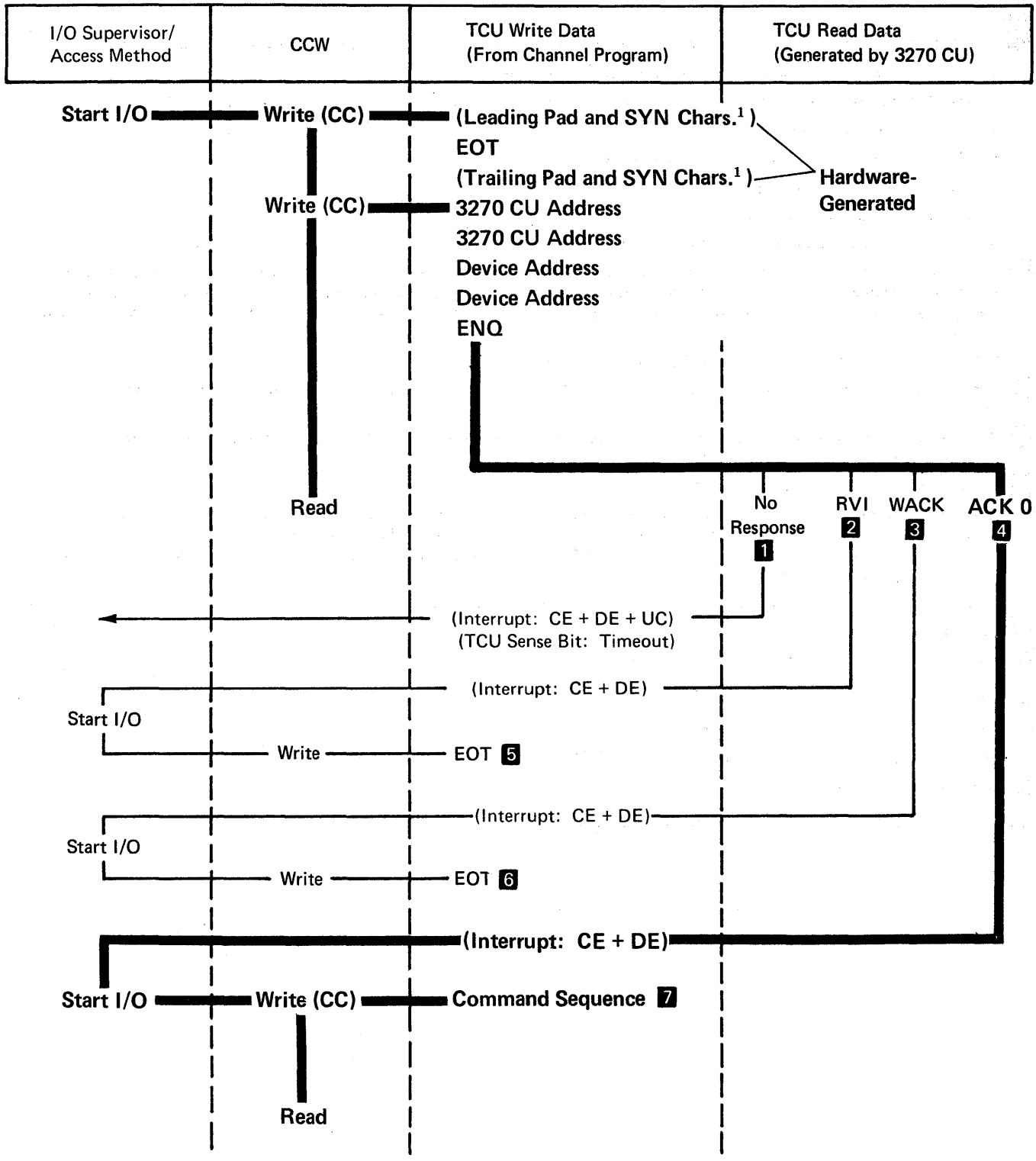
**Notes:**

1. A status message response is issued to a General or Specific Poll if (1) the 3270 CU has pending status (General Poll ignores Device Busy and device *unavailable* and the 3271 continues polling of next device) or (2) error status develops during execution of the poll. Status and sense bit assignments are described in Figure 5-7.
2. A Test Request Message response is issued to a General or Specific Poll if a TEST REQ key is pressed at the keyboard of a polled 3275 or 3277.
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 5-5 lists each operator action and the resulting read operation that will be performed. The read operations and the resulting data are described under "Read Modified Command" in Chapter 3.

LEGEND:
(Interrupt) = TCU-generated interruption.
1 Reversed numbers refer to notes.

**Note:** *This figure is referred to in Figures 5-1 and 5-6.*

Figure 5-3. 3270 CU Message Response to Polling or Read Modified Command




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

**Figure 5-4 (Part 1 of 2). Selection Addressing, 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 3271 is *unavailable* (has power off, is *offline*, or is not attached).
   - The 3275 is *unavailable* because the Security Keylock is in the off position).
   - Any character in the polling sequence is invalid.
   - The characters in the polling sequence are out of order.
   - The polling sequence is incomplete (fewer than 7 characters).
   - The 3270 CU address is incorrect in the write data stream.
   - The addressed 3270 CU was left selected from the previous transmission.
2. 3271: The addressed device has pending status (excluding Device Busy or a Device End) or is unavailable, the device-to-3271 buffer transfer was unsuccessful, the 3271 detected an internal parity or cursor check, or the addressed printer became *not ready* (out of paper, unrecoverable *hang*, power off, or cover open). The S/S information is stored in the 3271, and the internal 3271/device polling is stopped.

   3275: The 3275 has pending status, excluding Device Busy or Device End.
3. The addressed 3271 device or the 3275, including the 3284-3 Printer, 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, and, in the case of the 3271, the device-to-3271 buffer transfer is successfully completed.
5. Termination of attempted addressing sequence:

   3271: The 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 3271. Successful completion of a Specific Poll addressed to the responding device, a device selection addressed to any other device on the same 3271, or a General Poll addressed to the same 3271, is required to start the internal 3271 device polling operation.

   3275: A Specific Poll to the 3275 retrieves the status existing at the time the RVI response was made.
6. Termination of attempted addressing sequence.
7. See Figure 5-5 or 5-6 for the desired command sequence.

LEGEND:

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

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

8. Reversed numbers refer to notes.

**Figure 5-4 (Part 2 of 2). Selection Addressing, Sequence/Response Diagram**

address, but different I/O characters and hex codes are used to represent the CU address bytes. Column 1 in Figure 5-2 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 are given at the bottom of Figure 5-2.

For the 3270 CU, 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 3270 CU returns ACK 0 if the selection and buffer transfer were completed successfully.

When a 3275 is to be selected, note that device number 0 is always addressed (Figure 5-2, Note 2).

### *Write-Type and Control-Type Command Sequences*

The program initiates a Write, Erase/Write, Erase/Write Alternate, Copy, or EAU operation (Figure 5-5) by first writing a command and, except for EAU, a data sequence to the selected 3270 CU and then reading the response. A minimum of 1 data byte (the WCC or CCC byte) must follow all write-type and Copy commands. If the program reads a positive response (ACK) from the 3270 CU, it can terminate the operation or continue with another command. The program can write blocks of text to the 3270 CU by initiating, after receipt of each ACK, a Write command sequence for each block to be written.

The blocking of write data to devices attached to a 3271 control unit is accomplished as follows: Each time the 3271 receives a selection addressing sequence, it transfers the contents of the device buffer to the 3271 buffer before any data is received. After the 3271 has successfully completed execution of the Write command, the contents of the 3271 buffer are transferred to the device buffer. If the transfer of a block of write data to the 3271 is unsuccessful (for example, NAK reply), the 3271-to-device-buffer transfer is not performed. However, the 3271 can receive retransmission of that block; upon receipt of the command, the 3271 retrieves the device-buffer contents (which include any previous text blocks that were written successfully) before any write data is received.

The blocking of write data is of less value with a 3275 since the 3275 buffer is also the device buffer. Thus, if text-blocking is used and the 3275 fails to receive the block successfully, the buffer should be entirely written because orders within the unsuccessful data block may have affected data in any area of the buffer, possibly destroying the integrity of the buffer.

### *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 5-6) by first writing a command sequence to the selected 3270 CU and then reading the response. If the 3270 CU 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 3270 CU and then read the EOT reply. The three types of Read Modified message responses are shown in Figure 5-3.

### *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 5-7. If the sense bytes are transmitted in ASCII code, the EBCDIC code defined below is translated into ASCII before transmission.

The 3270 records status and sense conditions for each device. These conditions may include busy or ready status or detected errors. Figure 5-8 shows how these status and sense conditions are interpreted for each error response that the 3270 transmits in response to a poll sequence from the TCU.

## Error-Recovery Procedures

Errors detected at the 3270 system 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 5-1 and 5-3 through 5-6.

Figure 5-9 lists the various error combinations of sense/status bits [except for 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.

The error-recovery procedures recommended in Figure 5-9 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, after three retries, the operation is not successful, this should be considered a nonrecoverable error. Follow supplementary procedure A.

**Programming Note:** *A cursor check in the 3284 is indistinguishable from a data check that occurred in the 3271 or from a second selection to a 3277 with a cursor check. A selection addressing sequence or a poll sequence to another device on the same control unit should be attempted before the control unit is flagged as inoperative. A successful sequence indicates that the CU is probably satisfactory, and that the device requires manual intervention to reset it (for example, a 3277 with a nonrecoverable data check). An unsuccessful sequence indicates that the CU may be at fault and requires manual intervention to reset it.*

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.

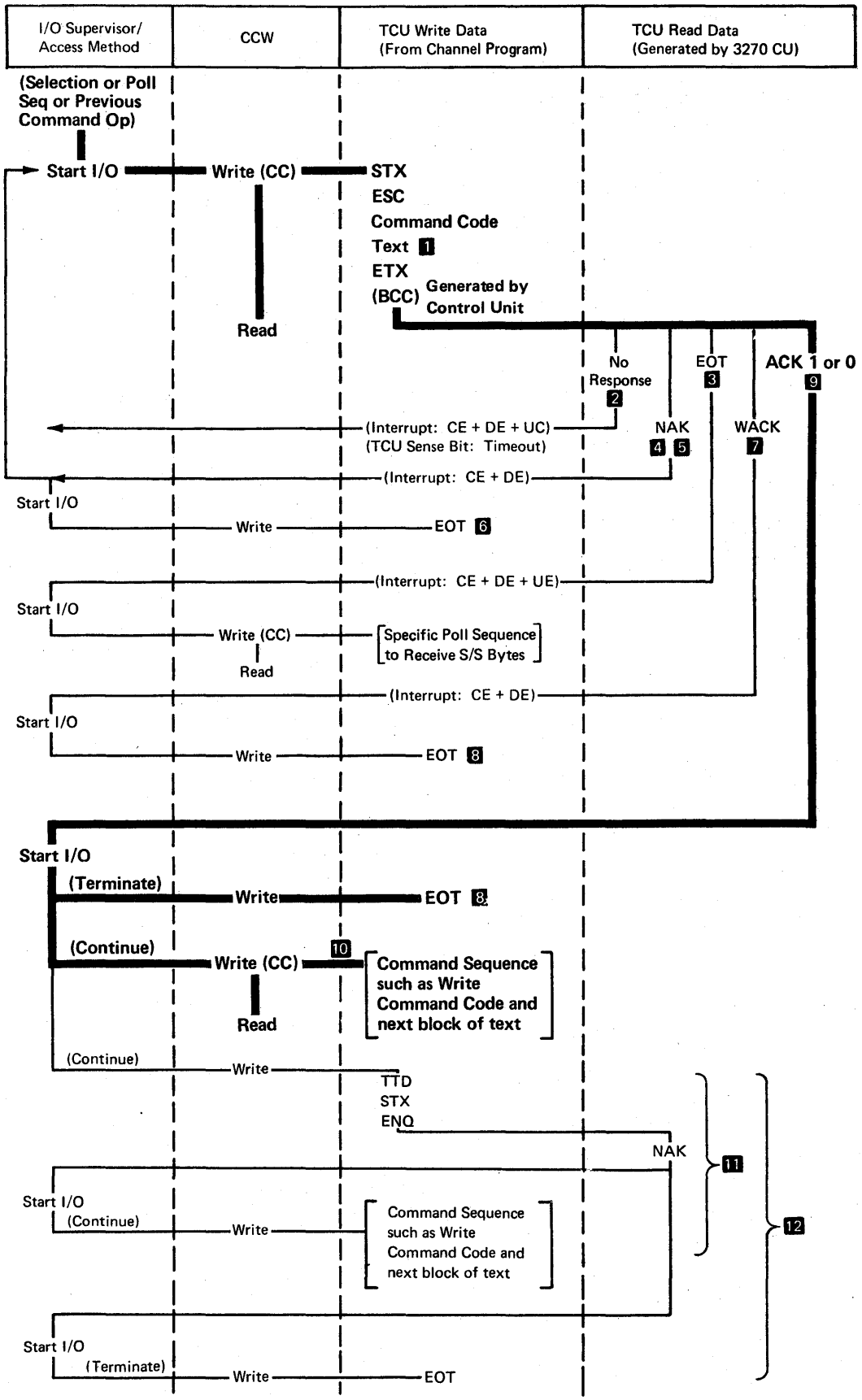



Figure 5-5 (Part 1 of 2). Write-Type and Control-Type Commands, Sequence/Response Diagram—BSC

**Notes:**

1. No text is transmitted on an EAU command transmission.
2. The command transmission was not successfully received, because of invalid framing (STX missing), causing a timeout at the TCU.
3. 3271: The control unit is unable to perform the operation indicated in the command transmission because of a busy/unavailable/not ready device or one of the following 3271-deteted check conditions:
   - Receipt of an illegal command/order sequence
   - Failure to decode a valid command
   - An I/O interface *overrun*
   - A parity/cursor check
   - An illegal buffer address
   - A locked buffer

   In the case of the Copy command: The Copy feature is not installed (3271 only), the *from* device is busy or has a locked buffer, or the CCC is missing.

   The EOT response to a command transmission indicates that status information is stored in the control unit and that internal 3271/ device polling is stopped. To ensure retrieval of valid status, the program must issue a Specific Poll (addressing the device that was selected when EOT was generated) as the next addressing sequence to this control unit. Successful completion of a Specific Poll addressed to the responding device, a device selection addressed to any other device on the same control unit, or a General Poll addressed to the same control unit, is required to restart the internal control unit device polling operation.

   3275: The 3275 is unable to perform the operation indicated in the command transmission because of (1) a BCC error, (2) a busy 3275 (including the attached 3284-3 Printer), or (3) a 3275-detected check condition (receipt of an illegal command/order sequence, failure to decode a valid command, an I/O interface *overrun*, a parity/cursor check, or missing ETX). A Specific Poll to the 3275 retrieves the status existing at the time the EOT response was made.
4. 3271: If a transmission problem causes both a 3271-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. 3271: BCC error or missing ETX has been detected. The NAK response requests the program to repeat its last transmission.

   **Note:** *The 3275 responds with EOT if it detects a BCC error or a missing ETX.*
6. Response issued by the program to terminate the operation if the 3270 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 (and, if 3271, that the 3271-to-device buffer transfer was successfully completed) 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 and, in the case of the 3271, the 3271-to-device buffer transfer is successfully completed.
10. Repeat the operation shown in this figure or in Figure 5-6 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 5-5 (Part 2 of 2). Write-Type and Control-Type Commands, Sequence/Response Diagram–BSC**

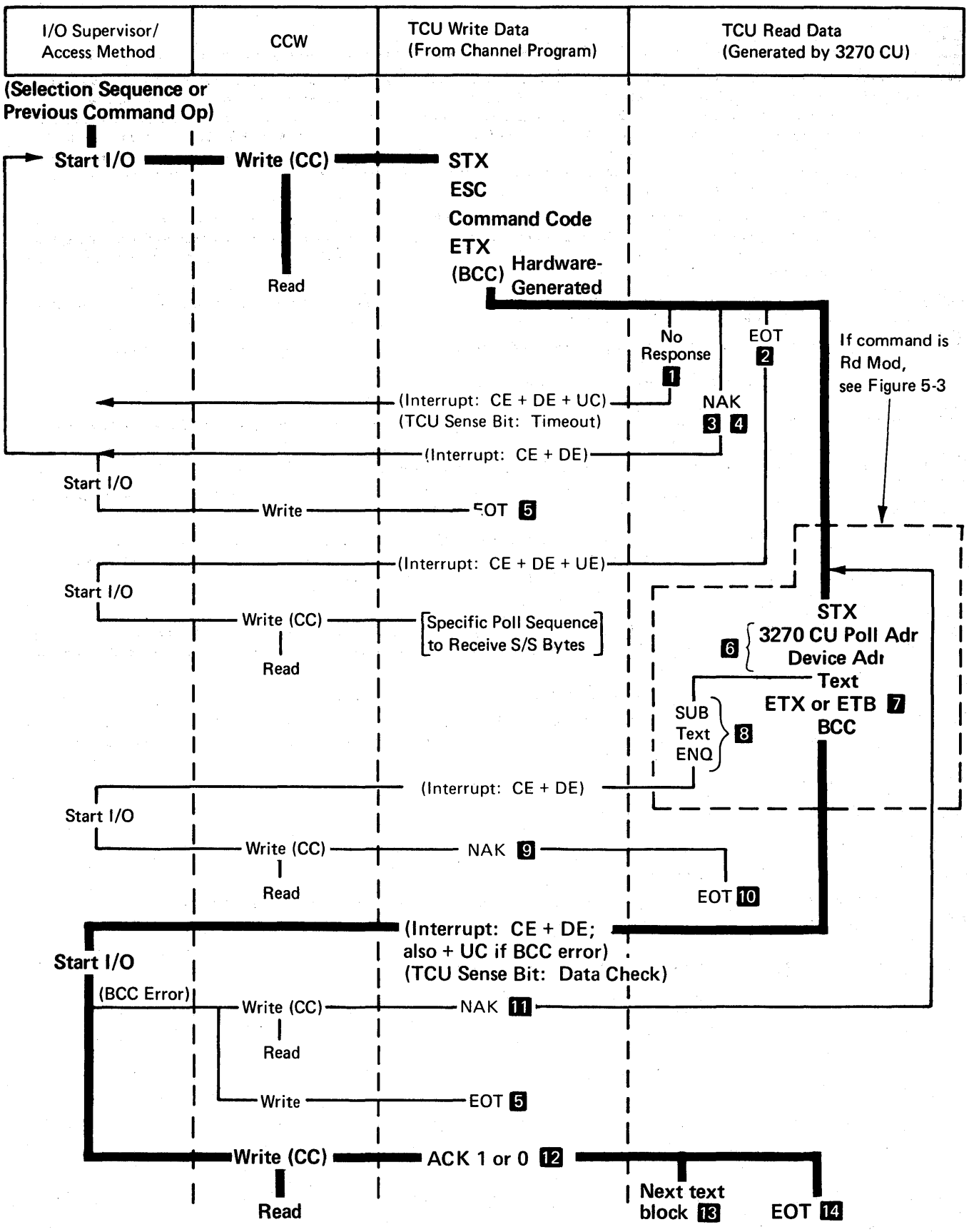



Figure 5-6 (Part 1 of 2). Read-Type Command, Sequence/Response Diagram—BSC

Notes:

1. The command transmission was not successfully received, because of invalid framing (STX missing), causing a timeout at the TCU.
2. 3271, 3275: The 3270 CU is unable to perform the operation indicated in the command transmission because of a busy/unavailable/not ready device or a 3270 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 3270 CU. To ensure retrieval of a valid status, a Specific Poll must be issued to the device-responding EOT as the next addressing sequence issued to this 3270 CU. Internal 3271/device polling is stopped. Restarting the internal 3271 polling operation requires the successful completion of a Specific Poll addressed to the responding device, a device selection addressed to any other device on the same 3271, or a General Poll addressed to the same 3271.

   3275: The 3275 is unable to perform the operation indicated in the command transmission because it (1) has detected a BCC error, (2) is busy (includes an attached 3284-3 Printer), (3) has detected a check condition (has received an illegal command/order sequence, has failed to decode a valid command, or has detected an I/O interface *overrun* or a missing ETX). A Specific Poll to the 3275 retrieves the status existing at the time the EOT response was made.
3. 3270 CU: If a transmission problem causes both a 3270 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. 3271: BCC error or missing ETX has been detected. The NAK response requests the program to repeat its last transmission.

   **Note:** *The 3275 responds with EOT if it detects a BCC error or a missing ETX.*
5. Response issued by the program to terminate the operation if the 3270 CU is unsuccessful in receiving a valid BCC following *n* attempts by the program to transmit the message.
6. This address sequence is included only in the first block of a blocked text message.
7. 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.
8. Upon detection of an internal parity check, the 3270 CU automatically substitutes the SUB character for the character in error. If a parity or cursor check is detected, ENQ is transmitted in place of ETX (or ETB) and BCC at the end of the text block and appropriate status and sense information is stored; also, internal 3271/device polling is stopped.
9. Mandatory program response to a text block terminated in ENQ.
10. 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 and that internal 3271/device polling is stopped. The status retrieval information included in Note 2 applies.
11. BCC error has been detected. The program issues NAK to cause the 3270 CU to repeat its last transmission.
12. Positive acknowledgment. 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. This response to a text block terminated in ETX turns on the device SYSTEM AVAILABLE indicator.
13. The second and all succeeding text blocks are framed as the first except that they do not include the 3270 CU/device address sequence.
14. 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 5-6 (Part 2 of 2). Read-Type Command, Sequence/Response Diagram–BSC**

| 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 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).<br><br>This bit is set with Operation Check when a Copy command is received that specifies a *busy* device with its *from* address.<br><br>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 that started a printer or by chaining a command to a Specific Poll addressed to a busy device. |
| 5 | *Unit Specity (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.<br><br>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 with the other pending status, and an RVI response is made. |
| 7 | *Transmission Check (TC)* — Not used by the 3271. This bit is set when the 3275 detects a BCC error on the TCU transmission. |
| | 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 (or 3271 Copy command if this feature is not installed). |
| 3 | *Intervention Required (IR)* — This bit is set if:<br>• A Copy command contains a *from* address in its data stream that specifies an unavailable device.<br>• A command attempted to start a printer but found it *not ready*. The printout is suppressed.<br>• The 3271 receives a selection-addressing sequence or a Specific Poll sequence for a device that is unavailable or that 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.<br>• The 3271 receives a command for a device that has been logged as unavailable or not ready. |
| 4 | *Equipment Check (EC)* — This bit indicates that a printer character generator or sync check error occurred, the printer became mechanically disabled, or a 3271 detected bad parity from the device. |
| 5 | *Data Check (DC)* — This bit indicates the detection of a parity or cursor check in either the 3271 or a device buffer or in the 3275 buffer, or a 3271 detected bad parity from the device. |
| 6 | *Control Check (CC)* — This bit is not used by the 3275. For the 3271, this bit indicates a timeout check. A timeout check occurs when a device fails to respond to 3271 communications within a specified time or when a device fails to complete an operation within a specified time. |
| 7 | *Operation Check (OC)* — This bit, when set alone, indicates one of the following:<br>• Receipt of an illegal buffer address or of an incomplete order sequence on a Write, Erase/Write, or Erase/Write Alternate command.<br>• The device did not receive a CCC or a *from* address on a Copy command.<br>• Receipt of an invalid command sequence. (ESC is not received in the second data character position of the sequence.)<br>• An I/O interface *overrun* is detected on a 3271. This occurs during a command when a data byte (character or order) is presented to the device by the TCU before the operation required by the previous data byte has been completed.<br><br>This bit is set with Control Check, Intervention Required, Data Check, Device Busy, or Data Check with Unit Specify to indicate that the errors that set these sense bits were detected while the 3271 was executing an operation with the *from* device during a Copy command. 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 5-7. 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.*<br>CC — A timeout check is caused by the addressed device. The operation is tried twice before this bit is set.<br>IR — The addressed device is unavailable.<br>DC, EC (either or both) — The 3271 detects bad parity on data received from the addressed device.<br>DE, EC, US — A character generator or syn check error has occurred, or the printer was mechanically disabled but the condition has been corrected.<br>DE, IR — The addressed printer is out of paper, its power has been turned off, or its cover is open.<br>DE, IR, EC, US — The addressed printer is mechanically disabled and cannot recover.<br>DE, DC, US — A parity error is detected at the printer.<br>DC, US — A parity check or cursor check is detected by the addressed device on the data it is sending to the control unit. |
| EOT | Read Commands | CR — Invalid 3270 command is received.<br>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.<br>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.<br>DB, US, DE — The addressed device becomes *not busy* before a Specific Poll is issued to retrieve the DB, US status.<br>IR — A command is addressed to an unavailable device. (This is not applicable to the 3275.)<br>DC<br>• A cursor check is detected at the 3271 before data transmission starts. The 3271 detects bad parity on data received from the addressed device. The operation is tried twice before this bit is set. No data is transmitted.<br>• A parity check is detected by the 3271 before it is transferred to the TCU. A SUB character is substituted for the error character during transmission. When the transmission is completed, the 3271 sends ENQ to indicate an error. When the TCU responds NAK, the 3271 responds EOT.<br>• A cursor check is detected by the 3271 during transmission to the TCU. When the transmission is completed, the 3271 sends ENQ to indicate an error. When the TCU responds NAK, the 3271 responds EOT.<br>DC, US — The addressed device detects a parity check or cursor check on the data it is sending to the control unit.<br>TC — A BCC error is detected at the 3275. |
| EOT | Write Commands | CR — An invalid or illegal 3270 command is received.<br>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.<br>TC — A BCC error is detected at the 3275.<br>DC — The 3271 detects a parity or cursor check on its buffer during command operation. The 3271 detects bad parity on data received from the addressed device. The operation is tried twice before this bit is set.<br>DC, US — The device detects a parity or cursor check on its buffer during the command operation.<br>CC — The device fails to complete an operation or respond to the 3271 in a certain time (timeout check).<br>DB, US — The addressed device is busy. The message is accepted, but not stored in the 3271 or 3275 buffer. The command is aborted.<br>DE, DB, US — The addressed device becomes *not busy* before a Specific Poll is issued to retrieve the DB, US status (described above). |

Figure 5-8 (Part 1 of 3). 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).<br>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.<br>IR, OC — The *from* device is not available.<br>OC, US — The *from* device has a locked buffer.<br>OC — The data stream contains other than 2 bytes (the CCC and the *from* address). The command is aborted.<br>OC — The *from* device buffer is larger than the *to* device buffer.<br>OC — The buffer of the *from* device (has 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.<br>DC,OC — The 3271 detects a parity check on the data transferred from the *from* device.<br>DC, OC, US — Set when the *from* device detects an internal parity or cursor check.<br>DB, US — The addressed *to* device is busy.<br>DB, US, OC — The addressed *to* device is also specified as the *from* device and is busy.<br>DB, US, OC, DE — The addressed device becomes *not busy* before a Specific Poll is issued to retrieve the DB, US, OC status (described above). |
| EOT | Write, Erase/Write, Erase/Write Alternate, Copy Commands | IR — The addressed device is not available, or the addressed printer is not ready.<br>IR, EC, US — A command attempted to start a printer operation, but the printer CARRIAGE MOTOR POWER switch (a CE service switch) is turned off. |
| EOT | Erase All Unprotected Command<br>Specific and General Poll | OC — One or more data bytes followed the command (buffer overrun).<br>DE, IR, EC, US — An unrecoverable mechanical failure is detected at the printer.<br>DE, EC, US — A character generator or sync check error or a mechanical failure is detected at a 3284/3286/3288 printer, but then recovered from.<br>DC, US — A parity check or cursor check is detected by the addressed device on the data it is sending to the control unit.<br>DC<br>• The 3271 detects a parity error on data to be transferred to the TCU. A SUB character is substituted for the error character during transmission. The transmission is completed, and the 3271 sends ENQ. When the TCU responds NAK, the 3271 responds EOT.<br>• A cursor check is detected at the 3271 before data transmission starts. (No data is transmitted.)<br>• The 3271 detects a cursor check during transmission to the TCU. The transmission is completed, and the 3271 sends ENQ. When the TCU responds NAK, the 3271 responds EOT.<br>DC, EC (either or both) — The 3271 detects a parity check on data received from the device.<br>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. (The 3271 record is updated.) **Note:** *When 3271 power is turned on, the DE bit is set for every available and ready device that is attached.*<br>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. (The 3271 record is updated.) |

**Figure 5-8 (Part 2 of 3). Remote Error Status and Sense Responses – BSC**

| Device Response | Command | S/S Explanation |
|---|---|---|
| EOT | Erase All Unprotected Command<br>Specific and General Poll | DC, US, DE – A parity error is detected at the printer.<br>CC (Specific Poll only) – The poll finds a device, previously recorded as unavailable, still unavailable (timeout check).<br>DC, DE – 3275 (only) detects an internal parity or cursor check on its buffer when the printer goes *not busy*.<br>IR, EC, DE (3275 only) – The printer CARRIAGE MOTOR POWER switch (a CE service switch) is turned off, or a mechanical *hang* condition is detected.<br>EC, DE (3275 only) – Character generator readout error. |
| | Specific Poll | CC – The poll finds a device, previously recorded as available and ready, now unavailable (timeout check). (The 3271 record is updated.)<br>DB – The addressed device is busy. |
| NAK | Read and Write Commands | The 3271 transmits NAK 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 a parity error are detected during the same command transmission, the parity error condition is reset, and a NAK response is set by the 3271. |

**Figure 5-8 (Part 3 of 3). Remote Error Status and Sense Responses – BSC**

| Sense/ Status Bits | Detected during 3270 Operation: Hex EBCDIC | Hex ASCII | Selection Addressing Sequence | Specific Poll Sequence | General Poll Sequence | A 3270 Command | Transmitted in Response to: Specific Poll | General Poll | Error Recovery Procedure: 3271 | 3275 | 3274 | 3276 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CR | 40 60 | 20 2D | | | | D, P | D, P | | 6 | 6 | 6 | 6 |
| OC | 40 C1 | 20 41 | | | | D, P | D, P | | 6 | 6 | 6 | 6 |
| OC, US | C4 C1 | 44 41 | | | | D, P | D, P | | 13 | NA | 13 | 13 |
| CC | 40 C2 | 20 42 | D, P | D, P | | D, P | D, P | | 1 | NA | NA | NA |
| CC, OC | 40 C3 | 20 43 | | | | D, P | D, P | | 1 | NA | NA | NA |
| IR | 40 50 | 20 26 | D, P | D, P | | D, P | D, P | | 4 | 4 | 4 | 4 |
| IR, OC | 40 D1 | 20 4A | | | | D, P | D, P | | 5 | NA | 5 | 5 |
| DC | 40 C4 | 20 44 | D, P | D, P | D, P | D, P | D, P | D, P | 1, 2 | 2 | 1 | 2 |
| EC | 40 C8 | 20 48 | D, P | D, P | D, P | D, P | D, P | D, P | 1, 2[1] | 2 | NA | NA |
| DC, EC | 40 4C | 20 3C | D, P | D, P | D, P | D, P | D, P | D, P | 1, 2[1] | 2 | NA | NA |
| DC, OC | 40 C5 | 20 45 | | | | D, P | D, P | | 1 | NA | NA | NA |
| DC, US | C4 C4 | 44 44 | D, P | D, P | D, P | D, P | D, P | D, P | 2 | NA | 2 | 2 |
| DC, OC, US | C4 C5 | 44 45 | | | | D, P | D, P | | 3 | NA | 3 | 3 |
| DC, DE | C2 C4 | 42 44 | | P | P | | | P | NA | 8 | NA | NA |
| DC, US, DE | C6 C4 | 46 44 | | P | P | | P | P | 8 | NA | 8 | 8 |
| IR, DE | C2 50 | 42 26 | | P | P | | P | P | 4 | 4 | 4 | 4 |
| IR, EC, DE | C2 D8 | 42 51 | | P | P | | P | P | NA | 7 | NA | NA |
| EC, DE | C2 C8 | 42 48 | | P | P | | P | P | NA | 7 | NA | NA |
| EC, US, DE | C6 C8 | 46 48 | | P | P | | P | P | 7 | NA | 7[2] | NA |
| IR, EC, US, DE | C6 D8 | 46 51 | | P | P | | P | P | 7 | NA | 7 | 7 |
| DB | C8 40 | 48 20 | D, P | D, P | | | D, P | | 9 | 9 | 9 | 9 |
| DB, DE[3] | 4A 40 | 54 20 | | | | | D | | 9 | NA | NA | NA |
| DB, US[4] | 4C 40 | 3C 20 | | | | D, P | D, P | | 10 | 10 | 10 | 10 |
| DB, US, DE | 4E 40 | 2B 20 | | | | D, P | D, P | | 1 | 1 | NA | NA |
| OC, DB[4] | C8 C1 | 48 41 | | | | D, P | D, P | | 11 | NA | 11 | 11 |
| TC | C1 40 | 41 20 | | | | D | D | | NA | 12 | NA | NA |
| TC, OC | C1 C1 | 41 41 | | | | D | D | | NA | 12 | NA | NA |
| TC, CR | C1 60 | 41 2D | | | | D | D | | NA | 12 | NA | NA |
| TC, DC | C1 C4 | 41 44 | | | | D | D | | NA | 12 | NA | NA |
| DE | C2 40 | 42 20 | | D, P | D, P | | D, P | D, P | None | None | None | None |
| IR, EC, US | C4 D8 | 44 51 | | | | P | P | | 7 | NA | NA | NA |
| CC, IR | 40 D2 | 20 4B | | D, P | D, P | D, P | D, P | | 1 | NA | NA | 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.*

[1] Perform error recovery procedure 1 if the error occurred during a read operation. Perform error recovery 2 if the error occurred during a write operation.

[2] Occurs only if the 3284, 3286, or 3288 printers are attached.

[3] The DB and DE S/S bits can occur together in response to a Specific Poll to a formatted 3277 if the operator has performed Backtab or Erase Input operations in rapid succession. Ignore Device End, and treat as Device Busy only.

[4] 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 *to* device.

Legend

NA — Not applicable

D — Display (3275, 3277)

P — Printer

**Figure 5-9. Remote Status and Sense Conditions–BSC**

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 eror indicates that the *from* device specified by a Copy command is unavailable. 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 not more than six times. Continued failure implies an application programming problem, which can be detected by analysis of 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, and 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, during an attempt 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. Indicates that the 3275 detected a BCC error during text transmission from the TCU. Follow procedure 2 if the failing command is a Write command with a data stream of more than 1 byte or if it is in a chain of commands and one of the previous commands in the chain is a Write command without an SBA order immediately following the WCC character. In all other cases, follow supplementary procedure D. If, after the recommended procedure has been tried six times, the problem is not corrected, follow supplementary procedure A.
13. 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 3271 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 1 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).

**Notes:**

1. *When the 3275 detects a BCC error, it will set the transmission-check (TC) sense/status bit and respond EOT.*
2. *An FF (hex) character in a data field will cause a BCC error.*

### *EOT to a Text Block*

The recovery procedure recommended depends upon the type of error detected. A Specific Poll must be issued immediately following the EOT to obtain the error sense/status information. (If the Dial feature is installed, a Specific Poll is not needed, because the 3275 automatically bids for the line present sense/status information.) Then the recovery procedures recommended in Figure 5-9 should be executed.

### *Errors Detected during a Specific or General Poll Sequence*

Any errors that result from execution of the poll sequence itself are listed in Figure 5-9, 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 5-9 should be followed.

## Point-to-Point (Switched Line) Data Link Control

A 3275 with the Dial feature operates on a point-to-point, switched communication line. Data exchange takes place between a 3275 and a TCU, but not between 3275s.

### *Terminal Identification*

Four terminal ID characters (4 bytes) are wired into each 3275 with the Dial feature. Only graphic characters can be assigned. The 1 character for 3270 devices is always f (for EBCDIC units) or F (for ASCII units). The remaining 3 characters can be assigned by IBM or by the customer at the customer's location. The non-IBM-assigned terminal ID characters consist of numbers and uppercase letters only. IBM-assigned terminal ID characters consist of lowercase letters and special graphics.

## *Contention Line Discipline*

### Bid Sequence

In switched-line operation, the stations are normally disconnected. When the TCU is dialed from a 3275, or a 3275 is dialed by the TCU and a connection is successfully made (with both stations in data mode), the data link is in point-to-point contention. Once a connection is made, either station can bid to become the control station by sending a terminal identification sequence. Normally, the control station would be the station that initiated the connection. The initial 3275 bid sequence is made up of the 4 terminal ID characters, followed by the character ENQ. Subsequent bids by the 3275 transmit only the ENQ character. The TCU bids for the line by sending the computer ID-ENQ sequence only during the first transmission and ENQ on the following bids. The bid sequence is used to maintain line discipline.

**Note:** *In the switched-line environment, the 3275 does not operate in transparent monitor mode.*

### 3275-Initiated Call

The telephone number of the desired computer system is dialed by the 3275 operator. Upon recognition of the answer tone from the called station, the modem (or line adapter) is automatically or manually switched into data mode. The 3275 operator then presses an attention ID (AID) key, usually ENTER, causing the following actions:

1. Disables the keyboard (except for the RESET key).
2. Turns on the INPUT INHIBITED indicator.
3. Initiates a bid for the line which, when successful, transfers a text message. The form of the message depends upon the key pressed (see Figure 3-2).

The SYSTEM AVAILABLE indicator's coming back on indicates to the operator that the 3275's message has been successfully transmitted. The operator can then press the RESET key, enabling the keyboard for transmitting another message, or disconnect, as desired. The keyboard can also be enabled by the computer's responding with a Write, Erase/Write, or EAU command and with the appropriate WCC.

### Computer-Initiated Call

A 3275 with the Dial feature can be called from the computer. If an external modem, wired for auto answer, or the 1200-bps integrated modem with the Auto Answer feature is used, the 3275 can answer a call unattended. This is of use when the 3275 is unattended and a printer is attached.

An external modem or the 1200-bps integrated modem with Auto Answer feature will, upon recognizing the ringing signal, initiate off-hook, send an answer tone to the TCU, and automatically switch into data mode. The computer then begins transmission by sending a bid sequence.

In manual operation, the 3275 operator recognizes the ringing signal, lifts the telephone receiver (goes off-hook), and activates the exclusion key on the handset.

In all cases, data mode is indicated to the 3275 operator. In the manual case, data mode is implied by the handset's being out of the cradle. In the automatic case, an OFF HOOK indicator on the 3275 implies data mode.

### Disconnection

Disconnection is the process of terminating a call. During this action, both stations should perform the disconnection. If only one station disconnects, the other station can stay connected and appear busy to incoming calls.

**Manual Disconnection.** To disconnect a 3275 manually, the operator must:

1. Raise and release the DISCONNECT switch on the 3275, causing the 3275 to send the disconnect sequence line control characters, DLE EOT. If the 3275 has an external modem wired for auto answer or a 1200-bps integrated modem with Auto Answer feature, the connection is automatically terminated.
2. On a 3275 without auto answer, the 3275 operator must replace the handset on-hook to achieve disconnection at the 3275. Replacing the handset restores the exclusion key to the talk position and disconnects the call. The handset should be cradled only following activation of the DISCONNECT switch, as confirmed by the SYSTEM READY indicator's turning off.

**Automatic Disconnection.** There are two ways to disconnect automatically, both requiring auto answer, either in an external modem or as part of the 2300-bps integrated modem:

1. By receipt of the disconnect sequence line control characters, DLE EOT.
2. By a 20-second timeout, which is enabled when the 3275 receives a ring signal from the CPU. The 20-second timeout is initiated each time a station transmits a valid header, text, response, or control transmission. It is reset each time a station receives two SYN characters from the line. Failing to reset the timer within 20 seconds causes the disconnect sequence of DLE EOT to be transmitted and causes the telephone to be hung up.

## *Data Link Control Characters*

The use of some link control characters in the 3275 with the Dial feature differs from the use of those in the basic 3275, as follows.

### ACK 0 and ACK 1 (Positive Acknowledgment)

When the 3275 responds to an initial bid for the line, the terminal ID precedes ACK 0. When an initial bid has been successfully completed, subsequent bids use only ACK 0.

The use of ACK 0 and ACK 1 to acknowledge data blocks positively is the same as for the basic 3275.

### NAK (Negative Acknowledgment)

When the 3275 is called by the computer, but has pending status other than *printer busy*, the 3275 responds to the initial bid for the line with the terminal ID preceding NAK. NAK alone precedes all further bids for the line when status is pending. The 3270 CU transmits NAK in response to a text transmission that contains a TTD sequence (STX ENQ). When the 3275 receives the NAK in response to a text transmission, the 3275 retransmits the last block of text.

### ENQ (Enquiry)

This character is transmitted by either station to bid for the line any time after it has transmitted or received EOT. However, ENQ is preceded by the terminal ID when the 3275 is making an initial bid for the line, and by the last character of a text message in which data check was detected by the 3275.

When the 3275 receives ENQ in response to a transmission, the last 3275 transmission to the TCU is repeated. The 3275 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).

**RVI (Reverse Interrupt)**

Upon receipt of the RVI character, the 3275 with the Dial feature completes its buffer transfer before sending EOT.

**EOT (End of Transmission)**

EOT is transmitted by the master station (usully the caller) to indicate end of transmission. Either station is free to bid for the line following the EOT character.

When used as a response to a text block, EOT indicates that status is pending.

**DLE EOT (Disconnect)**

The DLE EOT is the disconnect signal. Any 3275 with the Dial feature can transmit DLE EOT (initiated by activating the DISCONNECT switch). However, only units that are equipped with auto answer have the ability to disconnect automatically.

## Operational Sequences (Switched Line)

The following paragraphs describe the various data and control sequences that are unique to the 3275 with Dial feature operating on a switched line. Because operation is initiated differently from that of the basic 3275 operating on a leased line, neither selection nor polling applies to point-to-point contention operation. 3270 commands can be chained as described under "Remote Chaining of 3270 Commands."

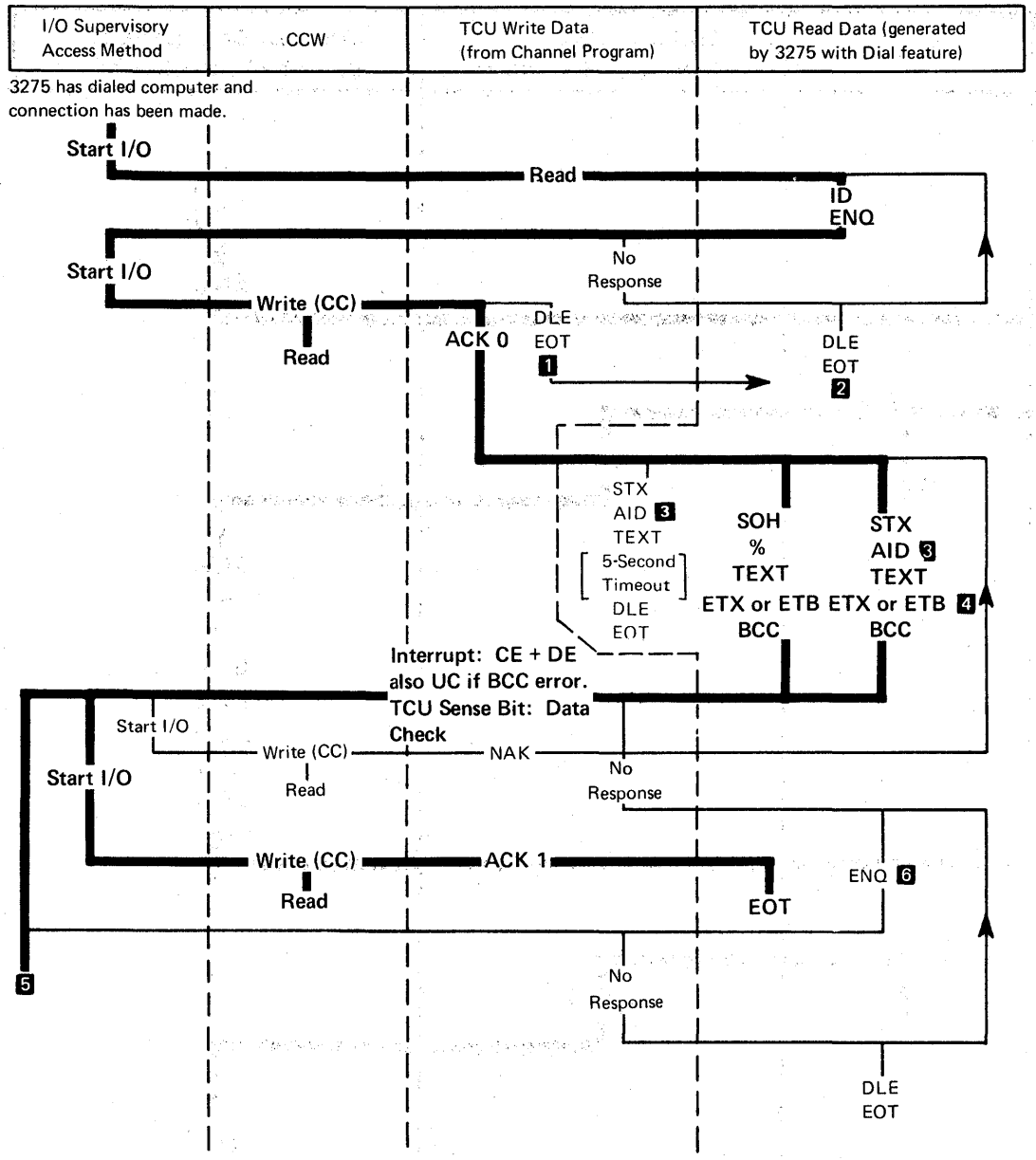
### *3275-Initiated Sequences*

The 3275 with the Dial feature does not need a read-type command, including a poll, to start transmission of text entered into the buffer or a status message. Normally, a 3275 operator who intends to transmit a text message to the computer enters this message by keyboard into the buffer. After correction of keying errors, the computer is dialed. After the connection has been made, the operator presses an attention key (Figure 3-2). Pressing this key causes the 3275 to bid for the line by sending its 4 assigned terminal ID characters and ENQ.

Receiving a positive acknowledgment ends the identification phase and allows the 3275 to enter the data exchange phase. In the latter phase, assuming no status is pending, the 3275 transmits a text message that is identical with messages generated by the read-modified operation in the basic 3275 (see Figure 5-10). If status is pending, the 3275 transmits a status message (see Figure 5-11).

### *TCU-Initiated Sequences*

The 3275 with the Dial feature can be called by the computer. The computer bids for the line with a computer ID-ENQ sequence or by sending ENQ only. (The computer ID of up to 15 characters is not decoded by the 3275.) When the 3275 responds with an ACK 0 or NAK to the initial line bid, the response character is prefaced by the 4 terminal ID characters. The program can then continue, as appropriate. Refer to Figure 5-12.




**Notes:**

1. Upon correct reception of an invalid terminal ID, the computer disconnects. The TCU may optionally send DLE EOT before disconnecting. This is defined in the BSC rules as an *unusual termination*.
2. The 3275 retries three times. When the number of retries is exhausted, the 3275 sends DLE EOT.
3. AID indicates which situation caused attention.
4. 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.
5. The remainder of this sequence/response diagram is the same as that for a General or Specific Poll, as shown in Figure 5-1.
6. The 3275 as the master station solicits a response by sending ENQ. After the number of retries is exhausted, the 3275 acts as described in Note 2

LEGEND:

1. Reversed numbers refer to notes.

Figure 5-10. 3275-Initiated Transmission, Sequence/Response Diagram

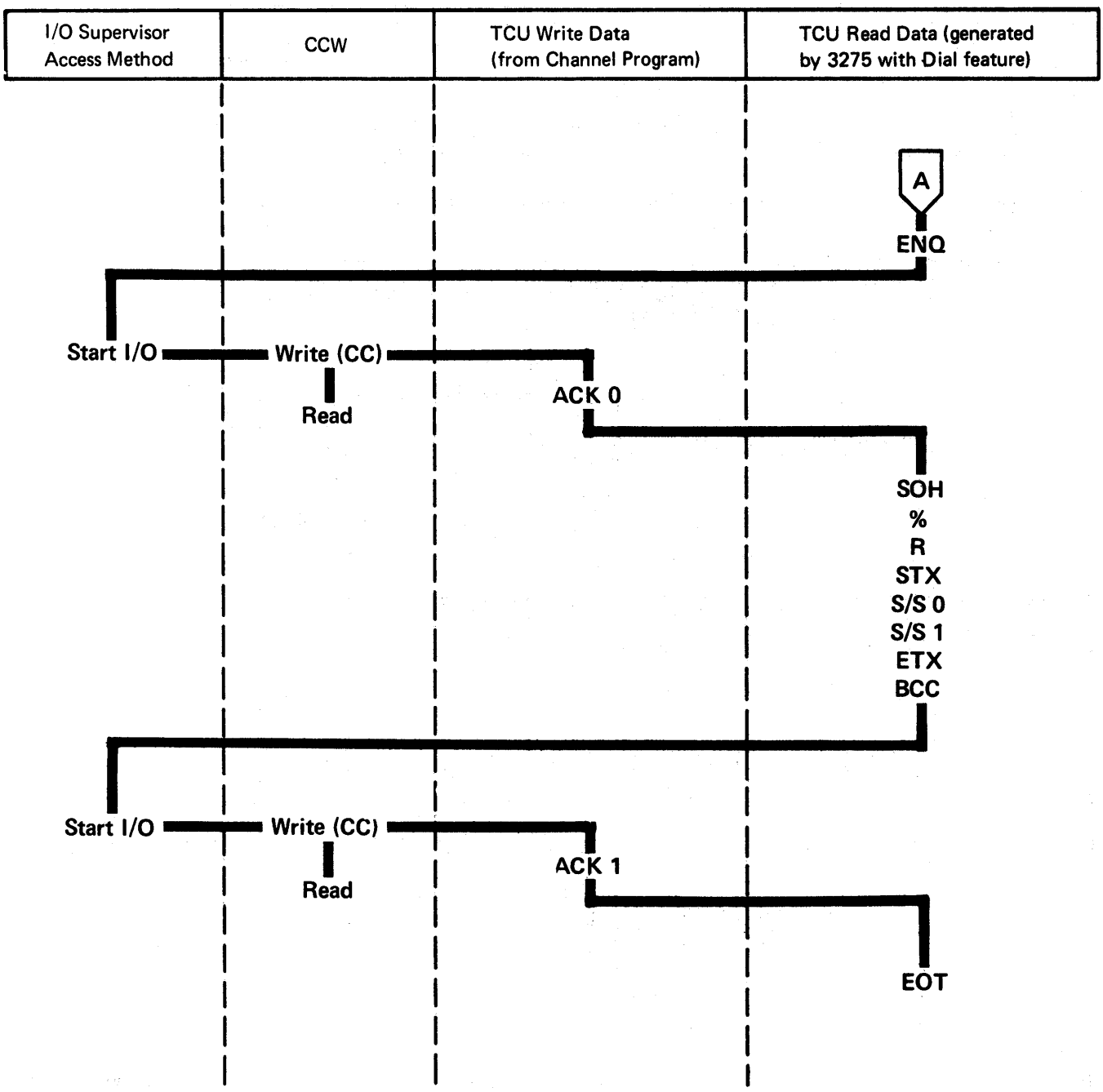
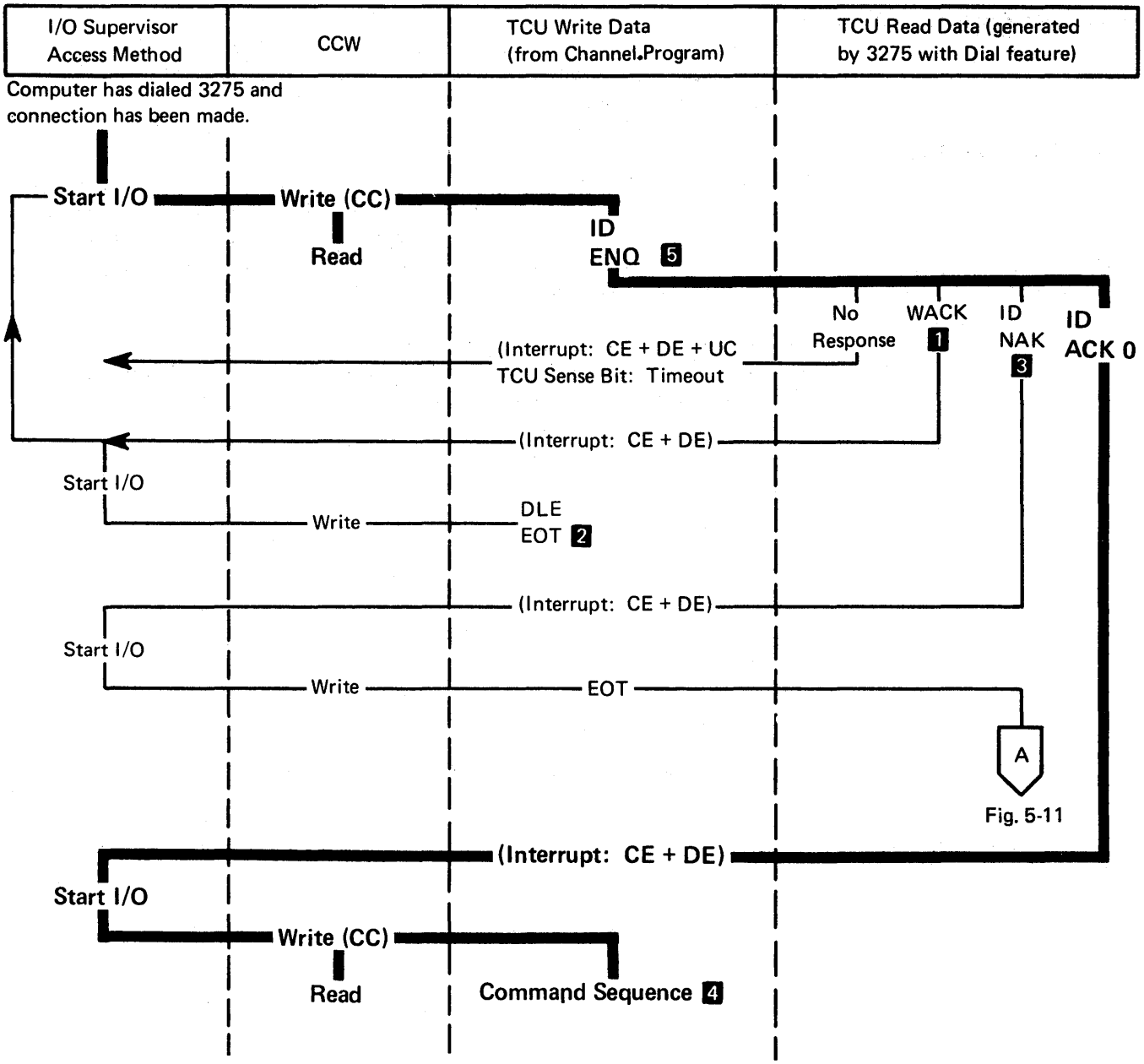



Figure 5-11 Status Message Transmission with Dial Feature, Sequence/Response Diagram




**Notes:**

1. The 3275 is not ready to receive, because of a printer, keyboard, or card reader operation.
2. The TCU should transmit DLE EOT before disconnecting. The 3275 with the Auto Answer feature will recognize DLE EOT and automatically disconnect.
3. The 3275 has status pending other than a busy printer and is not ready to receive. The 3275 monitors for EOT and prepares transmission of a status message.
4. Refer to Figure 5-5 or 5-6 for the desired command sequence.
5. Not decoded or used by the 3275.

LEGEND:

1 Reversed numbers refer to notes.

Figure 5-12. TCU-Initiated Transmission, Sequence/Response Diagram

### *Maintained Connection Sequences*

Once either station has signaled EOT, either station can bid for the line with ENQ without further use of a computer ID or terminal ID. The response to the bid need not be preceded by the ID either. See Figure 5-13 for an example.

### *Device Busy and Device End*

It is possible for a TCU line bid to find the terminal busy because of a printer, keyboard, or operator identification card reader operation. To an initial bid for the line, the busy 3275 responds WACK. The TCU might then either respond with a disconnection sequence DLE EOT or enter an ENQ/WACK loop, waiting for the busy-causing operation to end as indicated by a terminal ID-ACK response.

To a TCU line bid during a maintained connection, the busy 3275 also sends WACK. In this case, the program has a third choice of responding with just EOT. With EOT, the 3275 bids for the line and sends the device end status when the busy-causing operation ends.

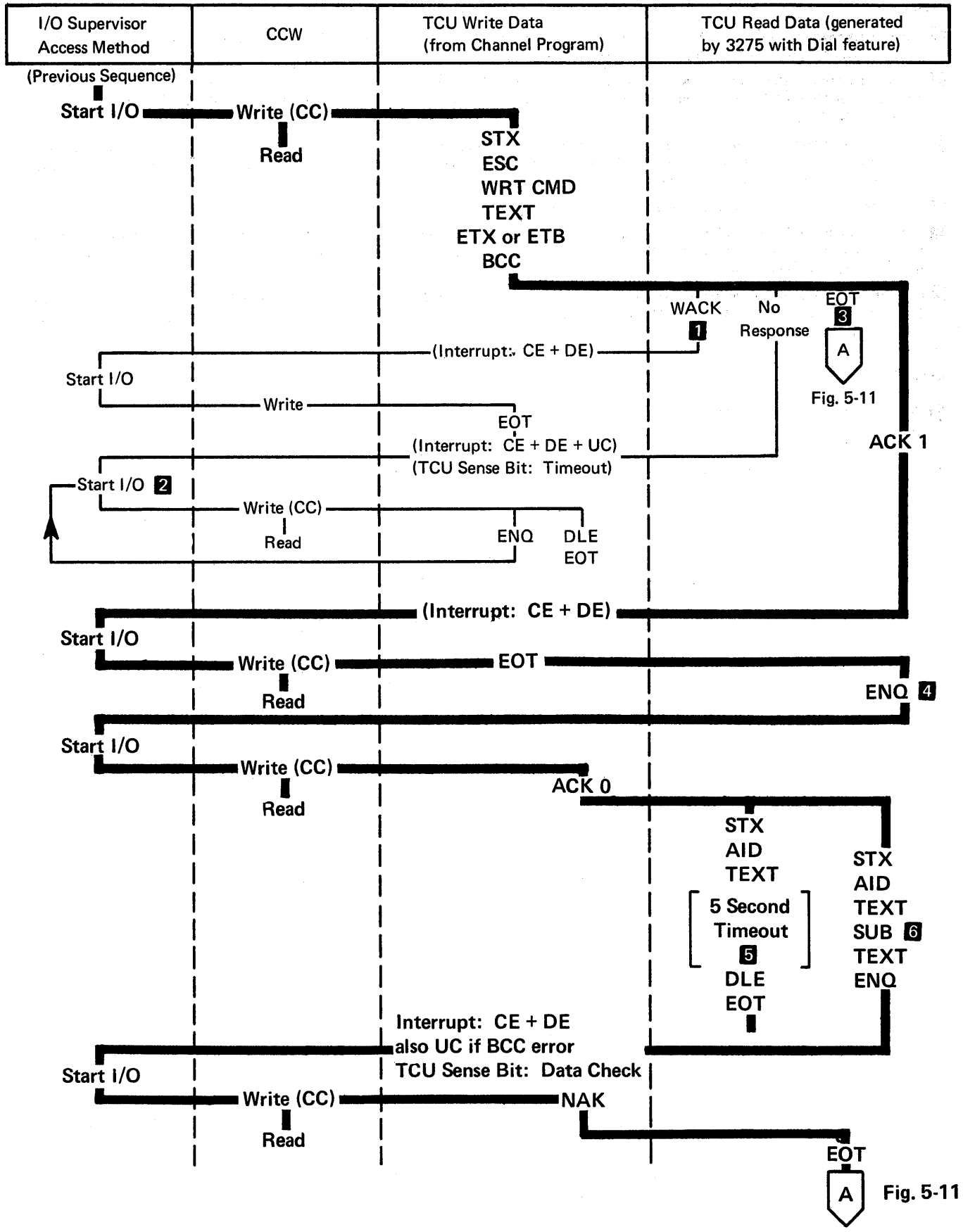



Figure 5-13 (Part 1 of 2). Example of Maintained Connection, Sequence/Response Diagram

**Notes:**

1. Positive acknowledgment, when the printer bit has been set in the Write Control Character (WCC) included with the Write command issued to a 3275 with attached printer. The printer is now busy.
2. The 3275 as the master station solicits a response by sending ENQ. After three retries, the 3275 that is equipped with the Auto Answer feature sends DLE EOT and disconnects automatically. The 3275 that is not so equipped sends DLE EOT. The operator should then manually disconnect.
3. The 3275 aborts because it is unable to receive or to execute the command. This condition causes status to be set and the transmission of a status message to be prepared. This situation could have been caused as the result of a command in a chain following a start-print operation or as the result of a BCC error.
4. The connection is still maintained. The 3275 has prepared another text message and bids for the line.
5. Here, it is assumed that the 3275 cannot complete transmission, because of a malfunction other than an internal parity check. A 5-second transmission timeout becomes effective, the uncompleted text transmission is terminated by DLE EOT, and, with auto answer installed. the telephone is automatically hung up.
6. Here, it is assumed that an internal parity error has been detected and the SUB character has been substituted for the character in error. The text block is terminated by ENQ. The mandatory response is NAK. In this situation, the 3275 is preparing for the transmission of a status message.

LEGEND:

1. Reversed numbers refer to notes.

**Figure 5-13 (Part 2 of 2). Example of Maintained Connection, Sequence/Response Diagram**

# Chapter 6. Remote Operations – SDLC (3271 and 3275 Models 11 and 12)

## Introduction

The 3271 Control Unit Models 11 and 12 and the 3275 Display Station Models 11 and 12 use synchronous data link control (SDLC) mode of operation and communicate, as terminal nodes, with the program via an IBM 3704 or 3705 Communications Controller and appropriate modems.

**Note:** *In the following paragraphs, the term* 3270 CU *is used in statements that apply to both a 3271 and a 3275. If a statement applies to only one 3270 unit, the appropriate unit number is used. The term* controller *is used in statements that apply to the 3704 and 3705 Communications Controllers.*

The 3270 CU that uses SDLC procedures provides half-duplex transmission over duplex or half-duplex facilities (nonswitched or privately owned). These communications use the multipoint data link mode of operation only.

When employing SDLC line discipline, the 3270 CU operates in extended binary-coded decimal interchange code (EBCDIC) or American National Standard Code for Information Interchange (ASCII) and performs as a PU type 1 unit.

## Related Publications

The line discipline for management of information transfer between the controller and the 3270 CU, SDLC is one of several logical elements that the total communication system network comprises. The remainder of that network consists of the controller and the host System/370. The operation of the total communication system network is governed by an overall group of procedures and protocols, referred to as Systems Network Architecture (SNA).

This chapter makes use of SDLC terms and a limited number of SNA terms. Only a few SDLC terms are defined herein. Readers who are unfamiliar with SDLC concepts and terminology should review the *IBM Synchronous Data Link Control General Information* manual, GA27-3093. Readers who require an understanding of SNA should refer to the *IBM Systems Network Architecture General Information* manual, GA27-3102. A functional description of the controllers is given in the *Introduction to the IBM 3704 and 3705 Communications Controller,* GA27-3051.

An aid to programming the 3270 in this discipline can be found in *Introduction to Programming the 3270 Information Display System*, GC27-6999.

## Multipoint (Nonswitched Line) Data Link Control

Each 3270 CU can operate on a nonswitched communication line with multiple stations. Time-sharing of the line is accomplished by interleaving transmissions between the controller and all units on the line. A 3271 Model 11 or 12 or 3275 Model 11 or 12 operates multidropped on the same line with properly featured units, such as other 3270 units employing SDLC, IBM 3601 Finance Communication Controllers, and IBM 3791 Controllers.

The controller is called the *primary* station of the multipoint network and controls operation of the communication link. All units attached by communication line to the controller are called *secondary* stations. The primary 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 primary station is either the transmitter or the receiver of every communication. Secondary stations receive primary station controls and information and, as a result, initiate transmissions (responses and information) depending upon the specific command.

## SDLC Transmission Blocks

SDLC transmission blocks are called *frames*. Frames, as defined for 3270 application, consist of a series of 8-bit, binary-coded bytes containing data and control information transmitted between the controller and the 3270 CU. Frames are subdivided into the following types of information, transmitted in the sequence listed:

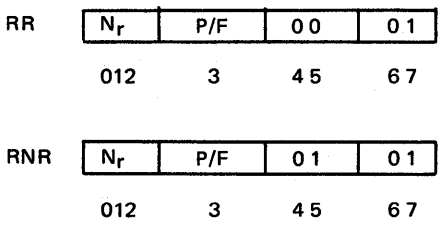
1. Flag (F) sequence – 1 byte
2. Control unit address (A) – 1 byte
3. Control (C) field – 1 byte
4. Information (I) field – up to 256 bytes of message data preceded by header information
5. Frame check sequence (FCS) – 2 bytes
6. Flag (F) sequence – 1 byte

When sending information to the host system, these units operate in modulo-3 mode – that is, up to two frames at a time. When receiving information from the host system, they operate in modulo-8 mode. Note that the information in any particular transmission must be associated with only one device.

An information field is required within the frame only when message data is to be transmitted. The descriptions of the components of the SDLC frame, as given in *IBM Synchronous Data Link Control General Information*, GA27-3093, are applicable to the 3270 system, with the following qualifications:

- The 3270 system makes use of the Receive Ready (RR) and Receive Not Ready (RNR) supervisory commands and responses only. RR and RNR responses are always sent by the 3270 CU with the final bit set to 1.

  The C-field byte formats for RR and RNR are as follows:



- The nonsequenced commands and responses employed by the 3270 system are limited to the following:

| Command/Response | C-Field | Hex Code | |
|---|---|---|---|
| Set Normal Response Mode (SNRM) command | 1 0 0 P 0 0 1 1<br>0 1 2 3 4 5 6 7 | 93 | * |
| Disconnect (DISC) command | 0 1 0 P 0 0 1 1<br>0 1 2 3 4 5 6 7 | 53 | * |
| Unnumbered Acknowledgment (UA) response | 0 1 1 F 0 0 1 1<br>0 1 2 3 4 5 6 7 | 73 | * |
| Disconnect Mode (DM) | 0 0 0 F 1 1 1 1<br>0 1 2 3 4 5 6 7 | 1F | * |
| Frame Reject (FRMR) response | 1 0 0 F 0 1 1 1<br>0 1 2 3 4 5 6 7 | 97 | * |
| Link Test | 1 1 1 P/F 0 0 1 1<br>0 1 2 3 4 5 6 7 | F3 | ** |

*Described in the *IBM Synchronous Data Link Control, General Information* manual, Form No. GA27-3093.
**Described in this section.

### *Link Test Command/Response*

The Link Test command/response is a basic test of the data link between the controller and the 3270 CU. When the controller sends the Link Test command, the 3270 CU checks that the FCS field is valid and that the C-field poll bit is set to 1. Data may be sent to the 3270 CU that is included in the nonsequenced frame. If the command is received correctly, the 3270 CU sends the Link Test response to the controller. Data is not sent by the 3270 CU.

### *Information (I) Field*

An information (I) field is required when message text is transmitted in either direction between the controller and the 3270 CU. The C-field format, which indicates that an I-field is being sent, is:

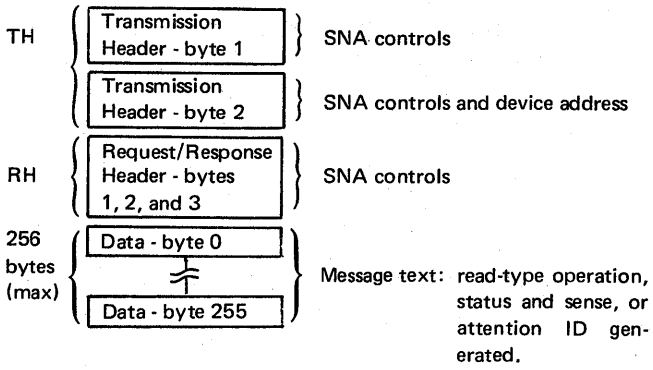
| $N_r$ | P/F | $N_s$ | 0 |
|---|---|---|---|
| 012 | 3 | 456 | 7 |

The I-field is transmitted as a series of 8-bit bytes in the following format:

*I-field sent from the controller to the 3270 CU*

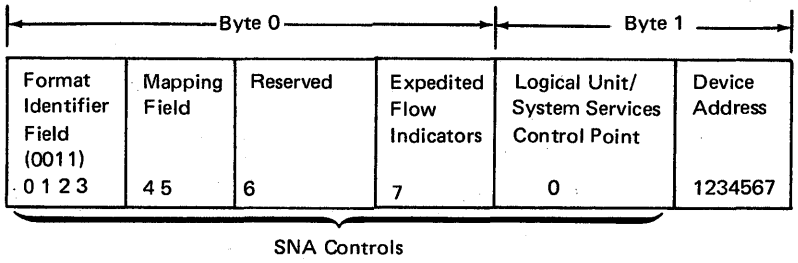
| | | |
|---|---|---|
| TH | Transmission Header - byte 0 | SNA controls |
| | Transmission Header - byte 1 | SNA controls and device address |
| RH | Request/Response Header - bytes 0, 1, and 2 | SNA controls |
| 256 bytes (max) | Command byte (0) | Command issued to the 3270 CU |
| | Data - byte 1 ... Data - byte 254 | Write-type operation message text |

*I-field sent from the 3270 CU to the controller*




**Transmission Header (TH)**

A transmission header is always included in an I-field. The 2 bytes of the transmission header contain four SNA fields in the following format:




The SNA controls are employed by higher-level network management. A description of these controls, as implemented by the 3270 system, follows.

*Bits 0, 1, 2, and 3 of byte 0* compose the format identifier (FID) field. The 3270 CU does not check these bits when they are received from the controller. When the TH is sent by the 3270 CU, FID 3 (0011) is used.

*Bits 4 and 5 of byte 0* are the mapping field, which records the text segment format that is used when read or write type operations are performed. Text segments contain a maximum of 256 bytes. Bit assignments for the mapping field are as follows:

11 – Indicates a complete basic information unit (BIU); that is, the segment associated with the TH is a complete unit.

10 – Indicates that the segment associated with the TH is the first segment in the BIU.

01 – Indicates that the segment associated with the TH is the last segment in the BIU.

00 – Indicates that the segment associated with the TH is an intermediate segment within the BIU.

*Bit 6* is reserved.

*Bit 7* is the expedited flow indicator (EFI) and is not checked by the 3270 CU when it is received from the controller. The 3270 CU sends the EFI as 0 in all cases except when sending a clear response, in which case the EFI is sent as 1. The Clear command and response are described under the heading "Control Functions."

Bit 0 of byte 1 is the Logical Unit/System Services Control Point (LU/SSCP) unit indicator. The 3270 CU stores this bit when it is received from the controller. When sending a response to the controller in reply to a request, the 3270 CU returns this bit as it was received. When an attention AID is generated, except when caused by a test request unit, the 3270 CU sets this bit to 1, indicating LU. A test request unit causes this bit to be set to 0, indicating SSCP.

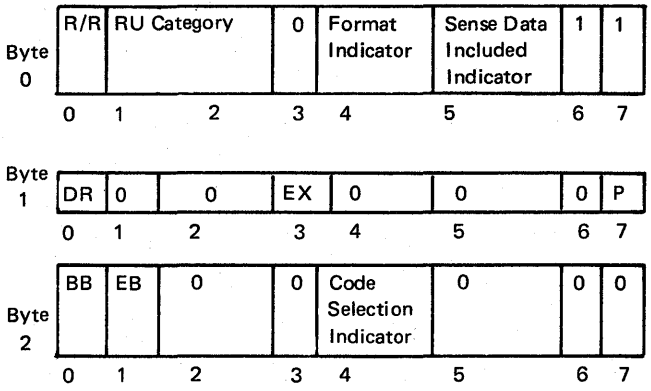
The device address is contained in bits 1 through 7 of byte 1. When received by the 3270 CU, the device address is decoded as the destination address for which the transmission is intended. When transmitted by the 3270 CU, the address indicates the device that initiated the transmission. Bit 1 is always set to 1, and bit 2 is always a 0. Up to 32 addresses, designated 0 through 31, are available for attachment of display station or printers to a 3271 control unit (Figure 6-1). Device address 0 is used when communicating with a 3275 display station.

### Request/Response Header (RH)

The request/response header contains 24 bits of SNA control information used by higher-level network controls to route and sequence transmissions and to indicate to the 3270 CU the form of response required. The RH and the message text contained in the text segment provide the basic exchange unit of control and data across the data link, called the *basic information unit* (BIU).

Each I-field may contain up to 256 bytes of message data. When the text of message exceeds 256 bytes, the message is segmented into a series of I-formatted frames. The first and all intermediate frames within the segmented group contain 256 bytes of message text. The last frame contains the remainder of the text being transmitted, up to 256 bytes. A request/response header is required when the message contains one I-frame (up to 256 bytes of message text) or within the initial I-frame of a segmented message.

The request/response header consists of 3 bytes with the following format:




R/R = Request/Response
RU = Request/Response Unit
DR = Definite Response 1
EX = Exception Response
P = Pacing
BB = Begin Brackets
EB = End Brackets

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

| CU or Device Number | EBCDIC I/O Char. | EBCDIC (Hex) (Note 1) | ASCII I/O Char. | ASCII (Hex) |
|---|---|---|---|---|
| 0 | SP (Note 2) | 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 | \| 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 1) | 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 3) | 7F | " | 22 |

Examples:

| 3271 Addressing | | | | 3275 Addressing | | | |
|---|---|---|---|---|---|---|---|
| | | EBCDIC | ASCII | | | EBCDIC | ASCII |
| General Poll CU5 | CU Address | C5 C5 | 45 45 | General Poll CU5 | CU Address | C5 C5 | 45 45 |
| | Device Address | 7F 7F | 22 22 | | Device Address | 7F 7F | 22 22 |
| Specific Poll Device 4 on CU5 | CU Address | C5 C5 | 45 45 | Specific Poll CU5 | CU Address | C5 C5 | 45 45 |
| | Device Address | C4 C4 | 44 44 | | Device Address | 40 40 | 20 20 |
| Select Device 4 on CU5 | CU Address | E5 E5 | 56 56 | Select CU5 | CU Address | E5 E5 | 56 56 |
| | Device Address | C4 C4 | 44 44 | | Device Address | 40 40 | 20 20 |

Notes:
1. *Graphic characters 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.*
2. *I/O character address (SP) is always used as the device address to select a 3275.*
3. *I/O character address (") is used as the device address to specify a General Poll operation.*

Figure 6-1. Remote Control Unit and Device Addressing – SDLC

Higher-level network controls determine the implementation and bit assignments within the RH. The response-indicating bits are definite response 1 (DR), exception response (EX), and pacing (P). Following are a detailed description of the response-indicating bits and a general description of the remaining RH bits.

The 3270 CU responds to combinations of response-indicating bits, specified in the RH received from the controller. The response generated by the 3270 CU consists of a frame containing the appropriate RH response bit(s) set to 1. (This is explained under the heading "3270 CU Responses.") Receipt of an RH by the 3270 CU with the DR bit set to 1 (byte 1, bit 0) indicates to the 3270 CU that a response must be sent when the specified command operation in the 3270 CU has been completed. An exception response (EX) is requested when RH bit 3 of byte 1 and the DR bit are set to 1. The 3270 CU generates an EX response if an error condition (other than an SDLC error) is detected during execution of a command. Error conditions are reported by the 3270 CU in the form of sense bytes contained within an I-field. If no error occurs, no response for the DR bit is sent. Pacing (P) is a response that allows the 3270 CU to indicate to the controller when message data can be sent for a device. The pacing response is returned when the requested operation is completed at the device.

The functions of the remaining 21 bits contained in the RH are summarized as follows:

*Bit 0 of byte 0* is the request/response (RR) bit. The 3270 CU does not check the RR bit when it is received from the controller. It sends the RR bit to the controller as a 0 to indicate a request. This occurs when message text is sent as part of a read-type command or read-by-poll operation, or when asynchronous status or sense information is transmitted. The 3270 CU sends the RR bit as a 1 to indicate a response in reply to a definite or exception response (with or without pacing) requested by the controller.

*Bits 1 and 2 of byte 0* are the request/response unit (RU Category) bits. They are stored, but not checked, by the 3270 CU when they are received from the controller. These bits are set, depending upon the contents of the RU, as follows:

| RU | RU Category |
|---|---|
| Function Management (FM) Data | 00 |
| Network Control | 01 |
| Data Flow Control | 10 |
| Session Control | 11 |

The 3270 CU sends FM data, except when sending a Clear or Pseudo Bid response, in which case bits 1 and 2 are sent in the same form in which they are received from the controller.

*Bit 3 of byte 0* (always 0) is not used by the 3270 CU.

*Bit 4 of byte 0* is the format indicator. The 3270 CU stores, but does not check, this bit when it is received from the controller. When the 3270 CU generates a request, the format indicator bit is sent as a 0; when sending a response, the 3270 CU sends this bit as it was received from the controller.

*Bit 5 of byte 0* is the sense-data-included indicator. The 3270 CU does not check this bit when it is received from the controller. The 3270 CU sends this bit as a 1 when sense data is transmitted and as a 0 when sense information is not sent to the controller.

*Bits 6 and 7 of byte 0* (always 1) are not used by the 3270 CU.

*Bits 1, 2, 4, 5, and 6 of byte 1* (always 0) are not used by the 3270 CU.

*Bit 0 of byte 2* is the begin bracket (BB) bit and is used by the 3270 CU in conjunction with the Pseudo Bid command (described under "Control Functions"). Receipt of the BB bit set decrements the poll counter in the 3270 CU.

*Bit 1 (end brackets), and bits 2, 3, 5, 6, and 7 of byte 2* are always 0 and not used by the 3270 CU.

*Bit 4 of byte 2* is the code selection indicator. This bit identifies the transmission code as EBCDIC (0), or ASCII (1).

### Command Byte

The command contained in the command byte is sent after the RH by the controller for execution by the 3275 or by a device attached to the 3271. A list of command codes and a description of 3270 command operations appear in Chapter 3. Order codes, when employed, are transmitted within the message text following a Write or Erase/Write command.

The following conditions must be met to allow command execution:

- The frame must have a valid FCS character.
- The I-field must be the initial I-field of a segmented message or must contain the entire text of the message.
- The addressed device must be in a ready state (not busy).

### 3270 CU Responses

The 3270 CU responds to combinations of DR, EX, and P bit settings received from the controller in byte 1 of the RH. Valid request and response formats are listed in Figure 6-2.

#### Definite Response with Pacing

1. *Write and Erase/Write Commands for Display Stations and Printer.* When a write-type operation is successfully completed, the 3270 CU responds with a frame containing byte sequence F, A, C, TH, RH, FCS, F, with DR=1, EX=0, and P=1. Successful completion of a write operation to a printer occurs when the printout is completed. When the Write command, Start Print, and buffer data is successfully transferred from the 3271 CU to the printer, the RR response is sent to the controller. DR1 with the pacing response bit set on is sent only when printing has been completed. In the interim, the 3271 CU can process other messages to other devices. If an error is detected (other than an SDLC error) during command execution, the 3270 CU sends a response frame with DR=1, EX=1, and P=1 within the RH, and inserts a text segment containing a 4-byte sense RU to report the error condition. Sense RU format is defined under the heading "Error Responses and Error Recovery."
2. *Read Modified and Read Buffer Commands for Display Stations.* Successful completion of a read-type command occurs when the data has been sent and acknowledged at link level by the controller. The 3270 CU then replies with a frame containing DR=1, EX=0, and P=1 within the RH. If an error is detected (other than an SDLC error) while the device buffer is being obtained, the 3270 CU sends a response frame containing DR=1, EX=1, and P=1, and includes a sense RU text segment. If an error is detected during transmission of the message data to the

| Response | Request format— sent by the controller: DR | EX | P | Response format— sent in reply by the 3270 CU: DR | EX | P | Explanation |
|---|---|---|---|---|---|---|---|
| Definite response with pacing | 1 | 0 | 1 | 1 | 0 | 1 | Indicates successful completion of a read or write type or Copy command by a display station; or a write type or Copy command by a printer.<br>1. Indicates that an error occurred during transmission of read data. In this case, the response may be preceded by a sense RU request containing an abort indication.<br>2. Indicates that an error was detected while a device buffer was being obtained.<br>**Note:** *The printer operates in definite response with pacing mode only. Therefore, when a command has been executed by a printer, the 3270 CU always responds with positive response with pacing (101 or 111), regardless of the request received.* |
| | | | | 0 | 0 | 0 | |
| | | | | followed by: | | | |
| | | | | 1 | 0 | 1 | |
| | | | | 1 | 1 | 1 | |
| Exception response with pacing | 1 | 1 | 1 | 0 | 0 | 1 | Indicates successful completion of a read or write type or Copy command by a display station.<br>1. Indicates that an error was detected while a device buffer was being obtained.<br>2. Indicates that an error occurred during transmission of read data. In this case, an exception request with an abort segment indication is transmitted before the response. |
| | | | | 1 | 1 | 1 | |
| | | | | 0 | 0 | 0 | |
| | | | | followed by: | | | |
| | | | | 0 | 0 | 1 | |
| No response with pacing | 0 | 0 | 1 | 0 | 0 | 1 | Applicable to commands executed by display stations only. An error response (EX = 1) is not sent, regardless of how the operation ends. The 3270 CU transmits only an isolated pacing response. |
| Definite response, no pacing | 1 | 0 | 0 | 1 | 0 | 0 | Applicable to display station command operations only. The response description is the same as described above for positive response with pacing, except that the pacing bit is always set to 0. |
| Exception response, no pacing | 1 | 1 | 0 | 0 | 0 | 0[1] | Applicable to display station command operations only. The response format is the same as explained above for exception response with pacing, except that the pacing bit is always set to 0. |
| | | | | 1 | 1 | 0 | |
| No response, no pacing | 0 | 0 | 0 | 0 | 0 | 0[1] | Applicable to display station command operations only. The 3270 CU does not send a response. |

[1] A response format 000 indicates that no response is sent.

**Figure 6-2. Request and Response Format**

controller, the 3270 sends an exception request to the controller with an abort segment structure (DR1=0, EXC=0, and P=0) as described under the heading "Error Responses and Error Recovery." Following the exception request, a frame is sent containing DR=1, EX=0, and P=1. In this case, the read operation is considered completed, but unsuccessful.

3. *Copy Command for Display Stations and Printers.* When buffer data has been transferred from the *from* device to the *to* device without detection of an error, the operation is considered completed. The 3270 CU then sends a response frame with DR=1, EX=0, and P=1. If the *to* device is a printer, the response is delayed until the printout is completed. When the Copy command, Start Print, and buffer data is successfully transferred from the 3271 CU to the printer, the RR response is sent to the controller. DR1 with the pacing response bit set on is sent only when printing has been completed. In the interim, the 3271 CU can

process other messages to other devices. If an error is detected while the *from* device buffer is being obtained, the 3270 CU sends a sense RU response with DR=1, EX=1, and P=1. The address in the TH is the address of the *to* device, but the sense RU indicates that the error is in the *from* device or in the 3270 CU. If an error is detected during the transfer of data to the *to* device, the 3270 CU responds with a sense RU response with DR=1, EX=1, and P=1, with sense indicating an error condition (see Copy command).

**Exception Responses with Pacing**

1. *Write and Erase/Write Commands for Display Stations and Printers.* During execution of a write-type command to a printer, when the transfer of message text from the 3270 CU to the printer has been completed, the CU may begin servicing other attached devices.

   When the printer operation has been successfully completed, the form of the response requested within the RH for the printer message is no longer present in the 3270 CU. In this situation, the 3270 CU replies by sending a definite response with pacing (DR=1, EX=0, P=1).

   When the addressed device is a display station, the operation is the same as that described for Write and Erase/Write commands under "Definite Response with Pacing," except that successful command execution results in the 3270 CU's sending an isolated pacing response that is DR=0, EX=0, P=1.

2. *Read Modified and Read Buffer Commands for Display Stations.* The operation is the same as that described for Read Modified and Read Buffer commands under "Positive Response with Pacing," except that the 3270 CU sends DR=0, EX=0, P=1 when the command has been successfully completed.

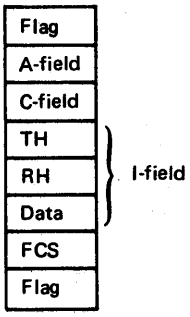
3. *Copy Command for Display Stations and Printers.* Since the printer must operate in positive response with pacing mode, the 3270 CU treats a request on Copy command operations to a printer as though positive response with pacing has been specified, regardless of the actual setting of the DR, EX, and P bits. The operation is the same as that described for the Copy command under "Definite Response with Pacing."

   When the *to* device is a display station, the operation is the same as that described under "Definite Response with Pacing," except that successful completion causes the 3270 CU to reply with an isolated pacing response, DR=0, EX=0, P=1.

**Definite or Exception Responses without Pacing; No Response with or without Pacing.** Definite or exception response without pacing, and no response with or without pacing, are four variations of responses based on the definite or exception response with pacing. The response formats and a description of the response operations are given in Figure 6-2.

**Data Transmissions by the 3270 CU**

Data transmitted by the 3270 CU can be message text, test request data, or status and sense information. Data is transmitted to the controller in the same SDLC frame format used by the controller except that a command code is not present within the text segment. The frame format is as follows:



**Message Text.** Message text can be transmitted following:

1. Receipt of a Read Buffer or Read Modified command with a poll bit set to 1 in the C-field, or when an RR command with the poll bit set to 1 is received after the frame containing the read-type command.
2. Receipt of an RR command with the poll bit set to 1 when an attention key is pressed (except the TEST REQ key).

The address contained in the TH is the address of the device that received the read-type command or the address of the device that had an attention key pressed when an RR command was received.

When more than 256 bytes of message text are transmitted, the data stream is segmented into 256-byte segments [as described previously for data transmissions to the 3270 CU under the heading "Request/Response Header (RH)"].

*Test Request Messages.* Test request messages can be entered from a display station keyboard when the operator has pressed the TEST REQ key and a Read Modified command is issued to the device. For a description of the test request operation, refer to the heading "Test Request Read" in Chapter 3.

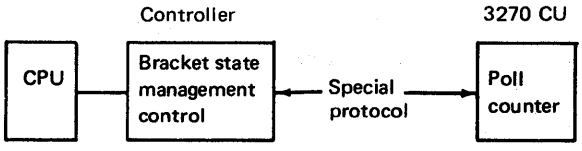
## Control Functions

The Clear and Pseudo Bid control functions take the form of control commands issued by the controller in information format. The 3270 CU sends a Clear response when replying to the Clear command.

*Clear Command/Response.* The controller sends the Clear command to cancel pending DR and (or) pacing responses from the 3270 CU. The Clear command format consists of a 1-byte RU (hex A1), with the DR request bit set in the RH. The 3270 CU replies with a response frame in the same format received from the controller.

*Pseudo Bid Command.* The controller sends the Pseudo Bid command to cause the 3270 CU to do a Specific Poll to a specific device and, if no attention ID is pending, to execute a write-type command at the selected device. The Pseudo Bid command is sent in information format as 1 byte of data (hex F8). If an attention ID is not present at the addressed display station or printer and a request was sent by the controller, the 3270 CU replies with a response frame. The controller then sends an information frame containing a write-type command and the BB bit set in the RH. The operation then proceeds as a write-type command. Bracket protocol for the application program and the terminal operator is given in *Introduction to Programming the IBM 3270 Information Display System,* GC27-6999.

Generally, the bracket state management function resides within the SNA CU. In the 3270 system employing SNA, however, the controller provides bracket state management control for the 3270 CU. As a result, the controller controls the operation of the poll counter in the 3270 CU and generates the Pseudo Bid command.




The poll counter is used for bracket state management. A successful Pseudo Bid command operation at the 3270 CU (that is, no attention ID pending) increments the poll counter. Receipt of the BB bit set, with no attention ID pending, decrements the poll counter. The poll counter must be equal to zero to allow the 3270 CU to present attention-generated information to the controller. When the poll counter is not equal to zero, communication with the devices is inhibited.

### *SDLC Sequence/Response Diagrams*

Figures 6-3, 6-4, and 6-5 are sequence/response diagrams showing, respectively, online/offline procedures, read-type commands, and write-type command operations. Only the portions of the SDLC frames essential to the operation are shown. The descriptive text in each diagram summarizes the flow of information between the controller and the 3270 CU.

### *Status and Sense (S/S) Bytes*

The 3270 CU records SNA and device status and sense (S/S) conditions for each attached device. All remote status and sense conditons are contained in 4 bytes, which are sent to the controller as an exception request or an exception response. Bytes 0 and 1 contain SNA S/S information, and bytes 2 and 3 contain device S/S information in the same format used for remotely attached BSC devices. An exception request or an exception response is returned to the controller when the 3270 CU has status pending and receives an RR command or I-formatted frame with the poll bit set to 1.

The status and sense message contains 9 bytes in the I-field in the following sequence:

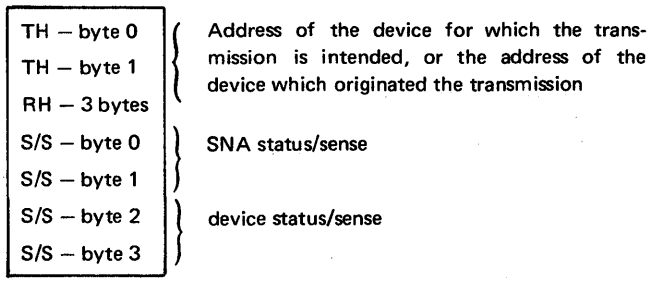
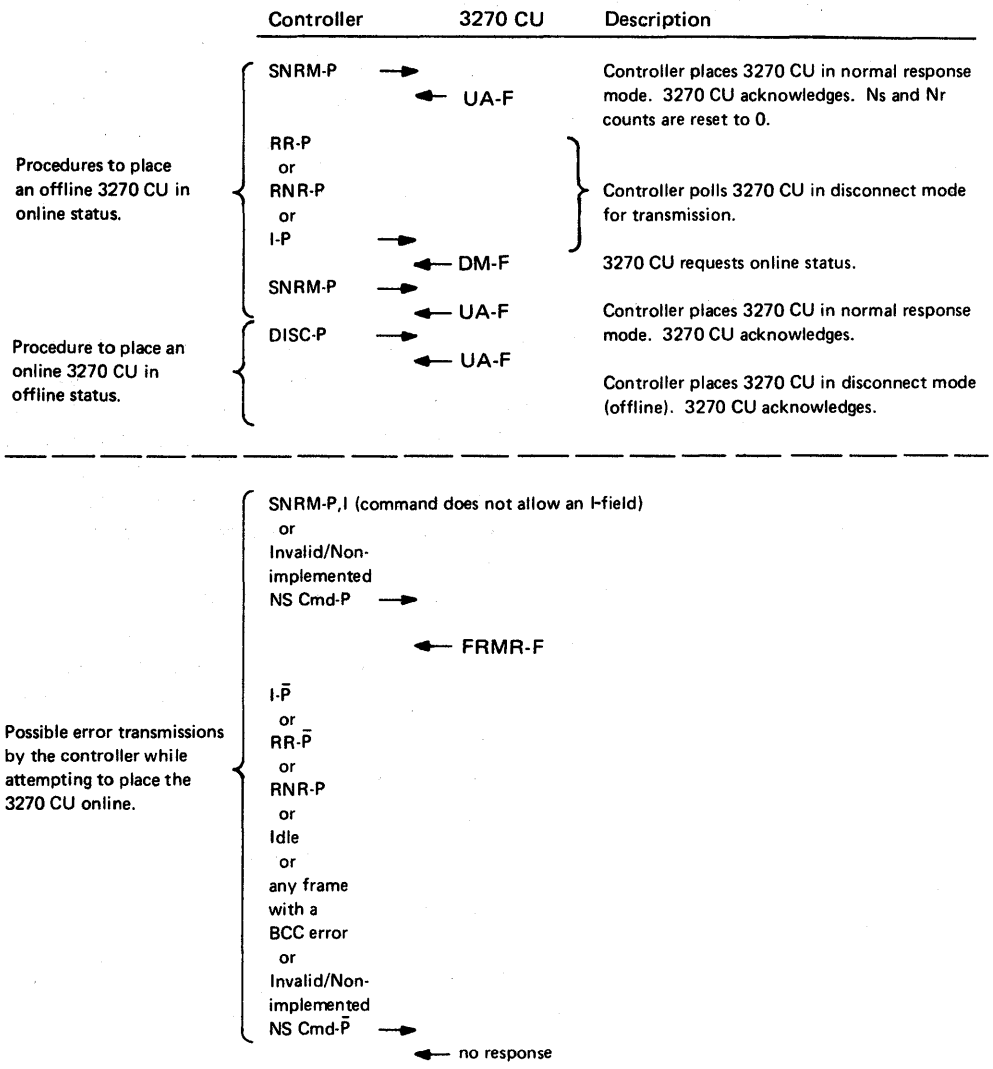



Figure 6-6 defines each S/S bit. Figure 6-7 shows how these status and sense conditions are interpreted for each error response or request transmitted by the 3270 CU.




**Note:** *Only SDLC bytes that are significant for the sequence being illustrated are shown in this figure.*

P/F=poll/final bit
I = information frame
dash (—) above a letter = not set to 1.

**Figure 6-3. Online and Offline Procedures, Sequence/Response Diagram**

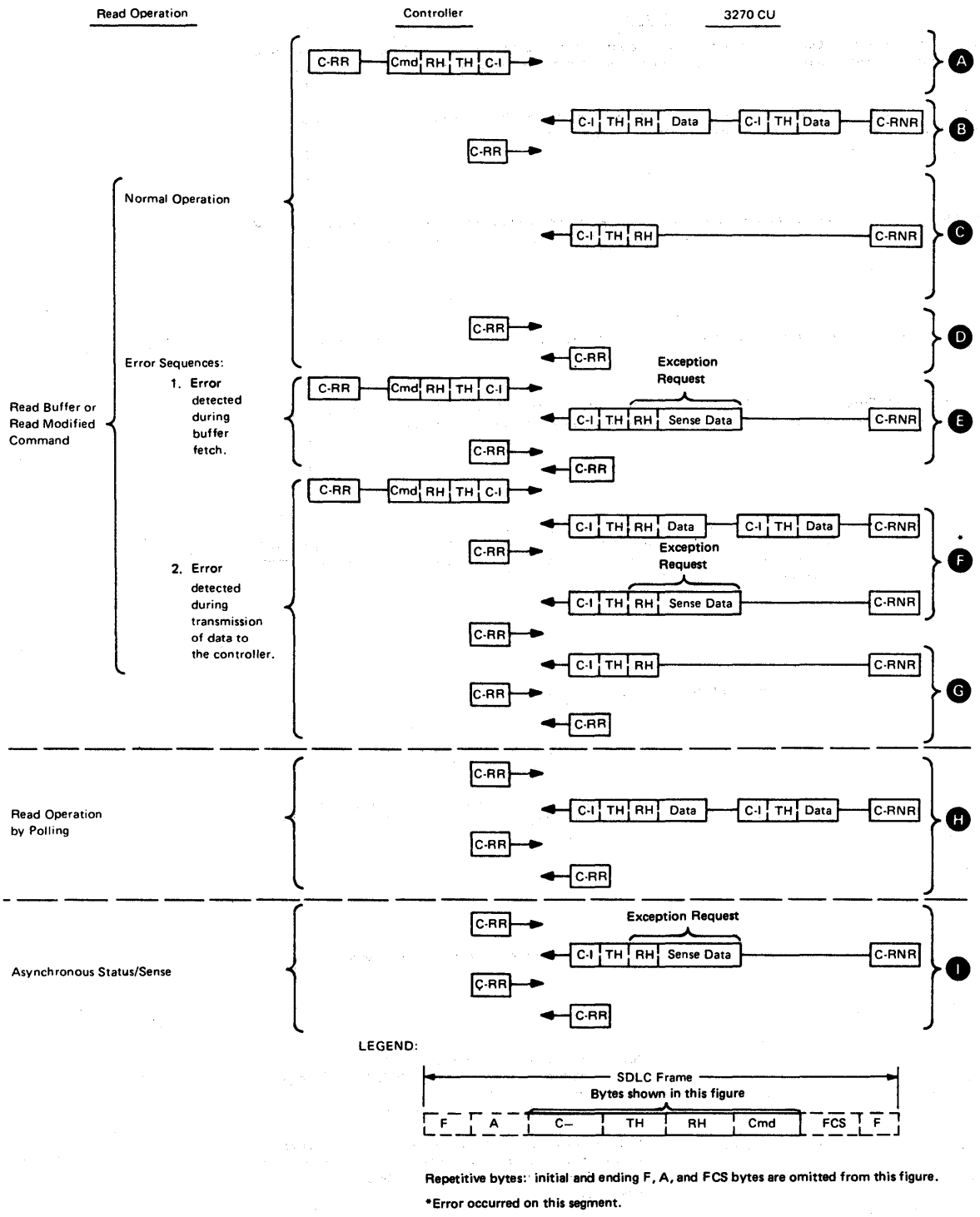



**Figure 6-4 (Part 1 of 2). Read-Type Command, Sequence/Response Diagram – SDLC**

Description

**A** Controller sends Read type command. Controller may send one I-frame with p=1, or send I-frame with p=0, followed by RR command frame with p=1. TH contains device address. If a response is required by the 3270 CU, the controller requests a response format in the RH.

**B** 3270 CU sends two I-frames of data followed by RNR with f=1. Controller acknowledges with RR command. Cycle is repeated until all I-frames have been sent. RH bytes are contained only in the first I-frame sent.

**C** If a 3270 CU RH response had been specified by the controller, an I-frame containing the response format is sent, unless an Exception response has been requested and no errors have occurred. In this case, the I-frame C-I TH RH is not needed and the 3270 CU sends RR only.

**D** Controller and CU RR exchange to complete the read operation. RR is sent by the 3270 CU only at the completion of a command.

**E** Read type command initiated. Error detected while fetching device buffer. If a response was requested by the controller in the RH of the I-frame containing the command, the 3270 CU responds with error response in the RH and 4 bytes of sense data.

**F** Data check sensed by CU for second I-frame. If a response was requested by the controller in the RH of the I-frame containing the read command, the 3270 CU transmits a frame containing the error, followed by an exception request frame; if no error occurs, the exception request is not sent, and read command is completed as in normal operation.

**G** If a 3270 CU response had been requested, an I-frame is sent containing an error response.

**H** This condition is caused by an Attention ID. When the controller polls, I-frames are transmitted and acknowledged in groups of two frames, as in the read-by-command operation. Error sequences and termination of read are the same as shown for read by command. Read by poll does not require 3270 CU responses.

**I** An exception request is returned to the controller when the 3270 CU is polled and has status pending, and a read, write, or print operation is not in process.

LEGEND: (continued)

**F**, 8-bit flag (01111110) sequence
**A**, 8-bit CU address field
**C**, 8-bit control field; specifies transmission format:
**C-I** (Information field)
**C-RR/RNR** (Supervisory Commands and Responses)
**C-SNRM/DISC/UA/DM/FRMR** (nonsequenced format)
**FCS**, 16-bit frame check sequence
**TH**, 16-bit transmission header; includes device address
**RH**, 24-bit request/response header; specifies definite response (DR), exception (EX), and pacing (P) request and response formats.
**P/F** bit, poll/final bit contained in the C-field

**Note:** *A number of DR, EX, and P request/response formats are available. For a description of the request/response formats applicable to each command, refer to the heading* "3270 CU Responses."

**Figure 6-4 (Part 2 of 2). Read-Type Command, Sequence/Response Diagram – SDLC**

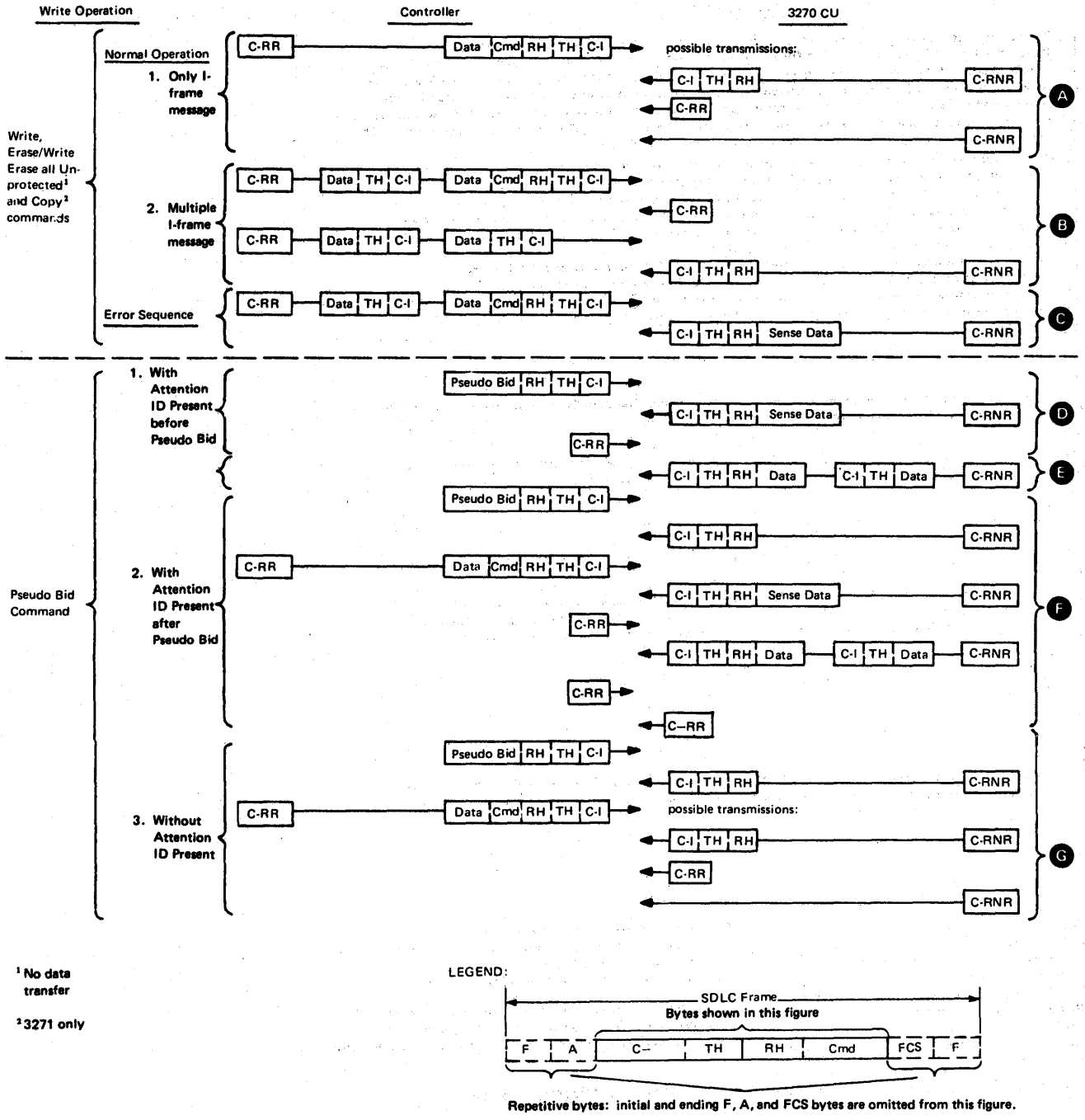



**Figure 6-5 (Part 1 of 2). Write-Type Command, Sequence/Response Diagram – SDLC**

Description

(A) The controller sends a request consisting of a write-type command containing one I-frame of message data. If the poll bit is not = 1 in the I-frame, a frame with RR and the poll bit set to 1 is sent. The 3270 CU can send one of three possible replies: If the controller requests a response in the RH, the 3270 CU sends the response frame followed by RNR, unless an exception response had been requested and no errors had occurred. In this case, the frame is not required and RR is sent. If the 3270 CU cannot perform the operation indicated within 75 ms, RNR is sent to the controller. When the controller does not request a response from the 3270 CU, RR is returned to the controller. The final bit is set to 1 in all 3270 CU transmissions, except I-frame transmissions.

(B) The controller sends a write-type command containing a series of I-frames. If the poll bit is not = 1 in the last data frame sent, a frame with RR and the poll bit = 1 is sent. The controller may send a partial transmission of data or the entire message. If a partial transmission occurs, the 3270 CU acknowledges with RR, and the controller then sends additional I-frames. The cycle is repeated until the controller has sent the last I-frame. If the controller had requested a response, the 3270 CU sends the response frame followed by RNR, unless an exception response was requested and no errors occurred. In this case, only RR is sent.
Note: *As in the example above, of a one I-frame message, the 3270 CU can reply with RNR if it cannot respond within 75 ms., or RR as the last transmission if a request was not sent by the controller.*

(C) A write type command is initiated. If a response was requested by the controller, and an error was detected during command execution, an exception response is sent by the 3270 CU after transmission of data from the controller is completed.

(D) The controller sends an I-frame containing the Pseudo Bid command. The 3270 CU, with an attention ID pending, replies with an exception response (Bracket Bid Reject).

(E) The 3270 CU sends read data in groups of two frames. If an error is detected, an exception response is sent to the controller, as shown in Figure 6-4, Read-Type Command, Sequence Response Diagram.

(F) The controller sends an I-frame containing the Pseudo Bid command. Because an attention ID is pending (Bracket Bid Reject) if a response had been requested, other than an exception response, a response frame is sent to the controller followed by RNR. Subsequently, the data that caused the attention is sent to the controller.

(G) The controller sends an I-frame containing a write type command with the BB bit set in the RH. The normal write type command operation takes place, as shown above in this diagram.

LEGEND: (continued)

F, 8-bit flag (01111110) sequence.
A, 8-bit CU address field.
C, 8-bit control field; specifies transmission format:
C-I (Information field)
C-RR/RNR (Supervisory Commands and Responses)
C-SNRM/DISC/UA/DM/FRMR (nonsequenced format)
FCS, 16-bit frame check sequence
TH, 16-bit transmission header; includes device address
RH, 24-bit request/response header; specifies definite response (DR), exception (EX), and pacing (P) request and response formats.
P/F bit, poll/final bit contained in the C field

Note: *A number of DR, EX, and P request/response formats are available. For a description of the request/response formats applicable to each command, refer to the heading* "3270 CU Responses."

**Figure 6-5 (Part 2 of 2). Write-Type Command, Sequence/Response Diagram – SDLC**

| Bit No. | Bit Definition |
|---|---|
| **S/S Byte 0:** | |
| 0 | *Path Error* —For the 3271, this bit is set if the device address received (bits 1 through 7 of TH byte 1) is invalid, or if the device adapter card for the indicated address is not installed.<br>For the 3275, this bit is set if the device address is not 1000000 (bits 1 through 7 of TH byte 1). |
| 1,2 | Reserved. |
| 3 | *Request Error* — This bit is set if the first byte of the RU is not recognized as a valid command or command function. Command Reject (CR), S/S byte 3, bit 2, is set when Request Error is set. |
| 4 | *Request Reject* — The bit is set if a Pseudo Bid command or begin bracket bit (set in the RH) is sent to a device that has an attention pending. |
| 5,6,7 | Reserved. |
| **S/S Byte 1:** | |
| 0,1,2,4,5 | Reserved. |
| 3,6,7 | These bits are set with *request reject* (bit 4, byte 0) |
| **S/S Byte 2:** | |
| 0,1,2,3 | Reserved. |
| 4 | *Device Busy (DB)* — This bit indicates that the addressed device is busy executing an operation. The device is busy when 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). |
| 5 | *Unit Specify (US)* — This bit is set if any S/S bit is set as a result of a device-detected error. |
| 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. When a printer goes from busy to not busy, a positive response with pacing is generated instead of DE. |
| 7 | Reserved. |
| **S/S Byte 3:** | |
| 0,1 | Reserved. |
| 2 | *Command Reject (CR)* — This bit is set upon receipt of an invalid 3270 command. |
| 3 | *Intervention Required (IR)* — This bit is set if:<br>• A Copy command contains a *from* device address in its data stream which specifies an unavailable device. |

| Bit No. | Bit Definition |
|---|---|
| **S/S Byte 3 (cont)** | |
| 3 | • A command attempted to start a printer but found it not ready. The printout is suppressed.<br>• The 3271 receives a Pseudo Bid sequence for a device which is unavailable or which became not ready during a printout.<br>• The 3270 CU receives a command for a device which the 3271 has logged as unavailable and not ready. |
| 4 | *Equipment Check (EC)* — This bit is set if:<br>• A printer character generator error occurred, or the printer became mechanically disabled.<br>• The 3270 CU detected bad parity from the device, or data transmitted in a device reply.<br>**Note:** *The data check (DC) bit may also be set.* |
| 5 | *Data Check (DC)* — This bit indicates detection of a parity or cursor check in either the 3271 or a device buffer, or in the 3275 buffer, or that the 3271 detected bad parity from the device. |
| 6 | *Control Check (CC)* — This bit is not used for the 3275. For the 3271, this bit indicates a timeout check. A timeout check occurs when a device fails to respond to 3271 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:<br>• An invalid buffer address or an incomplete order sequence was received on a Write or Erase/Write command.<br>• The device did not receive a CCC or a *from* address on a Copy command.<br>• A Read, Read Modified, Copy, or Erase All Unprotected command was received with TH mapping field bits not equal to 11 (i.e., a complete BIU).<br>• An I/O interface *overrun* is detected. This occurs if a data byte follows a Read Buffer, Read Modified, or Erase All Unprotected command, or if more than 2 data bytes follow a Copy command.<br>This bit is set with Control Check, Intervention Required, Data Check, or Data Check with Unit Specify, to indicate that the errors that set these sense bits were detected while the 3270 CU was executing an operation with the *from* device during a Copy command. |

Figure 6-6. Remote Status and Sense Byte Definitions–SDLC

| Status/Sense Bits | Explanation | |
|---|---|---|
| | Response | Request |
| PE (Address not available) | Bits 1 through 7 of TH byte 2 are not a valid device address or the device adapter card is not installed in the 3271. | NA |
| CC | A timeout check is caused by the addressed device. The operation is tried twice before the CC bit is set. | NA |
| CC, OC | The *from* device fails to complete an operation or to respond to the 3271 within a specified time period (timeout check) during a Copy command operation. | NA |
| DC | 1. The 3271 or 3275 detects a parity or cursor check in its buffer during a command operation.<br>2. The 3271 detects bad parity on data received from the addressed device. The operation is attempted twice before the DC bit is set. | A parity error is detected by the 3271 on a data transfer to the controller as a result of a poll or a parity error detected in the 3275, |
| DC, US (3271 only) | 1. A parity check or cursor check is detected by the addressed device on the data it is sending to the 3270 CU.<br>2. The device detects a parity or cursor check in its buffer during a command operation. | A parity check or cursor check is detected by the polled device on the data it is sending to the 3271 CU. |
| DC, OC (3271 only) | The 3271 detects a parity check on the data transferred from the *from* device during a Copy command operation. | NA |
| DC, OC, US | Sent when the *from* device detects an internal parity or cursor check while performing the Copy command. | NA |
| IR | The addressed device is not available or the addressed printer is not ready. | NA |
| IR, OC (3271 only) | The *from* device is not available on a Copy command. | NA |
| IR, EC, US (3271 only) | The addressed printer is mechanically disabled and cannot recover. | NA |
| OC | 1. The Copy command data stream contains more or less than 2 bytes (the CCC and the *from* device address). The Copy command is aborted. | NA |
| | 2. One or more data bytes followed an Erase All Unprotected command (buffer overrun). | NA |
| | 3. A data byte followed a read type command in the data stream received at the device. | NA |
| OC, US (3271 only) | The device has a locked buffer during a Copy command operation. (Refer to the heading "Copy Command" in Chapter 3, "Commands and Orders.") | NA |
| EC, US (3271 only) | A character generator error or a mechanical failure is detected at the printer, but recovery occurs. | NA |
| RE, CR | An invalid command is detected (first byte of data). For example, a Copy command is sent to the 3275. | NA |
| EC | Character generator error (3275 only) in printer. | Bad parity from a device (3271 only). |

**Figure 6-7 (Part 1 of 2). Remote Error Status and Sense Responses and Requests – SDLC**

| Status/Sense Bits | Explanation | |
|---|---|---|
| | Response | Request |
| EC, DC | Transmit parity error has occurred. If a buffer was obtained during the operation, the Data Check bit is also set. | NA |
| DE | The poll bit finds a device which was previously recorded as busy, as not busy. Transmission of an I-frame with read or write type data resets this bit. | The poll bit finds a device which was previously recorded as unavailable or not ready, as available and ready. |
| IR | The addressed printer is out of paper, power has been turned off, or the printer cover is open. | NA |
| IR, EC (3275 only) | Power is off at the 3284 Model 3 printer, or a malfunction is detected. | NA |
| OC, DB | The *from* device receiving a Copy command is busy. The device is busy performing an operation or a printout, reading data from the operator identification card reader, or performing a keyboard operation. | NA |
| DB | The addressed device is busy. | NA |

**Notes:**

1. *There are other conditions of multiple status that can occur which are not included here; for example, an unpredictable catastrophic card failure or multiple error conditions occurring simultaneously could cause an undefined combination of status and sense bits. If a multiple-status condition occurs, each bit must be checked separately to determine the cause(s) of the failure.*
2. *See Figure 6-8 for error-recovery procedures that are applicable for certain combinations of status/sense bits.*

Figure 6-7 (Part 2 of 2). Remote Error Status and Sense Responses and Requests – SDLC

## Error-Recovery Procedures

Errors detected by the 3270 CU are indicated to the system by a timeout, an FRMR response, or an exception request or response.

Figure 6-8 lists the various error combinations of sense/status bits (described in Figure 6-7) and refers to error-recovery procedures. The error-recovery procedures recommended in Figure 6-8 are as follows:

1. a. Any response other than NSA to a Set Mode command is discarded and results in *n* retries of the particular Set Mode command being attempted. If the time-out response persists, the system operator should take action to verify the link.
   b. Execute a new command sequence, starting with the command that was being executed when the error occurred. Executing a new command is a function of the access method or the application program and is the responsibility of the customer-written application program. If, after two retries, the operation is not successful, inform the system operator of the problem and follow procedure 4a.
   c. Perform procedure 1b, except, if operation is not successful, follow procedure 4b instead of 4a.
2. Notify the responsible application programmer that a nonrecoverable program error was detected.

| Sense/ Status Bit | Detected during 3270 Operation Transmitted as: Response | Request | Error Recovery Procedure: 3271 | 3275 |
|---|---|---|---|---|
| PE (Address not available) | D, P | | 2 | 2 |
| CC | D, P | | 1b | NA |
| CC, OC | D, P | | 1c | NA |
| DC | D, P | D, P | 1b, 3a[1] | 3a |
| DC, US | D, P | D, P | 3a | NA |
| DC, OC | D, P | | 1c | NA |
| DC, OC, US | D, P | | 3b | NA |
| IR | D, P | | 6a | 6a |
| IR, OC | D, P | | | NA |
| IR, EC, US | P | | 5 | NA |
| OC | D, P | | 2 | 2 |
| OC, US | D, P | | 7 | NA |
| EC, US | P | | 5 | NA |
| FIE, CR | D, P | | 2 | 2 |
| EC | D, P | D, P | 1b | 5 |
| DC, US, DE | D, P | | 3a | NA |
| IR, EC | D | | NA | 5 |
| DE | D, P | D, P | None | None |
| OC, DB | D, P | | 8a | NA |
| RR | D, P | | None | None |
| DB | D, P | | 8b | 8b |

Legend:
- NA – Not applicable
- D – Display (3277 or 3275)
- P – Printer

[1] Perform error-recovery procedure 1b if error occurred during a read operation. Perform error-recovery procedure 3a if error occurred during a write operation.

**Figure 6-8. Remote Status and Sense Conditions – SDLC**

3. a. Reconstruct the entire device buffer image, starting with the first segment if a multisegment transmission occurred, and retry the failing sequence of commands. The sequence of commands used to reconstruct the image should start with an Erase/Write command to correct a possible missing- or multiple-cursor condition in the device buffer. This procedure is the responsibility of the customer-written application program. If, after a series of retries, the problem is not corrected, inform the system operator of the problem and follow procedure 4a.

   b. The error occurred during execution of a Copy command. Follow procedure 3a, and reconstruct the entire image of the device buffer of the *from* device specified by the Copy command. If, after a series of retries, the operation is not successful, follow procedure 4b.

4. a. Request the system operator to request maintenance support. Following repair, reconstruct the buffer image. The sequence of commands used to reconstruct the image should start with an Erase/Write command to correct a possible missing- or multiple-cursor condition in the device buffer.

   b. The *from* device specified by the Copy command in the failing chain of commands is malfunctioning. The device should be identified from the customer- written application program, and the system operator should be requested to have this device repaired. After repair, reconstruct the device buffer image. The sequence of commands used to reconstruct the image should start with an Erase/Write command to correct a possible missing- or multiple-cursor condition in the device buffer.

5. 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 not want a new printout. In this case, the appropriate recovery procedure is the responsibility of the customer-written application program. If a new printout is required, follow procedure 6a.

6. a. 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) the terminal operator to ready the device. A Device End can be expected. 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 3a.

   b. The error indicates that the *from* device specified by a Copy command is unavailable. The device address associated with the error status and sense information is not the device that requires reading. The device that requires reading is the *from* address specified in a Copy command. The responsible customer application programmer should determine the *from* device address and inform the system operator.

7. An attempt was made to execute a Copy command in which access to the *from* device data was not authorized. Determine the appropriate customer-written application program, and notify the customer. The device address associated with the error status/sense bits in the sense RU is that of the Copy command *to* device.

8. a. This indicates that, in attempting to execute a Copy command, the *from* device was found to be busy. Follow procedure 1b when the *from* device becomes not busy. Note that the device address associated with the S/S bits in the sense RU is the address of the *to* device and not that of the busy *from* device. The *from* device will transmit Device End when it becomes not busy.

   b. This indicates that the addressed device is busy. If the device is a display station, it will transmit Device End when it becomes not busy. If the device is a printer, a positive response with pacing is sent.

### *Timeout to a Poll*

When the 3270 CU detects an FCS check, it initiates a timeout and does not respond to the controller. The controller retransmits the message several times, if necessary, in an attempt to correct the error.

### *CMDR Response to Invalid Nonsequenced Commands and I-Field Formats*

The 3270 sends the FRMR response for invalid nonsequenced command formats and I-formats. The recovery action for FRMR response is the responsibility of the controller's.

### *ROL Response to a Poll*

The 3270 CU sends a DM response upon receipt of an RR or RNR command with the poll bit set to 1, when it is in disconnect response mode. Disconnect response mode is a result of a DISC command's having been issued previously by the controller or of power's having been removed from the 3270 CU and then applied. The controller must issue an SNRM command to return the 3270 CU to online status.

### *Aborting an Inbound I-Frame*

Data checks are sensed by the 3270 CU before a segment of message data is transmitted to the controller. If the segment assembled for transmission was the first or the only segment, an exception request is transmitted with the mapping bits set to indicate one segment (whole) in place of the segment of message data that contained the error. If the segment assembled for transmission was an intermediate, or the last, segment and contained an error, an exception request is transmitted with mapping bits indicating one segment. In either case of this abort-segment structure, RH byte 1 is DR1=0, EXC=0, PAC=0, and the sense and status data check indicator is on. As a result, the host discards all segments received up to and including the segment containing the exception request.

# Chapter 7. Screen Design

## 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 think of *J* and *O* and *H* and *N* as useful, not 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: 936981, location: BOSTON, date of hire: 07/02/62.

Each data element has its own characteristics. In this example, the serial number is 6 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 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 it 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 basis and specify the characteristics, or *attributes,* of each individual field. The 3270 then provides program and data control on the basis of your individual field definitions.

### *How Fields Are Defined*

Each data field is established by writing a field attribute control code, or attribute character, as the first position of the field. A field is defined as the attribute character, plus all the data following it up to the next attribute character. The placement of attribute characters defines the field lengths, and the content of the attribute characters defines the other field characteristics. In the following examples, the symbol [ ] designates an attribute character.

All the characters in a field, except the attribute character itself, assume identical characteristics based on the specifications within the attribute character. In Figure 7-1 the characteristics of the field NAME: are controlled by the attribute [1], and terminated by the attribute [2]. The placement of attributes controls the length of the fields.

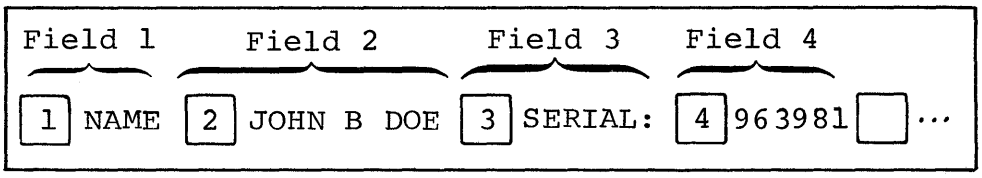



**Figure 7-1. Example of Four Fields and Attribute Bytes**

Field attributes can be modified or removed by a 3270 program. Removing the attribute character [2] causes NAME: JOHN DOE to be considered by the 3270 as a single field. Changing the content of the attribute [3] alters the characteristics of SERIAL: even though SERIAL: itself has not been altered and remains associated with that attribute.

### *What Attributes May Be Assigned to a Field*

Besides length, which is controlled by the position of attributes, you may specify these additional characteristics with the attribute character.

#### 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 7-1, NAME: would most likely be specified as protected. JOHN B DOE would be specified as protected if it were written by the computer and was to remain unchanged. If JOHN B DOE is to be entered or modified by the operator, the attribute [2] must specify unprotected.

#### Character Content

A field is either alphameric or numeric. 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 upon which keyboard is attached to the display. Figure 7-2 shows what characters may be entered with various combinations of keyboards and field types.

#### Visibility and Detectability

A field is either displayable or nondisplayable. When it is displayable and contains characters, those characters are displayed. When it is nondisplayable, any characters within that field are not displayed. The nondisplayable attribute is useful for entering classified or security information at a display unit that is in public view. Nondisplayable data is accepted by the 3270, but it is not visible on the screen.

All characters within a displayable field can be displayed at regular brightness or at a high intensity so that they stand out among regular display fields. High intensity may be used to call attention to error conditions or to highlight protected or format fields. Normal intensity may be used for all input fields, so the terminal operator can tell at a glance which fields require operator action. 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 or space 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. You are urged to designate all detectable fields as protected to prevent the operator's changing the content of the sensitive field.

| Keyboard Type | Keyboard Numeric Lock | Shift Key Pressed | Field Type | Protected | Resulting Characters | | |
|---|---|---|---|---|---|---|---|
| | | | | | In Buffer | Displayed on Screen | 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. | | |
| 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 7-2. Results of Keyboard and Field Combinations**

## Transmission

The most common operation of the 3270 (Read Modified) sends to the computer only those fields that have been entered, deleted, or changed by the operator. The 3270 keeps track of such modifications and uses that information to select data to send to the computer. If you wish to pass a field into the computer regardless of modification, you may assign the modified or modified data tag (MDT) attribute. You should note, however, that the operator can change the MDT attribute unless you also assign the protected attribute.

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 computer is not limited by attributes. The computer 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

You will find that these are the attributes most commonly used.

The attribute character for each field 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 7-3 illustrates a simple procedure in which the computer requests the operator to provide his name, location, and serial number.

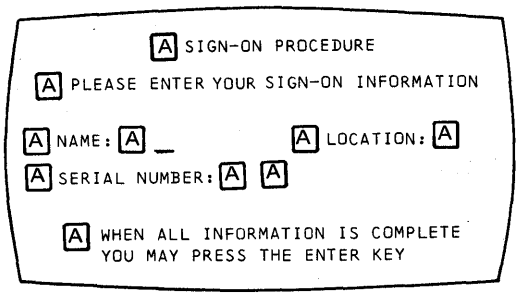



**Figure 7-3. Example of Attribute Specification**

**Field 1: SIGN-ON PROCEDURE**

This field is a heading that 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 attribute character at the beginning of the field. Field 1 and field 2 have identical attributes and are adjacent to each other. You may choose to define them separately and use 2 attribute characters, or you may choose to omit the attribute character at the beginning of field 2. In the latter case, the two headings combine to become a single field of greater length.

**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 3270 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 attribute character for field 4 and the attribute character for field 5.

**Field 5: LOCATION:**

The attribute character for this field is the same as that specified for field 3: protected and high intensity should be specified. This 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 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 attribute, but fails to do so because attribute characters 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 attribute character for this field were omitted, the work *LOCATION:* would become part of field 4 and would be of normal intensity and unprotected. This is undesirable because 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 must therefore 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 attribute character 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 of 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 filing 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 *input inhibited* 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 attribute for the next field. But the next field in the example is too far away ("WHEN ALL . . . KEY").

By placing an additional attribute character following input field 8, the operator cannot enter a serial number that is too long. If the positions allocated to the serial number are filled, the next keystroke locks the keyboard, as in the name and location fields.

This additional length check is used here because this is the last field to be entered. If you had another field to enter after SERIAL NUMBER, it might be more advantageous to omit this length check, as explained in field 9.

**Field 9: The Area between the Additional Attribute Described in Field 8 and "WHEN ALL . . . KEY"**

By definition, the additional attribute character 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 attribute defines a field as auto-skip. Auto-skip automatically positions the cursor at the location following the attribute character 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: WHEN ALL . . . KEY**

This field is a heading that the operator should not be allowed to change. It need not be of high intensity and, thus, may be defined as protected only. Field 10 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 attribute character is reached. This is called *wraparound.* Keep this in mind, particularly if you define the last field on a screen as unprotected!

Since fields 9, 10, and 1 are adjacent to each other (by wraparound) and have the same attributes, they may be combined into a single field by the omission of attributes before WHEN and SIGN-ON. The result is a single protected field beginning after the input area for serial number, 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 attribute characters 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.

## Panel Design

You can think of a panel as a single 3270 display screen image created by your program. (The term *screen* or *screen image* or *display image* could also have been used.)

If the terminal operator filled in the information requested in the panel in Figure 7-4, he might receive another panel, such as the one shown in Figure 7-5.

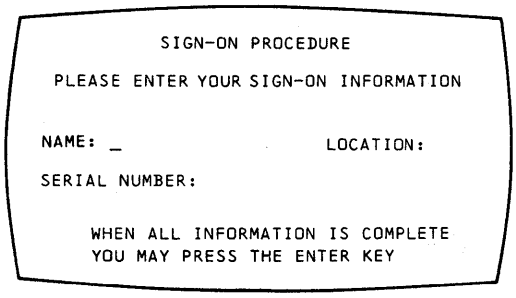



**Figure 7-4. An Example of a Panel**

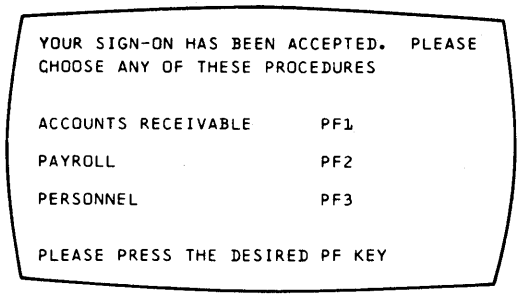



**Figure 7-5. Another Example of a Panel**

### *An Example of a Sequence of 3270 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 1,920-character panels that follow show one possible solution.

The first panel in the application is shown in Figure 7-6. 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 the invoice number are known, only Panel 1 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 7-7, 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 a corresponding program-function (PF) key. (See "Program Attention Keys" in this chapter.)

As a result of terminal operator response to Panel 2, Panel 3 (shown in Figure 7-8) displays all open invoices for the identified customer. The terminal operator can now use the selector pen 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; selection is verified immediately by the question mark's 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.

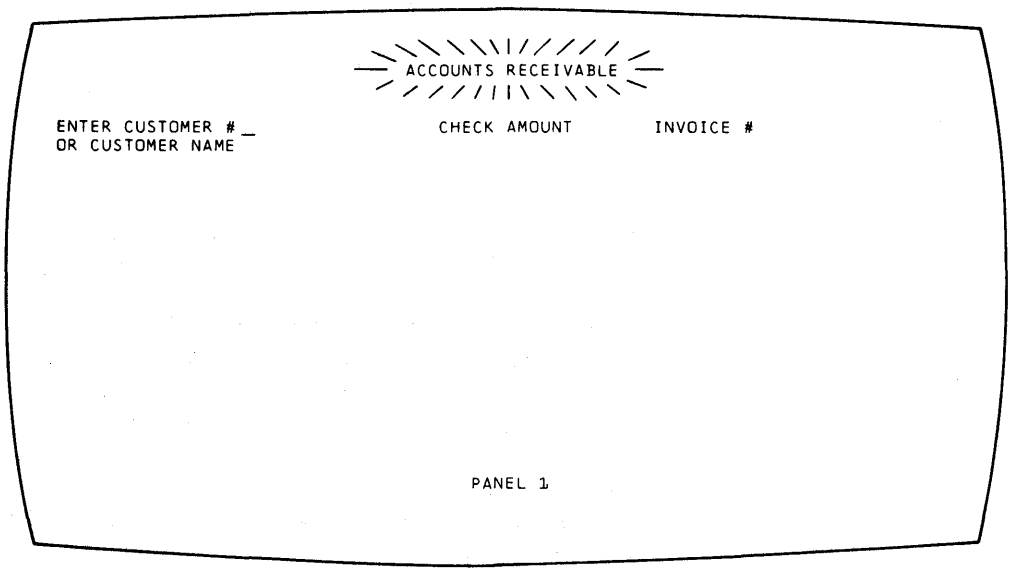



**Figure 7-6. Panel 1 of an Accounts Receivable Application**

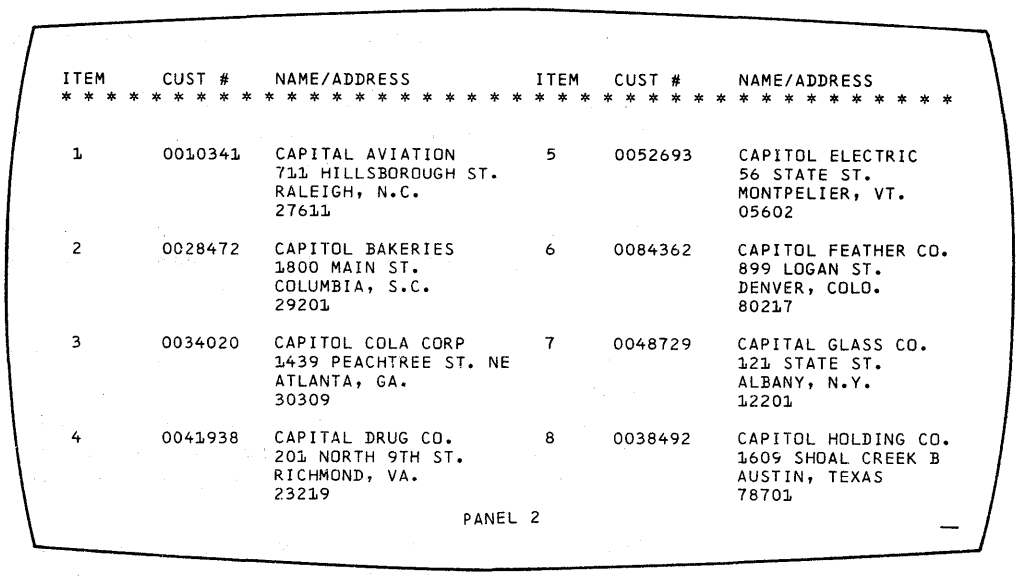



**Figure 7-7. Panel 2, Showing the Results of a Search on a Customer Name**

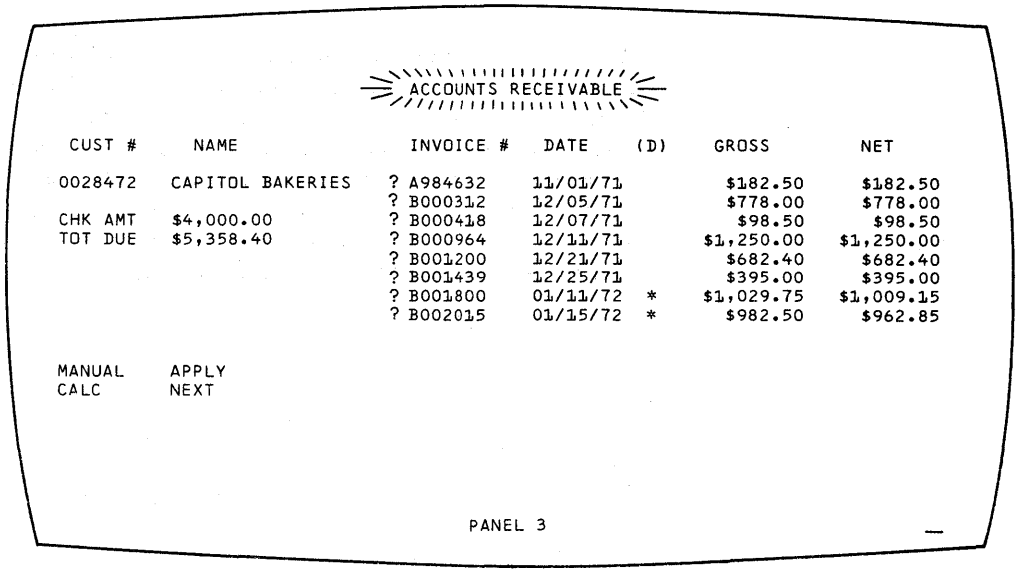



**Figure 7-8. Panel 3, Showing the Customer's Open Invoices**

Selecting CALC displays Panel 4 (Figure 7-9); this is the same as Panel 3 except that ACCOUNTS RECEIVABLE, which was of high intensity in Panel 3, is now of 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-hand 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*.

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 7-9) prior to selecting CALC and in another column show any balance remaining from the check amount after subtracting the selected invoice numbers. In Figure 7-9, 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 subtraction of the selected invoices.

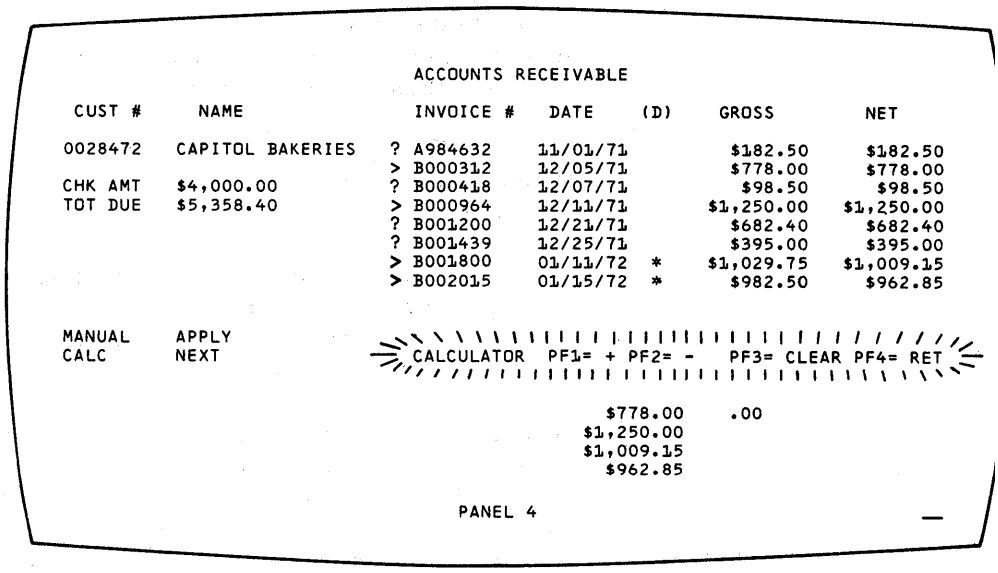



**Figure 7-9. Panel 4, Showing Use of the Calculator**

Panel 4 shows that the CALCULATOR could also allow the operator to key in amounts and to add them to 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 7-10.

Panel 5 is the same as Panel 4 except that, the operator having signaled completion of the CALCULATOR, that word now appears in normal intensity and ACCOUNTS RECEIVABLE once again appears in high intensity. The terminal operator can now, using the selector pen, select the invoices against which to apply the payment, and then select APPLY to post the payment.

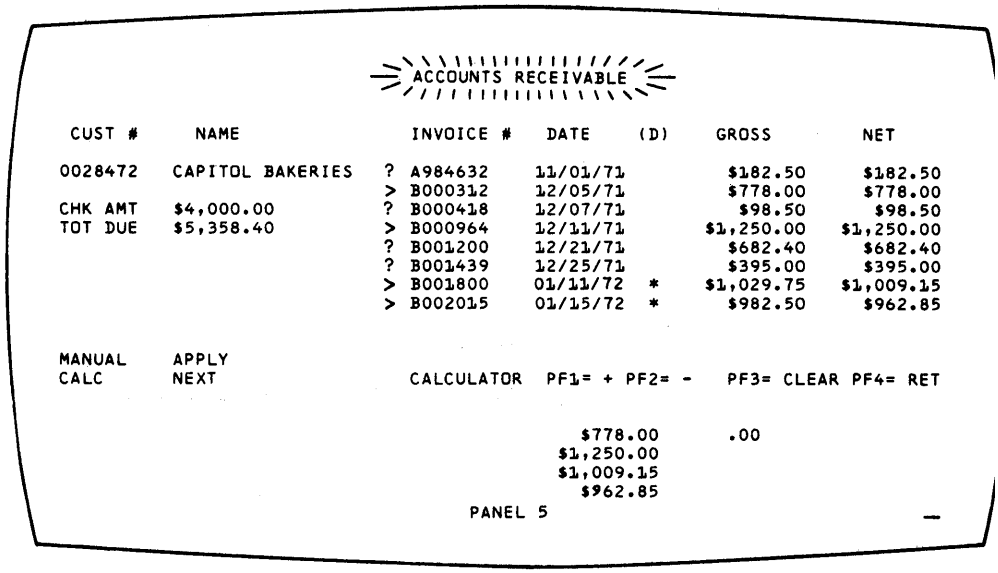



**Figure 7-10. Panel 5, Showing Selection of Invoices after Use of the Calculator**

Panel 6 (Figure 7-11) shows the ACCOUNTS RECEIVABLE file for the customer after posting of 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; 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.

## *Planning a Sequence of Panels*

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 response to panels. Usually, you will be able to approximate the sequence of panels. The exact sequence of output panels often depends on the input response to panels. The following discussion shows one way to define a sequence of panels.

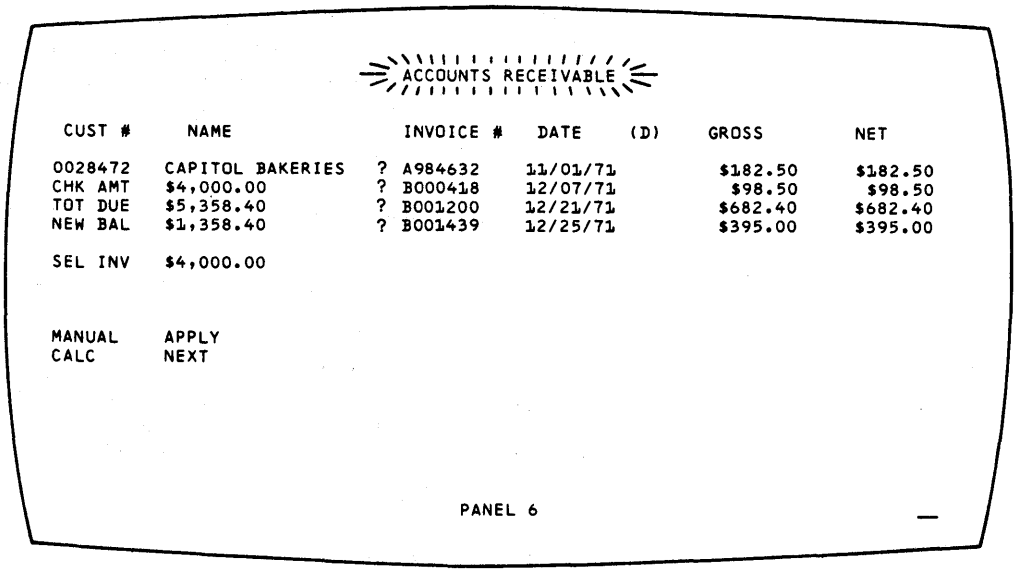



**Figure 7-11. Panel 6, Showing New Balance after Posting**

### *Defining the Purpose of Each Panel*

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 7-12.

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 7-13 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, is useful for layout.

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. With 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.

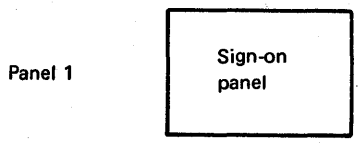



**Figure 7-12. Sign-On Panel Block Diagram**

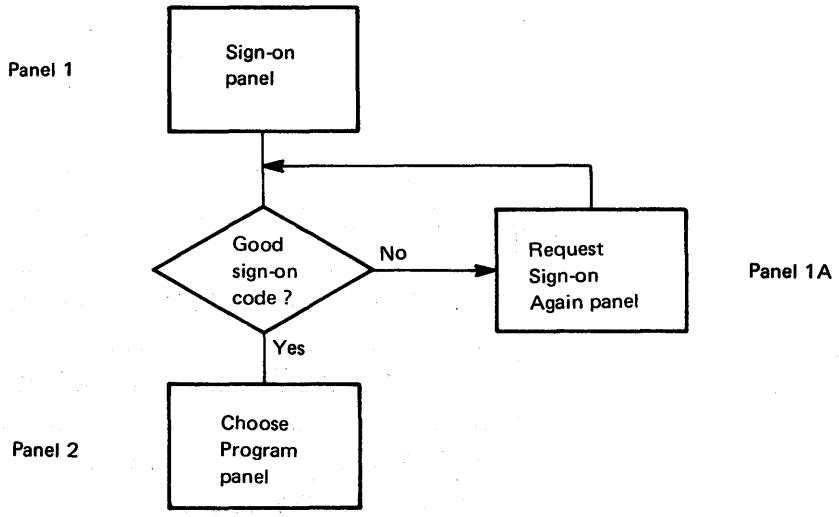



**Figure 7-13. Block Diagramming**

### *An Example of Laying Out a Panel*

To lay out a panel, consider the sign-on panel shown in Figure 7-14. 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 7-14 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. It is assumed you are using the 480-character display.

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 an attribute character. The attribute character 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 a character such as [A] on the layout sheet to indicate the space for the attribute character. As you get used to creating panels, you may want to enter the [A] 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 attribute characters and the cursor position have been added, the sign-on panel appears as shown in Figure 7-15.

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 in "What Attributes May Be Assigned to a Field."

For example, you might want the fields NAME:, LOCATION:, and SERIAL NUMBER: to have high-intensity attribute 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:, and SERIAL NUMBER: should be unprotected so that the operator can type in information. The operator input field following SERIAL NUMBER: can be numeric to allow some work station editing; the operator would not be allowed to enter an alphabetic character accidentally. 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.)

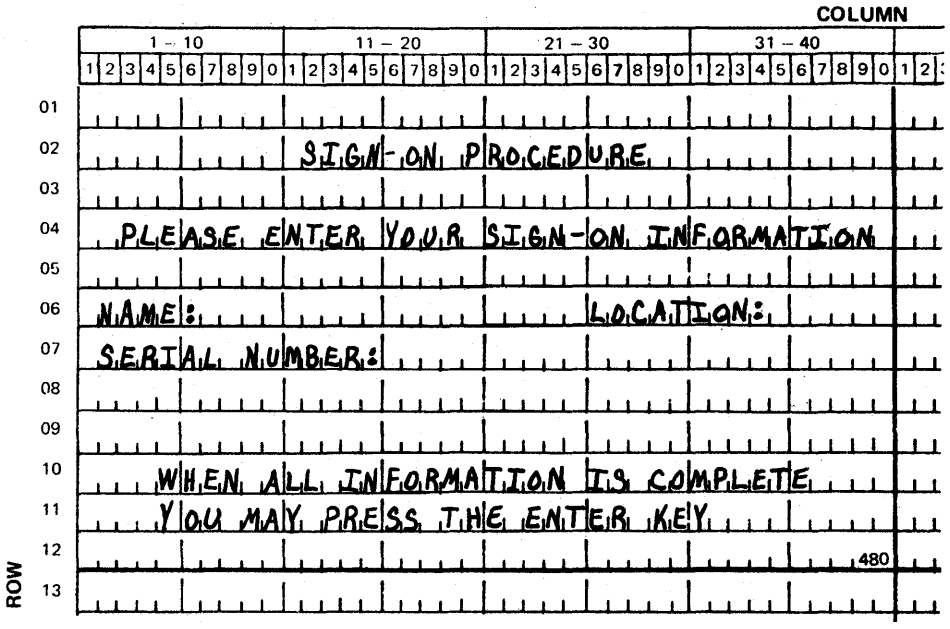



**Figure 7-14. Sign-On Panel as Written Out on Layout Sheet**

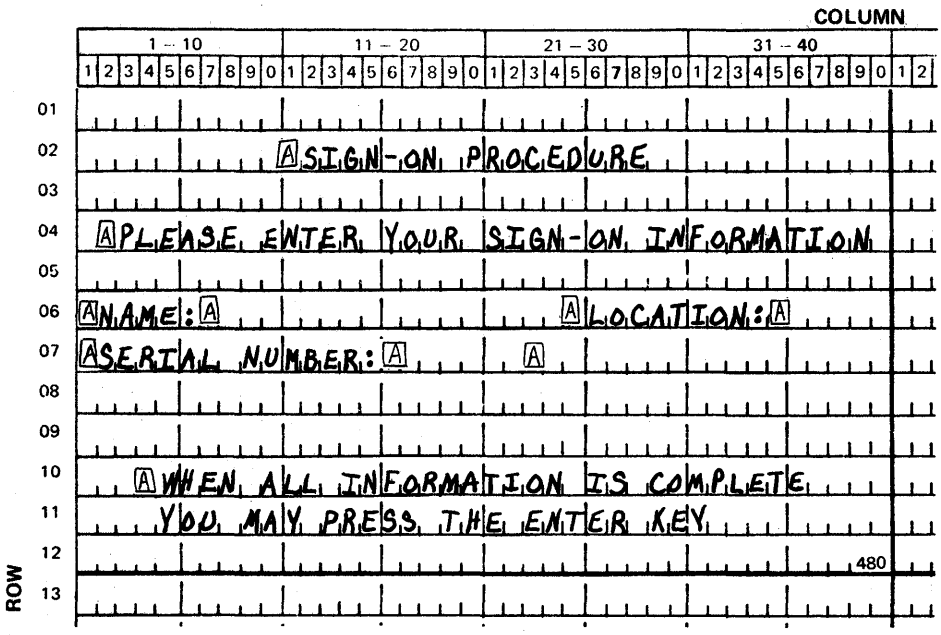



**Figure 7-15. Panel Layout, Including Attribute and Cursor Positions**

Having decided on these attributes, you can use the columns on the reverse side of the layout sheet to record the locations and attributes of the fields you have created. Your recording in these columns might appear as shown in Figure 7-16.

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. "Data Stream Coding" shows how the panel in this example is coded.

| | Display Printer | | Buffer Address | | | | Attribute | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Item | Row | Col | Dec | Hex | Orders | | Prot | No. | High Int | Sel Det | Non-Disp Prt | MDT On |
| 1 | 2 | 11 | | | | | ✓ | | | | | |
| 2 | 4 | 2 | | | | | ✓ | | | | | |
| 3 | 6 | 1 | | | | | ✓ | | ✓ | | | |
| 4 | 6 | 7 | | | | | | | | | | |
| 5 | 6 | 25 | | | | | ✓ | | ✓ | | | |
| 6 | 6 | 35 | | | | | | | | | | |
| 7 | 7 | 1 | | | | | ✓ | | ✓ | | | |
| 8 | 7 | 16 | | | | | | ✓ | | | | |
| 9 | 7 | 23 | | | | | ✓ | | | | | |
| 10 | 10 | 4 | | | | | ✓ | | | | | |

**Figure 7-16. Laying Out Field Attributes**

## Data Stream Coding

You must communicate certain information to a 3270 device or its control unit so that it can use the panels you have designed. This information includes commands, control characters, orders, 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. For the examples given below, assume that you begin with a clear screen: all writes to the 3270 are Erase/Write commands, and all positions are set to nulls. (Commands are discussed in more detail in this chapter under "The Relationship of One Data Stream to Another." Refer to Chapter 3 for the command codes.) 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. (Control characters are discussed later in this chapter.) Orders are instructions writtten to the 3270 to tell the display unit how to format your panel. They control the creation and placement of fields and data. You may reduce the size of your data streams by careful order selection. (Orders are discussed below.)

### *Orders*

Orders (1) position, define, and format data being written to the device, (2) erase selected unprotected data stored in the device, and (3) reposition the cursor.

Three orders provide enough instruction to format every panel:

- Start Field (SF) order: Specifies that the next character is an attribute character.
- Set Buffer Address (SBA) order: Specifies an address for data and successive orders.
- Insert Cursor (IC) order: Moves the cursor to the current buffer address.

These orders are included with the text, which is both the data you have in your computer for the terminal operator, such as field headings or inquiry responses, and the data that the operator has that must be provided to the computer, such as serial number, part number, or quantity desired. The orders and text are sent to the display unit and are interpreted by a control unit to which the display unit is attached. The control unit formats the panel text before it is actually displayed at the display station.

### *Adding Orders to the Panel Layout Sheet*

The back of the panel layout sheet is used for writing the panel orders. The column headings indicate what the columns should contain.

The first six columns, as shown in Figure 7-17, 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 11 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. WHEN ALL INFORMATION IS COMPLETE
  11. YOU MAY PRESS THE ENTER KEY

| Item | Display Printer Row | Col | Buffer Address Dec | Hex | Orders | |
|---|---|---|---|---|---|---|
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

**Figure 7-17. Text Items on Panel Layout Sheet**

It is only by coincidence that the number of items in this example equals the number of fields. 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 it item 11.

- 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, or IC.

As shown in Figure 7-18, the next six columns under the word Attribute provide the field attributes that can be defined with each attribute character. The programmer checks the appropriate columns of the attributes he is changing from the default values:

- Prot: Protected
- No.: Numeric
- High Int: High intensity
- Sel Det: Selector-pen-detectable
- Non-Disp/Prt: Not displayed (nor printed at printer)
- MDT On: Modified data tag on

| | Attribute | | | | | |
|---|---|---|---|---|---|---|
| Prot | No. | High Int | Sel Det | Non-Disp Prt | MDT On | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

**Figure 7-18. Field Attributes**

At the bottom of the six columns are the attribute values (Figure 7-19) that are automatically provided unless you specify a change. You must, however, specify a hexadecimal order value for the default attributes, as discussed under "Coding the Panel" in this chapter. The default values are:

- Unpr: Unprotected
- A/N: Alphameric (alphabetic and numeric)
- Norm: Displayed at regular brightness
- Non: Not detectable by the selector pen
- Norm: Displayed (at regular brightness)
- Off: Not modified

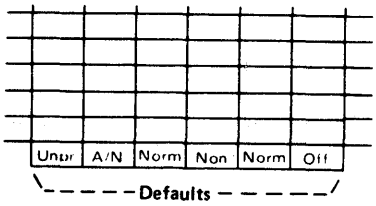



**Figure 7-19. Attribute Default Values**

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 7-20 shows a completed layout sheet containing all the orders to be sent with the sign-on panel. (The hexadecimal order values are discussed under "Coding the Panel" in this chapter.) Each item on the panel has been assigned a number to help you correlate the text with its associated orders.

| | Display Printer | | Buffer Address | | | | Attribute | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Item | Row | Col | Dec | Hex | Orders | | Prot | No. | High Int | Sel Det | Non-Disp Prt | MDT On |
| 1 | 02 | 11 | | | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | | | | |
| 2 | 04 | 02 | | | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | | | | |
| 3 | 06 | 01 | | | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | ✓ | | | |
| 4 | 06 | 07 | | | SF | ATT | | | | | | |
| | | | | | IC | | | | | | | |
| 5 | 06 | 25 | | | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | ✓ | | | |
| 6 | 06 | 35 | | | SF | ATT | | | | | | |
| 7 | 07 | 01 | | | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | ✓ | | | |
| 8 | 07 | 16 | | | SF | ATT | | ✓ | | | | |
| 9 | 07 | 23 | | | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | | | | |
| 10 | 10 | 03 | | | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | | | | |
| 11 | 11 | 05 | | | SBA | | | | | | | |

**Figure 7-20. Completed Order and Attribute Information**

Item 1. SIGN-ON PROCEDURE

To write this title, you must tell the control unit:

- Where you want the title displayed on the panel. The SBA order sets the buffer address (SBA) to the location at row 2, column 11 (R2, C11).
- That this location is the start of a field. The SF order tells the control unit that the location contains an attribute character and not a text character. You also indicate which attributes the attribute character is defining. In this case, the field is protected. The rest of the attributes for the field are default attributes and, therefore, do not have to be changed.

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 write an SBA order followed by the address R4, C2. This is also the beginning of a protected field, so you should include an SF order and a protected attribute.

Item 3. NAME:

As with item 2, you must idenfity where this text is displayed. Therefore, you must write an SBA order followed by the buffer address R6, C1, where the text begins. R6, C1 is also the beginning of a protected, high-intensity field and you should include an SF and an attribute as shown.

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 an attribute character next. The attribute character defines the input field as unprotected (U), alphameric (A), of normal intensity, not detectable by selector pen, and with 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 attribute character 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 that attribute character in location R6, C7, the new current address is R6, C8; this is the place where the cursor appears on the panel.

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 R6, C25, and (2) indicate that it is the start of a new field (SF), that it is protected, and that it has high intensity.

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.

Item 7. SERIAL NUMBER:

This field requires an SBA to location R7, C1, and an SF to begin a new field. The attribute is specified the same as that for item 5.

Item 8. Input field for serial number

The attribute character for this input field immediately follows the last character of the previous field, so an SBA is not required. The attribute is numeric only.

Item 9. An extra field created to limit the size of the serial number input field.

This follows the input field and is protected only. An SBA is required for location R7, C23, for proper placement of the attribute.

Item 10. WHEN ALL . . . COMPLETE.

The control unit must have two orders for this item: an SBA order that gives the starting address of R10, C3, and an SF order to indicate that it is the start of a new field. The attribute character defines a protected field, and the rest of the field attributes take the default values.

Item 11. YOU MAY . . . KEY.

All the words from "WHEN ALL" through "KEY" could have been treated as a single item, but 8 blank spaces would have to be sent between "COMPLETE" and "YOU" to position "YOU" properly at R11, C5. Use only the 3 characters required for an SBA order and its associated address, breaking the field into 2 items, to position "YOU" at R11, C5.

## *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 an assembler coding sheet or to any other form you find suitable.

On the coding sheet (and in your program), a panel is represented by a series of assembler DC statements, each with a name to which your program can refer. In the example given below, 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 command and designates SIGNPANL as the panel for display.

The display orders must be written in the DC statements in the hexadecimal codes listed in Figure 7-21. Thus, SF is represented by 1D, SEA by 11, and IC by 13.

| Order Sequence / Order | Byte 1 (Order Code) EBCDIC (Hex) | Byte 1 (Order Code) ASCII (Hex) | Byte 2 | Byte 3 | Byte 4 |
|---|---|---|---|---|---|
| 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. |
| Erase Unprotected to Address (EUA) | 12 | 12 | Address | Address | |
| Keyboard Only | | | | | |
| Duplicate (DUP) | 1C | 1C | | | |
| Field Mark (FM) | 1E | 1E | | | |

**Figure 7-21. Buffer Control Orders and Order Codes**

Each part of each order must be written in hexadecimal, including the attribute character that follows the SF order and the buffer address that follows the SBA order. The *IBM 3270 Reference Summary*, GX20-1878, contains the hexadecimal codes for all the attribute character combinations and the hexadecimal code for every buffer location in both EBCDIC and ASCII.

Begin coding with the first item on the panel layout sheet, the title: SIGN-ON PROCEDURE. Start with the orders for the panel text, which must always precede the text itself so that the control unit knows what to do with the text.

The first order for the title is the SBA order. Figure 7-21 shows that the SBA hexadecimal code is 11, so you write this code in a DC statement as:

```
DC X'11'
```

Now look up the R2, C11 address that must follow the SBA order. The EBCDIC address is 40F2, and it follows the SBA code in the DC statement:

```
DC X'1140F2'
```

You should also record this statement in the Buffer Address Hex column to the left of the SBA on the layout form for possible future reference. You may, if you prefer, look up all the addresses and record them in a similar manner before you begin to write your DC statements. See Figure 7-22 for an example.

The next order for the title is the SF order, which is followed by the attribute character. Attribute characters are shown in Figure 7-23. The SF code, 1D, and the attribute code, 60, are read from the table and added to the DC statement, which is then closed with a single quotation mark:

```
DC X'1140F21D60'
```

Following the DC statement containing the orders for the title is the DC statement containing the text for the title:

```
DC X'1140F21D60'
DC C'SIGN-ON PROCEDURE'
```

| | Display Printer | | Buffer Address | | | | Attribute | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Item | Row | Col | Dec | Hex | Orders | | Prot | No. | High Int | Sel Det | Non-Disp Prt | MDT On |
| 1 | 02 | 11 | | 40F2 | SBA | | | | | | | |
| | | | | | SF | Att | ✓ | | | | | |
| 2 | 04 | 02 | | C1F9 | SBA | | | | | | | |
| | | | | | SF | Att | ✓ | | | | | |
| 3 | 06 | 01 | | C3C8 | SBA | | | | | | | |
| | | | | | SF | Att | ✓ | | ✓ | | | |
| 4 | 06 | 07 | | | SF | Att | | | | | | |
| | | | | | IC | | | | | | | |
| 5 | 06 | 25 | | C360 | SBA | | | | | | | |
| | | | | | SF | Att | ✓ | | ✓ | | | |
| 6 | 06 | 35 | | | SF | Att | | | | | | |
| 7 | 07 | 01 | | C3F0 | SBA | | | | | | | |
| | | | | | SF | Att | ✓ | | ✓ | | | |
| 8 | 07 | 16 | | | SF | Att | | ✓ | | | | |
| 9 | 07 | 23 | | C4C6 | SBA | | | | | | | |
| | | | | | SF | Att | ✓ | | | | | |
| 10 | 10 | 03 | | C5A1 | SBA | | | | | | | |
| | | | | | SF | Att | ✓ | | | | | |
| 11 | 11 | 05 | | C6D9 | SBA | | | | | | | |
| | | | | | | | | | | | | |
| | | | | | | | | | | | | |

**Figure 7-22. Sign-On Procedure Panel Orders and Attributes**

**ATTRIBUTE CHARACTER BIT DEFINITIONS**

| Attribute | | | | | | Bits | Hex |
|---|---|---|---|---|---|---|---|
| Prot | A/N | MDT On | High Intens | Sel Det | Non-disp Prt | 23 4567 | |
| U | | | | | | 00 0000 | 40 |
| U | | Y | | | | 00 0001 | C1 |
| U | | | | Y | | 00 0100 | C4 |
| U | | Y | | Y | | 00 0101 | C5 |
| U | | | H | Y | | 00 1000 | C8 |
| U | | Y | H | Y | | 00 1001 | C9 |
| U | | | – | – | Y | 00 1100 | 4C |
| U | | Y | – | – | Y | 00 1101 | 4D |
| U | N | | | | | 01 0000 | 50 |
| U | N | Y | | | | 01 0001 | D1 |
| U | N | | | Y | | 01 0100 | D4 |
| U | N | Y | | Y | | 01 0101 | D5 |
| U | N | | H | Y | | 01 1000 | D8 |
| U | N | Y | H | Y | | 01 1001 | D9 |
| U | N | | – | – | Y | 01 1100 | 5C |
| U | N | Y | – | – | Y | 01 1101 | 5D |
| P | - | | | | | 10 0000 | 60 |
| P | | Y | | | | 10 0001 | 61 |
| P | | | | Y | | 10 0100 | E4 |
| P | | Y | | Y | | 10 0101 | E5 |
| P | | | H | Y | | 10 1000 | E8 |
| P | | Y | H | Y | | 10 1001 | E9 |
| P | | | – | – | Y | 10 1100 | 6C |
| P | | Y | – | – | Y | 10 1101 | 6D |
| P | S | | | | | 11 0000 | F0 |
| P | S | Y | | | | 11 0001 | F1 |
| P | S | | | Y | | 11 0100 | F4 |
| P | S | Y | | Y | | 11 0101 | F5 |
| P | S | | H | Y | | 11 1000 | F8 |
| P | S | Y | H | Y | | 11 1001 | F9 |
| P | S | | – | – | Y | 11 1100 | 7C |
| P | S | Y | – | – | Y | 11 1101 | 7D |

S = Skip  Y = Yes
U = Unprotected  H = High
P = Protected  N = Numeric

**Note:** *Hexadecimal values are given in EBCDIC.*

**Figure 7-23. Attribute Character Combinations in Hexadecimal**

To code an input field that contains no text, such as the input field for NAME:, write just one DC statement that contains the orders for that field:

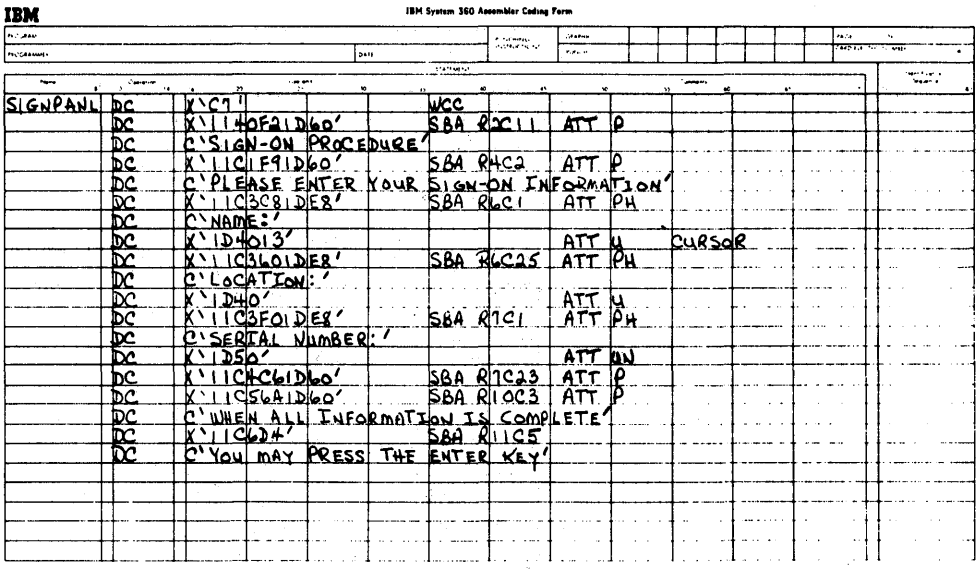
```
DC X'1D4013'
```

1D is the hexadecimal code for the SF order, 40 is the hexadecimal code for an attribute character that defines an unprotected field (and all other default attributes), and 13 is the hexadecimal code for the IC order.

A DC statement can be written as two or more statements. The DC statement above, for example, could be written as:

```
DC X'1D40'
DC X'13'
```

Each item from the panel layout sheet is coded in this fashion. Figure 7-24 shows the complete code required to display the sign-on panel. Except for one control character, it consists entirely of the panel text, preceded by the display orders for that text. (The control character is described under the heading "Write Control Character (WCC).")



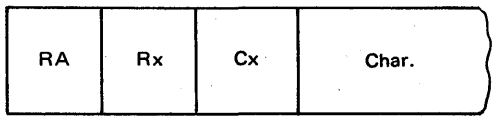
```
IBM                                IBM System 360 Assembler Coding Form
PROGRAMMER                  DATE
STATEMENT

SIGNPANL  DC  X'C7'                    WCC
          DC  X'11'40F21D60'           SBA R2C11     ATT P
          DC  C'SIGN-ON PROCEDURE'
          DC  X'11'C1F91D60'           SBA R4C2      ATT P
          DC  C'PLEASE ENTER YOUR SIGN-ON INFORMATION'
          DC  X'11'C3C81DE8'           SBA R6C1      ATT PH
          DC  C'NAME:'
          DC  X'1D4013'                              ATT N   CURSOR
          DC  X'11'C3L01DE8'           SBA R6C25     ATT PH
          DC  C'LOCATION:'
          DC  X'1D40'                                ATT N
          DC  X'11'C3F01DE8'           SBA R7C1      ATT PH
          DC  C'SERIAL NUMBER:'
          DC  X'1D50'                                ATT DN
          DC  X'11'C4C61D60'           SBA R7C23     ATT P
          DC  X'11'C5A41D60'           SBA R10C3     ATT P
          DC  C'WHEN ALL INFORMATION IS COMPLETE'
          DC  X'11'C6D4'               SBA R11C5
          DC  C'YOU MAY PRESS THE ENTER KEY'
```

**Figure 7-24. Assembler Language Statements for Sign-On Panel**

## *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. You specify the stop address immediately following the RA order, just as you specify an address after an SBA order. After the stop address, you specify the character that you want repeated. Symbolically this appears as:




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 YOU MAY PRESS THE ENTER KEY, you specify an SBA for R12, 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 7-25.

The order in the example is coded as:

```
DC X'3C4040'
DC C'*'
```

If you want to delete a field already on the screen, you can repeat the ***null*** character to delete it.

| Item | Display Printer Row | Col | Buffer Address Dec | Hex | Orders | | Attribute Prot | No. | High Int | Sel Det | Non-Disp Prt | MDT On |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 02 | 11 | | 40F2 | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | | | | |
| 2 | 04 | 02 | | C1F9 | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | | | | |
| 3 | 06 | 01 | | 83C8 | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | ✓ | | | |
| 4 | 06 | 07 | | | SF | ATT | | | | | | |
| | | | | | IC | | | | | | | |
| 5 | 06 | 25 | | C360 | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | ✓ | | | |
| 6 | 06 | 35 | | | SF | ATT | | | | | | |
| 7 | 07 | 01 | | C3F0 | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | ✓ | | | |
| 8 | 07 | 16 | | | SF | ATT | | ✓ | | | | |
| 9 | 07 | 23 | | C4C6 | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | | | | |
| 10 | 10 | 03 | | C5AA | SBA | | | | | | | |
| | | | | | SF | ATT | ✓ | | | | | |
| 11 | 11 | 05 | | C6D4 | SBA | | | | | | | |
| 12 | 01 | 01 | | | RA | * | | | | | | |

**Figure 7-25. Example of RA Order**

### *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 as the first character of 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 attribute characters 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 7-26.

The sign-on panel data is now complete and can be sent to the display unit.

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 |

**Note:** ***Hexadecimal codes are given in EBCDIC.***

**Figure 7-26. WCC Hexadecimal Codes**

## Analyzing Input Data

### *The Operator's Response*

When the sign-on panel is displayed, the operator responds by entering name, location, and serial number as shown in Figure 7-27. 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.

When the operator finishes entering the requested sign-on 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 fields

Figure 7-28 shows this sequence of input data, which is explained below.

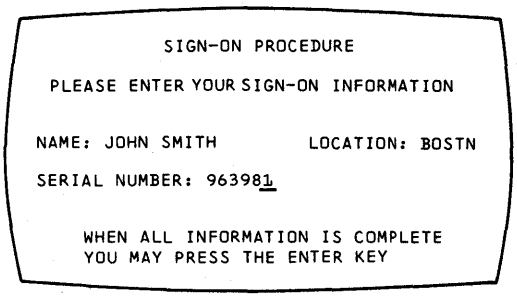



**Figure 7-27. Sign-On Panel with Operator's Input**

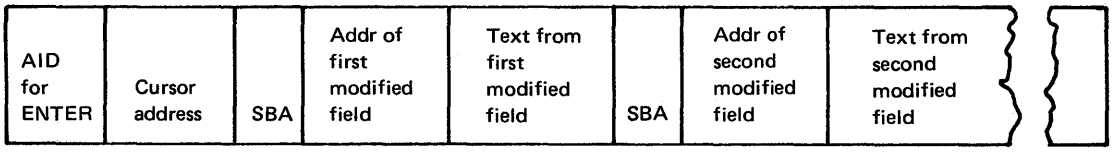



**Figure 7-28. Input Data Sequence**

### *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."

The AID code is always the first code your program receives from the display unit. The hexadecimal codes for all AID codes are shown in Figure 7-29.

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 attribute character 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 attribute character.

When any character location in an input field is modified by the operator, the MDT in the attribute character 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 attribute character whose code indicates that its MDT is on. Each time one is found, the display unit provides an SBA code and the starting address (the attribute character'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.

**Attention Identification (AID) Configuration**

**AID Values for Text Read**

| Graphic Character | EBCDIC (Hex) | Operator Action |
|---|---|---|
| – | 60 | No action by display operator |
| Y | E8 | No action (printer) |
| ' | 7D | ENTER key pressed |
| 1 | F1 | PF key 1 pressed |
| 2 | F2 | PF key 2 pressed |
| 3 | F3 | PF key 3 pressed |
| 4 | F4 | PF key 4 pressed |
| 5 | F5 | PF key 5 pressed |
| 6 | F6 | PF key 6 pressed |
| 7 | F7 | PF key 7 pressed |
| 8 | F8 | PF key 8 pressed |
| 9 | F9 | PF key 9 pressed |
| : | 7A | PF key 10 pressed |
| # | 7B | PF key 11 pressed |
| @ | 7C | PF key 12 pressed |
| = | 7E | Immediately detectable field selected |
| 0 | F0 | TEST REQUEST key pressed |
| W | E6 | Data transferred from card reader |

**AID Values for Short Read**

| | | |
|---|---|---|
| – | 6D | CLEAR key pressed (screen cleared) |
| % | 6C | PA1 key pressed |
| > | 6E | PA2 (cancel) key pressed |
| , | 6B | PA3 key pressed |

Figure 7-29. Attention Identifiers (AIDs) in Hexadecimal Codes (EBCDIC)

### *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 whether it is (X'C34F'). If it is, your program knows that the text following 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. (See Figure 7-29.)

C4C6: The cursor address R7, C23. The cursor is at the next character location after the entered serial number.

11: The SBA (Set Buffer Address) order code, which tells the program that the next 2 characters are addresses. (See Figure 7-21.)

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 Access Keys

### *Program Attention (PA) Keys*

Each 3270 keyboard has at least one program (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 7-29 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 the AID code only.

Your program should use these keys for operator requests for immediate action, such as trouble alerts, or for 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 timesaving 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 Input and Output

Positioning data for selector-pen (optional feature) use and setting the attribute characters are the same as for any other type of data, but the selector pen has additional data-stream requirements.

### *Selector-Pen Field Format*

A field for selector-pen operations must be defined as shown in Figure 7-30.

The attribute character, the designator character (described under the next heading), 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 attribute character can be detected by the selector pen.

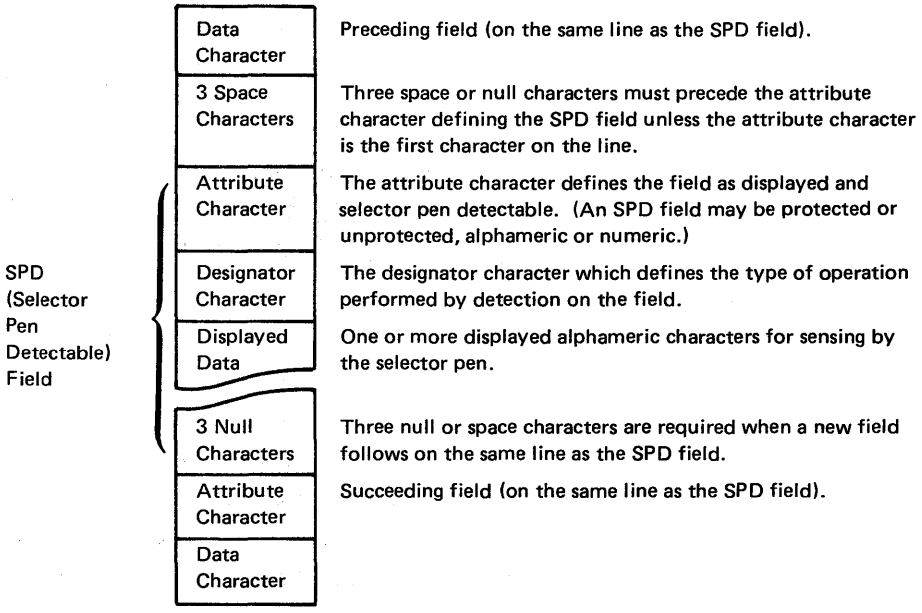



**Figure 7-30. Definition of Field for Selector-Pen Operation**

### *Designator Characters*

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

The selection field is defined by a question mark (?) designator character. When the selector pen detects a selection field, the MDT bit in the attribute character 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 the same field, the > reverts to a ? and the MDT is reset. The attention field is defined by a space or null designator character. Probing an attention field is similar to using an ENTER key. The input information is released to be read by the computer when it is ready to do so.

Figure 7-31 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.

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 3 required null characters and positions the SF, 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 the ENTER field releases the selection to the computer.

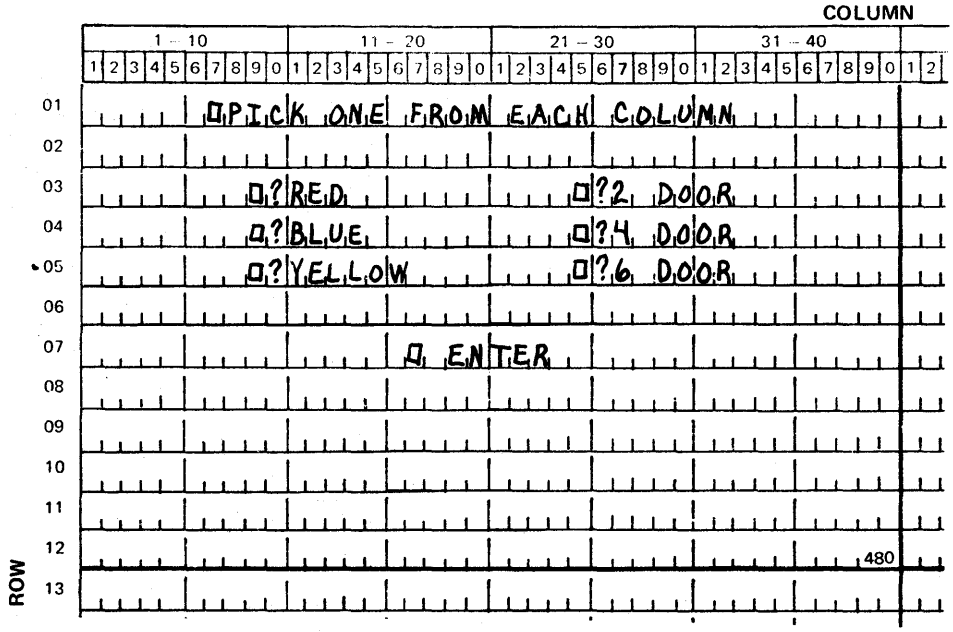



**Figure 7-31. Sample Panel for Selector-Pen Detection**

As the operator uses the selector pen, the program correlates the address of each selector-pen-detectable field with the data associated with it.

To combine selector-pen-detectable input with keyboard input, use the keyboard to release the data to the computer by pressing the ENTER key or a PF key. Use of the selector pen to release the data transmits only the addresses of the selector-pen-detected fields.

In this 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.

### *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 is has been read into the computer, the screen appears as shown in Figure 7-32. You would like the screen to look like Figure 7-33. Most of the information you want displayed is already there. An Erase/Write 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 7-32 to look like Figure 7-33, you would:

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.

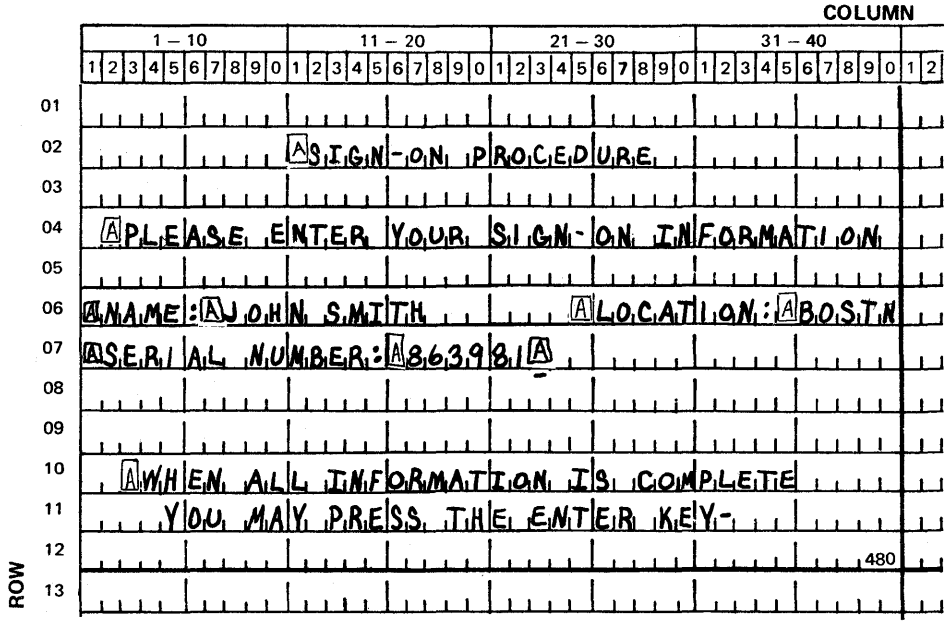



Figure 7-32. Modifying an Existing Panel – Basic Panel

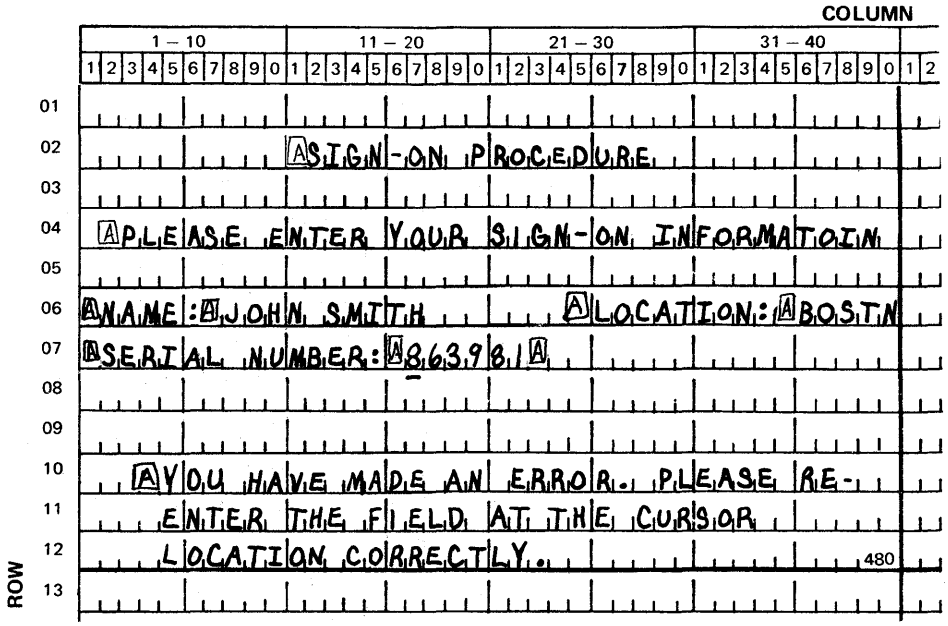



Figure 7-33. Existing Panel with Error Message

To do this the right side of your panel layout for the error panel might (in abbreviated form) look like Figure 7-34.

Item 1. Repositions the cursor to R7, C17.

Item 2. Changes the attribute at R10, C4 to protected and high intensity. If the designer of the sign-on panel had combined the original field at this location with the previous field, the SIGN-ON PROCEDURE field, and the following field by omitting the attributes at R10, C4, R2, C11, and R4, C2 (as you saw in the discussion of attributes), the result would have been 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 of 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 place the second line of the error message correctly: 3 characters are used instead of 6 null characters.

Item 4. Repositions the data flow for the third line of the error message.

Since there are two different types of Write commands, 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 persons responsible for the I/O portion of the program.

| | Display Printer | | Buffer Address | | | | Attribute | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Item | Row | Col | Dec | Hex | Orders | | Prot | No. | High Int | Sel Det | Non-Disp Prt | MDT On |
| 1 | 07 | 17 | | | SBA | | | | | | | |
| | | | | | IC | | | | | | | |
| 2 | 10 | 04 | | | SBA | | | | | | | |
| | | | | | SF | Att | ✓ | | ✓ | | | |
| | | "L | INE | / | OF | ERROR | MESSA | GE" | | | | |
| 3 | 11 | 05 | | | SBA | | | | | | | |
| | | "LI | NE | 2 | OF | ERROR | MESSA | GE" | | | | |
| 4 | 12 | 05 | | | SBA | | | | | | | |
| | | "LINE | | 3 | OF | ERROR | MESSA | GE" | | | | |

**Figure 7-34. Panel Layout Changes for Error Message (Keyed to Text)**

## Write Control Character (WCC)

When the operator presses the ENTER key after filling in the sign-on panel, the keyboard automatically locks, as it always does after an operator-initiated input operation. One function of the WCC, which was also discussed under "Coding the Panel," is to enable the keyboard. You should now decide whether you want the WCC at the beginning of the error panel data stream to enable the keyboard for the operator. Though it is normal to enable the keyboard at this point, you may not want to do it here. It might be better for the operator to press the RESET key, calling further attention to the error panel.

In Figure 7-32, 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 computer? 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 3270 to turn on the MDT for this field, and any field that 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. Though an Erase/Write resets all MDTs, a Write does not; therefore, if you do not reset them, all 3 input fields are returned to the computer. Since not all of them have changed, not all 3 should return to the computer. 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 7-26). You can now use the Write command to change the sign-on panel into the error message panel.

**Caution: 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.**

This 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 (see WCC above) 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. In addition, in both local and remote use, successive Write commands without an intervening READ may result in a *blinking* effect while you build up the panel. Blinking may be annoying to the operator.

Wherever possible, use a single Write command to avoid the inconveniences noted above.

### Erase Unprotected to Address

The error panel shown in Figure 7-33 displayed the erroneous serial number. All the operator had to do was 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 7-35.

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 Erase Unprotected to Address order, or EUA (watch the sequence of these letters so you do not confuse them with EAU, which is discussed next.) The EUA order inserts nulls (erases all unprotected positions, including attributes) from the current buffer address up to, but not including, the specified stop address.

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' EUA) is immediately followed by a row and column address.

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 3 characters of transmission (see Figure 7-36). The current buffer address is the stop address. Any data or SF order that follows goes into the buffer at this address.

The EUA order 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 7-37.

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 7-38).

You could have started at R6, C8 with an SBA to R6, C8, followed by the EUA to R10, C4. Sometime later in the data stream, however, you would have had to *back up*, probably with an SBA to insert the cursor.

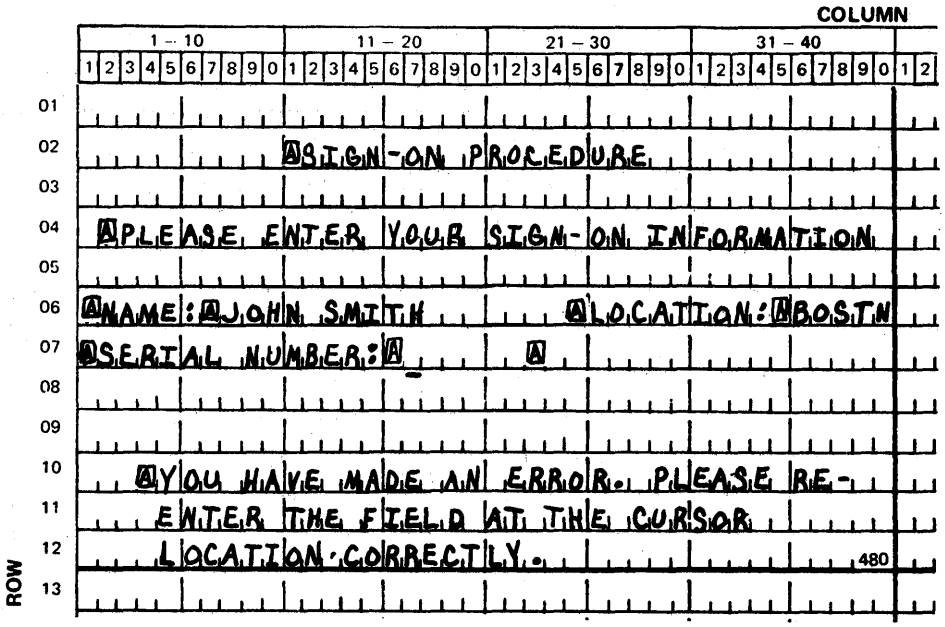



**Figure 7-35. Error Message Panel with Serial Number Field Erased**

| Item | Display/Printer Row | Display/Printer Col | Buffer Address Dec | Buffer Address Hex | Orders | | Attribute Prot | Attribute No. | Attribute High Int | Attribute Sel Det | Attribute Non-Disp/Prt | Attribute MDT On |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 07 | 17 | | | SBA | | | | | | | |
| | | | | | 1C | | | | | | | |
| | 10 | 04 | | | EUA | | | | | | | |
| 2 | | | | | SF | Att | ✓ | | ✓ | | | |
| | "LINE | 1 | OF | | ERROR MESSAGE" | | | | | | | |
| | | | | | • | | | | | | | |
| | | | | | • | | | | | | | |
| | | | | | • | | | | | | | |

**Figure 7-36. Example of EUA Use**

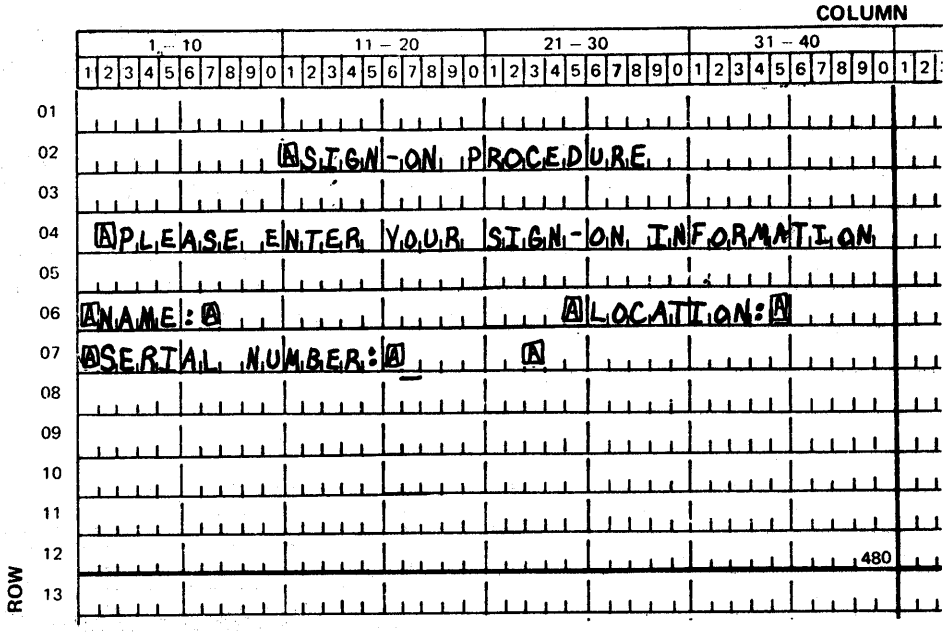



**Figure 7-37. Sign-On Panel with Three Erased Fields**

| Item | Display Printer Row | Display Printer Col | Buffer Address Dec | Buffer Address Hex | Orders | | Attribute Prot | Attribute No. | Attribute High Int | Attribute Sel Det | Attribute Non-Disp Prt | Attribute MDT On |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 01 | 07 | 17 | | | SBA | | | | | | | |
| | | | | | IC | | | | | | | |
| | 06 | 08 | | | SBA | | | | | | | |
| | 10 | 04 | | | EUA | | | | | | | |
| | | | | | SF | All | ✓ | | ✓ | | | |
| | | | | | • | | | | | | | |
| | | | | | • | | | | | | | |
| | | | | | • | | | | | | | |

**Figure 7-38. Erasing Multiple Fields with EUA**

### Erase All Unprotected Command

In the preceding example, you wanted to erase all unprotected data, reposition the cursor, and add some titles to the sign-on panel to make it an error panel. The Erase All Unprotected (EAU) command:

- Clears all unprotected fields (except attributes) to nulls.
- Resets MDTs in all unprotected fields.
- Unlocks the keyboard.
- Resets the AID (see "Program Access Keys").
- Repositions the cursor to the first character of the first unprotected field.

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 Erase All Unprotected 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 7-39.

The operator keys JOHN SMITH, presses TAB, keys BOSTN, presses TAB, keys 963981, and presses ENTER (Figure 7-40).

You simply send the 3270 and EAU command to unlock the keyboard. The operator then sees the same panel as in Figure 7-39. The operator may now key data for the next employee. You have used your knowledge of what is displayed already to arrive at the next panel or to re-create the present panel.

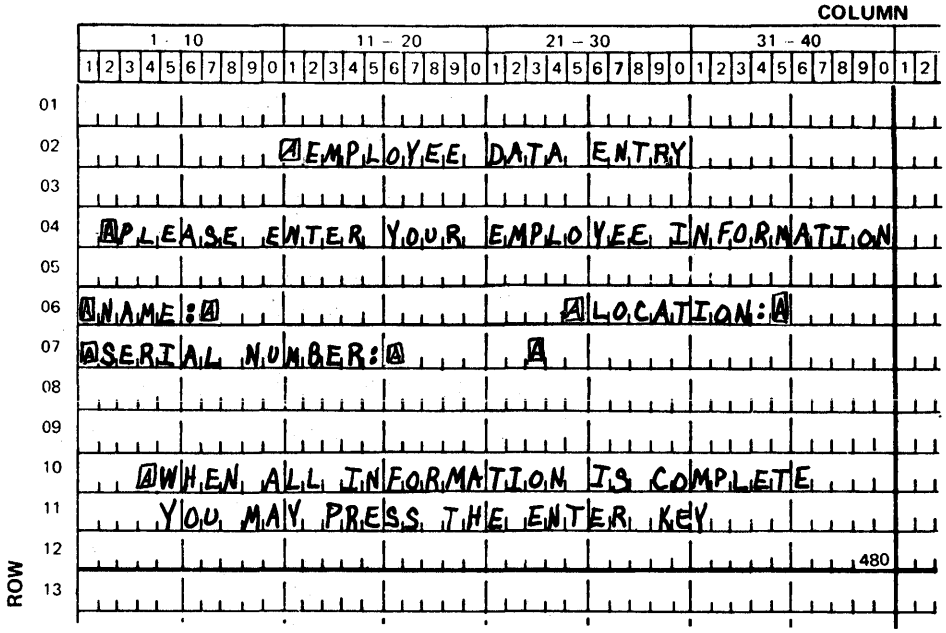



**Figure 7-39. Example of Data Entry Panel**

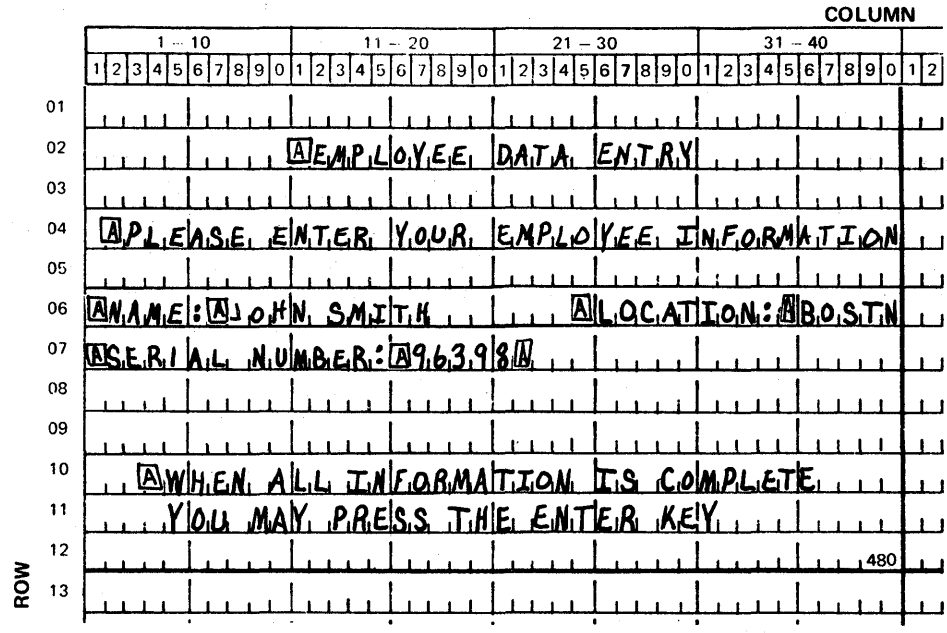



**Figure 7-40. Data Entry Panel with Entered Data**

### 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 7-41.

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 backwards. Remember, PA keys are assigned by the program. Program access 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; its use is therefore preferred to that of the ENTER or PF keys.

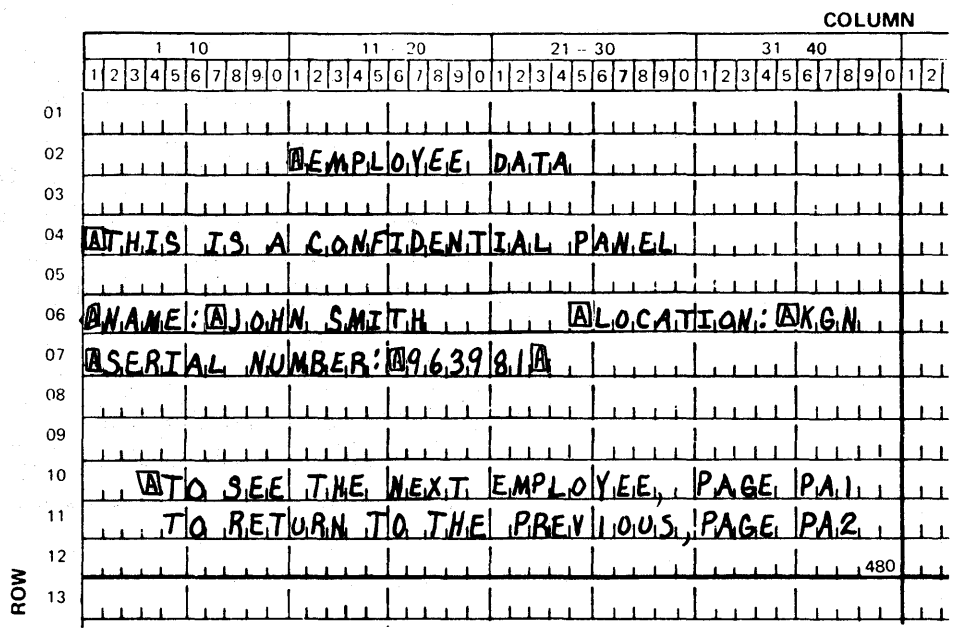
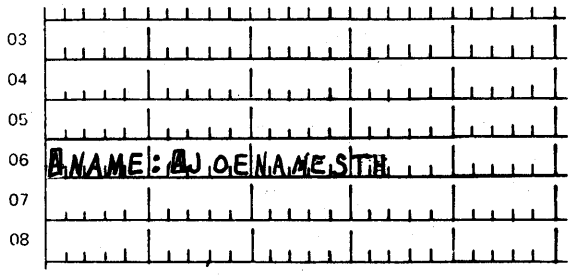



**Figure 7-41. Employee Data Panel**

### Program Tab

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.

When the operator is ready to view the information for the next employee, he presses the PA1 key. Since you want only to modify 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:

| WCA | SBA | R6 | C36 | J | O | 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 7-42.

As you have seen, you can write each new panel out in its entirety with an Erase/Write command. You should understand the relationship between the past data streams and the one you are building.

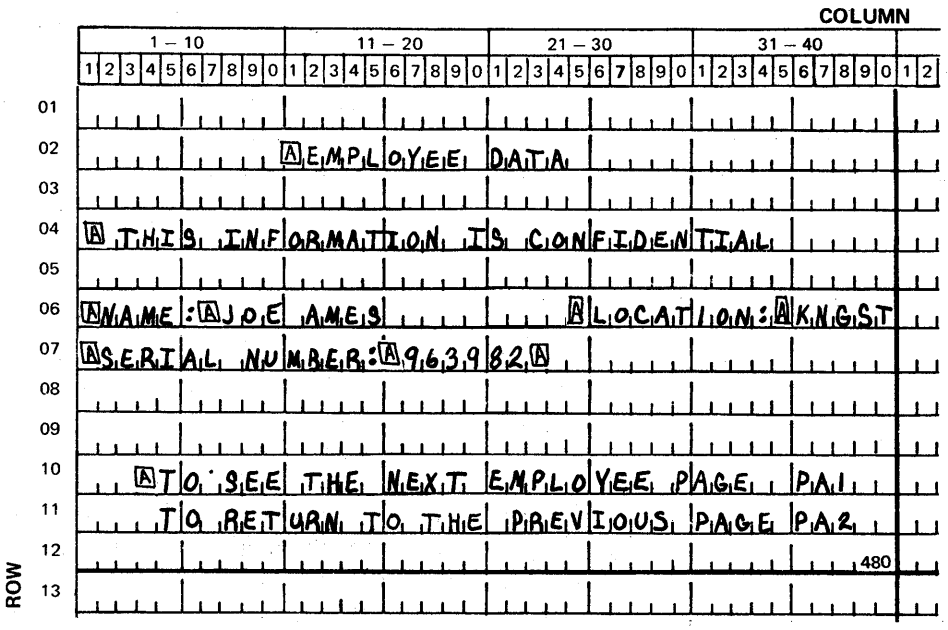



**Figure 7-42. Panel Defined with Program Tab**

# Chapter 8. Screen Management

A screen-management program module is a set of subroutines physically separate from application programs and from the telecommunication-management program module of an online 3270 system. Figure 8-1 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 the telecommunication-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 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 an attribute character with the protected option and the second buffer position contains a null character, the control unit rejects any attempt to copy from that device.

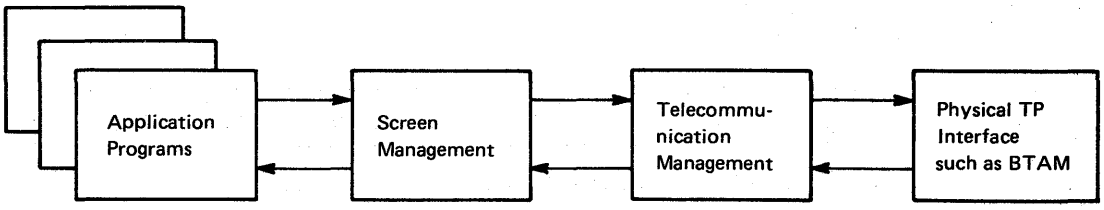



**Figure 8-1. Relationship of Screen Management to Telecommunication Management and Application Programs**

## 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.

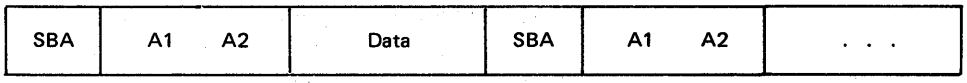
This chapter discusses several approaches to developing a screen-management module whose functions the application programmer can use to prepare output data streams and to decode input data streams. The approaches demonstrate how some 3270 device-dependent considerations can be removed from the application 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 BTAM 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:




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 into a binary value that 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 use of the above technique, several approaches can 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.

## Non-Selector-Pen Data Streams

**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 of the same size as the display buffer (480 or 1,920). The data read from the display is placed in the same relative position in the main storage buffer that 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 the operator can enter data, 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.

With 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 omitted input fields. However, you must be aware of variable-length numeric fields where the operator can omit leading 0's.

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 that corresponds to the first data position in a field, not to the buffer location of the attribute character 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 8-2 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 8-2:

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.

```
TABLE      DS  0H
ENTRY1     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
ENTRY2     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
ENTRY3     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 zero; relative offsets in the target buffer are shown relative to zero.*

**Figure 8-2. Table of Requirements**

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 8-2 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.*

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 chapter.
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 equals (field starting address determined in step 4 + maximum field length value in the entry found in the table) minus 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 whether 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:

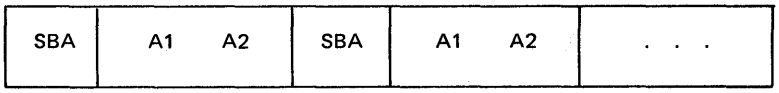
    a. Use the TRANSLATE instruction with the translation table built to convert lowercase alphabetic characters to uppercase.

    b. Use the OR instruction to place spaces in the field. This will change the DUP and FM characters. The FM appears as a semicolon (;) on the screen, but appears in the data stream as X'1E'. It will be converted to a true ; (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 * (X'5C').

**Immediate Selector-Pen Data Stream**

When a Read Modified command is executed for a display station as a result of an immediate detection by the selector pen, the resulting data stream consists of address strings that identify the fields on the screen that have the modified data tag set; no field data is transmitted in the data stream.

The data stream, ignoring the header information up to and including the AID character, appears as:




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 identifying the selected fields.

The list of indicators can be returned to the application program. A string of 1-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 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 8-3, 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

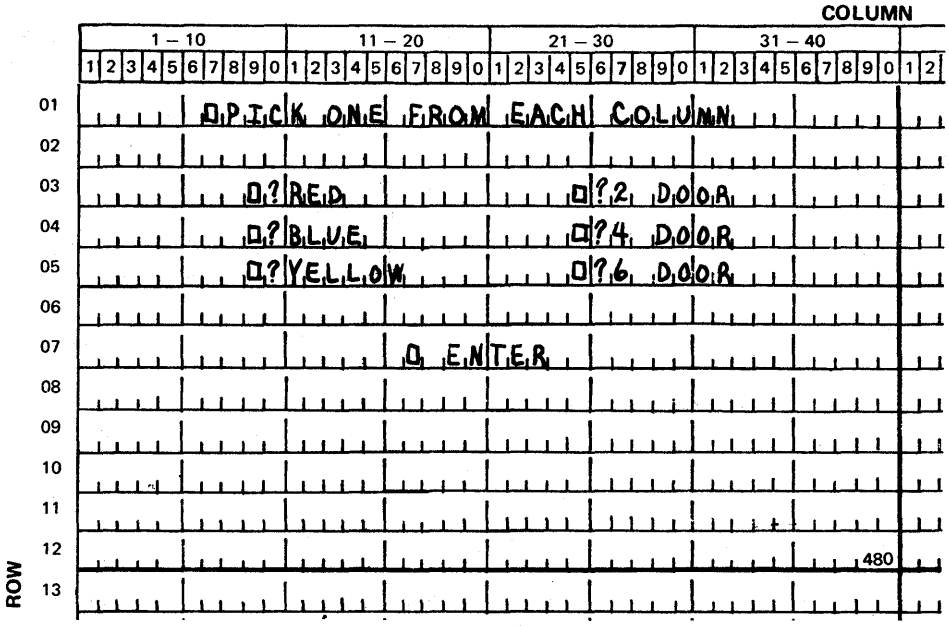



**Figure 8-3. Example of Selector-Pen Panel**

Using the sample table in Figure 8-4, the mapping routine returns a list in hexadecimal to the application program:

406F40406F406F

This list indicates that the 2nd, 5th, and 7th fields were selected. Note that the addresses of the selected fields appear in the data stream in the same sequence that 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 8-2 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 in the table correspond to the buffer position of the Selector Pen designator character in a field, not to the location of the attribute character that defines the field.*

**Figure 8-4. Sample Mapping Table**

### Mixed Read Modified Input Data Streams

When some keyed input and some delayed selector-pen detection occur in a panel during the same input operation from a display, you can use the table-driven mapping technique for non-selector-pen panels. Specify the table elements so that all delayed selector fields have a maximum length of 1 character. The mapping routine places the 1st character from the appropriate data stream field into the target field. The 1st character in a delayed selector-pen 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 whether 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 that 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. Following is a discussion of 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
```

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 calculate correctly the 3270 buffer addresses. The buffer address '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,NL EM)

generates the proper WCC, depending on the options selected in parentheses.

$CCC (PRINT,40CHAR,64CHAR,80CHAR,ALARM,ATT,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

is used with the Data Analysis feature to identify the keyboard providing 3277 Model 2 display input.

$SI generates the Suppress Index character, valid for the 3288 Model 2 printer. Other printers receive 4 (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 7-12 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:

| | | | |
|---|---|---|---|
| [symbol] | $MAC | op-type ,(row,column) ,(attributes) [,character] ,MODEL= | 1<br>2<br>3<br>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 24) and column (1 to 80) 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, and NONDISP.

Some valid values for WCC are RESET, RESTORE, ALARM, PRINT, 40CHAR, 64CHAR, 80CHAR, and NLEM.

Some valid values for CCC are PRINT, 40CHAR, 64CHAR, 80CHAR, ALARM, ATT, UNPROT, PROT, and 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 only 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.

After you have defined the macro instruction, the data stream required to create the sign-on panel shown in Figure 7-14 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)
          $MAC    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 refresh the panel completely.

### 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 the 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,ATT=(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 character, 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 Buffer:  ((R-1)X80) + (C-1)
```

If the row and column buffer addresses relative to 1 are in 2 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    R3,0
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'4C4D4E4F50D1D2D3D4D5D6D7'
         DC    X'D8D95A5B5C5D5E5F6061E2E3'
         DC    X'E4E5E6E7E8E96A6B6C6D6E6F'
         DC    X'F0F1F2F3F4F5F6F7F8F97A'
         DC    X'7B7C7D7E7F'
```

Since buffer address wrapping is dependent on screen size, application programs should not depend on buffer wrap during write operations. In addition, field attributes must be appropriately placed to delimit the end of the screen image.

### Copy Function for the 3271 and 3272

Many applications require complete and unaltered hard copy (printout) of the terminal's current screen contents for the display station operator. The printer on which the display contents are printed may support one or more display stations, depending on the 3270 configuration.

When using the copy function to obtain a printout on a 3288 Model 2 printer, remember that various print belts can be installed on the printer.

You should define a program-attention key so that a terminal operator can request hard copy on an assigned terminal printer. The screen-management program can be notified of the operator's request and perform the appropriate action.

When a data transfer to the computer occurs from pressing a program-attention key, a remote BSC 3277 or 3275 transmits AID and cursor address, and a local 3277 transfers only the AID character. The AID character identifies the key that transferred the data. No screen data is transmitted; so the program is notified of a specific request.

Once the request is identified by inspecting the AID character, the program must identify the type of unit that made the copy request. This can be done by examining the characteristics of the specific device in a terminal characteristics table that you can create. For example, depending on the type of device, the following procedures can be used to produce hard copy:

- To copy from a remote 3275 to the printer attached to the 3275, the program should send the WCC to the 3275. The WCC restores the keyboard, starts the printer, and prints 40 or 80 characters per line. Because the printer attached to the 3275 uses the same buffer as the display, all that is necessary to print the buffer (which contains the screen data) is the start-print bit in a WCC sent in a valid WRITE command sequence.

- To copy from a 3277 attached to a remote 3271 to a printer attached to the same 3271, the program should send the following data stream to the printer: STX, ESC, COPY command, CCC, from-device address, ETX. The CCC specifies start printer, the option to copy all data, and either 40 or 80 characters per line. A Model 2 display cannot be copied to a Model 1 printer, but all other copy combinations are valid. The device address following the CCC is a single-character address that identifies the device to be copied from and that is identical with the device address used to poll specifically the display requesting the copy function. The COPY command allows the buffer contents of a device attached to a 3271 to be copied to the buffer of another device attached to the same 3271, without moving the data to be copied to and from the computer. Once the prior data stream has been sent to the printer, the program should send the following data stream to the display station that requested the copy: STX, ESC, WRITE command, WCC, ETX. The WCC restores the keyboard. The operator has a positive response that the request has been honored, and the keyboard allows the operator to continue without manual intervention.
- To copy from a local 3277 to a local terminal printer, the program should execute a Read Buffer command to the display that made the copy request. The Read Buffer command is executed, and the display station transmits AID, a 2-byte cursor address, and the screen data to the computer. The program should then remove the AID character and the cursor address from the received data and, immediately preceding the remaining data, insert a WCC that specifies start printer and 40 or 80 characters per line. The altered data stream, beginning with WCC, should then be sent to the printer to copy the data. The program should then send a WCC with the restore keyboard option to the display that requested the copy function.

If the program determines that the receiving printer is busy, and the requested copy function cannot be immediately completed, one of the following actions should be taken:

3271: Notify the terminal operator of the situation, and ask the operator to wait or cancel the request.

3271 or 3272: Perform a Read Buffer to bring the screen data into the computer, where it can be queued until the printer is available, without delaying the operator.

# Appendix A. Indicators and Controls

The indicators and controls associated with each 3270 unit, except the 3287 Printer, are listed in Figure A-1 and described below (the indicators and controls associated with the 3287 Printer are described in Figure A-2):

**OFF-PUSH:** This triple-function concentric switch/control is used to control the application of power to the unit, and to control the brightness (outer knob) and contrast (inner knob) of the displayed image.

**BIT RATE:** This two-position toggle switch, added by the Dial feature, allows the 3275 Model 1 or 2 operator to select a transmission rate of 600 or 1,200 bps.

| | 3270 Unit | | | | | |
|---|---|---|---|---|---|---|
| **Indicator or Control** | **3277** | **3275** | **3272** | **3271** | **3284, 3286** | **3288** |
| OFF-PUSH (Sw, Ctl) | X | X | | | | |
| BIT RATE (Sw) | | D | | | | |
| DISCONNECT (Sw) | | D | | | | |
| INSERT MODE (Ind) | X | X | | | | |
| INPUT INHIBITED (Ind) | X | X | | | | |
| SYSTEM AVAILABLE (Ind) | X | X | | | | X |
| Sys Avl (Ind) | | | | | | X |
| SYSTEM READY (Ind) | | X | | X | | |
| SYNC SEARCH (Ind) | | X | | X | | |
| SELECTED (Ind) | | X | | X | | |
| FLAG DETECT | | S | | S | | |
| CU ACTIVE | | S | | S | | |
| OFF HOOK (Ind) | | D | | | | |
| TRANSMIT (Ind) | | X | | X | | |
| STATUS (Ind) | | X | | X | | |
| POWER ON LOCAL MODE (Sw) | | | X | | | |
| POWER OFF LOCAL MODE (Sw) | | | X | | | |
| MAIN LINE ON/OFF (Sw) | | | X | | | |
| LOC/REM (Sw) | | | X | | | |
| ON LINE/OFF LINE (Sw) | | | X | | | |
| I/O INTF DSBLD (Sw) | | | X | | | |
| POWER ON/OFF (Sw) | | | | X | X | X |
| Power On (I)/Power Off (O) (Sw) | | | | | | X |
| Carriage Restore (Pb) | | | | | | X |
| Start Test (Sw) | | | | | | X |
| VFC Selector (Sw) | | | | | | X |
| POWER ON (Ind) | | | X | | | |
| Ready (Ind) | | | | | | X |
| Ops Chk (Ind) | | | | | | X |
| Address I.D. (Label) | X | X | | | X | X |

**Key:**

Ctl – Control
Ind – Indicator
Pb – Pushbutton
Sw – Switch
X – Basic
D – Dial Feature
S – SDLC

**Note:** *3287 controls and indicators are shown in Figure A-2.*

**Figure A-1. Indicators and Controls**

**DISCONNECT:** This momentary-contact toggle switch, added to the 3275 Model 1 or 2 by the Dial feature, is used by the 3275 operator when terminating a call.

**INSERT MODE:** This indicator is turned on by the keyboard INS MODE key to show that the unit is in Insert Mode of operation. It is turned off by the keyboard RESET key.

**INPUT INHIBITED:** When lighted, this indicator shows that manual input to the unit from the keyboard, selector pen, or operator identification card reader is inhibited.

It is turned on by:

- Operation of any program-attention key.
- A selector-pen-attention operation that caused an I/O interruption to occur.
- An operator-identification-card-reader operation that caused an I/O interruption to occur.
- Turning of the Security Key Lock to the OFF position if the Security Key Lock feature is installed.
- Initiation of a printout at an unbuffered printer attached to the 3275 Display Station.
- A system-initiated I/O operation addressed to that unit.
- Operation of any alphameric key or the DUP, FIELD MARK, ERASE EOF, or DEL key when the cursor is in a protected field.
- Operation of any alphameric key not included in the numeric key grouping when the cursor is in a numeric field, without simultaneous operation of either the ALPHA or NUMERIC shift key, when the Numeric Lock special feature is installed.
- Detection of a parity or cursor check in the device buffer.

It is turned off by:

- Receipt and execution of a WCC with the keyboard-restore bit set.
- Receipt and execution of an Erase All Unprotected command.
- Turning of the Security Key Lock to the On position (if it was turned on because the Security Key Lock was in the Off position).
- Operation of the keyboard RESET key, with the following exceptions:
  - The device is selected and executing a command from the control unit.
  - The display station is in the process of reading a magnetic card from the operator identification card reader.
  - A printout is in process at the attached 3284 Printer Model 3.
  - A parity or cursor check has been detected.
- Termination of an unbuffered printer printout (if it was turned on because an unbuffered printer printout was initiated).
- Correction of a parity or cursor-check condition and resetting of the error status by a Write or Erase/Write command addressed to that device.

**SYSTEM AVAILABLE (3275 Models 1 and 2 and 3277), Sys Avl (3288):** When lighted, this indicator shows that the unit has had successful communication with the system and is available to accept an operator-initiated transmission to the system.

It is turned on by:

- Successful completion of a Write, Erase/Write, Erase All Unprotected, Copy, Read Modified, or Read Buffer command, in local or remote operation.
- On a 3275 (Models 1 and 2), receipt of an ACK from the TCU in response to an ETX at the completion of a General or Specific Poll sequence.

It is turned off by:

- Any operator-generated I/O interruption.
- A parity or cursor check and resulting I/O interruption.
- Turning of the Security Key Lock to the Off position.

**SYSTEM READY**: When lighted, this indicator shows that the Data Set carrier is on and that the TCU is online. With the Dial feature installed, this indicator lights when a transmission is first sent or received and extinguishes when a disconnect sequence is sent or received.

**SYNC SEARCH (3271 and 3275 Models 1 and 2 only)**: When lighted, this indicator shows that the unit is attempting to establish line synchronization.

**SELECTED (3271 and 3275 Models 1 and 2 only)**: When lighted, this indicator shows that the unit has been selected; that is, it is in the process of executing a command or a chain of commands.

**CU ACTIVE (3271 and 3275 Models 11 and 12 only)**: This indicator lights after selection, and remains set until the operation is completed.

**FLAG DETECT (3271 and 3275 Models 11 and 12 only)**: This indicator lights when a valid flag character (7E) is received.

**OFF HOOK/AUTO ANSWER**: This indicator replaces the SELECTED indicator when the IBM Line Adapter or external modem with Auto Answer feature is installed. When lit, it indicates that a communications link to the 3275 (Models 1 and 2) is active (that is, the data access arrangement is *off hook*). When the Auto Answer feature is not installed, the OFF HOOK/AUTO ANSWER indicator is always lit during unit operation.

**TRANSMIT**: When lighted, this indicator shows that the unit is transmitting to the TCU.

**STATUS**: When lighted, this indicator shows that an error-status condition exists within the unit.

**POWER ON LOCAL MODE**: This momentary-contact switch is used to turn on dc power for a 3272.

**POWER OFF LOCAL MODE**: This momentary-contact switch is used to turn off dc power for a 3272.

**MAIN LINE ON/OFF**: This two-position toggle switch is used to turn on and turn off ac power for the 3272.

**LOC/REM**: This two-position rotary switch on the 3272, when placed in the REM (remote) position, gives control of the power supply activation to the CPU to which the control unit is attached. When placed in the LOC (local) position, power is controlled at the 3272 by using the POWER ON LOCAL MODE and POWER OFF LOCAL MODE switches.

**ON LINE/OFF LINE**: This two-position toggle switch, when placed in the ON LINE position (the operating position),connects the 3272 to the channel interface.

**I/O INTF DSBLD**: This indicator lights when the ON LINE/OFF LINE switch on the 3272 is in the OFF LINE position.

**POWER ON**: When lighted, this indicator shows that power has been turned on for a 3272.

**POWER ON/OFF**: This two-position toggle switch is used to turn on and turn off power for 3271 control units and all printers.

**Ready**: When lighted, this indicator shows that the 3288 Line Printer is ready to receive transmissions from the contol unit. It is turned on after a successful power-on sequence, when the belt is up to speed and the printer is ready to print data.

It is turned off by:

- Open machine covers.
- Open print unit.
- Running out of forms.
- A paper motion failure (forms jam, torn forms, or missing feed holes).
- An overheated printer mechanism.
- A hardware failure requiring a repair action.

**Ops Chk**: When blinking, this indicator shows that the 3288 Line Printer not-ready condition (shown by the Ready indicator's being off) can be corrected by the operator.

It is turned on by:

- Open machine covers.
- Open print unit.
- Running out of forms.
- A paper motion failure.
- The TEST switch (on test switch panel) in other than the ON LN (On Line) position.

It is turned off when the condition that caused it to light is corrected.

**Address Identification**: Provision is made on each display station and printer to identify both the physical (hexadecimal) and symbolic addresses assigned to that unit at installation time.

**VFC Selector:** The VFC Selector switches on the 3288 Line Printer are set (00-99) by the operator to determine the number of lines skipped in a VFC operation.

**Carriage Restore:** The Carriage Restore pushbutton on the 3288 Line Printer advances the forms to a predetermined print line established by the initial forms positioning and the settings of the VFC selector switches.

**Power On/Power Off (Coded | and O):** This two-position rocker switch is used to control power to the 3288 Line Printer.

**Start Test:** This switch on the 3288 is used in conjunction with the test switches located on the test switch panel under the top cover to initiate offline test printouts.

Figure A-2 lists and explains the indicators and controls associated with the 3287 Printer.

| Indicator/Control | Explanation |
|---|---|
| Ready | This light indicates that the printer is available to print the data received from the controller. It goes off under any of the following conditions:<br>• Hold Print condition<br>• Test mode<br>• Check conditions<br>• Power off<br>• The printer runs out of paper<br>This light blinks when the Hold Print light is on to indicate that SCS data is being processed. |
| Hold Print | This light indicates that the hold-print or set-alternate condition has been entered. It remains on continuously in the hold-print condition and blinks in the set-alternate condition. |
| CU Signal | This light indicates that the 3287 is connected to a control unit and communication can take place. It goes off when the printer does not receive a signal from the control unit for 30 seconds or when the printer is in test mode. |
| 8 LPI | This light indicates that vertical line-spacing of eight lines per inch is being performed by the printer. If 6 LPI is selected with the Change LPI switch, and the control unit specifies 8 LPI, this light comes on only when printing is performed. The light shows the setting of the Change LPI switch when the printer is in the hold-print condition. |
| Check | This light indicates the detection of an error condition by the 3287. It goes off when all check conditions have been removed. The type of check condition is displayed in the Status indicator when the Check light comes on. |
| Double Space | This light indicates that double line-spacing is being performed by the printer. If single space is selected by the Change Space switch and the control unit specifies double space, this light comes on only when printing is performed. When the printer is in the hold-print condition, this light shows the setting of the Change Space switch. |
| Test | This light indicates that the automatic built-in tests are running in the 3287. It goes off at the error-free ending of all the tests. |

**Figure A-2 (Part 1 of 4). Indicators and Controls for 3287 Printer**

| Indicator/Control | Explanation |
|---|---|
| Dual Case | This light indicates that dual-case printing is being performed by the printer. If mono case (uppercase only) is selected by the Change Case switch and the control unit specifies dual case (both uppercase and lowercase), this light is on only when printing is being performed. When the printer is in the hold-print condition, this light shows the setting of the Change Case switch. |
| Status | The Status indicator displays a two-digit code that represents the current status of the 3287, such as:<br>• A check condition<br>• An end-of-forms condition<br>• Printer Status Information<br>• The result of a test operation in which an error has been detected.<br><br>The *IBM 3287 Printer Problem Determination Guide*, GA27-3151, contains a list of all the error codes and the actions the operator is to take when a code appears. |
| Hold Print/Enable Print | Pressing this switch to the Hold Print position causes the 3287 to stop printing after it has completed the function in process. The print head moves to the leftmost position, the Hold Print light comes on, the Ready light goes off, and data is held in the printer buffer for additional printing.<br><br>The Set Alternate, Change LPI, Change Space, Change Case, Form Feed, Setup, Index, Cancel Print, Buffer Reprint, PA1, and PA2 switches are operational only when the printer is in the hold-print condition.<br><br>Selecting Enable Print causes the Hold Print light to go off and the Ready light to come on. Printing then continues, following the preceding print position.<br><br>Pressing the Hold Print switch on and off within 10 minutes does not have any effect on communication with the control unit.<br><br>If the operator leaves the printer in the hold-print condition for more than 10 minutes, an *Intervention Required* message is sent to the control unit. The operator must then press the Enable Print switch to return to normal operation. |
| Change LPI | This switch is used to select vertical line-spacing between lines. When either 6 or 8 LPI is selected by the switch, the LPI selection by the host or the control unit supersedes the switch selection.<br><br>If printing is being done in the 8 LPI format, or if the 8 LPI switch is pressed while the 3287 is not printing data, the 8 LPI light comes on. When a power-on reset is performed, the printer is initialized to the 6 LPI condition (the 8 LPI light is off). Reset has no effect on the switch setting. When the printer is operating in the SCS mode, it is initialized to the current switch setting.<br><br>Note: *If the platen has been moved by hand, line-spacing from the first to the second print line may be out of specification since the platen does not have mechanical indexing, but all lines printed after the second line will be in specification. Care should be taken, therefore, when the platen is adjusted by hand to align first print line. Maladjustment can cause the first and second print lines to touch when the 8 LPI format is selected.* |

**Figure A-2 (Part 2 of 4). Indicators and Controls for 3287 Printer**

| Indicator/Control | Explanation |
|---|---|
| Set Alternate/Set Parameter/Reset Alternate Switches | Pressing the Set Alternate switch when the printer is in the hold-print condition activates the alternate function for all the operator panel switches and causes the Hold print light to flash on and off.<br><br>An operator can enter the maximum print position (MPP), using the hundreds, tens, and units alternate function switches, when the 3287 is in the alternate function mode of operation. Each time the Tens or Units switch is pressed, the Status indicator is incremented by 10 or 1, respectively. Pressing the Hundreds switch causes the Status indicator to flash for a 1XX selection and to remain on, continuously, for a 0XX selection. Once the MPP has been entered, pressing the Set Parameter switch causes the MPP selection to be saved for future use and to enter a hold-print condition.<br><br>Pressing the Reset Alternate switch before pressing Set Parameter causes the printer to return to the primary functions of the switches in the hold-print condition without storing a newly set MPP value (the Hold Print light is on continuously). The MPP is initialized to 132 when a power-on reset is performed. Reset and test modes have no effect on the MPP selection. The MPP selection is valid only when processing information data. |
| Change Space | This switch, when set to Double Space, causes the printer to perform double line-spacing during printing. When a power-on reset is performed, the printer is initialized to a single space condition (the Double Space light is off). Reset mode and test mode have no effect on the switch setting. |
| Change Case | Selecting mono case with this switch causes the printer to print in uppercase characters only. Selecting dual case causes the printer to print in dual case (both uppercase and lowercase characters). The Dual Case light comes on for dual case printing. During a power-on reset, the printer is initialized to a mono case condition (the Dual Case light is off). Reset mode and test mode have no effect on the switch setting. |
| PA1 and PA2 | These switches are operational only when the SCS Support feature is installed. Pressing either switch causes the printer to send a control code to the control unit and to display a function code in the Status indicator. The control unit and the printer communicate with each other and perform the operation the host program has defined for the PA1 and PA2 switches. When this is completed, the Status indicator light goes off. These switches are active only when the Hold Print light is on and the printer is operating in SCS mode. |
| Form Feed | This switch is operational only if the Page Length Control feature or the SCS Support feature is installed, and it is active only in the hold-print condition. The page size is defined by the operator using the Selector switches or by the host program in SCS mode. The page size defined by the host program supersedes that defined by the Selector switches.<br><br>Pressing this switch causes the printer to advance the forms until the first print line of the next page is reached, if the forms have been properly aligned and its page size has been properly defined. |
| Buffer Reprint | Not operational when attached to a 3271 or 3272. |
| Setup | This switch is used for forms alignment and can be activated only when the 3287 printer is in the hold-print condition. Pressing this switch causes the printer to print *H* characters continuously until the MPP is reached. The print head then returns to print position 1 without movement of the forms. When operating in SCS mode, the print head returns to the maximum print position. |

**Figure A-2 (Part 3 of 4). Indicators and Controls for 3287 Printer**

| Indicator/Control | Explanation |
|---|---|
| Index | Pressing the Index switch causes the printer to advance forms continuously. |
| Reset | This switch is used to reset a check condition and to turn off any error indications. The printer indexes one line and printing continues if allowed by the control unit. |
| Cancel Print | This switch is operational only if the SCS Support feature is installed. Pressing this switch when the Hold Print light is on causes the printer to stop printing, to display a *cancel selected* code in the Status indicator, and to send a code for canceling the print operation to the control unit if the printer was processing an SCS message.<br><br>If the printer was not printing SCS data, pressing the Cancel Print switch causes an *operator check* code to be displayed in the Status indicator. |
| Test | Pressing this switch causes the printer to enter test mode. When the 3287 is in test mode, it cannot communicate with the control unit. |
| Power (I/O) | The power switch controls power to the 3287. The I position is the *on* position, and the O is the *off* position. |
| Selector | The Selector switches are 2-digit, 10-position switches located on the operator's panel, used to specify the number of lines that can be printed on a form, from 00 through 99.<br><br>Forms feeding is performed when the Form Feed switch is pressed or a forms-feed control code is received in the data for the number of lines specified. The page-length value is read from the Selector switches during a power-on reset or when the Forms Feed switch is pressed while the 3287 is in the hold-print condition. The Page Length Control feature must be installed for these switches to be operational. These switches are not operable for SCS print operations. |
| Set Function | Reserved for future use. |

**Figure A-2 (Part 4 of 4). Indicators and Controls for 3287 Printer**

# Appendix B. Buffer Address I/O Interface Codes

| 40 Col | | 80 Col | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | | EBCDIC | | ASCII | |
| R | C | R | C | Dec | Hex | | | | |
| 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 | | 80 Col | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|---|---|
| R | C | R | C | Dec | 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 | | 80 Col | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|---|---|
| R | C | R | C | Dec | 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 | C4 | 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 | | 80 Col | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|---|---|
| R | C | R | C | Dec | 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 | 1[illegible] | 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 | | 80 Col | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|---|---|
| R | C | R | C | Dec | 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 | 2E |
| 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 | | 80 Col | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|---|---|
| R | C | R | C | Dec | 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 | | 80 Col | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|---|---|
| R | C | R | C | Dec | 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| | | | | EBCDIC | | ASCII | |
| R | C | Dec | Hex | | | | |
| 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| | | | | EBCDIC | | ASCII | |
| R | C | Dec | Hex | | | | |
| 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| R | C | Dec | 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| R | C | Dec | 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| R | C | Dec | Hex | EBCDIC | | 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 R | 80 Col C | Position Dec | Position Hex | Buffer Address (Hex) EBCDIC | Buffer Address (Hex) 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| R | C | Dec | 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| | | | | EBCDIC | | ASCII | |
| R | C | Dec | Hex | | | | |
| 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| R | C | Dec | 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| R | C | Dec | 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) | | | |
|---|---|---|---|---|---|---|---|
| | | | | EBCDIC | | ASCII | |
| R | C | Dec | Hex | | | | |
| 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| | | | | EBCDIC | | ASCII | |
| R | C | Dec | Hex | | | | |
| 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| R | C | Dec | 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 | F4 | 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| | | | | EBCDIC | | ASCII | |
| R | C | Dec | Hex | | | | |
| 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| R | C | Dec | 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 | | Position | | Buffer Address (Hex) | | | |
|---|---|---|---|---|---|---|---|
| | | | | EBCDIC | | ASCII | |
| R | C | Dec | Hex | | | | |
| 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 |

# Appendix C. Katakana Feature

This appendix contains Katakana unique information interface codes and the keyboard shift operations.

## Interface Codes

Figure C-1 show the Japanese Katakana EBCDIC interface codes. It corresponds to Figure 2-5 for U.S. codes.

| Bits 4567 | Hex 1 | 00 | 01 | 10 | 11 | 00 | 01 | 10 | 11 | 00 | 01 | 10 | 11 | 00 | 01 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| 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 | | | 。 | エ | / | | ア | タ | | | A | J | | 1 |
| 0010 | 2 | | EUA | | | 「 | オ | | | イ | チ | ヘ | | B | K | S | 2 |
| 0011 | 3 | | IC | | | 」 | ヤ | | | ウ | ツ | ホ | | C | L | T | 3 |
| 0100 | 4 | | | | | 、 | ユ | | | エ | テ | マ | | D | M | U | 4 |
| 0101 | 5 | PT | NL | | | ・ | ヨ | | | オ | ト | ミ | | E | N | V | 5 |
| 0110 | 6 | | | | | ヲ | ツ | | | カ | ナ | ム | | F | O | W | 6 |
| 0111 | 7 | | | | | ァ | | | | キ | ニ | メ | | G | P | X | 7 |
| 1000 | 8 | | | | | 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 | | DUP | | RA | < | * | % | @ | サ | | ヨ | ワ | | | | |
| 1101 | D | | SF | | | ( | ) | _ | ' | シ | ハ | ラ | ン | | | | |
| 1110 | E | | FM | | | + | ; | > | = | ス | ヒ | リ | ゛ | | | | |
| 1111 | F | | | | SUB | \| | ¬ | ? | | 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 not specified. The character displayed by the 3277 or 3275 for a given undefined character code may be different for other devices. IBM reserves the right to change at any time the character displayed for an undefined character code.*
2. *Hex codes 4A, 5A, 6A, and 7F are used for CU addressing, device addressing, buffer addressing, and control purposes (for example, WCC and CCC), but have no associated graphic characters.*
3. *The DUP and FM control characters are displayed or printed as * and; respectively.*
4. *NL and EM are stored in the buffer in two character locations. The Katakana hardware expands the NL and EM characters received from the program to the required 2-byte sequence. It also contracts the 2-byte buffer sequence to the single-byte EBCDIC NL or EM code on a subsequent read operation.*

   *NL and EM display or print as blank 5 and blank 9 respectively, except for a printer not operating under format control, which executes NL and EM and prints blank blank.*
5. *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 2-7.*
6. *For BSC data-link control characters, see Chapter 5.*

**Figure C-1. Japanese Katakana EBCDIC I/O Interface Code**

## Keyboard Shift Operations: LATIN SHIFT and KANA SHIFT Keys

The shift operations of the Katakana keyboards are different from those of the other EBCDIC keyboards described in Chapter 2. The following paragraphs discuss the unique keys and operations.

To place the keyboard in the lower shift of either Latin or Katakana (Kana) mode, press and release the desired mode shift key. This enables the characters on the lower portion of each character key to be generated. Holding the shift key depressed while operating the character keys causes the upper-shift characters of the selected mode to be generated.

In addition, a single depression of the Lock key locks the keyboard in the upper shift of the selected mode. A second depression of the Lock key returns the keyboard to the lower shift of the selected mode.

With two exceptions, once a mode is selected, the keyboard remains in that mode until the operator changes the mode by operating the Alternate Shift key. The exceptions are as follows:

- When power is initially applied, the keyboard is automatically placed in Latin mode.
- (Data entry keyboards only) – When the cursor enters a numeric field, the data entry keyboard is automatically placed in upper-shift Latin mode. Only 0–9, minus (-), decimal sign, and DUP may be entered when in this mode.

While the cursor remains in the numeric field, the upper-shift Latin mode can be overridden, one character at a time, by pressing the appropriate shift key as follows:

Upper-shift Kana mode – While holding the KANA SHIFT key depressed, press the selected character key.

Lower-shift Kana mode – Press and release the KANA SHIFT key; then press the selected character key.

Upper-shift Latin mode – While holding the LATIN SHIFT key depressed, press the selected character key. This permits keying in uppper-shift Latin mode characters other than 0–9, minus (-), decimal sign, and DUP.

Lower-shift Latin mode – Press and release the LATIN SHIFT key; then press the selected character key.

In all cases, when the selected character has been entered and the key (or keys) released, the keyboard returns to upper-shift Latin mode.

When the cursor leaves the numeric field, the keyboard returns to lower shift of the most recent Latin or Kana mode used by the operator. This is independent of whether the last mode was caused by an override by the operator or the mode being used just prior to entry of the cursor into the numeric field.

# Appendix D. Data Analysis – APL Feature

Figure D-1 shows the Data Analysis – APL feature and associated features; Figure D-2 shows the interface codes.

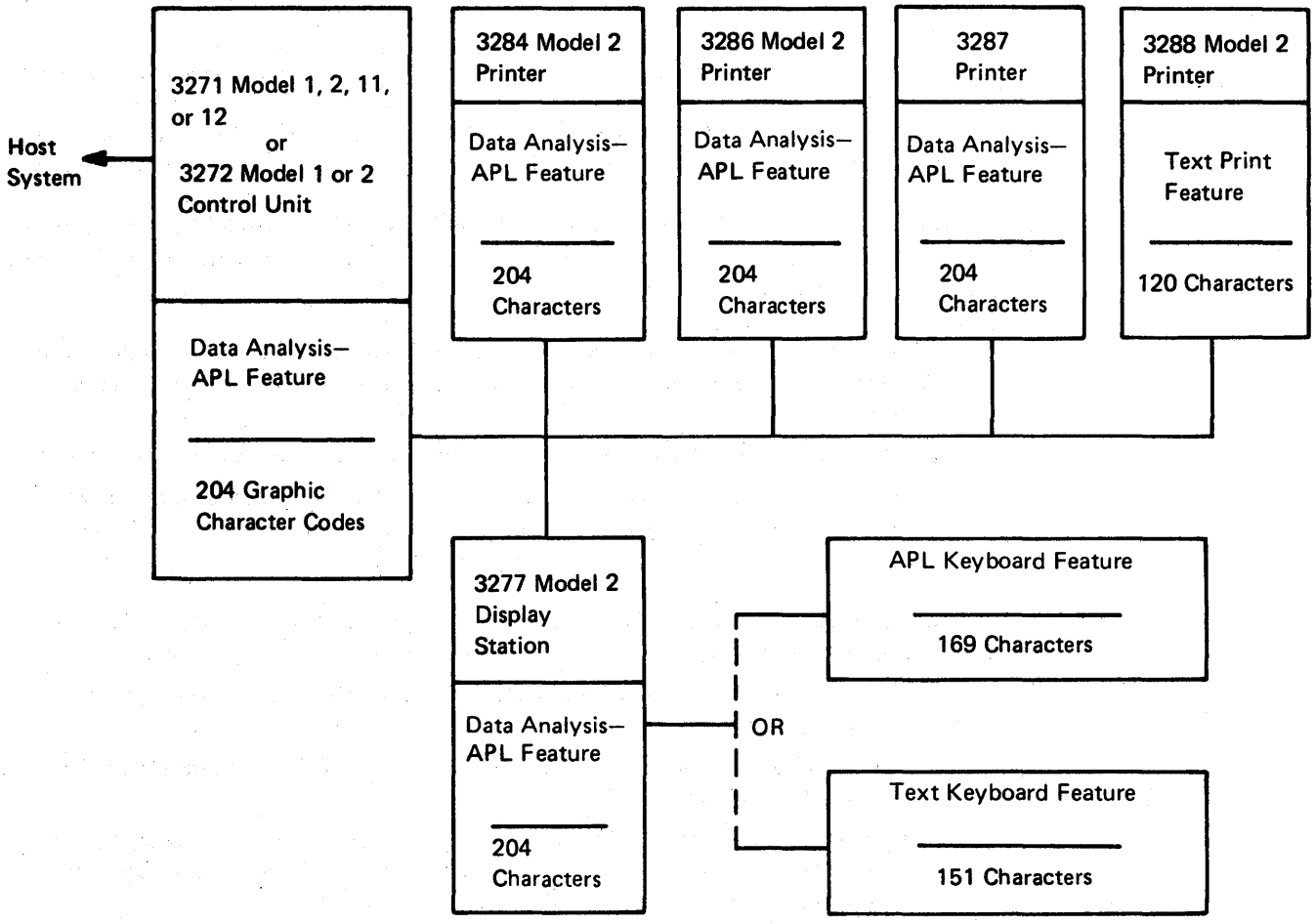



**Figure D-1. Data Analysis – APL Feature and Associated Features**

| 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 | | | | | SP | & | - | | | □ | ¯ | α | | | | 0 | |
| 0001 | 1 | | | | | | | / | | a | j | | ϵ | A | J | | 1 | |
| 0010 | 2 | | | | | | | | | b | k | s | ι | B | K | S | 2 | |
| 0011 | 3 | | | | | | | | | c | l | t | ρ | C | L | T | 3 | |
| 0100 | 4 | | | | | | | | | d | m | u | ω | D | M | U | 4 | |
| 0101 | 5 | | | | | | | | | 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 | | | | | | | | | h | q | y | ÷ | H | Q | Y | 8 | |
| 1001 | 9 | | | | | | | | | i | r | z | | I | R | Z | 9 | |
| 1010 | A | | | | | ¢ | ! | | : | ↑ | ⊃ | ∩ | ∇ | | | | | |
| 1011 | B | | | | | . | $ | , | # | | ⊂ | ∪ | Δ | | | | | |
| 1100 | C | | | | | < | * | % | @ | ≤ | | ⊥ | ⊤ | | | | | |
| 1101 | D | | | | | ( | ) | _ | ' | ⌈ | ○ | [ | ] | | | | | |
| 1110 | E | | | | | + | ; | > | = | ⌊ | | ≥ | ≠ | | | | | |
| 1111 | F | | | | | \| | ¬ | ? | " | → | ← | ∘ | \| | | | | | |

**Notes:**

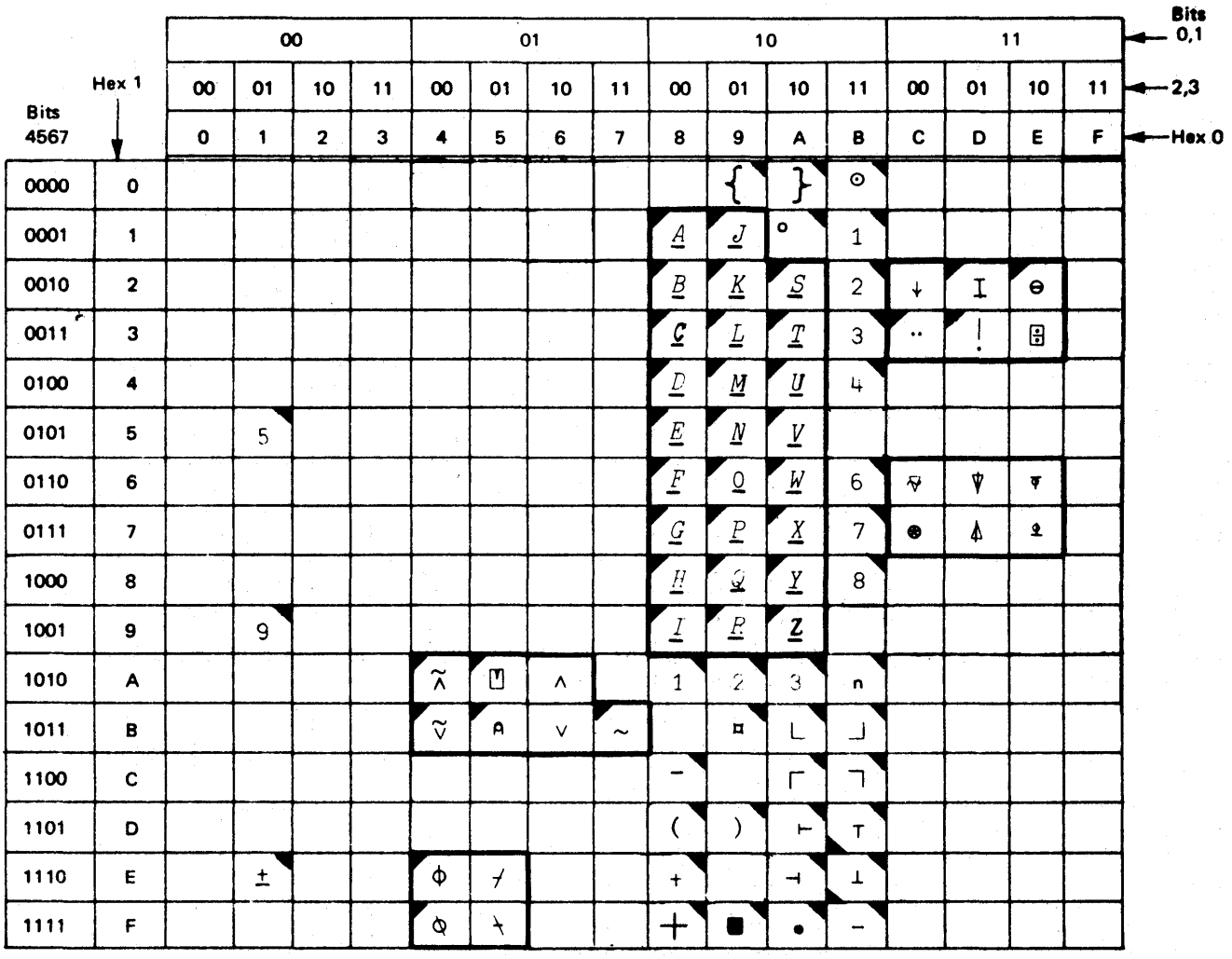
1. *NL, EM, DUP, and FM control characters are displayed or printed as 5, 9, *, and; characters, respectively, except by the printer under format control, in which case NL and EM do not result in a character's being printed.*
2. *The 89-character dual-case EBCDIC character set is shown within the bold outlines. All codes shown can be directly entered from the APL keyboard.*

**Legend:**

Codes that cannot be entered from the text keyboard.

Codes transmitted are unique to the APL keyboard. (See Part 2.)

Figure D-2 (Part 1 of 2). Data Analysis – APL Interface Codes




**Notes:**

1. *These codes, preceded by a hex 1D control character, transmit the graphics shown.*
2. *Codes B5, B9, and 9E or codes 15, 19, and 1E can be used in program-to-terminal messages — characters 5, 9, and ±.*

**Legend:**

Codes that cannot be entered from the text keyboard.

Codes that are not directly entered from the APL keyboard (APL characters are shown within the bold outline).

Codes transmitted are unique to the text keyboard. (See Part 1.)

**Figure D-2 (Part 2 of 2). Data Analysis – APL Interface Codes**

## APL Keyboard Special Feature Operation (3277 Display Station Model 2)

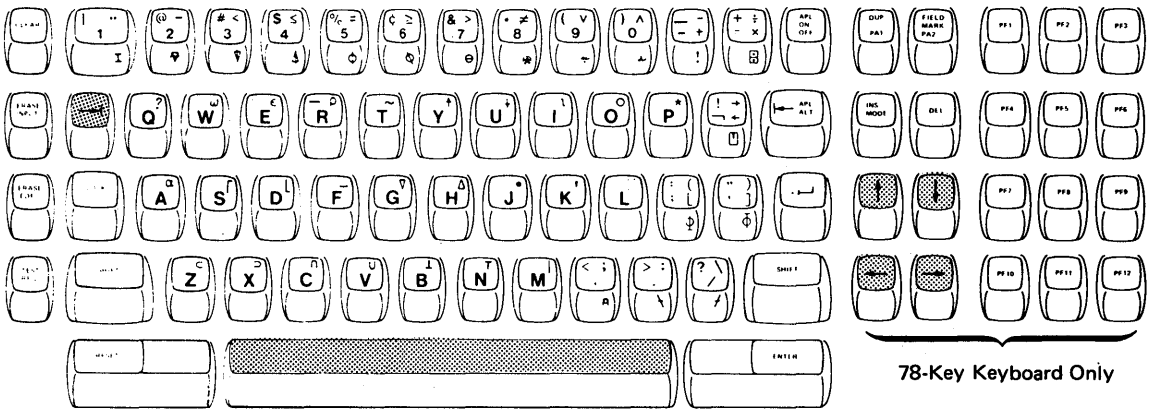
The APL keyboard (Figure D-3) allows the direct entry and display of the 133-character APL character set (Figure D-4). [For the Technical Notation (TN) character set, see Figure D-5.] In addition, this keyboard changes two standard typewriter keys: Backspace key to APL ON/OFF key, and Backtab key to Backtab/APL ALT key.

### *APL ON/OFF Key*

At initial power-on, the keyboard operates as a typewriter keyboard. Pressing the APL ON/OFF key once invokes the APL keyboard graphics. When the APL ON/OFF key is pressed once more, the keyboard reverts to the U.S. EBCDIC character set and standard operation. Keyboard status may be determined by pressing an APL character key; it will cause that character to be displayed only if APL is on. If the standard keyboard character is displayed, APL is off.

### *APL ALT Key*

In APL-ON status, the APL ALT key serves as an additional shift key. Holding it down while pressing a key that has a compound APL symbol on the front produces the corresponding character and output code. Holding down APL ALT while pressing an alphabetic character key produces an underscored uppercase character. In APL-OFF status, the APL ALT key retains its original backtab function.




Legend:

Typamatic Keys

**Figure D-3. APL Keyboard**

| APL Off | | APL On | | | | |
|---|---|---|---|---|---|---|
| Lowercase Shift | Uppercase Shift | Lowercase Shift | Uppercase Shift | | | APL On with APL ALT |
| a | A | A | α | (alpha) | | A |
| b | B | B | ⊥ | (base) | | B |
| c | C | C | ∩ | (cap) | | C |
| d | D | D | ⌊ | (downstile) | | D |
| e | E | E | ∈ | (epsilon) | | E |
| f | F | F | _ | (underbar) | | F |
| g | G | G | ∇ | (del) | | G |
| h | H | H | Δ | (delta) | | H |
| i | I | I | ⍳ | (iota) | | I |
| j | J | J | ∘ | (null) | | J |
| k | K | K | ' | (quote) | | K |
| l | L | L | ⍞ | (quad) | | L |
| m | M | M | \| | (stile) | | M |
| n | N | N | ⊤ | (top) | | N |
| o | O | O | ○ | (circle) | | O |
| p | P | P | * | (star) | | P |
| q | Q | Q | ? | (query) | | Q |
| r | R | R | ⍴ | (rho) | | R |
| s | S | S | ⌈ | (upstile) | | S |
| t | T | T | ~ | (tilde) | | T |
| u | U | U | ↓ | (down) | | U |
| v | V | V | ∪ | (cup) | | V |
| w | W | W | ⍵ | (omega) | | W |
| x | X | X | ⊃ | (close shoe) | | X |
| y | Y | Y | ↑ | (up) | | Y |
| z | Z | Z | ⊂ | (open shoe) | | Z |
| 1 | ! | 1 | ¨ | (dieresis) | | I (I-beam) |
| 2 | @ | 2 | ¯ | (overbar) | | ⍒ (del tilde) |
| 3 | # | 3 | < | (less) | | ⍖ (del stile) |
| 4 | $ | 4 | ≤ | (not greater) | | ⍙ (delta stile) |
| 5 | % | 5 | = | (equal) | | ⌽ (circle stile) |
| 6 | ¢ | 6 | ≥ | (not less) | | ⍉ (circle slope) |
| 7 | & | 7 | > | (greater) | | ⊖ (circle bar) |
| 8 | * (asterisk) | 8 | ≠ | (not equal) | | ⍟ (log) |
| 9 | ( | 9 | ∨ | (OR) | | ⍱ (NOR) |
| 0 | ) | 0 | ∧ | (AND) | | ⍲ (NAND) |
| - | _ | + | − | (bar) | | ! (quote dot) |
| = | + | X | ÷ | (divide) | | ⌹ (domino) |
| ¬ | ! | ← | → | (right) | | ⍞ (quote quad) |
| ; | : | [ | ( | (open paren) | | ⍬ (base null) |
| ' | " | ] | ) | (close paren) | | ⍤ (top null) |
| , | < | , | ; | (semicolon) | | ⍝ (cap null) |
| . | > | . | : | (colon) | | ⍀ (slope bar) |
| / | ? | / | \ | (slope) | | ⌿ (slant bar) |

**Figure D-4. APL Keyboard Feature Character Set**

| TN Character | Keyboard Substitute (All with APL On) | TN Character | Keyboard Substitute (All with APL On) |
|---|---|---|---|
| Subscript 1 | ⌊1 (APL downstile + digit) | ° (degree) | ○ (APL null) |
| Subscript 2 | ⌊2 (APL downstile + digit) | ± (plus or minus) | ⍬ (APL base null) |
| Subscript 3 | ⌊3 (APL downstile + digit) | ( (left brace) | ∇ (APL del) |
| Subscript n | ⌊N (APL downstile + char.) | ) (right brace) | Δ (APL delta) |
| Superscript 0 | ⌈0 (APL upstile + digit) | ¤ (lozenge) | ⍞ (APL quad) |
| Superscript 1 | ⌈1 (APL upstile + digit) | ■ (histogram) | ⍰ (APL domino) |
| Superscript 2 | ⌈2 (APL upstile + digit) | • (bullet) | ⍟ (APL log) |
| Superscript 3 | ⌈3 (APL upstile + digit) | ┐ (upper right corner) | ⊃ (APL close shoe) |
| Superscript 4 | ⌈4 (APL upstile + digit) | ┌ (upper left corner) | ∩ (APL cap) |
| Superscript 5 | ⌈5 (APL upstile + digit) | └ (lower left corner) | ⊂ (APL open shoe) |
| Superscript 6 | ⌈6 (APL upstile + digit) | ┘ (lower right corner) | ∪ (APL cup) |
| Superscript 7 | ⌈7 (APL upstile + digit) | ┬ (top junction) | ⊤ (APL top) |
| Superscript 8 | ⌈8 (APL upstile + digit) | ├ (left junction) | ⌿ (APL slant bar) |
| Superscript 9 | ⌈9 (APL upstile + digit) | ┴ (bottom junction) | ⊥ (APL base) |
| Superscript ( | ⌈( (APL upstile + char.) | ┤ (right junction) | ⍀ (APL slope bar) |
| Superscript + | ⌈+ (APL upstile + char.) | + (DA cross) | ÷ (APL divide) |
| Superscript ) | ⌈) (APL upstile + char.) | - (extended dash) | ¯ (APL overbar) |
| Superscript - | ⌈- (APL upstile + char.) | | |

**Notes:**

1. The 3270 with the Data Analysis – APL feature provides the capability of screen display and printer output of the TN character set and certain other special characters not shown in Figure D-3. In addition, most of the TN characters may be directly entered from the 3270 APL keyboard. Characters that compose the total TN character set are shown above. along with the means of directly entering each character or a recommended substitution of one or two characters to be used for each character.
2. The following subset of TN characters may be directly entered on the keyboard (see Figure D-3 of this appendix for proper setting of shift or APL ON/OFF keys).

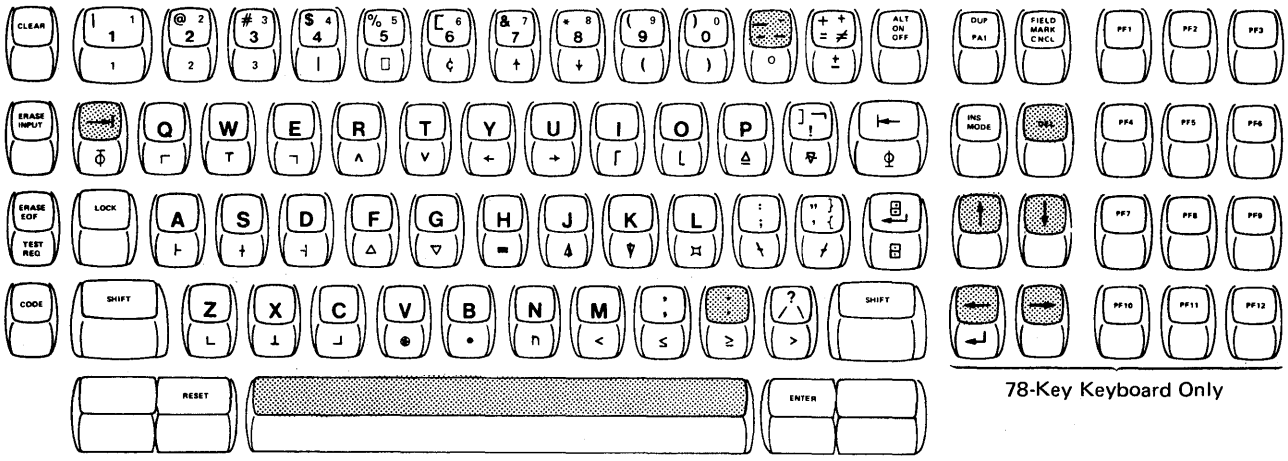
| | |
|---|---|
| space | 1 |
| A–Z (uppercase) | 26 |
| a–z (lowercase) | 26 |
| 0–9 (numeric) | 10 |
| ↓ ↑ ≤ ≥ ≠ [ ] ← → \ \| (APL special characters) | 11 |
| - = ¬ ; ' , . / $ % c & * ( ) – + ] : " < > ? (non-APL special characters) | 23 |

3. The subset of 35 TN characters, shown above, though they may not be directly entered, may be entered through use of the recommended substitution of APL characters not used in Note 2. Note that a user's application program would have the responsibility of decoding these substitutions. (Refer to Figure D-2, Data Analysis – APL Interface Codes.)

**Figure D-5. APL Keyboard TN Character Availability**

## Text Keyboard Special Feature Operation (3277 Display Station Model 2)

The text keyboard (Figure D-6) contains 78 keys that permit direct entry and display of the 151-character text keyboard character set (Figure D-7) when the appropriate shift is used. This keyboard also contains a shift indicator light to simplify operator control over shift modes. The text keyboard has changes to six normal typewriter keyboard control keys: RESET and ENTER keys have been repositioned to reduce confusion with the uppercase/lowercase SHIFT keys, the Backspace key is the ALT ON/OFF key, the TEST REQ key is the CODE key, the ERASE EOF key is the ERASE EOF/TEST REQ key, and the New Line key is both a character and function key.




Legend:

Typamatic keys

**Figure D-6. Text Keyboard**

### *ALT ON/OFF Key*

At initial power-on, alternate mode is inactive and the ALT indicator (above the ALT ON/OFF key) is off. This allows the dual-case EBCDIC character set to be entered. (See Alternate Mode in Figure D-7.) Pressing the ALT ON/OFF key once turns on alternate mode and the ALT indicator and allows the characters in the center column of Figure D-7 to be entered and displayed. When the ALT ON/OFF key is pressed again, the ALT indicator is turned off and alternate mode becomes inactive.

When alternate mode is inactive and the keyboard is in lowercase (SHIFT key inactive) shift, the character in the lower left or the lowercase of the character in the center of the appropriate keytop can be entered. When alternate mode is inactive and the keyboard is in uppercase shift, the character in the upper left or the uppercase of the character in the center of the appropriate keytop can be entered.

When alternate mode is active and the keyboard is in lowercase (SHIFT key inactive), the character in the lower right or the lowercase of the character in the center of the appropriate keytop can be entered. When alternate mode is active and the keyboard is in uppercase, the character in the upper right or the uppercase of the character in the center of the appropriate keytop can be entered.

### *CODE Key*

When the CODE key is held down, code shift is active in both ALT ON and ALT OFF conditions. Code shift active allows the character on the front face of each key to be entered and displayed. (See the column on the right side of Figure D-7.) Code shift becomes inactive when the CODE key is released.

### *ERASE EOF/TEST REQ Key*

When this key is pressed and the code shift is inactive, the erase EOF function is inputted. When this key is pressed and the code shift is active, the test request function is inputted. The ERASE EOF and TEST REQ keys are described in Chapter 2 under "Keyboard Operations."

| Standard Mode | | | Alternate Mode | | | | | |
|---|---|---|---|---|---|---|---|---|
| Lowercase Shift | | Uppercase Shift | Lowercase Shift | | Uppercase Shift | | | Code Mode |
| a | | A | a | | A | | | ⊢ (left junction) |
| b | | B | b | | B | | | • (bullet) |
| c | | C | c | | C | | | ⌋ (lower right corner) |
| d | | D | d | | D | | | ⊣ (right junction) |
| e | | E | e | | E | | | ⌉ (upper right corner) |
| f | | F | f | | F | | | Δ (delta) |
| g | | G | g | | G | | | ∇ (del) |
| h | | H | h | | H | | | ■ (histogram) |
| i | | I | i | | I | | | ⌈ (upstile) |
| j | | J | j | | J | | | ⍙ (delta stile) |
| k | | K | k | | K | | | ⍛ (del stile) |
| l | | L | l | | L | | | ⌑ (lozenge) |
| m | | M | m | | M | | | < (less) |
| n | | N | n | | N | | | n (subscript) |
| o | | O | o | | O | | | ⌊ (downstile) |
| p | | P | p | | P | | | ⍙ (delta underscore) |
| q | | Q | q | | Q | | | ⌜ (upper left corner) |
| r | | R | r | | R | | | ∧ (AND) |
| s | | S | s | | S | | | ⍭ (DA cross) |
| t | | T | t | | T | | | v (OR) |
| u | | U | u | | U | | | → (right) |
| v | | V | v | | V | | | ⊛ (log) |
| w | | W | w | | W | | | ⊤ (top junction) |
| x | | X | x | | X | | | ⊥ (bottom junction) |
| y | | Y | y | | Y | | | ← (left) |
| z | | Z | z | | Z | | | ⌞ (lower left corner) |
| 1 | | ! | 1 | | 1 | | subscript | 1 |
| 2 | | @ | 2 | | 2 | | subscript | 2 |
| 3 | | # | 3 | | 3 | | subscript | 3 |
| 4 | | $ | 4 | | 4 | | | \| (stile) |
| 5 | | % | 5 | super- | 5 | | | ⍞ (quad) |
| 6 | | [ | 6 | script | 6 | | | ⍋ |
| 7 | | & | 7 | | 7 | | | ↑ (up) |
| 8 | | * | 8 | | 8 | | | ↓ (down) |
| 9 | | ( | 9 | | 9 | | superscript | ( |
| 0 | | ) | 0 | | 0 | | superscript | ) |
| - | (hyphen) | _ | – | (extended | - | (super- | | ° |
| = | | + | ≠ | dash) | + | script) | | ± |
| ! | | ] | ! | | ¬ | | | ⍒ (del tilde) |
| ; | | : | ; | | : | | | ⍳ |
| ' | | " | ( | | ) | | | ⍷ |
| , | | < | , | | < | | | ≤ |
| . | | > | . | | > | | | ≥ |
| / | | ? | \ | | ? | | | > |
| ⍞ | (domino) | ⍞ | ⍞ | | ⍞ | | | ⍤ (top null) |
| | | | | | | | | ⍦ (base null) |

**Figure D-7. Text Keyboard Feature Character Set**

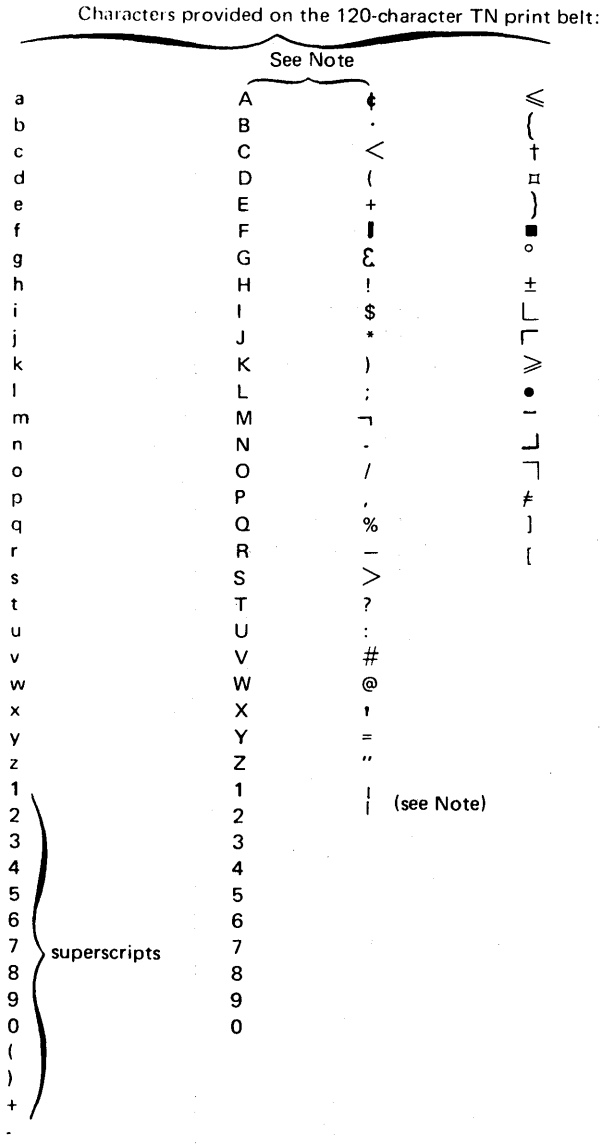
## *New Line Key*

Pressing the New Line key (with alternate mode either ON or OFF) causes the new line character, :, to display and the new line function (index and cursor return to left margin) to occur. In code shift, the New Line key only enters the domino; the new line function is provided by the cursor left key in code shift only. The New Line key is not typamatic on the text keyboard.

## *Tab and Backtab Functions*

The Tab and Backtab keys on the text keyboard operate in the same way as those on the typewriter keyboard. (See "Keyboard Operations" in Chapter 2.) In code shift, however, the functions are not performed; instead, the tab (Φ) and backtab (Φ) symbols are entered and displayed for text application programs to format printed output. The Tab key is not typamatic in code shift.

Figures D-8, D-9, and D-10 note the characters available for printing on the 3288 under various print modes.




**Note:** *The characters listed in the second and third columns constitute the character set for the (optional) 64-character EBCDIC print belt. The broken vertical bar ( ¦ ) is not on the 120-character TN print belt and cannot be entered from the text keyboard.*

**Figure D-8. Text Print Character Set for 3288 Printer Model 2**

| 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 | | | | | SP | & | – | | | 5 | − | 0 | | | | 0 | |
| 0001 | 1 | | | | | | | / | | a | j | 0 | 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 | | 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 | | I | R | Z | 9 | |
| 1010 | A | | | | | ¢ | ! | ¦ 9 | : | ≤ | ¤ | ± | – | | | | | |
| 1011 | B | | | | | . | $ | , | # | ( | ) | ⌊ | ⌋ | | | | | |
| 1100 | C | | | | | < | * | % | @ | | | ⌜ | ⌝ | | | | | |
| 1101 | D | | | | | ( | ) | _ | ' | ( | ) | [ | ] | | | | | |
| 1110 | E | | | | | + | ; | > | = | + | | ≥ | ≠ | | | | | |
| 1111 | F | | | | | ¦ | ¬ | ? | " | ┼ | ■ | • | SI | | | | | |

**Notes:**

1. Only those data characters shown within the bold outlines can be printed by the 3288 printer with the Text Print feature installed, using the 64-character EBCDIC print belt.
2. NL (hex 15), EM (hex 19), DUP (hex 1C), FM (hex 1E), and NUL (hex 00), and SI control characters are printed as 5, 9, *, ; and space characters, respectively, except when line length format is not specified, in which case NL and EM do not result in a character's being printed.
3. Hex 6A, superscript 9 shown above, causes a broken vertical bar ( ¦ ) to be printed when using the 64-character EBCDIC print belt.
4. SI (BF) is suppress index.

Superscript

**Figure D-9. 3288 Variant of EBCDIC for Text Print Feature**

| | | | | |
|---|---|---|---|---|
| A | 1 | \| | a | [ |
| B | 2 | @ | b | ] |
| C | 3 | # | c | ⩾ |
| D | 4 | $ | d | ≠ |
| E | 5 | % | e | NULL |
| F | 6 | ¢ | f | FF |
| G | 7 | & | g | DUP |
| H | 8 | * | h | FM |
| I | 9 | ( | i | NL |
| J | 0 | ) | j | EM |
| K | . | — | k | |
| L | = | + | l | |
| M | ¬ | ! | m | |
| N | ; | : | n | |
| O | ' | " | o | |
| P | , | < | p | |
| Q | . | > | q | |
| R | / | ? | r | |
| S | SPACE | | s | |
| T | | | t | |
| U | | | u | |
| V | | | v | |
| W | | | w | |
| X | | | x | |
| Y | | | y | |
| Z | | | z | |

**Notes:**

1. During execution of a Copy command, only the characters shown above are printed by the 3288 equipped with the Text Print feature and using the 120-character TN print belt.
2. If the 120-character TN print belt is replaced with a 64-character EBCDIC print belt, only the characters in the first 3 columns are printed.
3. The control codes NULL, FF, DUP, FM, NL, and EM are printed as space, <, *, ;, 5 and 9, respectively, regardless of which print belt is installed.
4. When additional character and control codes not shown above appear in the data stream, printing of undefined characters or erroneous printer operation results.

**Figure D-10. 3288 Text Print Restricted Character Set (Copy Command)**

# Appendix E. Abbreviations

**A.** Attention.
**ACK.** Positive acknowledge.
**AID.** Attention identification.
**ALPHA.** Alphameric.
**A/N.** Alphameric (alphabetic/numeric)
**ASCII.** American Standard Code for Information Interchange.
**async.** Asynchronous.
**atb.** Attribute.

**B.** Busy.
**BB.** Begin bracket.
**BCC.** Block check character.
**BIU.** Basic information unit.
**BOC.** Bus out check.
**bps.** Bits per second.
**BSC.** Binary synchronous communications.
**BTAM.** Basic telecommunications access method.

**C.** Column.
**CAW.** Channel address word.
**CC.** Chain Command (flag), control check.
**CCC.** Copy control character.
**CCW.** Channel control word.
**CE.** Channel end.
**char.** Character.
**cmd.** Command.
**cncl.** Cancel.
**cps.** Characters per second.
**CPU.** Central processing unit.
**CR.** Command Reject.
**CRT.** Cathode-ray tube.
**CSW.** Channel status word.
**ctl.** Control.
**CU.** Control unit.
**CUE.** Control Unit End.

**D.** Display.
**DAA.** Data access arrangement.
**DB.** Device busy.
**DC.** Data check.
**DE.** Device end.
**dec.** Decimal.
**del.** Delete.
**disc.** Disconnect.
**DLE.** Data link escape.
**DR.** Definite response.
**dup.** Duplicate.

**EAU.** Erase All Unprotected.
**EB.** End brackets.
**EBCDIC.** Extended binary-coded-decimal interchange code.
**EC.** Equipment check.
**EFI.** expedited flow indicator.
**EM.** end of message.
**ENQ.** enquiry.
**EOF.** end of field.
**EOI.** end of inquiry.
**EOR.** end of record.
**EOT.** end of transmission.
**ERP.** error-recovery procedure.
**ESC.** escape.
**ETB.** end of transmission (block).
**ETX.** end of text.
**EUA.** Erase Unprotected to Address.
**EX.** Exception (response)

**FF.** Forms feed.
**FID.** Format identifier.
**FIE.** Function interpret error.
**FM.** Field mark, function management.

**GP.** General Poll.

**hex.** Hexadecimal.
**Hz.** Hertz.

**I.** Information (format).
**IC.** Insert Cursor.
**ID.** Identity.
**ident.** Identification.
**ind.** Indicator.
**INS.** Insert.
**I/O.** input/output.
**IOS.** Input/Output Supervisor.
**IR.** Intervention Required.
**ITB.** Intermediate transmission block.

**kbd.** Keyboard.

**LRC.** Longitudinal redundancy check.
**LU/SSCP.** Logical unit/system services control point.

**MDT.** Modified data tag.

**NA.** Not applicable.
**NAK.** Negative acknowledgment.
**NCP.** Network control program.
**NL.** New Line.
**NS.** Nonsequenced (format).
**NSA.** Nonsequenced acknowledgment.
**NUL.** Null.

**OC.** Operation Check.
**OICR.** Operator identification card reader.

**P.** Printer, protected.
**PA.** Program access.
**PF.** Program function.
**PSI.** Primary to secondary indicator.
**PT.** Program tab.

**R.** Row.
**RA.** Repeat to Address.
**Rd Mod.** Read Modified.
**req.** Request.
**RH.** Request/response header.
**RNR.** Request not ready.
**ROL.** Request online.
**R/R.** Request/response.
**RR.** Request ready.
**RU.** Request response unit.
**RVI.** Reverse interrupt.

**S.** Sequenced (format).
**SA.** Selection addressing.
**SBA.** Set Buffer Address.

**SDLC.** Synchronous data link control.
**SF.** Start Field.
**SIOF.** Start I/O Fast Release.
**SM.** Status modifier.
**SNA.** Systems network architecture.
**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.
**STX.** Start of text.
**SUB.** Substitute.
**sw.** Switch.
**SYN.** Synchronous idle.

**TC.** Transmission check.
**TCU.** Transmission control unit.
**TH.** Transmission header.
**TTD.** Temporary text delay.

**U.** Unpotected.
**UC.** Unit check.
**UE.** Unit exception.
**US.** Unit specify.

**V.** Volts
**VFC.** Vertical forms control.
**VTAM.** Virtual telecommunications access method.

**WACK.** Wait before transmit positive acknowledgment.
**WCC.** Write control character.
**WT.** World Trade.

# Appendix F. Glossary

This glossary defines data processing and communication terms used in this publication and other terms as they apply to the 3270 Information Display System. For definitions of terms not included in this glossary, see *IBM Data Processing Glossary*, GC20-1699.

**access method.** A technique for moving data between main storage and input/output devices.

**AID.** *See* attention identifier.

**alphameric field.** A field that may contain any alphabetic, numeric, or special character that is available on any of the 3270 keyboards.

**alphameric keyboard.** A typewriter-like keyboard used to enter letters, numbers, and special characters into a display station buffer; also used to perform special functions (such as backspacing) and to produce special control signals.

**attention.** An occurrence, external to an operation, that would cause an interruption of the operation.

**attention identifier (AID).** A code that is recorded in the display station when the operator takes an action that produces an interruption.

**attribute.** A characteristic of a display field. The attributes of a display field include protected or unprotected (against manual input and copy operations); numeric-only or alphameric input control: displayed, nondisplayed, display-intensified; selector-pen-detectable or -nondetectable; and modified or not modified.

**attribute character.** A code that defines the attributes of the display field that follows. An attribute character is the first character in a display field, but it is not a displayable character.

**audible alarm.** An alarm that is activated when predetermined events occur that require operator attention or intervention or system operation.

**auto-poll.** A machine feature of a transmission control unit that permits it to handle negative responses to polling without interrupting the processing unit.

**automatic polling.** *See* auto-poll.

**automatic skip.** After entry of a character into the last character position of an unprotected display field, automatic repositioning of the cursor from a protected and numeric field to the first character position of the next unprotected display field.

**automatic upshift.** Automatic shift of the Data-Entry Keyboard when the cursor enters an unprotected numeric field to allow entry of only the upper symbols on dual-character keys.

**available/unavailable.** A device is available for CU-channel operation if (1) ac power is on at the device, (2) it is online, (3) it is physically attached to the CU, and (4) its security lock is turned on. The device is unavailable if any one of these conditions does not exist.

**basic mode.** A set of facilities (including the macro instructions needed to use them) that enable the application program to communicate with BSC and start-stop terminals, including the locally attached 3270 Information Display System. READ, WRITE, SOLICIT, RESET, DO, and LDO macro instructions are basic-mode macro instructions.

**basic telecommunications access method (BTAM).** An access method that permits read/write communications with remote devices.

**binary synchronous communications (BSC).** Communication using binary synchronous transmission.

**bracket.** In VTAM, an exchange of data between an application program and a logical unit that accomplishes some task defined by the user as uninterruptible.

**BSC.** *See* binary synchronous communications.

**BTAM.** *See* basic telecommunications access method.

**buffer.** The hardware portion of a display station, control unit, or buffered printer in which display or print data is stored.

**buffer address.** The address of a location in the buffer at which one character can be stored.

**busy/not busy.** The CU considers a device busy if (1) it is performing an operation initiated by the CU (namely, an erase-all-unprotected operation or a printing operation) or (2) the CU attempted to perform a command with the device but found the device busy executing a manually initiated operation. A manual operation can be initiated at the keyboard, operator identification card reader, or selector pen.

**cathode-ray tube (CRT).** A vacuum tube in which a slender beam of electrons is projected upon a fluorescent screen to produce a luminous glow corresponding to the path of the beam.

**CCC.** *See* copy control character.

**character addressing.** The capability of gaining access to any character position in the buffer by using an address.

**character generator.** A hardware unit contained in each 3275, 3277, and printer. It converts the digital code for a character into signals that cause the character to be printed or displayed.

**character position.** A location on the screen at which one character can be displayed; also, an addressed location in the buffer at which one character can be stored.

**clear indicator.** In VTAM, a SESSIONC indicator sent by one node to another that prevents the exchange of messages and responses.

**cluster control unit.** (1) A device that can control the input/output operations of more than one device. A remote cluster control unit can be attached to a host CPU only via a communication controller. A cluster control unit may be controlled by a program stored and executed in the unit, or it may be controlled entirely by hardware. (2) *See also* communication controller.

**command.** A request from a terminal for the performance of an operation or the execution of a particular program.

**communication controller.** (1) A type of communication control unit whose operations are controlled by a program stored and executed in the unit. Examples are the IBM 3704 and 3705 Communications Controllers. (2) *See also* cluster control unit.

**communication facilities.** Any media, such as a telephone circuit, that connects a remote 3270 unit (3271 or 3275) with a computer.

**connection.** In VTAM, in response to a request from an application program, the linking of VTAM control blocks in such a way that the program can communicate with a particular terminal. The connection process includes establishing and preparing the network path between the program and the terminal.

**control character.** A character whose occurrence in a particular context initiates, modifies, or stops a control operation.

**conventional 3270.** A locally attached 3270 terminal or a remotely attached 3270 terminal that uses the BSC line discipline.

**copy control character (CCC).** A character used in conjunction with the Copy command to specify the type of data to be copied.

**copy operation.** An operation that copies the contents of the buffer from one display station or printer to another display station or printer attached to the same control unit.

**cursor.** A visible, movable mark used to indicate a position on a display surface.

**cursor check.** An error condition that occurs when 3275 or 3277 circuitry detects no cursor or more than one cursor in the display buffer.

**Data-Entry Keyboard.** A standard typewriter keyboard on which the numeric keys are grouped in a format similar to the numeric keys on a card punch keyboard (to facilitate entry of numeric data). Other features include (1) automatic upshift of the keyboard when the cursor enters a numeric-only display field and (2) automatic prevention of entry of nonnumeric characters into a numeric-only display field, when the special Numeric Lock feature is installed.

**data set.** *See* modem.

**data stream.** All data transmitted through a channel in a single read cr write operation to a display station or printer.

**data transfer.** In data communications, the sending of data from a data source and the receiving of the data at a data sink.

**data-transfer mode.** (1) A set of facilities (including the macro instructions needed to use them) that enable the application program to communicate with terminals. (2) *See also* basic mode and record mode.

**definite response 1.** In VTAM, a response that indicates whether its associated message was successfully forwarded to its final destination (such as the display screen of an output device).

**definite response 2.** In VTAM, a response that indicates that the node sending the response has accepted recovery responsibility for the associated message.

**definition statement.** In VTAM, the means of describing an element of the telecommunication system.

**designator character.** A character that immediately follows the attribute character in a selector-pen-detectable field. The designator character controls whether a detect on the field will or will not cause an attention. For a nonattention-producing field, the designator character also determines whether the modified data tag for the field is to be set or reset as the result of a selector-pen detect.

**detect.** *See* selector-pen detect.

**detectable.** An attribute of a display field; determines whether the field can be sensed by the selector pen.

**disconnection.** In VTAM, the disassociation of VTAM control blocks in such a way as to end communication between the program and a connected terminal. The disconnection process includes suspending the use of the network path between the program and the terminal.

**display field.** An area in the display buffer, or on a screen, that contains a set of characters, manipulated or operated upon as a unit.

**display operator.** A person who uses the keyboard to perform operations at a display station.

**erase all unprotected (EAU) command.** A command that clears all unprotected fields to nulls, resets modified data tags in all unprotected fields, unlocks the keyboard, resets the attention identifier, and repositions the cursor to the first character of the first unprotected field.

**erase unprotected to address (EUA) order.** An order that erases all unprotected positions (inserts nulls) from the current buffer address up to, but not including, the specified stop address.

**escape command sequence.** A 2-character sequence used in remote operations that consists of ESC (27 hex in EBCDIC and 1B hex in ASCII) and the command character that follows and specifies the 3270 command.

**field.** *See* display field.

**FME response.** *See* definite response 1.

**formatted display.** A screen display in which a display field, or fields, has been defined as a result of storing at least one attribute character in the display buffer.

**general polling.** (1) An input technique for remote 3270 devices in which special invitation characters are sent to a device control unit instructing that control unit to begin transmission from all devices ready to enter data. (2) *See also* polling and specific polling.

**incoming group.** (1) In systems with TCAM, that portion of a message handler designed to handle messages arriving for handling by the message control program. (2) *See also* outgoing group.

**input field.** An unprotected field in which data can be entered, modified, or erased manually.

**insert cursor (IC).** An order that moves the cursor, if necessary, to the current buffer address.

**intensified display.** An attribute of a display field; causes data in that field to be displayed at a brighter level than other data displayed on the screen.

**intensified field.** Data in a field displayed at a brighter level than that for a nonintensified field.

**interpret table.** In VTAM, an installation-defined correlation list that translates an argument into a string of 8 characters. Interpret tables can be used to translate a logon message into the name of an application program for which the logon request is intended.

**invitation list.** In systems with the telecommunications access method (TCAM), a sequence of polling characters or identification sequences associated with the stations online; the order in which the characters are specified determines the order in which the stations are invited to enter a message.

**I/O pending.** The condition that results (1) in the generation of the attention status in a locally attached display station and (2) in a response to a polling operation in a remotely attached display station.

**keyboard numeric lock.** A special feature that allows entry of 0–9, minus (-), period (.), or DUP only; otherwise, the keyboard will be disabled.

**leased line.** *See* nonswitched line.

**line adapter.** 1200-bps Integrated Modem.

**line control characters.** Characters that regulate the transmission of data over a line; for example, delimiting messages, checking for transmission errors, and indicating whether a station has data to send or is ready to receive data.

**line group.** One or more communication lines, of the same type, that can be activated and deactivated as a unit.

**local.** Pertaining to the attachment of devices directly by channels to a host CPU. Contrast with *remote.*

**logical unit.** The combination of programming and hardware of a teleprocessing subsystem that constitutes a terminal for VTAM.

**logoff.** In VTAM, a request from a terminal to be disconnected from an application program.

**logon.** In VTAM, a request by or on behalf of a terminal to be connected to an application program.

**logon message.** In VTAM, the data that can accompany a logon request received by the application program to which the request is directed.

**major node.** A set of one or more minor nodes represented by a single symbolic name. A major node can be a set of local terminals, a set of application programs, or a network control program.

**MCP.** *See* message control program.

**MDT.** *See* modified data tag.

**message control program (MCP).** In TCAM, a program that is used to control the sending or receiving of messages to or from remote terminals.

**message handler.** In systems with the telecommunications access method (TCAM), a sequence of user-specified macro instructions that examine and process control information in message headers, and that perform functions necessary to prepare message segments for forwarding to their destinations. One message handler is required for each line group having unique message-handling requirements.

**modem.** A device that modulates and demodulates signals transmitted over communication facilities.

**modified data tag (MDT).** A bit in the field attribute of a display field, which, when set to 1, causes that field to be transferred to the host during a read modified operation.

**multidrop.** A line or circuit interconnecting several stations; synonymous with *multipoint line.*

**NIB.** *See* node initialization block.

**node.** A point in a telecommunication system defined to VTAM by a symbolic name. *See also* major node.

**node initialization block (NIB).** In VTAM, a control block, associated with a particular node, that contains information used by the application program to identify a node and indicate how communication requests directed at the node are to be implemented.

**nonswitched line.** A connection that does not have to be established by dialing.

**null character.** An all-0 character that occupies a position in the storage buffer and is displayed as a blank.

**null suppression.** In reading the contents of the buffer for a display or printer, the bypassing of all null characters in order to reduce the amount of data to be transmitted or printed.

**order code.** A code that may be included in the write data stream transmitted for a display station or printer; provides additional formatting or definition of the write data.

**order sequence.** A sequence in the data stream that starts with an order code and includes a character address and/or data characters related to the order code.

**outgoing group.** (1) In systems with TCAM, that section of a message handler that manipulates outgoing messages after they have been removed from their destination queues. (2) *See also* incoming group.

**parity check.** An error condition that occurs when 3270 system circuitry detects one or more characters with bad parity in a 3270 unit buffer.

**PCI.** *See* program-controlled interruption.

**polling.** A technique by which each of the terminals sharing a communication line is periodically interrogated to determine whether it requires servicing.

**printer hang (3284/3286 only).** This condition exists when the print mechanism is unable to advance successfully. This condition can occur at any time during a printout through to, and including, the carriage return and new line advance. The printer will try to recover, that is, mechanically restore its print mechanism to the starting position. This hang condition may be caused by a mechanical malfunction or by loss of ac power at the carriage motor.

**program access (PA) key.** A program attention key that may be defined to solicit program action that does not require data to be read from the buffer of the display station. If a Read Modified command is issued in response to the program-attention-key interruption, only the attention identification (AID) character is transferred to the program; no data from the buffer is transferred.

**program attention key.** On a display keyboard, a key that produces an interruption to solicit program action.

**program-controlled interruption (PCI).** An interruption that allows buffers to be deallocated continuously, replenishing the available unit pool.

**program function (PF) key.** On a display keyboard, a key that passes a signal to a program to solicit a particular program operation.

**Program Tab (PT).** An order that advances the current buffer address to the address of the first character location of the next unprotected field.

**protected field.** A display field for which the display operator cannot use the keyboard or operator identification card reader to enter, modify, or erase data.

**read-modified operation.** An operation in which only those display fields in which the modified data tag is set are read.

**ready/not ready.** The only devices that can be *not ready* are the attached printers. Thus, a printer is not ready to operate with the CU when (1) the printer's cover is open, (2) it is out of paper, or (3) a *hang* condition exists in the printer. (*See* Printer Hang.)

**record mode.** A set of facilities (and the macro instructions needed to use them) that enable the application program to communicate with logical units or with the locally or remotely attached 3270 Information Display System. SEND and RECEIVE are record-mode macro instructions.

**remote.** Pertaining to the attachment of devices to a central computer through a communication control unit. Contrast with *local.*

**Repeat to Address (RA).** An order that stores a specified alphameric or null character, starting at the current buffer address and ending at, but not including, the specified stop address.

**request parameter list (RPL).** In VTAM, a control block that contains the parameters necessary for processing a request for connection or communication, or a request for an operation related to connection or communication.

**RPL.** *See* request parameter list.

**RRN response.** *See* definite response 2.

**SDLC.** Synchronous data link control.

**security key lock.** A special feature that disables all input functions and blanks the display, except when the key is inserted in the lock and turned.

**selector pen.** A pen-like instrument that may be attached to the display station as a special feature. When pointed at a detectable portion of an image and then activated, the selector pen senses the presence of light at a display field and produces a selector-pen detect.

**selector-pen attention.** An interruption generated when a selector-pen detect occurs on a display field that has a null or space designator character. The attention concludes the selector-pen operation.

**selector-pen detect.** The sensing by the selector pen of the presence of light from data in a display field that has the detectable attribute. Depending on the designator character of that display field, the detection and location information is identified on the screeen (and stored in the buffer) or may produce an interruption that is transmitted to the CPU.

**SESSIONC indicators.** In VTAM, indicators that can be sent from one node to another without using SEND or RECEIVE macro instructions. SDT, clear, and STSN are SESSIONC indicators. All SESSIONC indicators are sent with a SESSIONC macro instruction.

**Set Buffer Address (SBA).** An order that sets the buffer address to a specified location.

**Short Read.** A Read Modifed command sent in reply to depression of the CLEAR CNCL key or of a PA key at a display station. Only an AID byte is transferrred to main storage.

**SNA 3270.** A 3270 terminal that uses synchronous data link control (SDLC) and is treated as a logical unit by VTAM.

**specified polling.** (1) A polling technique that sends invitation characters to a device to find out whether the device is ready to enter data. (2) *See also* general polling and polling.

**Start Field (SF).** An order that defines the start of a data field for display or printing.

**Structured Data, 6-Bit.** The low-order, 6-bit, binary-coded characters used internally by the CU. The 6-bit code is applicable to all characters received by the CU: graphic, AID, attribute, write control (WCC), copy control (CCC), CU and device address, buffer address, status and sense.

**suppress index (SI) order.** An order that generates the suppress index character, valid only for the 3288 Printer Model 2 (other printers receive I, an OR bar). This character inhibits a line index to allow overprinting.

**switched line.** A communication line in which the connection between the computer and a remote terminal is established by dialing.

**TCAM.** *See* telecommunications access method.

**telecommunication network.** In a telecommunication system, the combination of all terminals and other telecommunication devices and the lines that connect them.

**telecommunications access method (TCAM).** A method used to transfer data between main storage and remote or local terminals. Application programs use either GET and PUT or READ and WRITE macro instructions to request the transfer of data, which is performed by a message-control program.

**terminal.** (1) *A point in a system or communication network at which data can either enter or leave. (2) Any device capable of sending and receiving information over a communication channel.

**terminal-initiated logon.** A logon request that originates from the terminal.

**Test Request Read.** A Read Modified command resulting from the operator's pressing the TEST REQ key to allow entry of a predefined test-request data format.

**unformatted display.** A screen display in which no attribute character (and, therefore, no display field) has been defined.

**unprotected field.** A display field for which the display station operator can manually enter, modify, or erase data.

**virtual telecommunications access method (VTAM).** A set of IBM programs that control communication between terminals and application programs running under DOS/VS, OS/VS1, and OS/VS2.

**VTAM.** *See* virtual telecommunications access method.

*American National Standards Institute (ANSI), *American National Dictionary for Information Processing.*

**VTAM definition library.** The DOS/VS files or OS/VS data sets that contain the VTAM definition statements filed during VTAM definition. These statements describe the telecommunication system to VTAM and can be used to tailor VTAM and the system to suit the needs of the installation.

**WCC.** *See* write control character.

**wraparound.** The continuation of an operation (for example, a read operation or a cursor movement operation) from the last character position in a buffer to the first character position in the buffer.

**write control character (WCC).** A character used in conjunction with a Write-type command to specify that a particular operation, or combination of operations, is to be performed at a display station or printer.

**1200-bps Integrated Modem.** A feature for the 3275 that provides a modem capable of operating at a speed of 1,200 bps over nonswitched communication facilities, or at speeds of 600/1,200 bps over switched communication facilities via a similarly equipped 2701 or 3705.

# Index

IBM 3270 IDS 3271, 3272, 3275 Description and Programmer's Guide (File No. S360/S370/S3/4300-09) Printed in U.S.A. GA23-0060-0

GA23-0060-0

IBM 3270 Information Display System
3271 Control Unit, 3272 Control Unit, 3275 Display Station
Description and Programmer's Guide

**READER'S COMMENT FORM**

Order No. GA23-0060-0

**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.*

**How did you use this publication?**

[ ] As an introduction
[ ] As a reference manual
[ ] For another purpose (explain) _____

[ ] As a text (student)
[ ] As a text (instructor)

_____

**Is there anything you especially like or dislike about the organization, presentation, or writing in this manual? Helpful comments include general usefulness of the book; possible additions, deletions, and clarifications; specific errors and omissions.**

**Page Number:** **Comment:**

**What is your occupation?** _____

**Newsletter number of latest Technical Newsletter (if any) concerning this publication:** _____

**If you wish a reply, give your name and address:**

**IBM branch office serving you** _____

**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 or you may mail directly to the address in the Edition Notice on the back of the title page.)**

Please use pressure sensitive or other gummed tape to seal this form.

GA23-0060-0

**Reader's Comment Form**

Cut or Fold Along Line

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.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department 52Q
Neighborhood Road
Kingston, New York 12401

Fold and tape | **Please Do Not Staple** | Fold and tape

IBM®

IBM 3270 Information Display System
3271 Control Unit, 3272 Control Unit, 3275 Display Station
Description and Programmer's Guide

**READER'S COMMENT FORM**

Order No. GA23-0060-0

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.*

How did you use this publication?

[ ] As an introduction
[ ] As a reference manual
[ ] For another purpose (explain) _____

[ ] As a text (student)
[ ] As a text (instructor)

_____

Is there anything you especially like or dislike about the organization, presentation, or writing in this manual? Helpful comments include general usefulness of the book; possible additions, deletions, and clarifications; specific errors and omissions.

Page Number: Comment:

What is your occupation? _____

Newsletter number of latest Technical Newsletter (if any) concerning this publication: _____

If you wish a reply, give your name and address:

IBM branch office serving you _____

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 or you may mail directly to the address in the Edition Notice on the back of the title page.)

Please use pressure sensitive or other gummed tape to seal this form.

GA23-0060-0

**Reader's Comment Form**

Cut or Fold Along Line

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.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department 52Q
Neighborhood Road
Kingston, New York 12401

Fold and tape | **Please Do Not Staple** | Fold and tape

IBM®

GA23-0060-0