Alfaskop System 41

Reference Manual
IBM 3270 Emulation

DATASAAB

FE411-810C
1981-02-20

Printed in Sweden by Text & Tryck AB, Spånga/1981

Preface

"Reference Manual – IBM 3270 Emulation" presents the Alfaskop System 41 as used in IBM 3270 terminal system applications. Its purpose is to provide you with reference material on the Alfaskop System 41 and the interrelationships among system components.

Chapters 1 and 2 present basic information about system components and terminal system configurations.

Chapter 3 describes the operational characteristics. The commands and orders are defined in chapter 4.

Chapter 5 defines the communication protocol for binary synchronous communication (BSC), remote connection (IBM 3274 model 1 C).

Chapter 6 defines the SDLC communication protocol and the SNA commands used with this protocol when the IBM 3274 Control Unit (model 1 C, SNA/SDLC version) is emulated.

Chapter 7 defines the communication protocol for local connection (IBM 3274 model 1 B and 1 D).

The specifications in this publication are subject to change or supplementation without notice.

To find out more about the Alfaskop System 41, please contact your Datasaab representative.

This is a revised edition of publication FE411-810B.

Datasaab AB
Alfaskop Division
Documentation Department
S-175 86 Järfälla, Sweden

Reader comments

You can help improve future editions of this document by answering the following questions and sending us your comments.

1. Document name and number: ____________________

____

2. My job: ____________________

3. I have used this document

☐ To acquire general information (for ____________________

____________________)

☐ As a technical manual (reference book)

☐ As a textbook in a course (trainee)

☐ As source material for teaching a course (instructor)

Comments: ____________________

____

4. I think this document is

☐ Easy to find things in

☐ Easy to find things in with some reservations

☐ Difficult to find things in

Comments: ____________________

____

5. I think this document is

☐ Easy to understand

☐ Easy to understand with some reservations

☐ Difficult to understand

Comments: ____________________

____

6. I think this document is

☐ Well illustrated

☐ Poorly illustrated

Comments: ____________________

____

7. I think that this document provides

☐ Full coverage of the subject at hand

☐ Poor coverage (essential parts are lacking)

Comments: ______________________________

____

8. I think that this document is

☐ Well adapted to my skills and knowledge

☐ Poorly adapted to my skills and knowledge

Comments: ______________________________

____

9. Other comments: ______________________________

____

____

____

We appreciate your cooperation.
Please send your answers and comments to

Datasaab AB
Alfaskop Division
Documentation Department
S-175 86 Järfälla, Sweden

Contents

System Components

Introduction

The Alfaskop System 41 was made possible by a long-term development program that embraced ergonomics, state-of-the-art terminal system architecture and sophisticated software systems.

The Alfaskop System 41 used in IBM 3270 emulations is an intelligent, programmable terminal system that prepares and sends information to an IBM host computer and obtains information in return.

The Alfaskop System 41 hardware consists of communication processors, display units, printer units and flexible disk units which can be combined in various configurations.

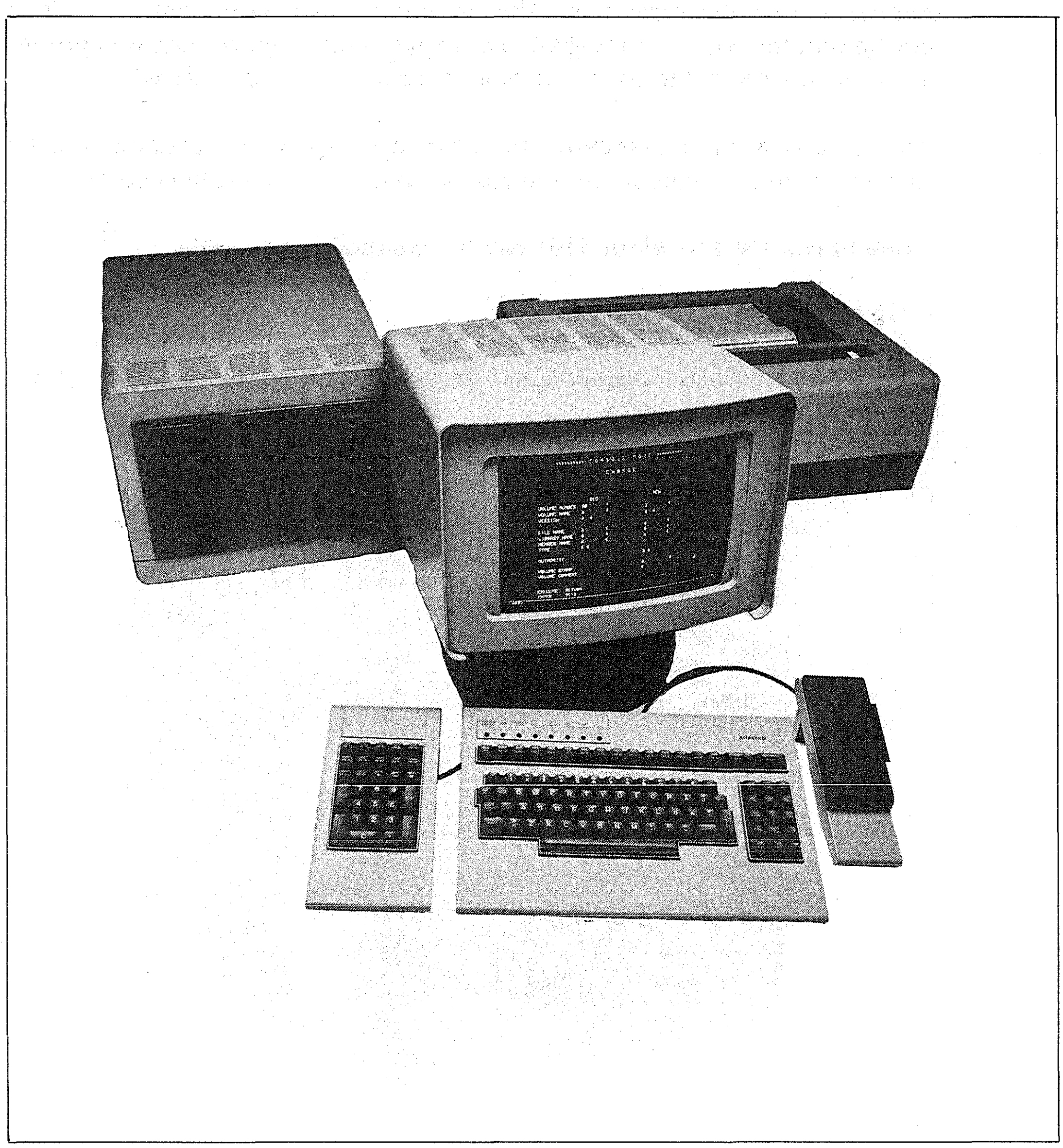


Fig. 1. Alfaskop System 41 hardware

A system component is a functional hardware or software unit. The system components mentioned in this section can be used in a terminal system intended for IBM 3270 emulations.

Alfaskop System 41 features are categorized as specify features or extra features. A specify feature must be included in a functional system. An extra feature is an optional feature that improves performance, provides additional operational capability, or permits expansion of the system.

Hardware

Communication Processor 4101

Communication Processor 4101 is intended for modem connection via the CCITT V 24/28 interface. Transfer rates range up to 9600 bps.

The communication processor controls communication between the host computer and the terminals. The communication processor is used in configurations where more than one display unit or more than one printer are to be connected to the same modem (cluster configurations).

The basic version of Communication Processor 4101 can accommodate up to eight terminal connections and can be connected to one host computer.

Communication Processor 4101 can be expanded to provide

- Up to 32 terminal connections

- Two host computer connections via modem/communication lines (Software on an RPQ basis)

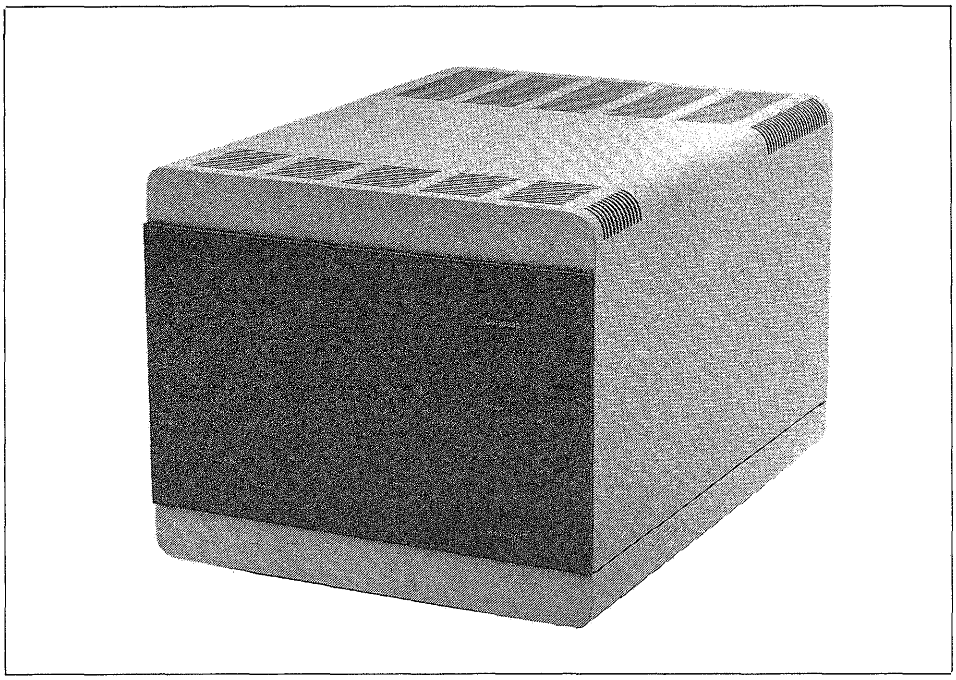


Fig. 2. Communication Processor 4101

Functionally, Communication Processor 4101 can be subdivided as follows

- Synchronous Communication Adapter 4194
- Microprocessor
- Memory
- Memory expansion
- Direct memory access
- Terminal selection control
- Terminal unit adapter

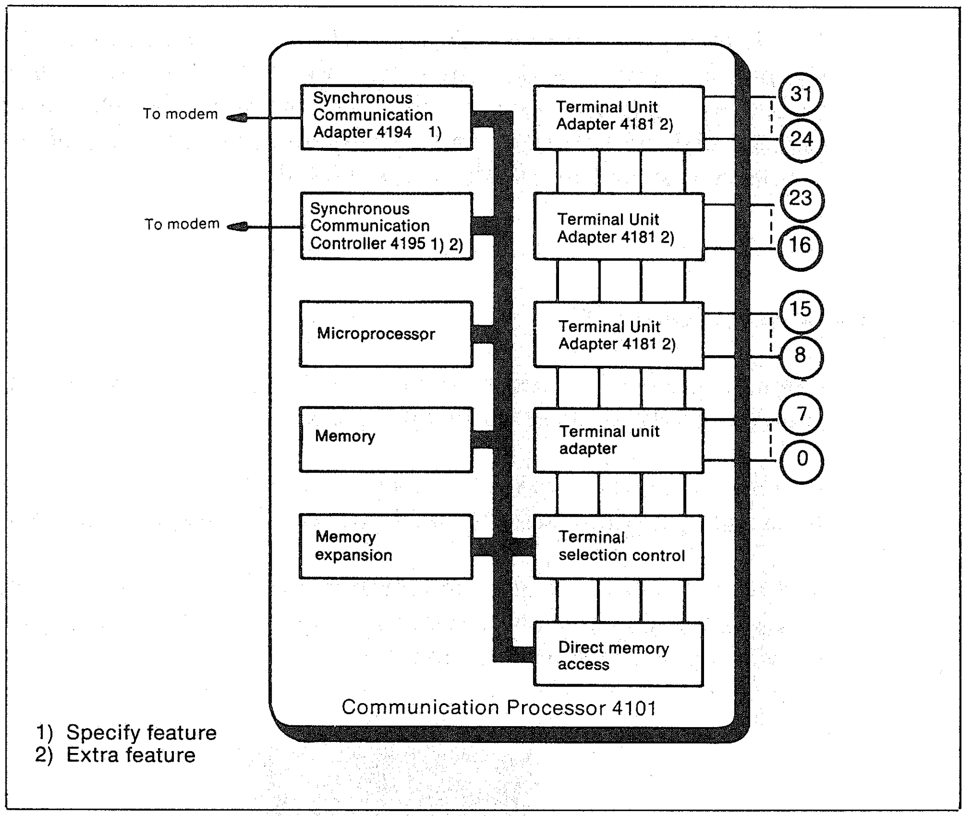


Fig. 3. Block diagram of Communication Processor 4101

The synchronous communication adapter converts serial data, received from the communication line, to parallel data used in the communication processor and vice versa. Line transmission is synchronous and can comply with either the BSC or the SDLC protocol.

The microprocessor is an 8-bit LSI processor with an addressing capability of 64 kilobytes.

The basic memory (read/write memory, RWM) has a capacity of 32 kilobytes and it can be expanded to a maximum of 64 kilobytes. The software that controls the operation of the Alfaskop System 41 is stored on diskettes. When power is turned on, the software is loaded into the memory.

One terminal unit adapter is included in the basic version of the communication processor. It enables eight two-wire cables (or coaxial cables) to be connected to the communication processor. Each additional Terminal Unit Adapter 4181 expands the number of two-wire (or coaxial) connections by eight. The maximum number of the connections is 32.

The terminal unit adapter consists mainly of a switch matrix that is controlled by the microprocessor via the terminal selection control.

Direct memory access (DMA) transfers are initiated by the microprocessor. DMA transfer includes

- Internal polling of terminals
- Host computer communication
- Internal communication between terminals

If a display unit requests a data transfer to an assigned flexible disk unit, the microprocessor makes the connection by sending control information to the terminal selection control. The display unit then addresses the flexible disk unit directly and the microprocessor is asked to break the connection when the transfer is completed.

Communication Processor 4102

Communication Processor 4102 is intended for local connection to an IBM selector, block multiplexer or multiplexer channel.

The maximum instantaneous data transfer rate is 650,000 bytes per second for write operations and 400,000 bytes per second for read operations.

The basic version of Communication Processor 4102 can accommodate up to eight terminals. Communication Processor 4102 can be expanded to provide up to 32 terminal connections.

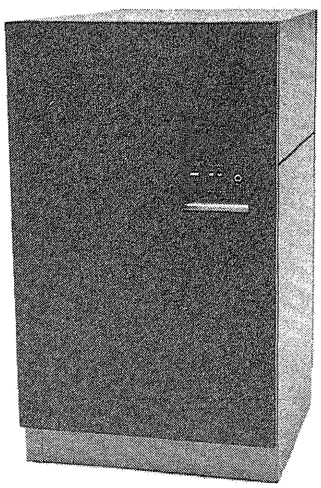


Fig. 4. Communication Processor 4102

Functionally, Communication Processor 4102 can be subdivided as follows

- Channel communication controller
- Microprocessor
- Memory
- Memory expansion
- Direct memory access
- Terminal selection control
- Terminal unit adapter

Communication Processor 4102 is built up on the same principles as Communication Processor 4101. It differs only with regard to the communication interface.

The channel communication controller is designed for direct connection to an IBM System 360/370 or IBM 3030/4300 Series data channel.

The rest of the functional blocks are the same in the two communication processors.

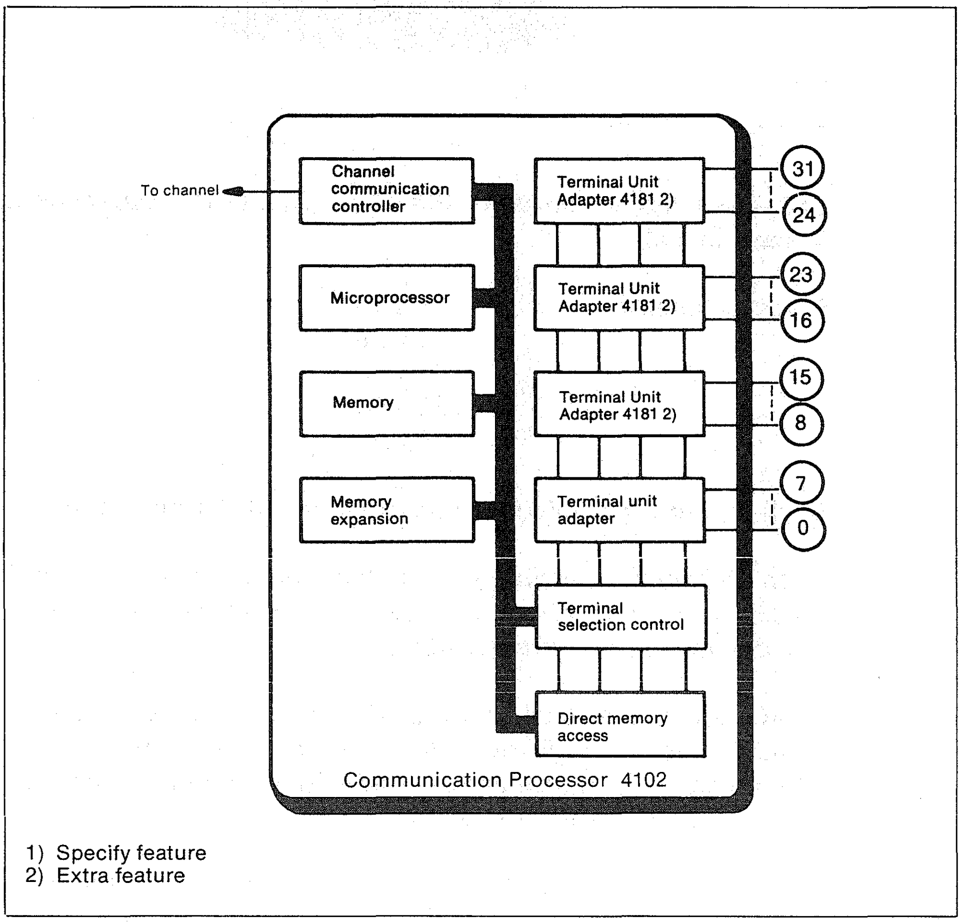


Fig. 5. Block diagram of Communication Processor 4102

Display Unit 4110

Display Unit 4110 is an attractive, operator-oriented cathode ray tube display.

The display unit features an anti-glare screen, and it is also equipped with a pull-out hood which eliminates reflections encountered in poor lighting. The display unit can be tilted and swiveled, thus ensuring comfortable, convenient viewing for each individual operator.

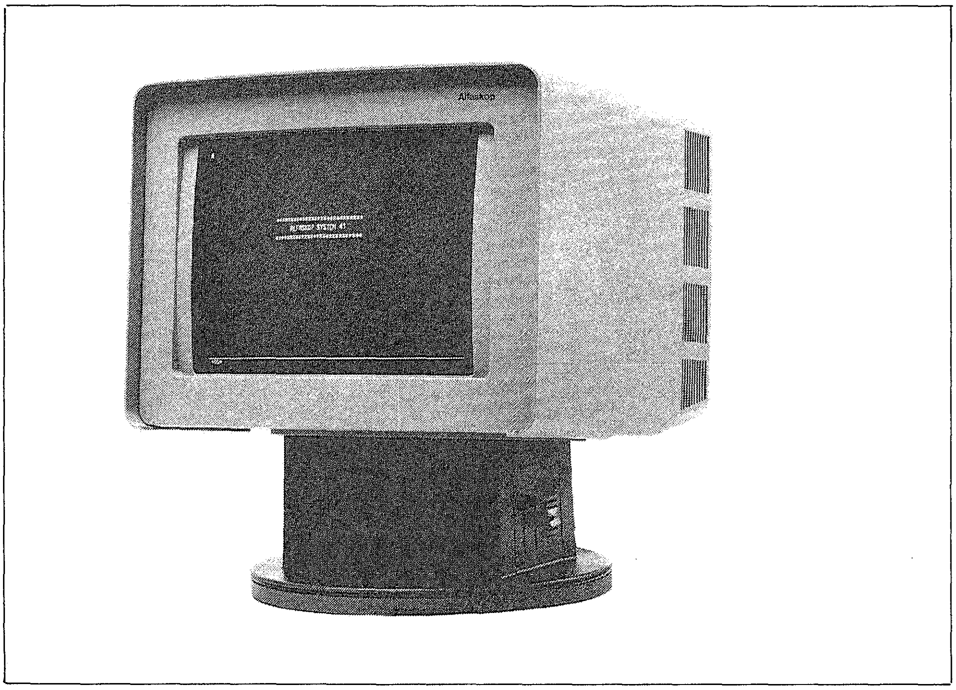


Fig. 6. Display Unit 4110

The screen has a capacity of 480-3440 characters. Program-defined screen formats include

Lines x characters:	12 x 40
	12 x 80
	24 x 80
	32 x 80
	43 x 80

Each format is provided with an extra line for terminal system messages.

Each character is displayed in a character cell consisting of 9 x 16 dots (except in 32- and 43-line formats), thus ensuring crisp, clear characters and minimizing risk of misreading.

Fields can be displayed with high or normal brightness, and they can also be made nonvisible. Flashing text and underlining are also provided. The cursor can be displayed either as an underscore or a filled rectangle. The cursor can flash or glow steadily.

Display units are available for cluster configurations (connection to the communication processor) and single display-unit configurations (modem connections).

Display Unit Characteristics

Effective screen size:	Height 180 mm Width 258 mm
Character cells:	9 horizontal x 16 vertical dots 9 x 12 for 32-line formats 9 x 9 for 43-line formats
Dot matrices:	7 x 13, 7 x 9 and 7 x 8
Refresh rate:	50 Hz
Screen colour:	Amber

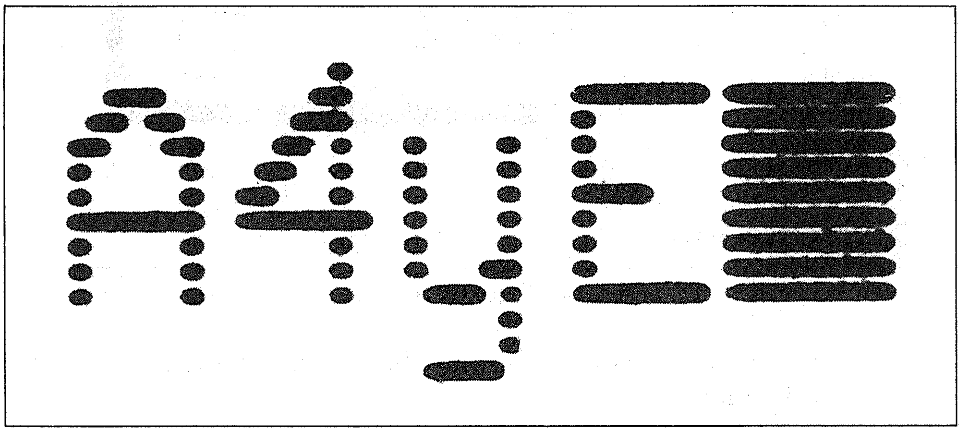


Fig. 7. Examples of character layouts

Display Unit Operation

Functionally, Display Unit 4110 can be subdivided as follows

- Microprocessor
- Memory
- Memory expansion
- Two-wire interface adapter
- Keyboard interface
- Cathode ray tube unit (CRU)
- CRU interface
- Selector pen adapter
- Synchronous Communication Adapter 4194 (single display-unit configuration)
- Asynchronous Communication Adapter 4193, if a V24/28 printer is to be connected.

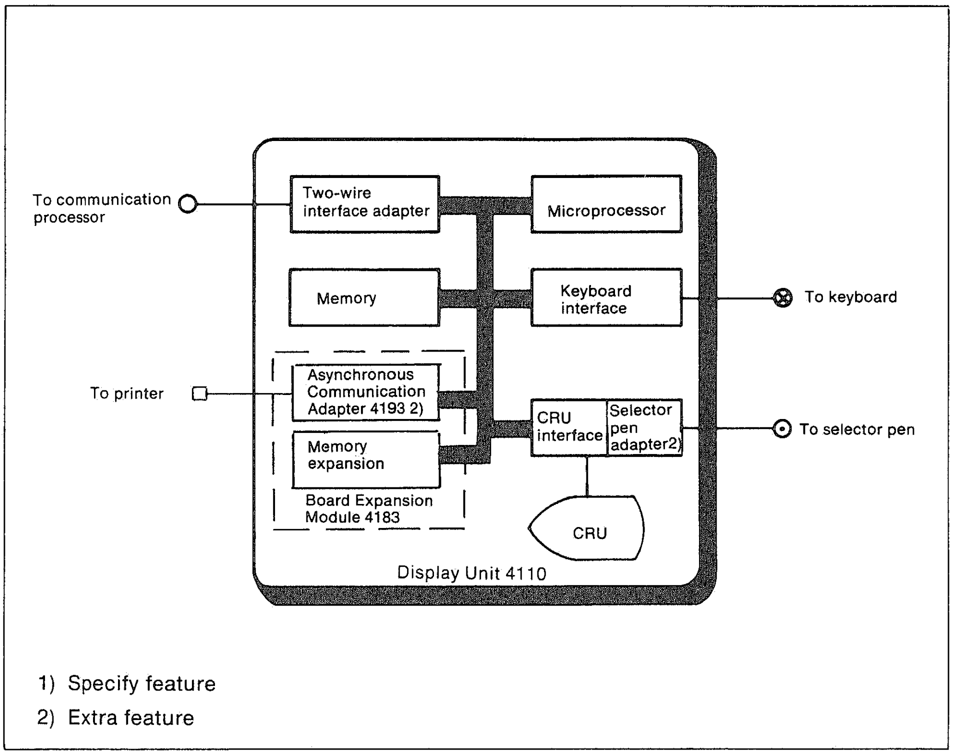


Fig. 8. Block diagram of Display Unit 4110 in cluster configuration

The microprocessor is an 8-bit LSI processor with an addressing capability of 64 kilobytes.

The basic memory has a capacity of 32 kilobytes and it can be expanded to a maximum of 64 kilobytes. Board Expansion Module 4183 is needed if the memory is to be expanded.

The display unit is connected to the communication processor via a two-wire cable. Coaxial cable can also be used. The two-wire interface adapter includes circuits that provide galvanic isolation from the communication processor.

The two-wire interface adapter converts serial information, sent to/from the communication processor, to parallel data used in the display unit. The data sent on the two-wire cable is frequency-modulated and the transfer rate is 300 kilobits per second. The interface adapter also includes circuits for direct memory access.

When the display unit is used in single display-unit configurations the interface adapter serves as an adapter for the flexible disk unit.

Synchronous Communication Adapter 4194 is the same adapter that is used in Communication Processor 4101. The adapter converts serial data received from the communication line to parallel data used in the display unit and vice versa. Line transmission is synchronous and can comply with either the BSC or the SDLC protocol.

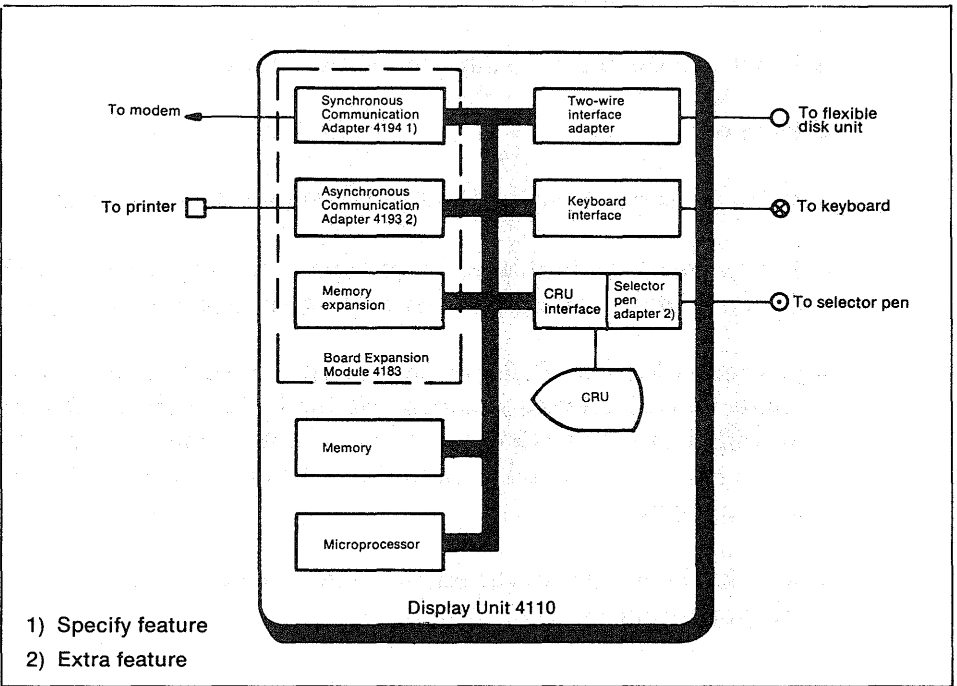


Fig. 9. Block diagram of Display Unit 4110 in single display-unit configuration

Keyboard Unit 4140 with Keyboard Expansion Unit 4141

The keyboard consists of a basic module plus an optional expansion unit which can be located either to the right or left of the basic module.

The keyboard is connected to the display unit via a 1-metre cable and can thus be positioned as desired relative to the display unit.

The keyboard incorporates a microcomputer which controls and supervises its operations.

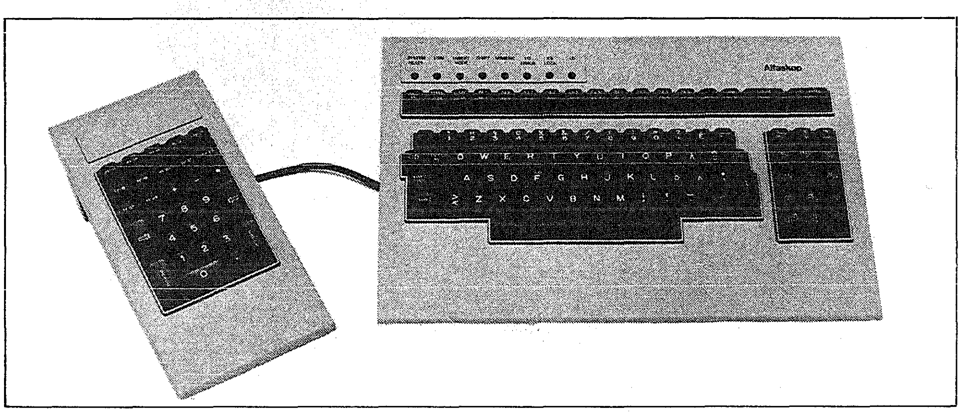


Fig. 10. Keyboard Unit 4140 with Keyboard Expansion Unit 4141

Keyboard features include

- Up to 127 keys
- Clicking sound (adjustable volume) which acknowledges keyed entries
- Alarm signal (adjustable volume and pitch) for illegal keyed entries
- Indicator lamps that are lighted and extinguished by signals from the terminal system and/or host computer
- Keyboard slope can be adjusted in two steps
- Palm rest

The following basic types of keyboards are available for Display Unit 4110

- Typewriter keyboard with basic typewriter key layout. Alphanumeric keys are encoded with both lower case and upper case characters.
- Data entry keyboard with basic data-entry type of layout. When numeric characters are entered in a numeric field, the keyboard is automatically upshifted to take advantage of the grouped numeric keys. When lower shift characters are to be entered in a numeric field the ALPHA SHIFT key has to be used.
- APL keyboard allows the entry of characters oriented to APL programming applications.

Magnetic Identification Device 4131

The magnetic identification device reads magnetically encoded ID-cards containing up to 37 ID-characters plus three control characters.

The magnetic identification device is a separate unit connected to the keyboard via a 1-metre cable.

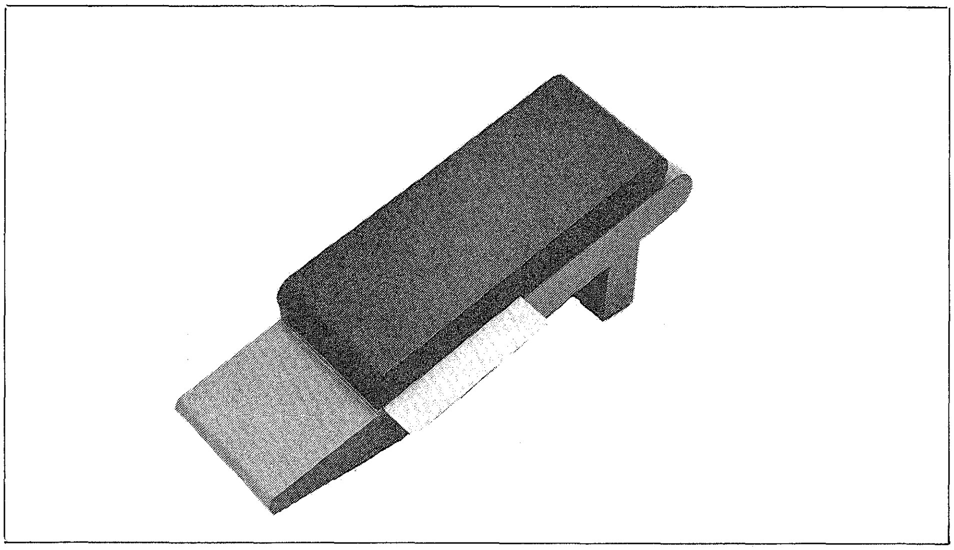


Fig. 11. Magnetic Identification Device 4131

Selector Pen Device 4130

The Selector Pen Device 4130 consists of a selector pen with a fibre optic cable and a selector pen adapter.

The selector pen is connected to the display unit via the fibre optic cable. The selector pen adapter is inserted into the display unit.

The selector pen is used to select among a number of predefined fields on the screen, whereupon the computer and/or terminal system can carry out certain operations that have been defined for the field in question.

Printer Unit 4154

The free-standing, pedestal-mounted printer unit provides hard copies of information appearing on the display screen or data obtained from the host computer system.

The buffered printer unit is connected to the display unit, and it can be located at a distance of up to 600 metres from the display unit.

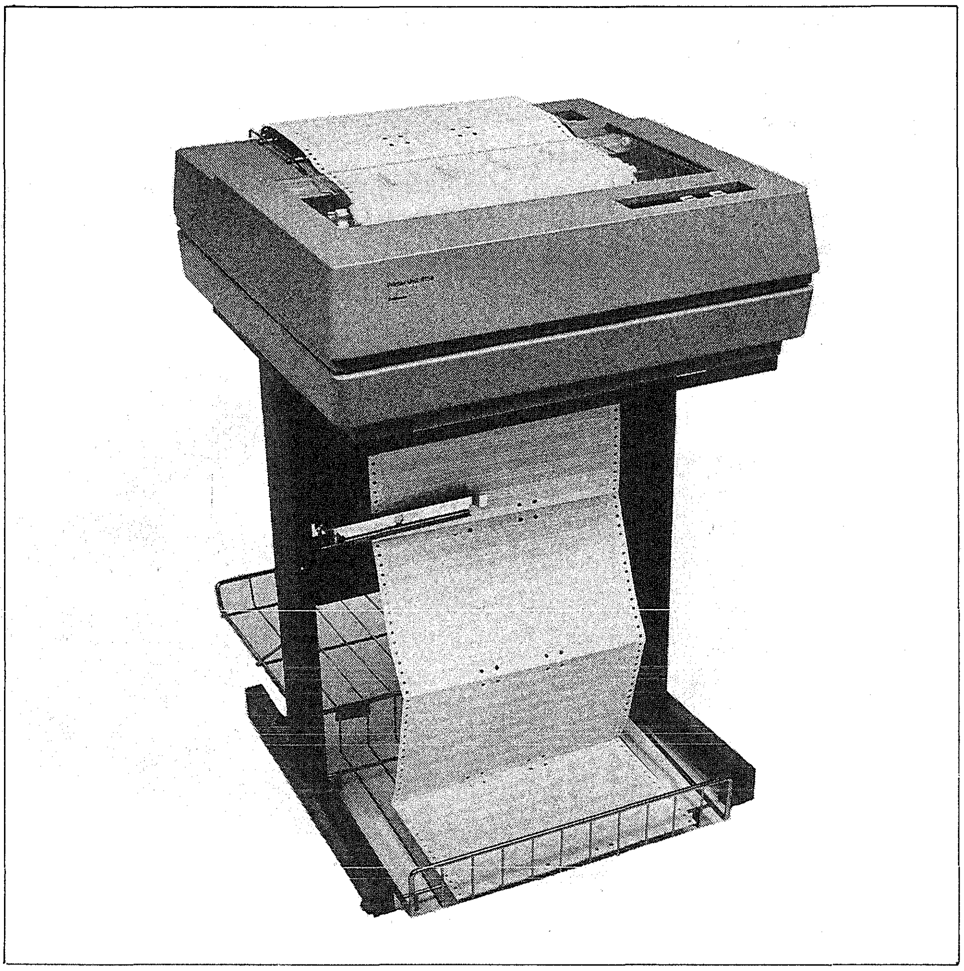


Fig. 12. Printer Unit 4154

Printer Unit 4154 Characteristics

Dot matrix:	9 x 7
Character set:	95 graphic symbols including upper/lower case letters, numerals, special characters and space.
Printing speed:	100–125 lines per minute
Characters per line:	132
Horizontal spacing:	10 characters per inch
Vertical spacing:	6 or 8 lines per inch
Paper feed:	Tractor
Paper format:	Min width, 4 inch (105 mm) Max width, 16 inch (406 mm)
Paper copies:	Top copy plus 4 on 45 g/m^2 printing paper. Max thickness of paper plus carbon, 0.4 mm

Printer Unit 4154 incorporates the form feed function as standard. The vertical tabulation function is an extra feature.

Printer Unit 4153

Printer Unit 4153 is a free-standing desk-top unit. This buffered printer is connected to the display unit, and it can be located at a distance of up to 600 metres from the display unit.

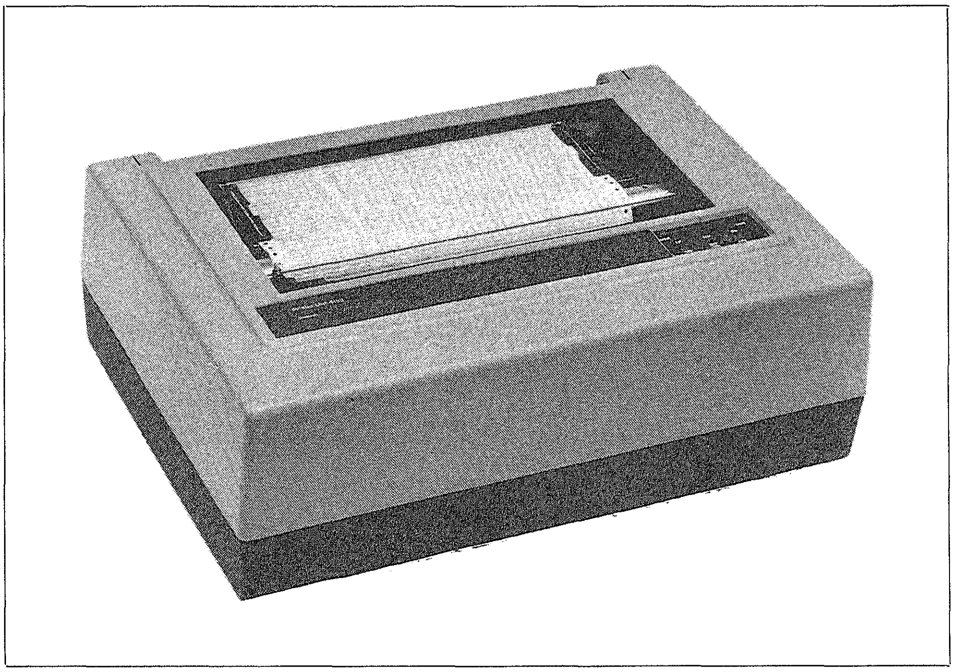


Fig. 13. Printer Unit 4153

Printer Unit 4153 Characteristics

Character structure:	9 x 9 dot matrix
Printing speed:	Up to 250 characters per second 90-130 lines per minute
Characters per line:	Max 154
Character set:	95 graphic symbols including upper/lower case letters, numerals, special characters and space
Horizontal spacing:	10 characters per inch
Vertical spacing:	6 lines per inch
Paper feed:	Tractor
Paper format:	Min width, 2'' (50 mm) Max width, 17.8'' (450 mm)
Paper copies:	Top copy plus 5 on 50 g/m^2 printing paper. Max thickness of paper plus carbon, 0.55 mm.

Printer Unit 4153 incorporates the form feed and vertical tabulation functions as standard.

Flexible Disk Unit 4120

The flexible disk unit, which provides secondary storage for the display system, can be connected to the communication processor.

The flexible disk unit can also be connected to a single display-unit configuration.

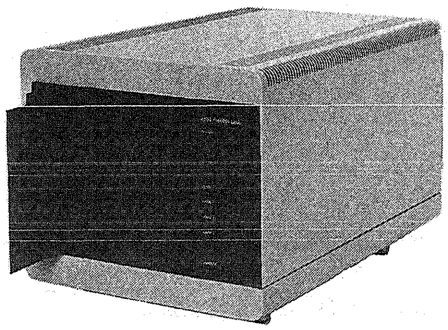


Fig. 14. Flexible Disk Unit 4120

Flexible Disk Unit Characteristics

Number of diskettes:	Max 2
Capacity:	256 kilobytes per diskette with IBM 3740 sectoring (128, 256, 512 or 1024 bytes per sector)
Access time:	Track-to-track, 8 ms Average (including settling), 93 ms Settling time, 10 ms Head load time, 40 ms
Transfer rate:	250 kilobits per sec
Rotational speed:	360 r/min
Recording density:	3408 bpi
Track density:	48 tpi
Tracks:	77 (one side)
Media requirements:	IBM diskette or equivalent

Flexible Disk Unit Operation

The flexible disk unit contains a microcomputer which controls and supervises diskette storage and readout operations.

The flexible disk unit also incorporates a file handling system, thus limiting tasks carried out by the communication processor or display unit to opening and defining files, after which data can be read or stored. Moreover, data can be read from one diskette to another within the flexible disk unit.

Functionally, Flexible Disk Unit 4120 can be subdivided as follows

- Microprocessor
- Memory
- Flexible disk adapter with direct memory access transfer
- Flexible disk drive

The microprocessor is an 8-bit LSI processor.

Memory capacity is 32 kilobytes.

The flexible disk adapter provides direct memory access at a rate of 300 kilobits per second. The flexible disk adapter also includes circuits that provide galvanic isolation from the communication processor or display unit.

As standard, the flexible disk unit is equipped with one flexible disk drive. An extra Flexible Disk Drive 4121 can be inserted into the flexible disk unit.

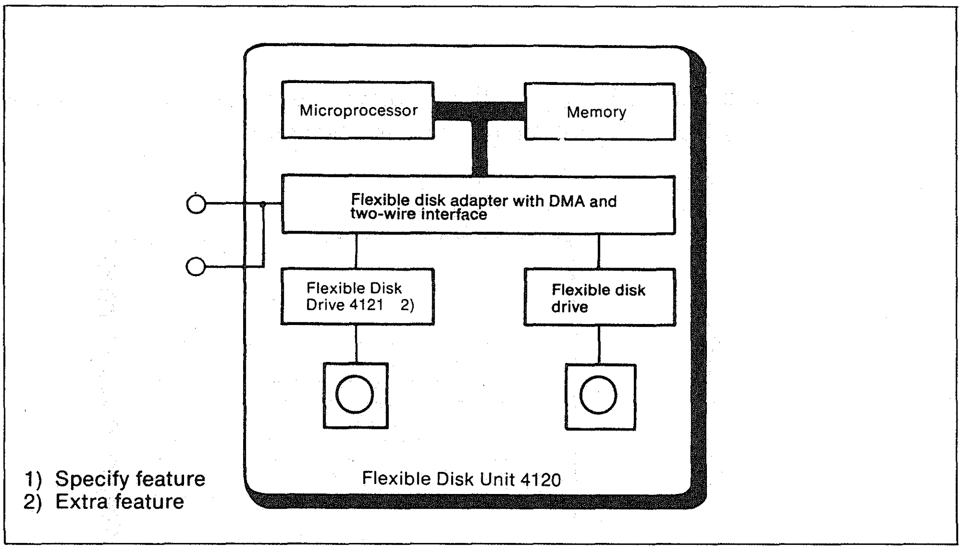


Fig. 15. Block diagram of Flexible Disk Unit 4120

Software

The Alfaskop System 41 software is divided into three main parts

- Operating system
- System software
- Application software

Operating System

All software components operate under the control of, and with the assistance of, an operating system that permits a number of programs, communication lines and terminals to be handled and processed concurrently.

The operating system carries out

- CPU scheduling
- Memory management
- Interrupt-initiated program control
- I/O processing
- Start/restart initialization
- Error detection

The operating system that is used in the terminal and the communication processor consists of modules used by both.

System Software

The terminal-console-functions software, the emulation software, and the Alfaform package are called system software.

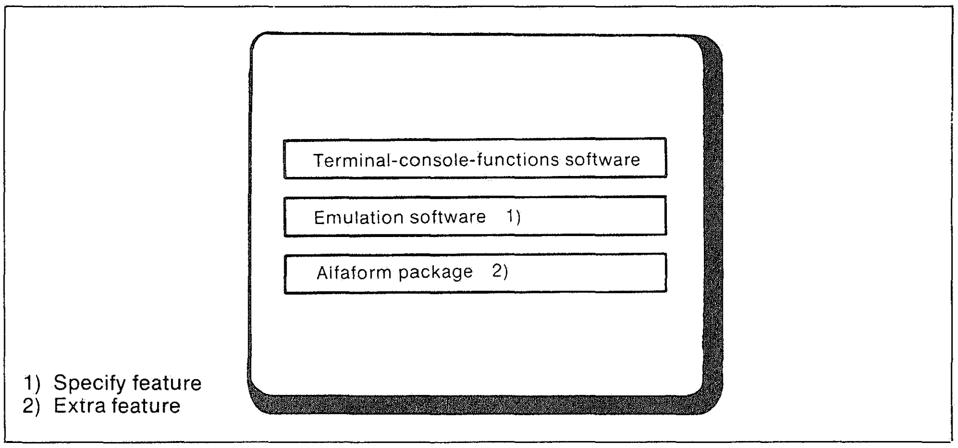


Fig. 16. System software

Terminal-Console-Functions Software

The terminal-console-functions software provides numerous functions such as

- Dump functions
- Test functions
- Internal and external communication line monitoring
- System utility functions (including flexible disk utilities)
- System customizing

To ensure privacy and security, there are five degrees of authorization implemented using combinations of customer and Datasaab passwords.

Emulation Software

The following IBM 3270 system emulations are available with the Alfaskop System 41

- Control Unit 3274 1B and 1D with Display Station 3278, models 1, 2, 3 and 4.

- Control Unit 3274 1C, remote operation BSC, with Display Station 3278, models 1, 2, 3 and 4.

- Control Unit 3274 1C, remote operation SNA/SDLC, with Display Station, models 1, 2, 3 and 4.

Alfaform Package

The Alfaform package is designed to reduce loads on the communication line and host computer by handling part of the processing within the terminals. This is achieved by means of the following

- Forms defined and maintained by terminal user in cooperation with the host computer system. A form is a definition of a display layout which specifies the fixed data and the checks that are to be carried out on entered (variable) data. These checks can be of the following types: numeric/alphanumeric, field length, check digit, max/min value, cross-field, etc.

- Fixed texts stored locally to supplement messages received from the host computer

- Numerous editing and checking functions carried out on data entered via the keyboard before it is sent to the host computer

- Operator training carried out off-line

The Alfaform package which executes these functions has been designed so that they can be implemented with minimum impact on existing application programs in the host computer. Alfaform also permits different terminals within the same cluster configuration to work concurrently on different applications.

Application Software

The application programs include forms defined in Alfaform.

The application programs (user-written) are stored in the flexible disk unit together with the operating system and the system programs.

Contents

Illustrations

Configurations

A single display unit can communicate directly with the host computer. A display unit included in a cluster configuration (comprising a number of display units, printer units or flexible disk units) can communicate with the host computer that serves the entire cluster under the supervision of a communication processor.

A cluster configuration can be connected remotely to the computer via modem lines or connected locally to an IBM System 360/370 or IBM 3030/4300 Series selector, block multiplexer or multiplexer channel.

A maximum of 32 display units and printer units in any combination can be connected to the same communication processor. One display unit, one printer unit and one flexible disk unit can be connected to the same two-wire cable (or coaxial cable). Two or three peripheral units of the same type *cannot* be connected to the communication processor via a single two-wire (or coaxial) cable.

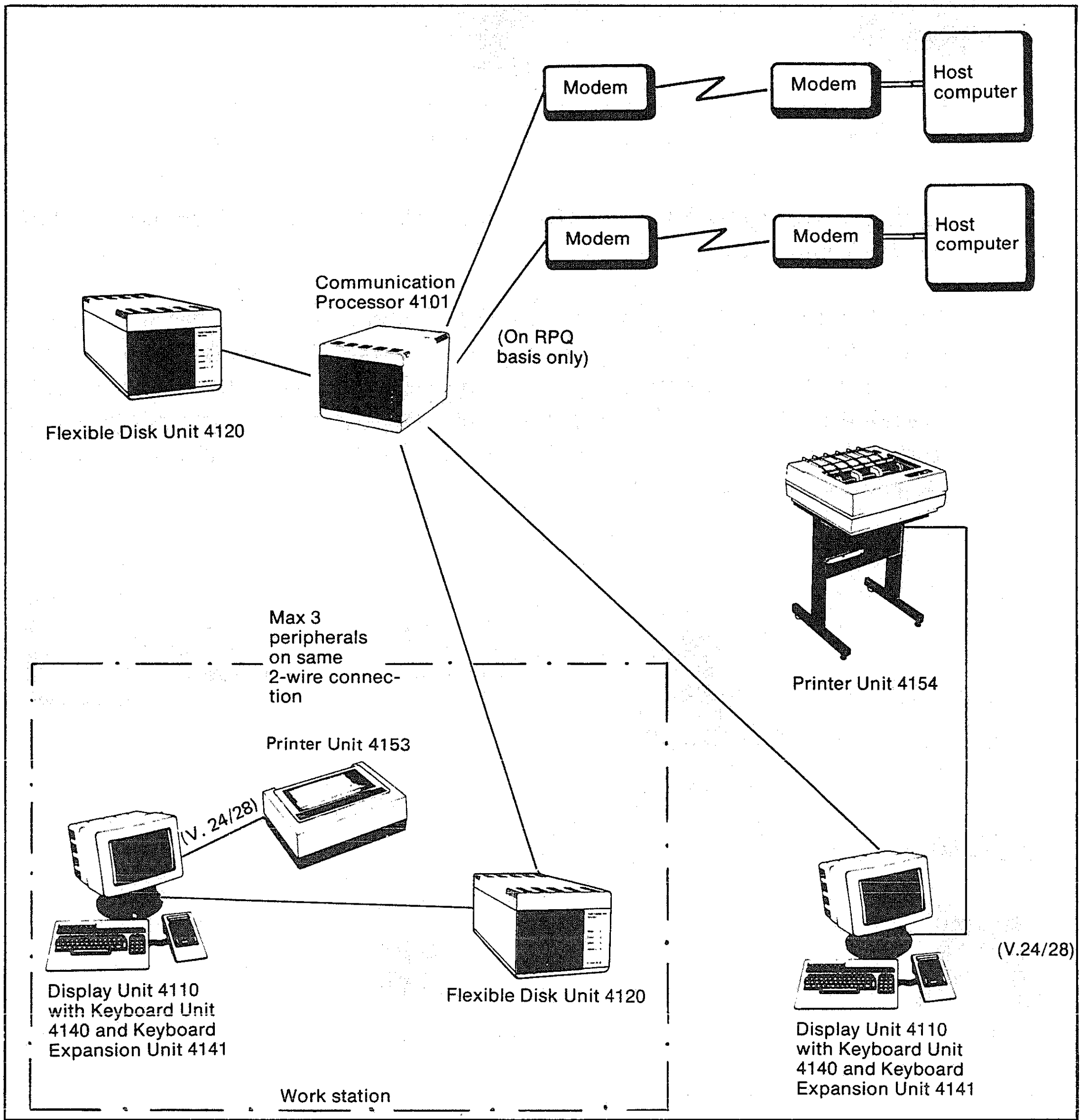


Fig. 1. Cluster configuration connected to remote host computers

The maximum overall length of the cable from the communication processor to the display unit is 1500 m for the two-wire cable and 600 m for the coaxial cable. The printer unit is connected to the display unit via a V24/28 interface. The maximum overall length of the printer cable is 600 m.

The display unit and the flexible disk unit are connected in a chain and the display unit must be at the end of the chain. This is because the flexible disk unit has two connectors, one for the cable to the communication processor and one for the cable to the display unit in the chain. The display unit has only one connector.

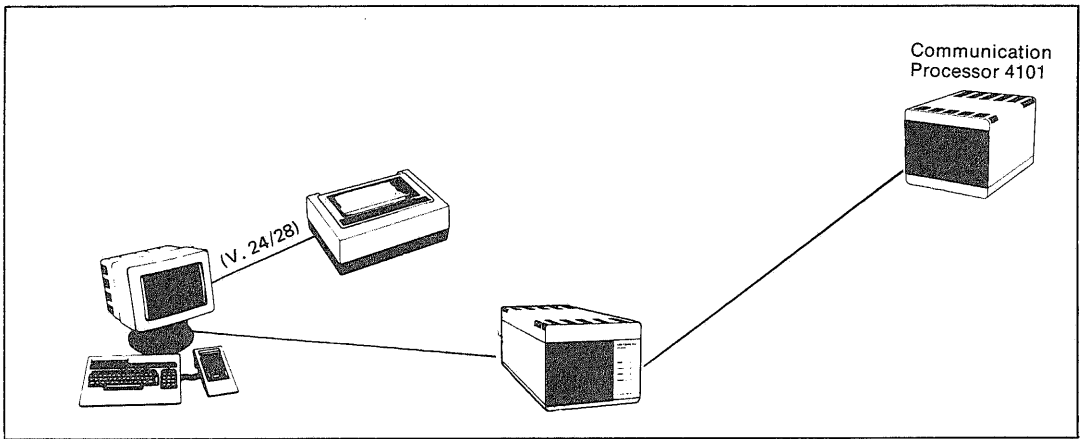


Fig. 2. Chain of peripherals connected to a communication processor

A single display-unit work station can communicate with the host computer via a modem line.

The display unit is connected to the modem. The flexible disk unit is connected to the display unit via a two-wire (or coaxial) cable. The maximum overall length of the two-wire cable is 1500 m and the maximum overall length of the coaxial cable is 600 m. The printer unit is connected to the display unit via a V24/28 interface. The maximum overall length of the printer cable is 600 m.

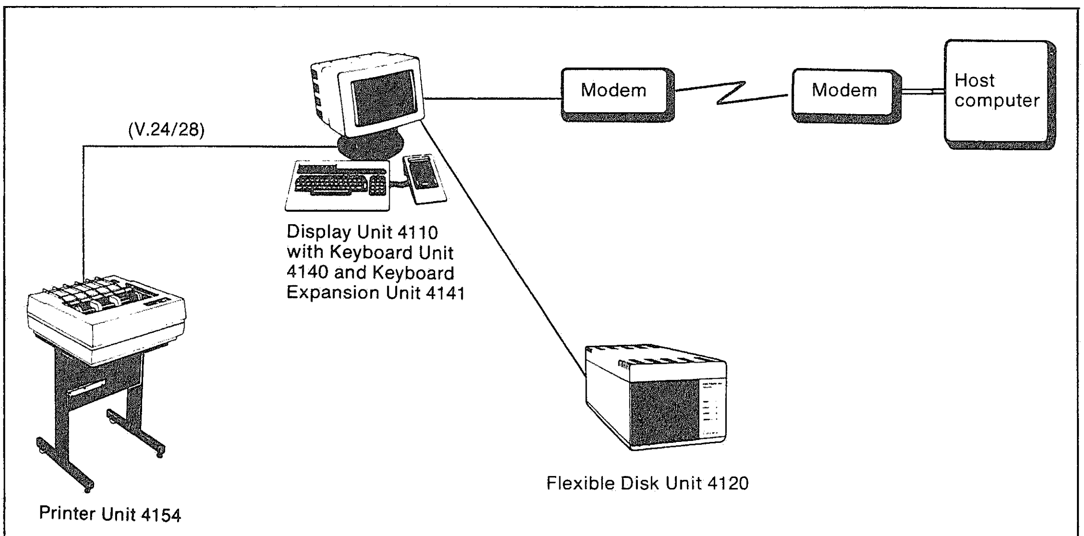


Fig. 3. Example of single display-unit configuration with remote connection to host computer

Examples of Terminal System Configurations

Single Display-unit Configurations

Figs. 4 and 5 show how the system components used in a single display-unit configuration can be arranged.

Cluster Configurations

Figs. 6 and 7 show how the system components used in a remote cluster configuration can be arranged and Figs. 8 and 9 show how the system components used in a local cluster configuration can be arranged.

The sum of the display units and printer units that can be connected to a *single* communication processor must not exceed 32.

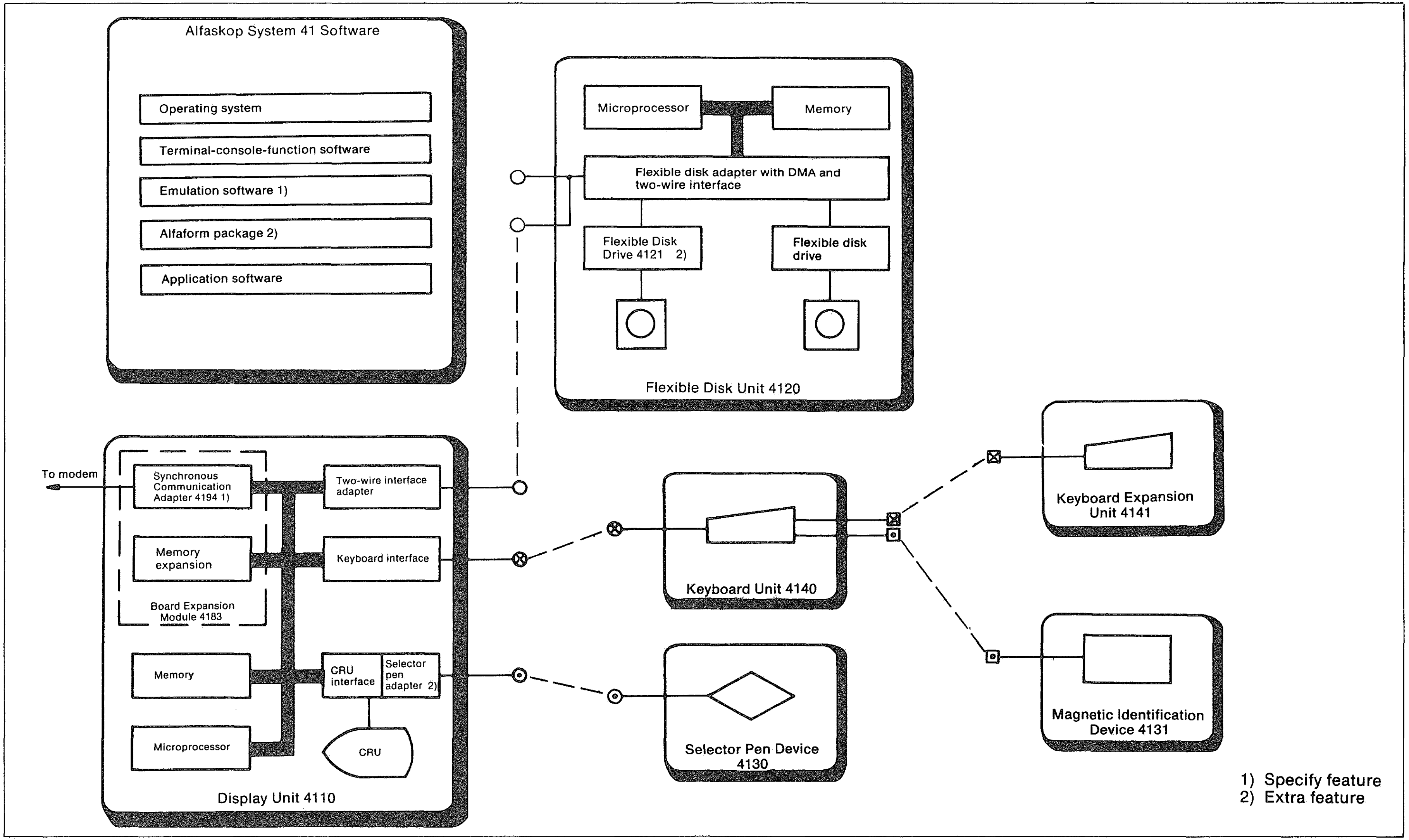


Fig. 4. Single display-unit configuration

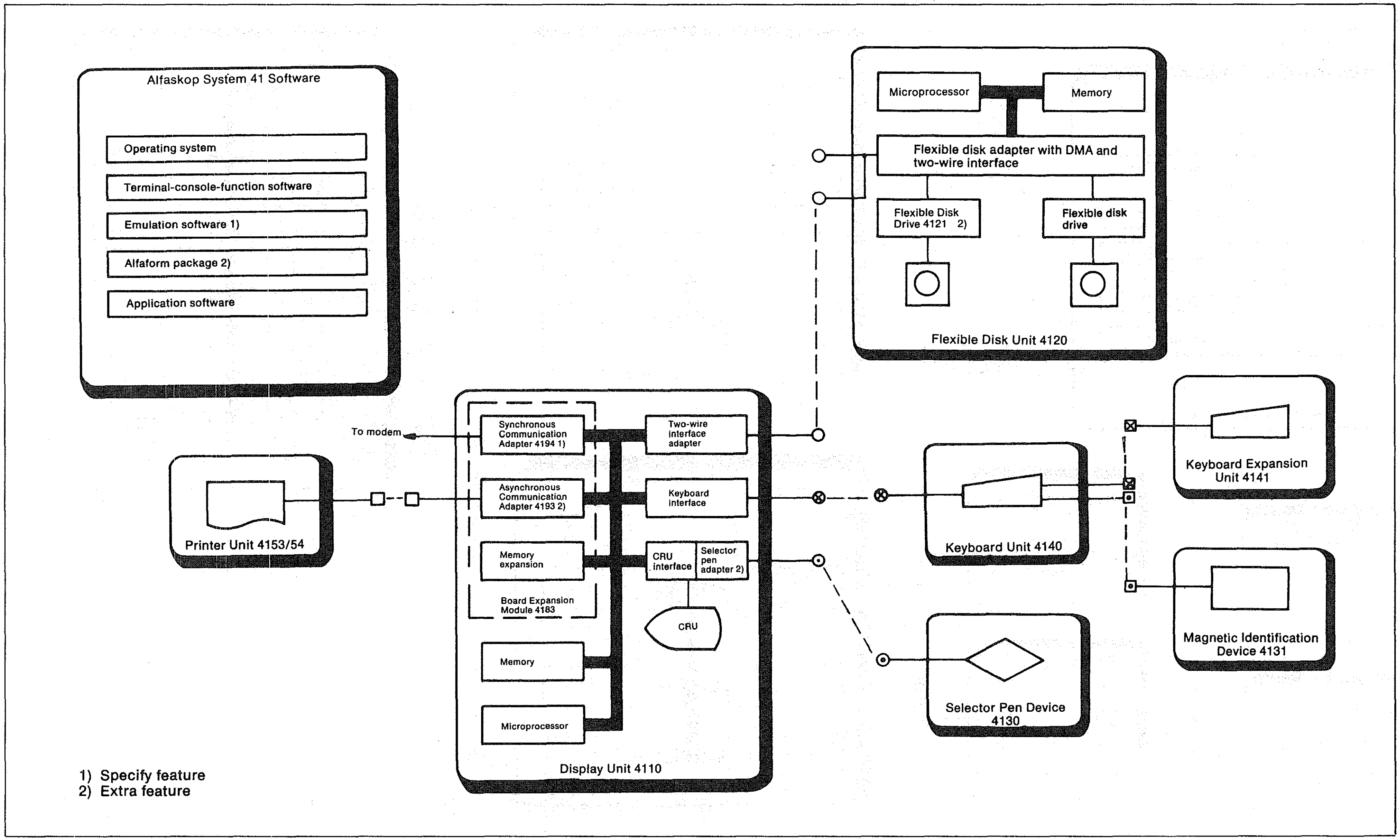


Fig. 5. Single display-unit configuration that includes Printer Unit 4153/54

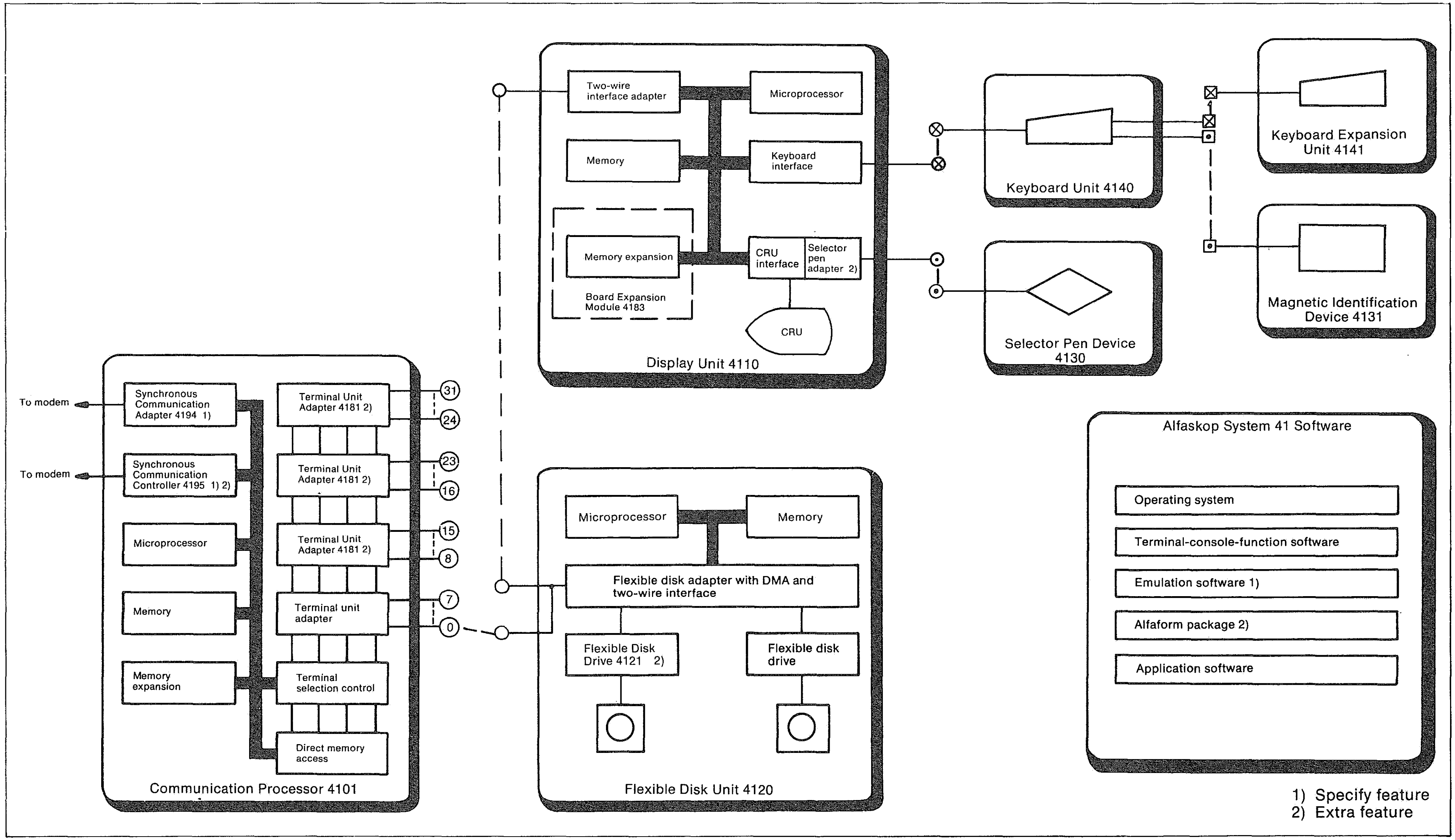


Fig. 6. Remote cluster configuration

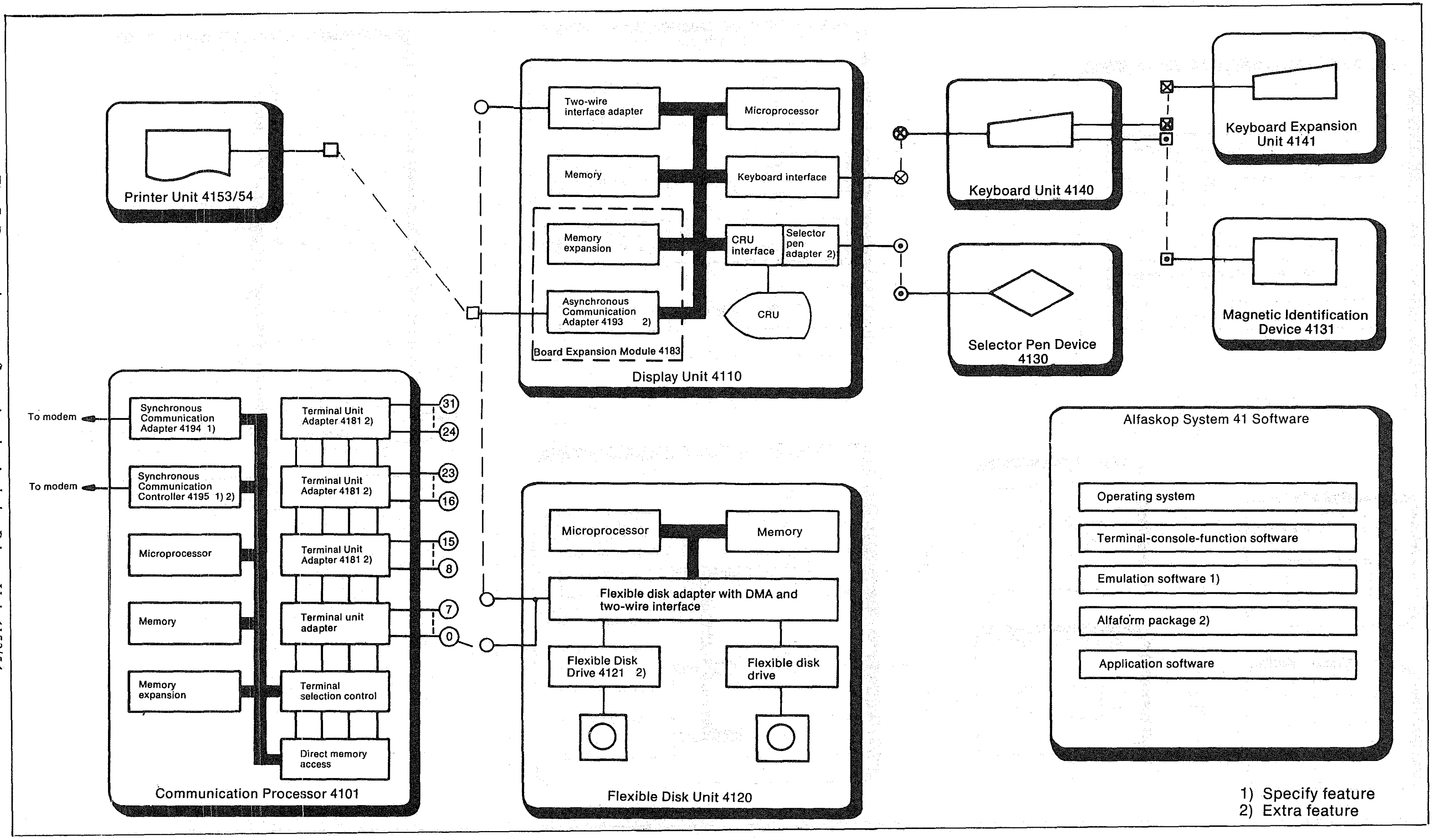


Fig. 7. Remote cluster configuration that includes Printer Unit 4153/54

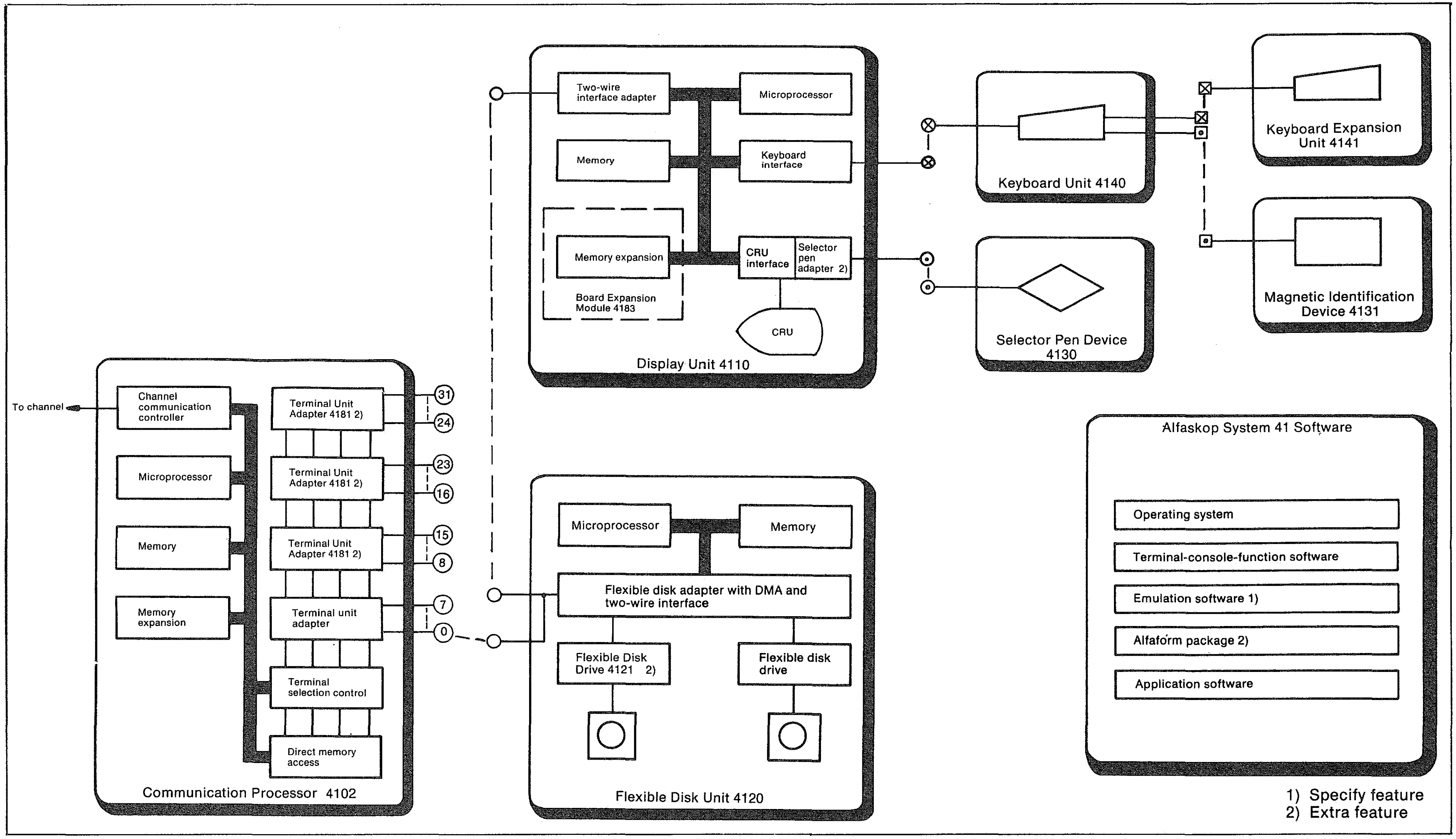


Fig. 8. Local cluster configuration

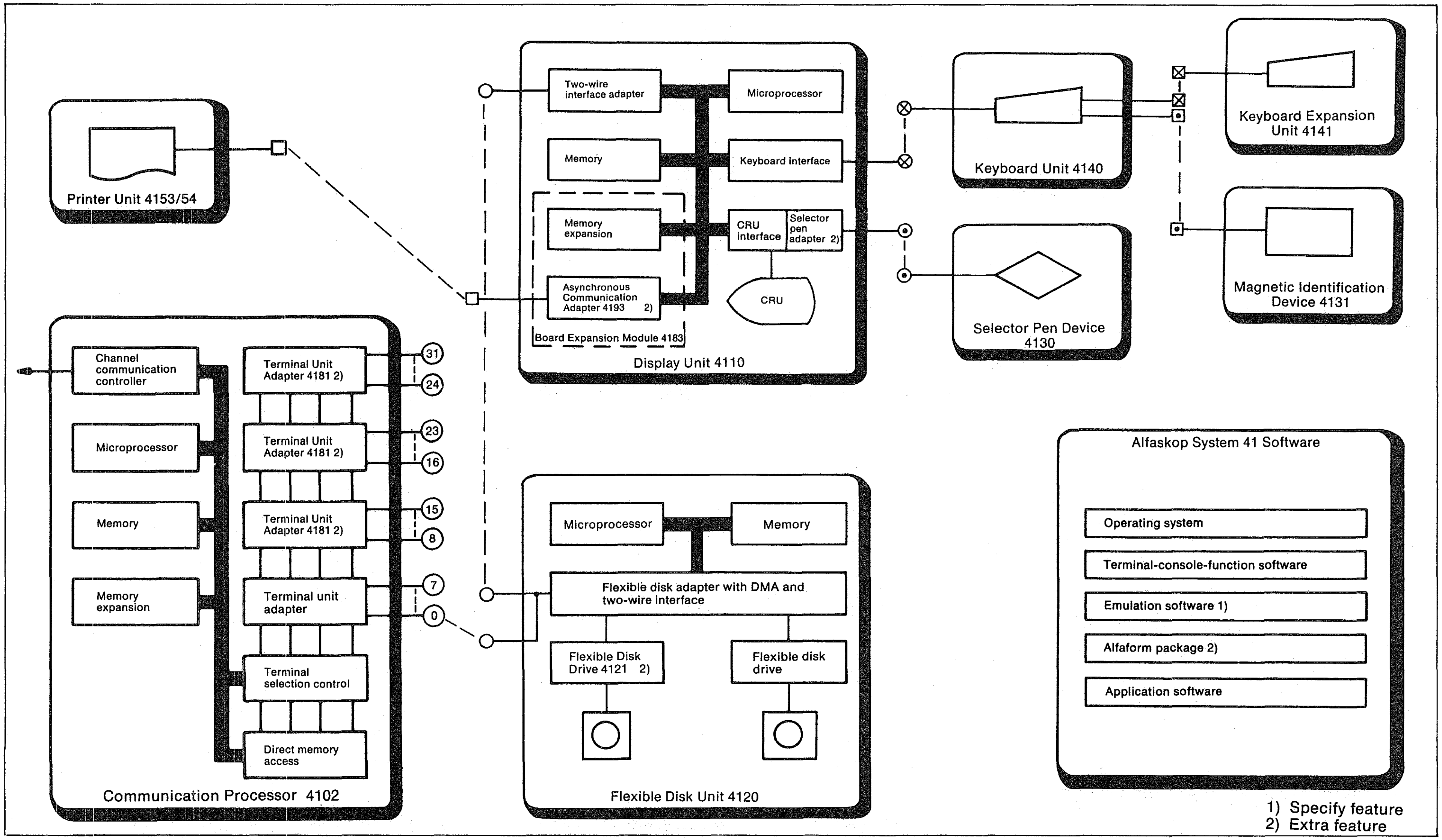


Fig. 9. Local cluster configuration that includes Printer Unit 4153/54

Contents

Operational Characteristics

Display Unit

Presentation

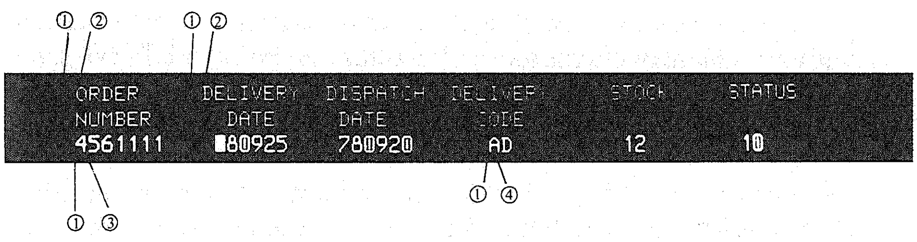
Data stored in the display area is presented on the screen in the form of alphanumeric characters and symbols. When a keyboard is connected, text can be entered from the keyboard and presented on the screen.

The computer can communicate with the operator in two ways

- The computer or Alfaform formats the display on the screen and the operator is thus provided with a form into which he enters the message.

- The screen display is left unformatted by the computer and the operator himself determines how the message will be arranged on the screen.

The display on the screen is formatted by the use of attribute characters to define the various data fields on the screen. A data field can start at any position on the screen. It is established by an attribute character which defines the display area extending forward to the next attribute character. Each attribute character occupies one position and is presented as a space on the screen.



① Attribute character. Not displayed.
② Protected field. Displayed at normal brightness.
③ Unprotected field. Numeric entries only. High brightness.
④ Unprotected field. Alphanumeric entries. High brightness.

Fig. 1. Example of display formatting

The attribute character contains information indicating

- Whether the field is protected or unprotected from operator entry

- The type of entries which the operator can make into unprotected fields
  - Numeric entries only
  - Alphanumeric entries

- Which one of the following brightness levels is called for
  - Off (field not displayed)
  - Normal
  - High

- Whether or not the field is selector-pen-detectable

- Whether or not the field has been modified by the operator using the MDT bit.

All positions on the screen can be addressed individually, and the computer can thus write at any character position. Position addressing can be repeated within a message so that data can be written at different parts of the screen without having to transmit all the text on the screen.

Cursor

A special symbol, called a *cursor,* is displayed on the display screen to indicate where the next character entered from the keyboard will be stored.

The cursor is automatically generated by the terminal system and may appear as an underscore, as a flashing underscore, or as a rectangular or flashing rectangular symbol imposed over a character. The character within the rectangular cursor remains visible (inverse video). The operator can change the cursor from an underscore to a rectangular symbol, or vice versa, by pressing the alternate cursor (CU AL) key. The same operator can cause either type of cursor to flash by using the cursor blink (CU BL) key. When the cursor is displayed under one character in a line of characters, the character can be changed or deleted by keyboard action. Also, if the cursor is displayed under (or within) an unprotected position without a display character, a character can be entered in that position by keyboard action. When a character is entered from the keyboard, the cursor moves one position to the right.

The cursor is not affected by keyboard locking but is extinguished for part of the time during which the display unit communicates with the computer.

The cursor is nondestructive and can be advanced, backspaced, or moved up or down without erasing the characters it passes. If a character is entered in the last position of a line, the cursor advances to the first position on the next line, or if the line is the last line of the screen the cursor is moved to the first position on the top line. If the backspace key is used when the cursor is in the first position of the top line, the cursor will appear in the last position of the bottom text line (automatic wraparound). When data is transmitted to or from the computer, the cursor is not affected unless the computer data contains an order calling for cursor movement.

Automatic Skip

When formatted presentation is used and the operator enters a character in the last position of an unprotected field, an automatic cursor movement determined by the succeeding attribute character will be performed in the following manner

- If the attribute character specifies the next field as numeric and protected, the cursor is automatically positioned at the first character position in the next unprotected field.

- For other attribute characters, the cursor moves to the position following the attribute character, i.e. the first character position in that field.

Controls and Indicators

The display unit is provided with the following controls and indicators

- Power on/off switch with indicator that lights up when display unit power is switched on

- Brightness control for regulating display brightness

- Contrast control for regulating the difference between text fields having high and normal brightness

- Reset switch. Should not be used by the operator

- Different texts are displayed on message line, e.g. while the program is being loaded into the display unit.

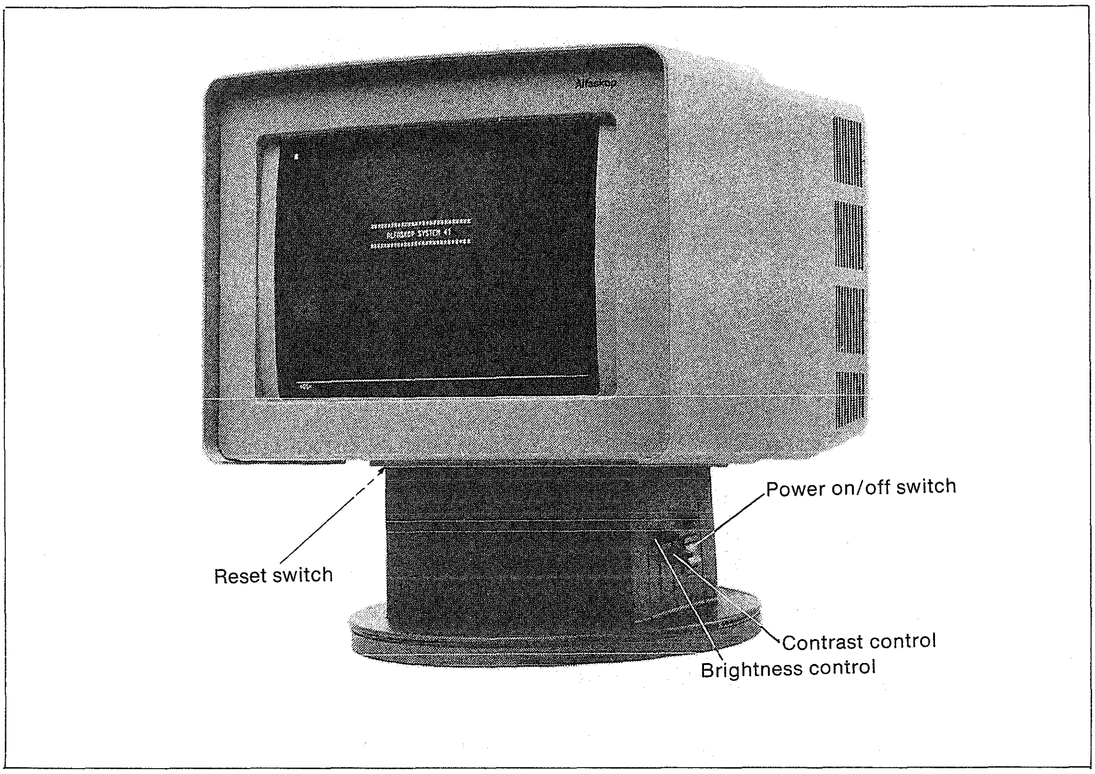


Fig. 2. Display unit controls and indicators

An indicator panel located at the top of the keyboard informs the operator of the terminal status. This panel contains the following indicator lamps (in all versions of the keyboard)

• SYSTEM READY	Indicates readiness for traffic with the computer
• USM	Indicates that the computer has an unsolicited message for the operator
• INSERT MODE	Indicates that the INS MODE key has been depressed, i.e. that the display unit is operating in the insert mode
• SHIFT	Indicates that the keyboard is in the upper shift mode, i.e. shift key is depressed or SHIFT LOCK key has been depressed
• NUMERIC	Indicates that the cursor is in a numeric entry field.
• I/O ERROR	Indicates an error in an external unit (a printer for example)
• KB LOCK	Indicates that the keyboard is locked
• ID	Used only when the magnetic identification device is included and in such case it indicates that the ID-card has been read correctly

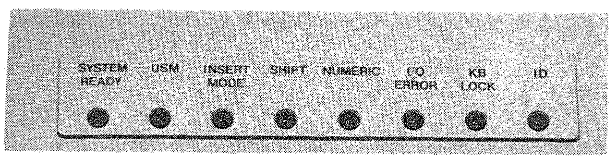


Fig. 3. Keyboard indicator panel

Three controls are located on the keyboard

- Audible alarm volume control
- Audible alarm pitch control
- Click volume control for regulating the volume of the clicking sound that acknowledges keyed entries

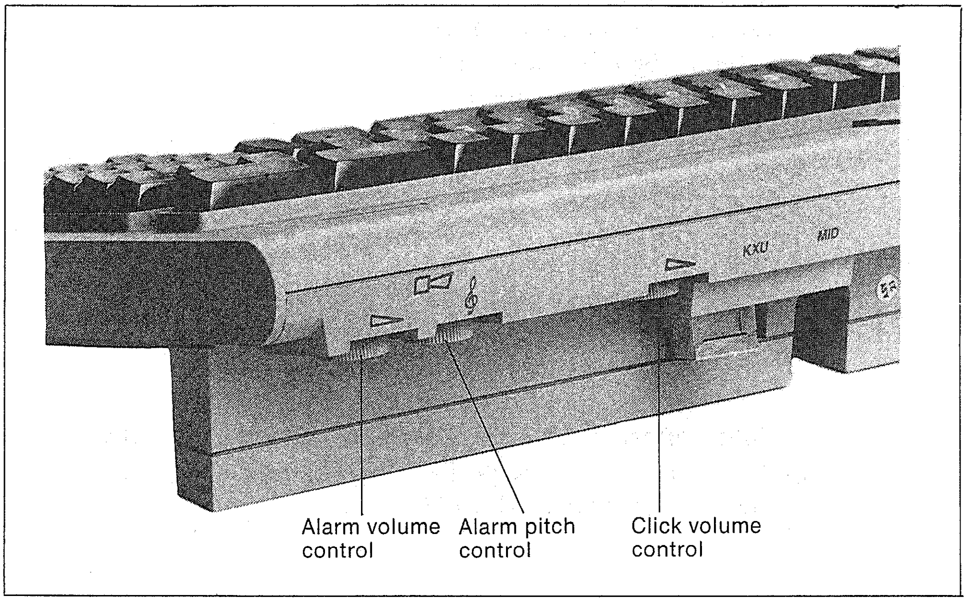


Fig. 4. Keyboard controls

Message Line

The bottom line on the screen is used to present detailed information about the status of the terminal system.

Appendix 4 presents the messages that can appear on the message line.

Typewriter Keyboard Keys

Functionally, the keyboard keys can be subdivided as follows

- Alphanumeric keys for
  - Letters, numerals and symbols

- Edit keys for
  - Cursor control
  - Tabulation
  - New line
  - Erasing
  - Insertion or deletion of data
  - Cursor to message line
  - Cursor blink
  - Alternate cursor

- Operational keys for
  - Cancellation of transmission requests and keyboard locking
  - Reception of unsolicited message
  - Rolling the message line
  - Cancellation of print requests
  - Definition of printer for local printout
  - Initiation of hard copy printing

- Program attention keys for
  - Initiation of data transmission
  - Selector pen detection function
  - Clear the display screen
  - Test requests
  - Program access
  - Program function

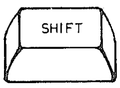
- Keyboard expansion unit with
  - Program function keys
  - 10-key numeric pad, +, –, . and * keys
  - Tabulation, enter and space keys

All alphanumeric, special symbol, and move cursor keys have typamatic capability. If the key is kept depressed more than 0.5 s, the key function is repeated at a frequency of 10 Hz.

Alphanumeric Keys

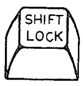
The keys in the large keybank at left are used to enter alphanumeric characters including special symbols. Their functions are tabulated below.

Shift



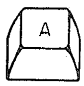
The shift keys function in the same way as on an ordinary typewriter, thus enabling the operator to select either upper or lower case characters.

Shift lock



Locks the shift key at its depressed position. This lock is cancelled by depressing the SHIFT key.

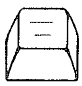
Character



One of two modes can be selected inside the terminal. In the dual case mode, lower case characters are displayed when the shift key is up and upper case characters when the shift key is down. In the mono case mode, all alphabetic keys provide upper case characters regardless of the shift key.

The character is entered at the cursor position after which the cursor is moved forward to the next position.

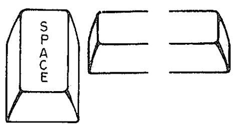
Minus-sign/
Underline



When the shift key is up this key enters the minus sign (–).

When the shift key is down this key instantly underlines a full word by entering a space at the cursor position and underlining the succeeding characters as far as the next space.

Space

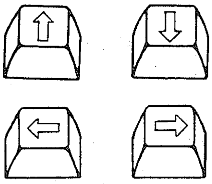


The keyboard unit has one space key and the keyboard expansion unit has one space key. Both function identically, causing the cursor to be moved one position to the right and any character located at the cursor's original position to be erased – thus leaving a blank space.

Edit Keys

The edit keys carry out keyboard editing functions. Operation of these keys does not affect the MDT (modified data tag) bit. These keys and their functions are tabulated below.

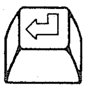
Cursor control



Moves the cursor one character position in the direction indicated by the arrow on the key. The cursor may be moved into any character location, including unprotected and protected numeric and alphanumeric character and attribute character locations, through the use of these keys.

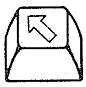
The cursor can wrap around in response to these keys. Horizontal wraparound always involves vertical movement. The cursor repositions to the next or preceding line (or from the bottom to the top line or vice versa). Vertical wraparound in response to the ↑ or ↓ key involves no horizontal movement. The cursor stays in the same character column.

New line



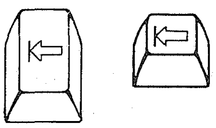
Moves the cursor to the first unprotected position on the next line. If the next line contains no unprotected field, subsequent lines are searched and wraparound takes place at the last line. If no unprotected field is found on the screen, the cursor moves to the first position in the first line.

Home



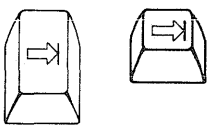
Moves the cursor to the first position in the first unprotected field on the screen. If no unprotected field is found on the screen, the cursor is moved to the first position in the first line.

Back tab



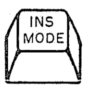
Moves the cursor to the previous start position in an unprotected field. If no unprotected field or no attribute character is found, the cursor is moved to the first position in the first line.

Tab



Moves the cursor to the first position in the next unprotected field. If no unprotected field or no attribute character is found, the cursor is moved to the first position in the first line.

Insert mode



When this key is depressed, the terminal goes into the insert mode, which is indicated by a lamp on the keyboard. Any alphanumeric character that is then entered in an unprotected field will appear at the cursor position and if, before entry, this position contained a character, it will be moved one position to the right together with all subsequent characters in the field. As a result, the new character will have been inserted between two characters previously present on the screen. If the display is formatted, the attribute character is flagged to indicate that the content of the field has been changed by the operator.

Insert mode operations can only be carried out in unprotected fields. Moreover, a null character must be present either at the cursor position or to the right of it within the same field. If all character positions are occupied (no remaining null characters), the operation is inhibited and an alarm is issued to the operator.

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

Operation of the RESET key, ENTÉR key, or any other host communication initiating key returns the keyboard to normal mode. (Operation of the magnetic identification device, the selector pen, or the CU SE (cursor select) key also returns the keyboard to normal mode.)

Delete

Erases the character at the cursor position in an unprotected field and moves subsequent characters in the same field and on the same line one position to the left. Vacated character locations at the end of the line will be filled with nulls. If the display is formatted, the attribute character is flagged to indicate that the content of the field has been changed by the operator. If the unprotected field encompasses more than one line, characters in lines other than the line identified by the cursor will not be affected.

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

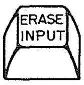
Erase to end of field

When the shift key is up this key erases all characters starting at the cursor position and extending to the last position in an unprotected field. The cursor position is not changed. The attribute character is flagged to indicate that the field has been changed by the operator. If the field is on more than one line, this operation wraps around from line to line.

Operation of this key when the cursor is located in an attribute character location or is within a protected data field generates an audible alarm. No character locations are cleared, the cursor is not moved, and the MDT bit is not set.

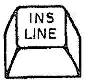
When the shift key is down this key enters the host computer generated printer authorization matrix into the communication processor.

Erase input



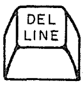
Erases all unprotected fields on the screen and moves the cursor to the first unprotected position on the screen. If there is no unprotected field on the screen, the cursor is moved to the first position on the first line. If the display is formatted, all MDT bits for unprotected fields are reset to indicate that the content of the field has not been modified by the operator.

Insert line



Moves the line on which the cursor is positioned and all following lines down one line, thus providing a blank line at the cursor position. The content of the last line is lost. If the display is formatted, the last attribute character before the cursor position line is flagged to indicate that the content of the field has been changed by the operator. No insert line operation is carried out if the line on which the cursor is located, or any line below it, contains an attribute character. An alarm is issued to the operator if this function cannot be carried out.

Delete line



Deletes the line at the cursor position and moves all following lines up one line, thus providing a blank bottom line. If the display is formatted, the last attribute character before the deleted line is flagged to indicate that the content of the field has been changed by the operator. No delete line operation is carried out if the line on which the cursor is located or any line below contains an attribute character. An alarm is issued to the operator if this function cannot be carried out.

Cursor-blink/
Cursor-alternate

When this key is depressed and the shift key is up (CU BL), the cursor (either the underscore or the filled rectangular cursor) will change from steady glow to flashing and vice versa.

When this key is depressed and the shift key is down (CU AL), the cursor display is changed, i.e. the rectangular cursor is changed to an underscore and vice versa.

Cursor to
message line

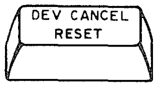
When this key is depressed, the cursor is moved to the first unprotected position on the message line.

If it is depressed again, the cursor is moved back to its original position.

Operational Keys

The functions of the operational keys are tabulated below.

Reset/Device cancel

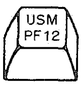


When this key is depressed and the shift key is up (RESET), it will inhibit initiated, but not yet executed, data transmission. When this key is depressed, the keyboard will be unlocked if data transmission is not in progress.

When this key is depressed and the shift key is down (DEV CANCEL), it will cancel a current outstanding print request to a printer if the printer is busy or inoperable. A request initiated by the PRINT key is dequeued and the keyboard unlocked.

The DEV CANCEL key also switches off the I/O ERROR lamp if it is lighted. This key has no function while a command is being executed.

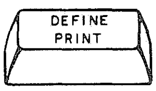
USM



When the computer has an unsolicited message to send to the display unit, the USM lamp on the keyboard is lit. Depressing this key, when the shift key is down, indicates to the computer that the operator is ready to receive the message. The lamp is switched off when the indication is sent to the computer.

For information about what happens when the shift key is up (PF12) see section on "Program Attention Keys".

Print/Define



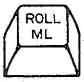
When the shift key is up (PRINT), this key is used to initiate local printout of the display image. Printout always starts at the first position on the first line and ends at the last position on the last line of the screen. Spaces or unoccupied character positions at the end of each line are suppressed in order to increase effective printing speed. Attribute characters are converted to spaces and the cursor is not printed out. An initiated, but not yet executed, printout request can be inhibited using the DEV CANCEL key.

When this key is depressed and the shift key is down (DEFINE), the cursor is moved to indicate a 2-position field on the message line. The operator enters the printer number or printer class into these positions and depresses the ENTER key.

If the specified printer is not authorized (that is, the matrix does not permit the display to copy to the selected device or class of devices), an error indication is displayed on the message line.

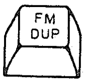
If the selected printer class or printer is valid and authorized for this display, the connection indicator will change to indicate the new connection, and the print ID mode is terminated. The cursor reappears in its original position.

Roll message line



When this key is depressed the content of the message line is changed. There are three alternative message line contents, one comprising information obtained from the operating system, one comprising information obtained from emulation and one comprising information obtained from Alfaform.

Duplicate/ Field-mark



When the shift key is up (DUP), this key enters a unique character that is represented on the screen by an asterisk with an overbar. Moreover, tabulation to the first position in the next unprotected field takes place, and the MDT bit is set to 1. The DUP character provides a means of informing the application program that a "duplicate" operation is indicated for the rest of the current field. The DUP character is transferred as a DUP code when the data is read from the display unit to the program. No duplicate operation is performed by the Alfaskop system.

When this key is depressed and the cursor is located in an attribute character location or is within a protected data field, an audible alarm is generated. No character locations are changed, the cursor is not moved, and the MDT bit is not set.

When the shift key is down (FM), this key enters a unique character that is represented on the screen by a semicolon with an overbar, and the MDT bit is set to 1. The field mark character provides a means of informing the application program of the end of a field in an unformatted buffer or subfield in a formatted buffer. The field mark character is transferred as an FM code when the data is sent from the display unit to the host computer.

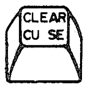
When this key is depressed and the cursor is located in an attribute character location, or is within a protected data field, an audible alarm is generated. No character locations are changed, the cursor is not moved, and the MDT bit is not set.

Program Attention Keys

These keys solicit program action by evoking I/O Pending at the display terminal.

The functions of the program attention keys are tabulated below.

Cursor-select/ Clear



When this key is depressed and the shift key is up a selector-pen-detection function will be carried out. The CU SE key can be used on any field defined as a selector-pen-detectable field (as desribed in the section on "Selector Pen Device"). However, a cursor-select field does not impose the space or null character padding constraints associated with the selector-pen-detectable field. A cursor-select operation can be carried out within a field that is on a different line from that of the attribute character that describes the field.

Cursor select operations may be immediate or deferred (as defined for selector-pen-detectable fields).

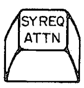
The field used for a cursor-select operation can also have the following format

- Attribute character as defined for selector pen
- Designator character as defined for selector pen
- Data character(s), optional
- Attribute character for next field

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

When this key is depressed with the shift key down (CLEAR), the entire screen is erased including attribute characters and protected fields and the cursor is moved to the first position in the first line. The computer is simultaneously notified by means of a clear sequence that the entire screen has been erased.

Attention/ System-request

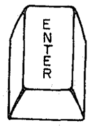


The ATTN key is inoperative in BSC. In SNA/SDLC the ATTN key is used to initiate sending of the Signal command. Further information is presented in the chapter on Remote Operation – SNA/SDLC.

The SYREQ key is in BSC used to send a test request to the host computer which can initiate a test program that checks the terminal functions. Before depressing the SYREQ key, the operator normally has to enter data in a predefined format on the screen. The format is determined by the access method being used in the computer.

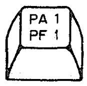
The SYREQ key is in SNA/SDLC used to transfer device ownership between sessions. Further information is presented in the chapter on Remote Operation SNA/SDLC.

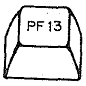
Enter



This key initiates transmission to the computer. An initiated, but not yet executed, transmission can be inhibited using the RESET key.

Program function and Program access keys





There are a total of 24 program function keys designated PF1 through PF24 on the keyboard unit and the keyboard expansion unit. Functions PF1 through PF12 are located on the keyboard unit and PF13 through PF24 on the keyboard expansion unit. Keys PF11 and PF13 through P24 are independent of the shift key.

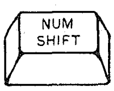
When a program function key is depressed, transmission to the computer is initiated. Information about which program function key was depressed is sent along with the transmitted text.

When the shift key is down, the operator has access to the 10 program access keys designated PA1 through PA10. When any of the program access keys is depressed, transmission to the computer is initiated. The message that is sent contains only information specifying the program access key that was depressed.

Data Entry Keyboard Keys

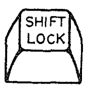
The following additional keys and key functions are provided on the data entry keyboard

Numeric shift



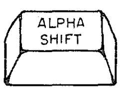
The NUM SHIFT key functions in about the same way as on an ordinary typewriter (upper character on keytop is entered when the NUM SHIFT key is depressed).
Note: This key disables the numeric lock function.

Shift lock



Locks the NUM SHIFT key at its depressed position. This lock is cancelled by depressing the NUM SHIFT or the ALPHA SHIFT key. This key does not disable the numeric lock function.

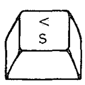
Alpha shift



When this key is depressed, the lower character on the keytop (Q, W, U, J for example) can be entered into a numeric entry field. (Numeric lock function disabled.)

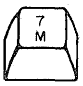
For edit, operational and program attention keys, this key functions as an ordinary shift key, irrespective of cursor position.

Character



When no shift key is depressed, S is entered at the cursor position, after which the cursor is moved to the next position. When the NUM SHIFT key is depressed, < is entered.

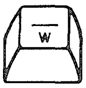
Alphanumeric character



When the cursor is in an alphanumeric entry field, and the NUM SHIFT key is up, M is entered. When the cursor is in a numeric entry field, and the ALPHA SHIFT key is up, 7 is entered. When the NUMERIC lamp is lighted, it indicates that the cursor is in a numeric entry field.
The characters defined as numerics are

- Digits 0–9, minus sign and duplicate character DUP (valid for all keyboard versions)
- Period character (valid for Swedish/Finnish, English and North American keyboard versions)
- Comma character (valid for Danish, Norwegian, Austrian/German, Spanish, Belgian and French keyboard versions)

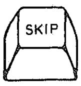
W/ Underline



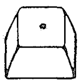
When the shift keys are up, a W is entered.
(French and Belgian keyboards have a Q in this position.)

When the NUM SHIFT key is depressed, this key instantly underlines a full word by entering a space at the cursor position and underlining succeeding characters as far as the next space.

Skip



Performs the same function as the forward tab keys.



When this key is depressed on Danish, Norwegian, Austrian/German, Spanish, Belgian and French keyboards, a comma will be entered.

The rest of the keys on the data entry keyboard have the same functions as corresponding keys on the typewriter keyboard.

10-Key Numeric Keybank

To enable the operator to enter numeric information more easily, a separate 10-key numeric keybank is provided on the keyboard expansion unit.

Numeric Lock

Whether or not the numeric lock feature is to be implemented is determined at customizing time. This feature is controlled by the attribute character and facilitates entry of exclusively numeric data. If the operator tries to enter an alphanumeric character into a numeric field on a terminal selected for numeric lock, the input is inhibited and an alarm is issued. The operator can then continue the input operation immediately by entering the correct numeric character. The characters defined as numerics are

- Digits 0–9, minus sign and duplicate character DUP (valid for all keyboard versions)

- Period (valid for Swedish/Finnish, English (GB) and North American (US) keyboard versions)

- Comma (valid for Danish, Norwegian, Austrian/German, Spanish, Belgian and French keyboard versions)

Keyboard Locking

The keyboard is locked on the following occasions

- During reception of a message from the computer

- When any of the program attention keys has been depressed, until transmission is completed by the computer

- When a Write command is completed without a keyboard unlock order being issued

- When a transmission is initiated by the selector pen or the magnetic identification device, until the transmission is completed by the computer

- When the PRINT key has been depressed, until data has been transferred from the display buffer to the print buffer or until the printout request has been cancelled

Alarm

The alarm function can be triggered either locally within the terminal or from the computer. An alarm signal is obtained

- In response to a special alarm order from the computer

- When an attempt is made to enter non-numeric information into a numeric field

- When an attempt is made to enter information into a protected field or an attribute character location

- When a key is depressed while the keyboard is locked

Magnetic Identification Device

The magnetic identification device reads magnetically encoded ID-cards. The ID-card is inserted into the identification device manually, and when the card has been read correctly the ID lamp on the keyboard lights.

When the card is removed from the identification device, the computer is informed that the previously sent ID code is no longer valid.

The ID-cards are encoded in accordance with ISO 3554. The characters (max. 40) are stored on track 2. Each character consists of a parity check bit (odd parity) and four data bits. Only the characters shown below are used.

On the card, the start character (1011) is followed by up to 37 ID characters, and these are followed by a stop character (1111) and an LRC character. The LRC calculation embraces all characters including the start and stop characters. There must be an even number of ones (total) at each bit position. The LRC character itself has odd parity.

Character on ID-card						
Bits					Character	I/O Interface
P	2^3	2^2	2^1	2^0		EBCDIC
1	0	0	0	0	0	F0
0	0	0	0	1	1	F1
0	0	0	1	0	2	F2
1	0	0	1	1	3	F3
0	0	1	0	0	4	F4
1	0	1	0	1	5	F5
1	0	1	1	0	6	F6
0	0	1	1	1	7	F7
0	1	0	0	0	8	F8
1	1	0	0	1	9	F9
0	1	0	1	1	Start	7B
0	1	1	0	1	Field separator	7D
1	1	1	1	1	Stop	7F

Selector Pen Device

The selector pen can be used to

- Select selector-pen-detectable data fields on the screen
- Initiate a transmission request

These two functions make it possible for the operator to use the display unit without using a keyboard. In situations where the pen is used together with the keyboard, transmission requests can be initiated either with the pen or from the keyboard.

A selector-pen-detectable data field is defined by means of an attribute character. The data field must comprise at least three characters including the attribute character. If the attribute character is not the first character on the line, it must be preceded by at least three space or null characters. The data field is selector-pen-detectable only to the end of the line. If the field extends beyond one line, only those characters on the same line as the attribute character can be detected by the pen. A maximum of 256 selector-pen-detectable fields are permitted.

The selector pen is activated when the operator actuates a switch on the tip of the pen by pressing the pen against the screen, thereby indicating the selector-pen-detectable data fields on the screen with horizontal lines and high brightness. When the tip of the pen is held close to the screen on a selector-pen-detectable field, a field-found indication is obtained. The field in question is identified by the program and the action that is taken is determined by the character following the attribute character. Only one field detection can be made each time the pen is activated. A return to the normal display mode takes place when a field-found indication is obtained or when the pen switch is released after no field has been found.

Selector-pen-detectable fields that are to be used for selection operations and thus not to initiate transmission requests are defined by displaying a ? immediately after the attribute character. When a field-found indication is obtained for a particular field, the ? for this particular field is changed to a >, and the attribute character is then flagged (MDT bit set) to indicate that the field has been changed by the operator. If a mistake was made and the operator activates the pen and again detects that same field, the previous indication is erased. The character is changed from > back to ?, and the MDT bit in the attribute character is removed.

The selector-pen-detectable field that initiates a transmission request is defined by providing a space or null character immediately after the attribute character, plus at least one display character within the field. This request to the host computer can result in the computer carrying out a read operation to obtain the addresses of the fields that were selected or modified by the operator.

A selector-pen-detectable field indicated by an & immediately after the attribute character also initiates a transmission request. When such a field is detected, the MDT bit is set. The display unit responds to a poll or Read Modified command by providing both addresses and data in the fields that were modified by the operator.

If the character immediately following the attribute character is not ?, >, space, null or &, no operation is carried out.

If the terminal is provided with a keyboard, the keyboard will be locked when the transmission request is initiated from the selector pen. In such case, an initiated, but not yet executed, transmission can be inhibited by means of the RESET key on the keyboard.

The selector pen can operate on both protected and unprotected selector-pen-detectable fields.

Printer Unit

It is possible to carry out copy operations (for remote operation only), computer-initiated and local printout operations.

In a copy operation, initiated by e.g. a program access key, the image displayed on the screen can be printed out on any printer connected to the same communication processor. The Copy command and the printer and display unit addresses are issued from the computer, although buffer-to-buffer transmission takes place locally within the terminal system.

In a computer-initiated operation, the computer first addresses the printer in order to determine whether the printer is ready to accept a message or not. If the response is positive, the message is transferred from the computer, stored in the print buffer and then printed out.

In a local printout operation, initiated by the PRINT key, the image displayed on the screen is printed out on a specific printer. The command is issued locally and buffer-to-buffer transmission takes place locally within the terminal system. Consequently, this operation proceeds at high speed. Since the display unit is released after the completion of the buffer-to-buffer transmission, the operator can start a new transaction immediately. The execution of printout requests is based on a first-in/first-out queue procedure.

The local printout function is directed by a printer authorization matrix. To obtain a local printout function, the printer authorization matrix must be loaded into the communication processor. When power to the communication processor is turned on, and program load has been performed, the attached printers are available for printout operations according to a default printer matrix defined in Console Mode and transferred from the flexible disk unit to the communication processor. The printer authorization matrix can also be loaded into the communication processor via a display unit from an application program residing in the host system.

The printer authorization matrix defines the operating characteristics of the printers attached to the communication processor. In this regard, the matrix serves a three-fold purpose

- Printer mode. A printer may be reserved for exclusive use of either the host computer or the local printout function. A third mode allows it to be shared by these two functions. The three modes are called the system local, and share modes.

- Printer classes. A printer class is a way of grouping printers for use by the local printout function. A local printout request directed to a printer class is then serviced by one of the printers assigned to that group.

- Define display unit. The printer authorization matrix specifies which display units can use any given printer for local printout.

Printer Modes

A printer in the local mode can be used for local printout functions regardless of host attachment or communications protocol. This means that display units within the cluster can contend for use of printers but the host computer cannot. The printer is not available for print operations implemented directly from the host computer.

A printer in the system mode is entirely under host system control. The printer cannot be used for operator-initiated local printout requests.

In the shared mode, both host-directed printing operations and local printout operations are permitted on the same printer. In the system mode, the printer is protected from local printout operations. In the local mode, the printer is protected from host-initiated operations. However, when in shared mode, the subsystem does not guarantee this type of integrity. The user must assume the responsibility for integrity of his printed data by applying application-specific rules and proper programming practices when using a printer in the shared mode.

Printer Classes

The printer authorization matrix enables printer classes to be defined. The definition of a class of printers is provided by the customer, and may be based on type, character subset, type of forms used, location, etc. For example, in a particular installation, class "72" may have been defined as referring to all printers with preprinted forms. Thus, an operator may select an authorized printer on the basis of these characteristics rather than by address.

The printer authorization matrix allows a maximum of 16 printer classes to be defined in each subsystem. A display operator may select a printer by class by using the DEFINE key and keying in a number ranging from 70 through 85 corresponding with one of the 16 classes. In any configuration, a single printer may be in one or several classes, or not in any class. Several printers may be members of a single class.

Printer Authorization Matrix

The printer authorization matrix specifies how display units can use printers attached to the same communication processor. Consider the following example of a matrix:

Printer																		
No.	Mode	Class								Display unit No.								
		70	71	72	·	·	83	84	85	0	1	2	·	·	·	29	30	31
0	Local	0	0	0	·	·	0	0	0	1	1	0	·	·	·	0	0	0
1	Local	1	0	1	·	·	0	0	0	0	0	1	·	·	·	0	0	0
2	Local	0	1	0	·	·	0	0	0	0	0	1	·	·	·	0	0	0
·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
29	Share	1	0	1	·	·	0	0	0	0	0	0	·	·	·	0	1	1
30	System	0	0	0	·	·	0	0	0	0	0	0	·	·	·	0	0	0
31	Local	0	0	1	·	·	0	0	0	0	0	0	·	·	·	1	0	1

A maximum of 32 printers and display units in any combination can be connected to the same communication processor. The logical addresses of the individual printers and display units are determined when the system is customized.

On the first line in the matrix, display units 0 and 1 have been assigned printer 0 for local printout. Printer 0 cannot be addressed by class since it has not been assigned any class.

The matrix indicates that printer 1 can be used by display unit 2 for local printout. Display unit 2 can also use printer 2 for local printout. Printer 1 can be addressed by specifying classes 70 or 72 and printer 2 by class 71. The matrix defines the printer assignment options that are desired for local printout operations.

After the printer authorization matrix has been transferred from the diskette or from the host computer to the terminal system, the operator must (via the message line) specify which of the printers (or printer classes) appearing in the matrix is to be used for local printout. To do this, the operator depresses the DEFINE key on the keyboard belonging to the display unit that is to be assigned the printer (or printer class). After the DEFINE key has been depressed, a 2-position field is indicated by the cursor on the message line. The operator must enter the printer number or printer class into these positions. If necessary, the operator can change the printer assignment within the scope of the printer numbers and printer classes appearing in the matrix. The printer assignment is made when the operator depresses the ENTER key.

Controls and Indicators for Printer Unit 4154

- ON/OFF — The ON/OFF switch is located at right on the top cover.
- FEED — This pushbutton has a built-in lamp. Depressing this pushbutton momentarily will advance the paper one line, and when it is kept depressed, it will advance the paper continuously until the top of the next form is reached. When the FEED lamp glows steadily, the printer is in the off-line mode. A flashing FEED lamp indicates either a paper jam, paper out or a fuse alarm condition. After the paper fault condition is corrected, depressing the FEED pushbutton will reset the alarm condition. Note that when a paper out alarm is in effect, keeping this pushbutton depressed will advance paper to its trailing end.
- AUTO — This pushbutton has a built-in lamp. Depressing this pushbutton puts the printer in the on-line mode, so that the data source can select this printer for data transmission as necessary. Depressing the pushbutton a second time will return the printer to the off-line mode (data source cannot select this printer for data transmission). When the AUTO lamp glows steadily, the printer is in the on-line mode (i.e. selectable by the data source). A flashing AUTO lamp indicates that an illegal Vertical Tabulation command was found in the text received from the data source.

If the top cover on Printer Unit 4154 is raised, the following switches are accessible

- FINE — This switch functions only when the printer is in the off-line mode (lamp in FEED pushbutton lighted). The FINE switch is used for fine adjustment of a line on a form (or the like). When this switch is moved forward (up) the paper is moved up in steps of about 0.007 inches. When it is moved back again (down) the paper is moved down. Make certain that this does not wrinkle the paper. The paper continues moving as long as the pushbutton is kept depressed.

- LPI — This switch is used to select a line spacing of 6 or 8 lines per inch.

- TOF SET — This switch is used to position the first line on the form.

- FORM LENGTH — This switch enables the operator to select form lengths ranging from 4 inches to 12 inches as follows

Form Length position	1	2	3	4	5	6	7	8	9	10	11
Form length (inch), 6 LPI	4	4 1/6	5 3/6	5 5/6	6	8	8 2/6	8 3/6	11	11 4/6	12
Form length (inch), 8 LPI	4	4 1/8	5 4/8	5 5/8	6	8	8 2/8	8 4/8	11	11 4/8	12

NOTE:
Changing from 6 LPI (lines per inch) to 8 LPI sometimes entails certain difficulties since form lengths do not comply with the ISO/R216 standard (A-formats).

- CHANNEL NO — This switch does not function unless the vertical tabulation function is being used. The setting of this switch determines the lines in a form on which the tab stops will be set in compliance with the customer-specified contents of a ROM. See the information on the nameplate adjacent to this switch.

Controls and Indicators for Printer Unit 4153

- ON/OFF — The power ON/OFF switch is located on the rear at the right-hand side of the printer.

- LINE ON/OFF — When this switch is at the OFF position (off line) orders from the display unit or communication processor are inhibited and operator actions such as "line feed" are enabled.

  When this switch is at the ON position (on line), the printer is available for printouts.

- PAPER LF/TOF — This spring-return rocker switch can be used for several purposes when the printer is off line.

  - Line by line paper feed
    Depress the upper part of the switch (LF) momentarily.

  - Continuous paper feed
    Depress the upper part of the switch (LF) and keep it depressed as long as desired.

  - Feeding to top of form (TOF)
    Depress the lower part of the switch (TOF) and the paper will be advanced to the top of the next form.

- EOP OFF LINE — This lamp is lighted if the LINE switch is in the OFF position, if the paper runs out (End of Paper) or if a hardware fault has occurred.

- Acoustic alarm — An alarm is sounded if, for example, the operator tries to start the printer without the format tape in place, the paper runs out when the printer is in the on line mode, or a hardware fault occurs.

Communication Processor 4101

Controls and Indicators

The power ON/OFF switch is located at the lower, rear, right-hand (viewed from front) corner of the communication processor.

The RESET pushbutton is located at the lower, front, right-hand (viewed from front) corner of the communication processor. The RESET pushbutton is accessible from beneath the cabinet. The RESET pushbutton should not be used by the operator.

The communication processor has four indicators on its front

- READY — Starts flashing when power is turned on. Continues flashing until the operating system has been loaded into the communication processor, whereupon it commences to glow steadily.

- LINE 1 — Lights when communication with the computer can proceed via modem line 1.

- LINE 2 — Lights when communication with the computer can proceed via modem line 2.

- POWER ON — Lights when power is switched on to the communication processor.

Flexible Disk Unit 4120

Controls and Indicators

The power ON/OFF switch is located at the lower, rear, right-hand (viewed from front) corner of the flexible disk unit.

The RESET pushbutton is located at the lower, front, right-hand (viewed from front) corner of the flexible disk unit. The RESET pushbutton is accessible from beneath the cabinet. This pushbutton should not be used by the operator.
The flexible disk unit has a number of indicators on its front.

- READY — Lights when the program has been loaded into the flexible disk unit.
- POWER ON — Lights when the power is switched on to the flexible disk unit.

The three indicators designated ERROR 1, 2 and 3 are all intended for service personnel.

The indicator located on the flexible disk drive lights when reading or writing is not performed. When the indicator lights the diskette can be removed from the drive.

Contents

Illustrations

Tables

Commands and Orders

The host computer controls the Alfaskop System 41 terminals by means of a set of commands and orders.

Commands are used by the host computer to initiate operations such as writing, reading and erasing data in the buffer in a selected terminal.

Orders are used in write-type command operations, either alone or together with data that is to be displayed or printed. Orders are used to position, define, and format data being written into the buffer, to erase selected unprotected data in the buffer, to reposition the cursor and to specify printout format.

For remote operation-BSC and local operation, commands can be chained so that more than one command can be executed using only one output instruction in the host. After each command, the direction of transmission on the communication line is reversed so that an acknowledgement or reply message can be sent back.

Commands

The commands executed by the terminal system are of four main types

- Read-type commands which transfer data from the terminal buffer and, in remote configurations, transfer status/sense information to the host computer.
- Write-type commands that are used to transmit orders and data from the host computer to the terminal system.
- Control commands which are used to execute certain printer or display operations.
- Sense-type commands (only in local configurations) which transfer sense data to the computer. The sense bytes specify certain check conditions in the addressed terminal or specify the control unit model that is being emulated.

Table 1 presents all non-SNA commands and the associated codes used for both local and remote operation.

The following description of the commands presents general information concerning these four main types and detailed information concerning commands used in more than one of the operations described later. Detailed information about the commands used in only one operation (e.g. Remote Operation SNA/SDLC) is presented in the individual descriptions.

Table 1. Local and remote command codes

Command	Local operation $EBCDIC_{16}$	Remote operation $EBCDIC_{16}$
Read-type commands		
Read Buffer	02	F2
Read Modified	06	F6
Read Modified All	–	6E
Write-type commands		
Write	01	F1
Erase/Write	05	F5
Erase/Write Alternate	0D	7E
Control commands		
Select	0B	–
Select RM	0B	–
Select RB	1B	–
Select RMP	2B	–
Select RBP	3B	–
Select WRT	4B	–
Copy	–	F7
Erase All Unprotected	0F	6F
No Operation	03	–
Test I/O	00	–
Sense-type commands		
Sense	04	–
Sense ID	E4	–

Note: Several of the commands listed above are applicable to a restricted number of operations and emulations. Please refer to the individual descriptions for information about how they can be used.

Read-Type Commands

The read-type commands which the terminal system can execute in all operations are Read Buffer and Read Modified. The Read Buffer command is used for transmitting the entire contents of the buffer in the selected terminal to the host computer. The Read Modified command is used for transmitting data generated by operator activities at the terminal. The information that is transmitted can consist of data fields that have been changed by entries made via the keyboard, data entered from the magnetic identification device, data from selector-pen-detectable fields selected using the selector pen, or CU SE key, or codes generated by any of the function keys which initiate transmission request.

In local configurations, operator activity which requires some form of collaboration with the computer causes Attention status to be sent to the computer which responds by sending a read-type command to the terminal.

In remote configurations, on the other hand, read-type operations are normally carried out in connection with polling sequences which are sent out from the computer at certain intervals. A remote terminal which receives a polling sequence responds by sending back status and sense information if such information exists. If a message is waiting, it is sent to the computer in response to the polling sequence by means of a Read Modified command operation generated in the terminal. The Read Modified command is not sent from the computer. If neither status/sense information nor a message is waiting, a "no message to send" indicator is sent.

A read-type operation can also be called for by the host computer. In such case, it follows a selection addressing sequence either as the first/only command or as a command chained from another command. However, a read-type command can never be chained from a command which starts a printout operation. Note that the chaining of read-type commands is only permitted after a selection addressing sequence and not after a polling sequence. The communication procedures are described in greater detail in the sections on Remote Operation.

Buffer Start and Stop Addresses

The start position is the buffer address at which a Read Buffer command commences reading and at which a Read Modified command commences scanning for data to send to the host computer. A read-type operation commences at buffer address 0 in the following situations

- When the command is unchained, i.e. when the command follows immediately after a selection addressing sequence
- When the command is chained to a Sense, Select, No Operation or Copy command
- When the command (chained or unchained) is sent to an unformatted screen

If the read-type command is chained to a Write, Erase/Write or a read-type command, the read-type operation always starts at the current buffer address.

A Read Buffer command always reads to the last position in the buffer area, and this also applies to a Read Modified command when the buffer is unformatted, i.e. does not contain any attribute characters. When a Read Modified command is issued to a formatted buffer, scanning/reading continues to the last position in the last field. If the first position in the first line does not contain an attribute character, it means that the last field continues into the first line (wraparound) and scanning/reading can thus continue to the next attribute character.

After a read-type operation is concluded, the buffer address points to position 0, i.e. the first position on the first line unless wraparound took place in connection with a read modified operation. If wraparound took place, the buffer address points to the position of the next attribute character.

Attention Identification

Most replies that follow a read-type command or a poll include a text part introduced by an AID byte (Attention Identification). The AID byte informs the host computer as to which type of activity caused the read-type operation. See Table 2. An AID code other than $(60)_{16}$ or $(E8)_{16}$ indicates that the operator at the selected terminal has carried out an operation which requires collaboration with the computer. Examples of such operations include

- Depressing an ENTER, USM, CLEAR, SYREQ, program function or program access key
- Insertion or removal of a card in the magnetic identification device
- Using the selector pen or CU SE key to select a selector-pen-detectable field that initiates transmission

Table 2. Attention Identification (AID) Codes

Operator action	AID codes $EBCDIC_{16}$	Command operation	Notes
ENTER key and Selector pen attention (&)	7D	Read Modified	
PF 1 key	F1		
PF 2 key	F2		
PF 3 key	F3		
PF 4 key	F4		
PF 5 key	F5		
PF 6 key	F6		
PF 7 key	F7		
PF 8 key	F8		
PF 9 key	F9		
PF 10 key	7A		
PF 11 key	7B		The message that is transmitted comprises an AID code, a cursor address, an SBA order (only for formatted screen), the field address + 1 and the text. Nulls are suppressed.
PF 12 key	7C		
PF 13 key	C1		
PF 14 key	C2		
PF 15 key	C3		
PF 16 key	C4		
PF 17 key	C5		
PF 18 key	C6		
PF 19 key	C7		
PF 20 key	C8		
PF 21 key	C9		
PF 22 key	4A		
PF 23 key	4B		
PF 24 key	4C	Read Modified	
Magnetic identification device attention:			
Standard:	E6	Read Modified	When the card is inserted the message comprises the AID code and identification code. When the card is removed only the AID code is transmitted.
IMS:			
Card inserted	7D		
Card removed	6D	Read Modified	
Selector pen attention (space, null)	7E	Read Modified	The message comprises only the AID code and the field addresses. No data is transmitted.
PA 1 key	6C	Short Read	
PA 2 key	6E		
PA 3 key	6B		
PA 4 key	84		
PA 5 key	85		The message comprises only the AID code.
PA 6 key	86		
PA 7 key	87		
PA 8 key	88		
PA 9 key	89		
PA 10 key	91		
USM key	92		
CLEAR key	6D	Short Read	
SYREQ key	F0	Test Request Read	Test message from terminal to computer. The AID code is transmitted only when a Read Buffer command is received from the computer. It is not transmitted for a Read Modified command or polling.
No action at display unit	60	Read Buffer or Read Modified	Codes are transmitted when a read-type operation is called for by the computer (not polling) and the terminal has not received a transmission request from the operator
No action at printer unit	E8	Read Buffer or Read Modified	

Read Buffer Command

The computer can issue a Read Buffer command regardless of operator activity. This command is used primarily for testing purposes and it causes all data (including nulls) from the buffer position at which reading starts to the last buffer position inclusive to be transmitted from the addressed terminal to the computer. Fig. 1 illustrates the format of a response to a Read Buffer command.

	Link header	
AID	Attention Identification	
b_1 b_2	Cursor address, see Appendix 3	
SF ATB	Start Field order Attribute character	Only for formatted screen
TEXT	First data field, including nulls	
SF ATB TEXT	Second data field, including nulls Only for formatted screen	
	Link trailer	

Fig. 1. Message from the terminal issued in connection with a Read Buffer command

Read Modified Command

A Read Modified command normally initiates a read modified operation but, depending upon operator actions at the terminal, a short read or a test request read operation can also be initiated. The operations caused by specific operator actions are presented in Table 2 which shows the AID byte codes. In remote configurations, Read Modified commands are not normally sent from the computer due to the fact that the polling sequence initiates a Read Modified command in the terminal if a transmission request is present [AID ≠ $(60)_{16}$ or $(E8)_{16}$] and no status message is waiting.

For an *ENTER key attention, program function key attention, selector pen attention (&) or a Read Modified command obtained from the computer,* all data fields that have been modified by the keyboard, the selector pen or by a previous Write command are transmitted to the computer. Insignificant information is suppressed, i.e. the null characters included in the fields are not transmitted. When a data field is modified by the operator, the Modified Data Tag (MDT) bit is set in the attribute character for the modified data field. When a read modified operation is then carried out, the MDT bit in each attribute character is examined and the data fields for which MDT bits are set are transmitted to the computer. In the message to the computer, each modified data field is preceded by a terminal-generated Set Buffer Address order (SBA), followed by the 2-byte buffer address of the first character position in the field (attribute character address + 1). See Fig. 2.

	Link header	
AID	Attention Identification	
b_1 b_2	Cursor address, see Appendix 3	
SBA	Set Buffer Address order	First modified data field* (nulls suppressed)
b_1 b_2	Attribute character address + 1	
TEXT	Alphanumeric data	
SBA b_1 b_2 TEXT		Second modified data field (nulls suppressed)
		Remaining modified data fields (nulls suppressed)
	Link trailer	

* Note: The SBA order and attribute character address + 1 are not obtained for an unformatted screen. Instead, all text in the buffer (nulls suppressed) follows the cursor address.

Fig. 2. Message from the terminal issued in connection with an ENTER key attention, program function key attention, selector-pen attention (&) or a Read Modified command

For a *magnetic identification device attention,* a read modified operation is initiated which, when the identification card is inserted, transmits AID and the identification code that was read. When the identification card is removed, only AID is transmitted to the computer. Other data in the terminal buffer is not affected by this read modified operation. Fig. 3 illustrates the format of the message.

	Link header	
AID	Attention Identification	
b_1 b_2	Cursor address	
C9	I	Only in IMS version
70	Special character	
TEXT	Identification code, 0–37 bytes	
	Link trailer	

Fig. 3. Message from the terminal issued in connection with a magnetic identification device attention

For a *selector-pen attention,* resulting from operation of the selector pen in a field that starts with a space or null character, a read modified operation is initiated which transmits only information indicating which fields were modified by the operator via the keyboard or selector pen. For each selector-pen-detectable field in which the MDT bit is set, a Set Buffer Address order is transmitted as well as the buffer address of the first character position in the data field (attribute character address + 1). See Fig. 4. No alphanumeric data is transmitted from the modified selector-pen-detectable fields. If any characters have been entered from the keyboard into non-selector-pen-detectable data fields, the addresses of these fields are sent in the same way.

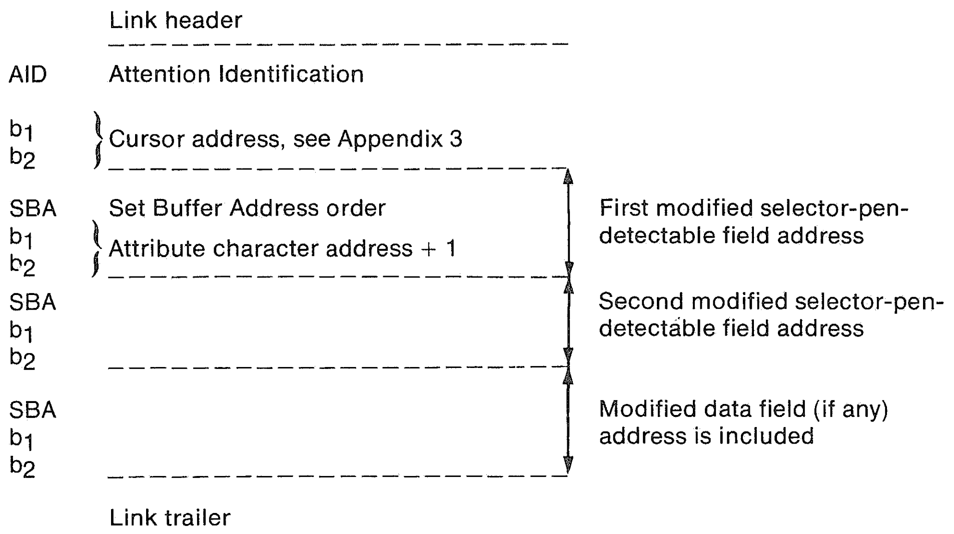


Fig. 4. Message from the terminal issued in connection with a selector pen attention (space or null)

For a *program access key, USM key or CLEAR key attention,* a short read operation is initiated. This means that only AID is transmitted to the computer. In remote configurations, this is, of course, supplemented with data link control characters etc (STX, CU, DV and ETX, for BSC).

For a *test request attention,* a test request read operation is initiated. This operation, just as a read modified operation, transmits to the computer all data fields that have been modified (MDT = 1) by the keyboard or selector pen. However, in this case, the message is introduced by a terminal-generated Test Request Read heading. See Fig 5. The test request read operation can only be used for terminals that do not use the SNA protocol.

	Link header	
SOH $6C_{16}$ 61_{16}	Start of Heading character	Test Request Read heading
STX	Start of Text character	
SBA	Set Buffer Address order	First modified data field* (nulls suppressed)
b_1 b_2	Attribute character address + 1	
TEXT	Alphanumeric data	
SBA b_1 b_2 TEXT		Second modified data field (nulls suppressed)
		Remaining modified data fields (nulls suppressed)
	Link trailer	

* Note: For an unformatted screen the SBA order and attribute character address + 1 are not obtained. Instead, all text in the buffer (nulls suppressed) follows STX.

Fig. 5. Message from the terminal issued in connection with a test request attention

Write-Type Commands

The write-type commands which the terminal system can execute are Write, Erase/Write and Erase/Write Alternate. The computer uses these commands to format and load buffer data into a selected terminal and to initiate certain terminal oper-tions such as starting the printer, unlocking the keyboard and sounding the audible alarm. The Erase/Write command erases the entire buffer before the write operation commences and this command is thus suitable for use initiate certain fer, i.e. when the entire buffer area is to be loaded with new data. To modify existing buffer data, the Write command is used instead. The third write-type command, Erase/Write Alternate, carries out an erase/write function. It is also used to switch the display unit into the large screen mode. With the exception of the erasing function and screen size switching, the three commands provide identical functions.

A write-type command can follow as the first/only command after a selection addressing sequence, or it can be chained to other commands which are preceded by a selection addressing or polling sequence. The restrictions imposed in connection with chaining are that the Erase All Unprotected command must not be chained from a Write command and that a Write, Erase/Write or Erase/Write Alternate command which starts the printer must be the last command on the chain, i.e. further chaining is not permitted.

Buffer Addressing

The text part of the message following a write-type command can comprise both buffer data and orders. The writing of buffer data can start at any specified position in the buffer area and the data characters are then stored successively in the following positions until a new positioning order is obtained (Set Buffer Address order) or until all data has been stored. In connection with a write-type operation, the buffer address is incremented by one for each character that is stored.

In connection with a Write command, the buffer position at which the write operation is to start is determined as follows

- A Set Buffer Address order (SBA) specifies the start position

- When the command is chained to a Read command or to another Write command, the current buffer address is used as the start position

- In other situations, i.e. when an SBA order is lacking or when the command is unchained or chained to a control or Sense command, the cursor indicates the start position to be used, i.e. the buffer address is set equal to the cursor address.

In connection with an Erase/Write or Erase/Write Alternate command, both the cursor and buffer addresses are reset to 0 after an erase operation so that a write-type operation will start at the first position in the first line.

An invalid buffer address specified within a buffer control order is converted to a valid buffer address.

Write Control Character, WCC

Each write-type command is followed by a write control character, WCC, which can define the terminal function both before and after the operation specified by the command. Table 3 presents the bit assignments for WCC and explains the functions of these bits. The first byte following the command byte is always interpreted as a WCC.

Table 3. Write control character, WCC, bit definitions

Bit*	Function
7, 6	Determined by bits 0–5, see Appendix 1 for translation to EBCDIC.
5, 4	*Printout Format.* Defines the printout format as follows 00 – NL and EM ordersin the text determine the printout format. Defaults to a line of a length defined when customizing when these orders (NL, EM) are not present. 01 – Specifies 40-character print line 10 – Specifies 64-character print line 11 – Specifies 80-character print line
3	*Start Printer.* When this bit is set to 1, it indicates that the printout operation shall be started when the write-type operation in the buffer has been completed.
2	*Sound Alarm.* When this bit is set to 1, the audible alarm is sounded in the selected display unit when the write-type operation has been completed.
1	*Keyboard Restore.* When this bit is set to 1, the keyboard is restored (unlocked) when the write-type operation has been completed.
0	*Reset MDT bits.* When this bit is set to 1, all MDT (Modified Data Tag) bits in the selected terminal's buffer data are reset. Resetting is carried out before the write-type operation is started.

* Note: IBM designates the most significant bit (Bit No. 7) as No. 0.

Write Command

The Write command is used by the computer to modify existing data in the buffer in a selected terminal. A write-type message sent to a terminal comprises a command code (CMD), write control character (WCC), orders and the data that will modify the buffer content. For a remote operation, it also includes the necessary BSC or SDLC part of the transmission.

	Link header

CMD	Command code, see Table 1*
WCC	Write control character, see Table 3
TEXT	Orders and alphanumeric data

	Link trailer

* Note: In local operation, the command code is included in the link header and distributed via the CCW.

Fig. 6. Write, Erase/Write and Erase/Write Alternate commands, message layout

Erase/Write Command

The Erase/Write command is used by the computer when the entire buffer area in a selected terminal is to be loaded with new data. This command causes two different operations to be carried out in the terminal – an erasing operation followed by a write operation. The erasing operation erases the entire buffer (fills it with null characters), positions the cursor at the first position in the first line and resets the buffer address to 0. The write operation is then carried out in exactly the same way as for a Write command. If WCC is not sent, only the erasing operation is carried out.

Erase/Write Alternate Command

The purpose of this command is to permit existing host computer software to be used. Display units and printers having 960-character buffers function as 480-character devices on a default basis. Display units and printers having 1920-, 2560- or 3440 character buffers function as 1920-character devices on a default basis. The Erase/Write Alternate Command can be used to specify the non-default buffer utilization capacities set forth in the table below. Host application programs written for display units and printers having 480- or 1920-character buffers can thus be used without alteration for 960-, 1920-, 2560- and 3440-character display units and printers.

The following default and alternate buffer utilization capacities can be selected when the system is set up and customized.

Default character capacity	Alternate character capacity
480	960
1920	1920
1920	2560
1920	3440

The Erase/Write Alternate command also operates as an Erase/Write command. Once the display unit or printer is placed in alternate mode, operation continues in the alternate mode until the operator depresses the CLEAR, or SYREQ key or until an Erase/Write command is received, power fails at the display unit, or printer or, in locally attached systems, a system reset sequence occurs. Only these conditions return the display unit or printer to the default-value screen-size or character print capacity.

Control Commands

The control commands are used by the computer to initiate certain terminal operations that do not require any transmission of data between the computer and the terminal. The control commands include Copy, Select-type commands, Erase All Unprotected, Test I/O and No Operation. All but the Copy command can be executed in connection with local operation. For remote, non-SNA operation, only Copy and Erase All Unprotected commands can be executed. For SNA operation, only Erase All Unprotected command can be executed.

Copy Command

The Copy command, which can only be used for remote, non-SNA operation, transmits buffer data from one terminal to another terminal that is connected to the same communication processor. The selected terminal is the "to" terminal, i.e. terminal to which buffer data is to be transmitted. The command code is followed by

- Copy control character (CCC) that specifies which buffer data is to be transmitted
- The "from" terminal address which specifies the terminal from which the buffer data is to be fetched.

	Link header

CMD	Command code, see Table 1
CCC	Copy control character, Table 4
FDV	"From" terminal address

	Link trailer

Fig. 7. Copy command, message layout

Table 4 presents the bit assignments for the copy control character, CCC, and explains the functions of these bits. The command code (CMD) must always be followed by both CCC and a "from" address. If one or both are lacking, the command is not executed. Instead, the terminal generates error status.

In a Copy command, the selected terminal ("to" terminal) address can also be specified as the "from" terminal address. In such case, nulls are inserted in all locations that do not contain the data specified by CCC bits 1 and 0.

Table 4. Copy control character, CCC, bit definitions

Bit*	Function
7, 6	Determined by bits 0–5, see Appendix 1 for translation to EBCDIC.
5, 4	*Printout Format.* Defines the printout format in accordance with the following 00 – NL and EM orders in the text determine the print line length. Defaults to a line of a length defined when customizing when these orders are not present. 01 – Specifies 40-character print line 10 – Specifies 64-character print line 11 – Specifies 80-character print line
3	*Start Printer.* When this bit is set to 1, printout is started in the "to" terminal when the transmission of the buffer content has been completed.
2	*Sound Alarm.* When this bit is set to 1, the audible alarm in the "to" terminal is sounded when the transmission of the buffer content has been completed.
1, 0	*Type of data to be copied.* Defines the data that is to be copied as follows 00 – Only attribute characters are copied 01 – Attribute characters and unprotected data fields (including nulls) are copied. All character positions within protected data fields are replaced with null characters 10 – All attribute characters and protected data fields (including nulls) are copied. All character positions within unprotected data fields are replaced with null characters 11 – The entire content (including nulls) of the "from" terminal buffer is copied

* Note: IBM designates the most significant bit (Bit No. 7) as No. 0.

A Copy command can be chained from read-type and control commands. (If a Copy command is chained from a write-type command, the data loaded into the buffer is destroyed by the Copy command.) When a Copy command which starts the printer is chained from another command, the Copy command shall be the last command in the chain.

A Copy command will not be effective if the first position of the "from" terminal buffer contains an attribute character defining a protected alphanumeric field.

Select-Type Commands

The select-type commands are immediate commands that are used only for local operation. Please refer to section on "Local Operation".

Erase All Unprotected

The Erase All Unprotected command is used for both local and remote operation. This command causes the following operations to be performed in the addressed terminal

- All unprotected data fields are erased (cleared to nulls)
- The MDT bit in each attribute character which specifies an unprotected field is reset to 0
- The keyboard is unlocked
- Attention identification, AID, is reset
- The cursor is positioned at the first character position in the first unprotected field
- The buffer address is set to the cursor position address

When there is no unprotected field (entire buffer area protected), no erasing is obtained in the buffer and the MDT bits are not changed. However, the other operations are performed, i.e. the keyboard is unlocked (restored), AID is reset and the cursor is repositioned. In this case, the cursor is repositioned to the first position in the first line which is also set as the buffer address.

The Erase All Unprotected command is an immediate command when used for local operation. This means that the communication processor responds with Channel End as initial status, thus freeing the channel for other operations. When the erasing operation has been completed, the communication processor sends Device End as asynchronous status to the channel and then becomes not busy. An Erase All Unprotected command can be chained from read-type commands.

	Link header
CMD	Command code, see Table 1*
	Link trailer

* Note: In local operation, the command code is included in the link header and distributed via the CCW.

Fig. 8. Erase All Unprotected command, message layout

No Operation

No Operation is an immediate command which is used only for local operation. This command does not give rise to any operation in the communication processor. However, it can be used to fetch waiting status. If there is no waiting status, the communication processor combines initial and ending status by sending Channel End and Device End to the channel, thus indicating that the operation is completed.

Test I/O

Test I/O is an immediate command which is used only for local operation. It can be used to fetch waiting status.

Sense Commands

The Sense commands are used only for local operation.

The computer sends the Sense command to a communication processor which has sent status with the Unit Check bit set. The communication processor responds to a Sense command by sending a sense byte which defines in greater detail the error indicated by the status message. When Unit Check is obtained in the status byte, sense information shall always be fetched by means of a Sense command. This is necessary due to the fact that all commands except Sense, No Operation and Test I/O cause the sense data to be reset. See the section on Local Operation for a detailed description of the sense byte.

The Sense ID command is used to ascertain which control unit model is being emulated.

Orders

Orders are used in write-type command operations, either alone or together with display or print data. There are two types of orders: buffer control orders and printout format orders. Buffer control orders are used to position, define and format data that is to be loaded into the buffer; to erase selected unprotected positions in the buffer; to position the cursor; to reserve the printer for a computer output message: and to inform the operator that an unsolicited message is waiting. These orders are executed when they are found in the data stream and they are not stored in the buffer area. However, certain orders, those which control formatting and repetitive operations for example, are followed by specific data that is stored in the buffer. Printout format orders are always written in the buffer and stored until printout starts, at which time they are executed in the sequence in which they are found. Each printout format order occupies a character position in the buffer.

Table 5. Buffer control orders

Order	Byte 1 Order Code $EBCDIC_{16}$	Byte 2	Byte 3	Byte 4
Start Field, SF	1D	Attribute[1] character ATB	–	–
Set Buffer Address, SBA	11	1st address byte b_1[2]	2nd address byte b_2[2]	–
Insert Cursor, IC	13	–	–	–
Program Tabulation, PT	05	–	–	–
Repeat to Address, RA	3C	1st stop address byte b_1[2]	2nd stop address byte b_2[2]	Character to be repeated[3]
Erase Unprotected to Address, EUA	12	1st stop address byte b_1[2]	2nd stop address byte b_2[2]	–
Underline	6D	–	–	–
Unsolicited Message, USM	2F	–	–	–

Notes: 1 See Table 6.
2 See list of buffer addresses in Appendix 3.
3 See EBCDIC code table in Appendix 2.

Buffer Control Orders

Start Field Order, SF, with Attribute Character

The Start Field order, which is not stored in the buffer memory, indicates that the next character in the data stream is an attribute character. The terminal then stores the following character, i.e. the attribute character, at the current buffer address.

The attribute character defines the data area extending to the next attribute character and specifies, among other things, which type of data the operator can enter, how the presentation shall be carried out and whether the data field has been modified by the operator. Aside from the Modified Data Tag bit, which indicates whether or not the data field has been modified, the attribute character is protected from being affected by operator activities. The only terminal operation which can affect the attribute character is depression of the CLEAR key which erases the entire buffer content including the format data. The computer can send an attribute character containing any combination of bits. The attribute character bit assignments are presented in Table 6 which also explains the functions of the different bits.

Table 6. Attribute character, ATB, bit definitions

Bit*	Function
7, 6	Determined by bits 0–5, see Appendix 1 for translation to EBCDIC
5	0 – Unprotected field 1 – Protected field
4	0 – Alphanumeric input 1 – Numeric input (if the numeric lock feature is present) Note: If bits 4 and 5 are set to 11 automatic skip to the next unprotected position is obtained if the operator keys a character into the position preceding the attribute character.
3, 2	00 – Normal Brightness and not selector-pen-detectable 01 – Normal brightness and selector-pen-detectable 10 – High brightness and selector-pen-detectable 11 – Off, no printout and not selector-pen-detectable
1	Unused, shall always be 0
0	Modified Data Tag (MDT). Identifies modified data fields during Read Modified command operations 0 – Data field has not been modified 1 – Data field has been modified by the operator (can also be initially set by the computer) The MDT bit is reset by the computer or the ERASE INPUT key

* Note: IBM designates the most significant bit (Bit No. 7) as No. 0.

Set Buffer Address Order, SBA

Set Buffer Address (SBA) is a 3-byte order that specifies the address at which the following data shall be stored. An SBA order can also precede another order to specify

- The start address for a Program Tabulation (PT), Repeat to Address (RA) or Erase Unprotected to Address (EUA) order
- The address at which an attribute character shall be stored by a subsequent Start Field (SF) order
- The address at which the cursor shall be positioned by a subsequent Insert Cursor (IC) order

Insert Cursor Order, IC

The IC order positions the cursor at the position specified by the current buffer address. This operation does not change the current buffer address or the content of the position it specifies. Only one IC order should be used. If more IC orders are sent, only the last one will be effective.

If IC is lacking, the following applies

- A Write operation does not affect the cursor position
- Erase/Write operations move the cursor to the first position on the screen.

Program Tabulation Order, PT

The PT order, which can be used for tabulating operations in a display unit or printer buffer, moves the current buffer address ahead to the address of the first character in the next unprotected field. If PT is obtained when the current buffer address points to an attribute character that defines the following field as unprotected, the current buffer address is advanced to the first character position in the field. When the PT order follows running text in a write data stream obtained from the computer, null characters are entered into the remaining positions as far as the end of the field. In the event that the PT order is sent to an unformatted screen, null characters are entered into the remaining positions to the end of the screen, and the buffer address is reset to 0. In a PT operation on a formatted screen, the buffer address becomes the first position on the first line if no attribute character is found before the end of screen.

Repeat to Address Order, RA

Repeat to Address is a 4-byte order which stores a specified character in all positions starting at and including the current buffer address and extending to but not including the specified stop address (b_1, b_2). See Table 5. The specified character is repeated, regardless of its code, and when the operation is completed, the current buffer address will be equal to the stop address.

When an RA order specifies a stop address that is equal to or lower than the current buffer address, wraparound occurs during the operation. An RA order which specifies a stop address that is equal to the current buffer address causes the specified character to be stored in all buffer positions.

Note that an attribute character can be overwritten by an RA order.

Erase Unprotected to Address Order, EUA

An EUA order is a 3-byte order that enters null characters into all unprotected buffer positions starting at and including the current buffer address and extending to, but not including, the specified stop address (b_1, b_2). See Table 5. When the operation is completed, the current buffer address is equal to the stop address. This operation does not affect the content of any attribute characters.

When an EUA order specifies a stop address that is equal to or lower than the current buffer address, wraparound takes place during the operation. An EUA order which specifies a stop address that is equal to the current buffer address causes null characters to be entered into all unprotected buffer positions.

Underline Order, __

The underline order (6D) occupies one position in the buffer and is displayed as a space on the screen. The underline order initiates underlining of the succeeding characters as far as the next space, NUL or attribute character.

Unsolicited Message Order, USM

When sent to a display unit, the USM order lights an indicator lamp on the operator's keyboard. This function can be used to indicate that a message to the terminal is waiting in the computer. When the operator is ready to accept the message he depresses the USM key and a transmission request is initiated. On the next poll a short read operation is executed so that only the AID code is transferred to the computer, see Table 2. At the same time the USM indicator lamp goes out. Now the computer has received an indication that the operator is ready to accept the message and thus an ordinary write message is sent.

The USM order may also be sent to a printer causing a printer reservation for the computer, see below.

Printout Format Orders

Several of the orders listed in Table 5 can also be used in the printer buffer.

New Line and End of Message Orders, NL EM

The length of the printed line is determined by the control character (WCC, CCC). The following four formats are available

- 40 characters per line
- 64 characters per line
- 80 characters per line
- Transparent format

When any of the first three fixed formats listed above is used, carriage-return/line-feed (CR LF) is generated automatically when the specified number of characters has been reached. When the transparent format is used, the message layout is determined by the special printout format orders (new line NL and end of message EM) which must be included in the text message that is to be printed out. The printing of a space is all that happens in response to NL and EM orders when any of the first three fixed formats is used. The maximum length of the printed line must be determined when customizing.

The transparent format is handled as follows

- When an NL order is found in the text, CR LF is generated

- When an EM order is found, CR LF is generated automatically whereupon the printout operation is concluded. Thus, any remaining charaters in the printer buffer will not be printed out. If the EM order occurs in the first position of a line, the second automatically-generated CR LF function is inhibited.

- If the NL order is missing, CR LF will be generated automatically when the maximum printed line length has been reached

The following applies to all formats

Attribute characters, NUL characters and the text in non-displayed fields (all represented by NUL below) are handled as follows during printout

- A line containing only spaces and NULs results in CR LF. This results in a blank line in the printout

- NULs between text sections are printed as spaces

- The printer does not take any action when a line containing all NULs is encountered

The printer has a repertoire of 94 printable characters. In addition, the following can be printed out

- NL and EM which are printed as spaces, except in a transparent format where they are not printed at all

- DUP is printed as *

- FM is printed as ;

Table 7. Printout format orders

Order	Order Code $EBCDIC_{16}$
New Line, NL	15
End of Message, EM	19
Vertical Tabulation, VT	0B
Form Feed, FF	0C
Unsolicited Message, USM	2F

Vertical Tabulation Order, VT

The VT character causes the printer to advance the paper to the next preset line within a particular form or, if already at the end of the form, to a new form. The preset line positions on the form are determined when installing the printer.

As a special feature, vertical tabulation can also be ordered by means of a 2-byte sequence, NL DUP ($151C_{16}$).

Form Feed Order, FF

The FF order causes the printer to advance the paper to the beginning of a new form (page). The start position for a new form is determined by the TOF switch on the printer (PU 4154) or format tape (PU 4153/59).

As a special feature, form feed can also be ordered by means of a 2-byte sequence, NL FM ($151E_{16}$).

Unsolicited Message Order, USM

When the USM order is included in a message sent to the printer, the printer will be reserved for computer initiated printouts and operator initiated printouts will be inhibited. The printer is reserved until a new computer message that does not contain a USM order is received.

Contents

Illustrations

Tables

Remote Operation – Binary Synchronous Communication (BSC)

In remote configurations, the Alfaskop System 41 can utilize either the Binary Synchronous Communication (BSC) protocol or the Synchronous Data Link Control (SDLC) protocol. Only the BSC operating mode is described in this chapter.

Transmission Interface

The interface used in the Alfaskop System 41 complies with the requirements for signal levels, impedances and logic specified for the CCITT V24/28 interface.

The Alfaskop System 41 communicates with a central computer, a front end computer or a concentrator via modem equipment and a 2-wire/4-wire point-to-point or multipoint communication network. The transmission is bit and character synchronous and the transfer rate can range up to 9600 bps.

Data Link Control

The computer controls all traffic to/from the terminals via a poll/select system, i.e. it polls the terminals for input messages and selects a particular terminal for an output operation. Each terminal in the network has a unique address which is used in connection with polling or selection. All text and all data link control characters are transmitted in EBCDIC code.

Modes of Operation

The terminal is always in one of three modes of operation: Control mode, Text mode or Transparent Monitor mod e.

Control Mode

The terminal always enters the Control mode when it receives a valid end of transmission sequence (see description of EOT in the section headed "Data Link Control Characters"). When the terminal is in the Control mode, it monitors the communication line to detect

- A valid polling or selection addressing sequence used to select the terminal for entry into the Text mode

- A sequence that contains characters DLE STX, which place the communication processor in the Transparent Monitor mode

Text Mode

In the Text mode, the terminal operates either as a master or slave station. The terminal that sends a message is called the master station and the terminal that receives the message is called the slave station.

The terminal becomes the master station when it sends STX in response to a read-type command or a poll operation. As the master station, it can send ENQ in order to request a reply or request a retransmission from the computer. When the transmission of a message is concluded, i.e. when EOT is sent, the terminal returns to the Control mode.

The terminal is the slave station while it receives a message from the computer, i.e. while a write-type command is being executed. As the slave station, the terminal replies in a definite manner to transmissions from the master station.

Transparent Monitor Mode

The terminal itself cannot operate in the Transparent mode, but it can be connected to a line together with other types of units that can operate in the Transparent mode.

When the terminal is receiving character sequence DLE STX, it switches to the Transparent Monitor mode. In this mode, the terminal shall ignore all characters except

- A transparent text sync sequence (DLE SYN)
- A transparent text terminating sequence (DLE ITB, DLE ETB, DLE ETX or DLE ENQ)

The terminal shall leave the Transparent Monitor mode and return to the Control mode if

- A transparent text sync sequence is not received within three seconds
- A transparent text terminating sequence is received

Text Blocking

Messages from the terminal system are sent in blocks to the computer. Each text block can contain a maximum of 256 characters, addressing and data link control characters included (SYNs excluded). Each block starts with STX and ends with ETB except for the last block of the message which is terminated by ETX. Since only the first text block includes the terminal address (CU/DV) this block may include up to 252 text and order characters while the remaining blocks of the message may include up to 254 characters each.

Possible AID (Attention Identification) and cursor address are included in the first text block only. Furthermore, the three bytes of an SBA (Set Buffer Address) order and the two bytes of an SF (Start Field) order are always sent together in one text block.

Transmission Checking

Cyclic redundancy block checking is carried out on the following types of data

- Terminal commands sent out from computer (including text which follows a write-type command)
- Data that is sent to the computer in response to a read-type command or a polling operation

A cyclic redundancy check (CRC) character is accumulated in both the computer and the terminal system. The master unit sends its CRC character at the end of the message for comparison with the slave station's CRC character.

Accumulation commences when STX or SOH is found, but it does not include this character. All characters following this STX or SOH are a part of the accumulation, up to and including ETB or ETX. After ETB or ETX, the CRC character (two bytes) is transmitted.

When the terminal detects a CRC error in a received message, it responds by sending NAK, thus requesting retransmission. During reception, the CRC character is separated from the message and not stored in the terminal buffer.

Data Link Control Characters

The data link control characters are generated automatically by the computer/terminal system and used to establish connections and control all traffic on the communication network between the computer and the terminal system. For example, these characters are used for message framing, acknowledgement indicating that a message was received correctly, etc. The data link control characters perform the following functions

PAD $(FF)_{16}$	The PAD character is generated by the computer/terminal and used to ensure complete transmission and reception of the first and last character in a message.

SYN Synchronous Idle (32)16

The synchronization section of the message consists of two consecutive SYN characters which are used to achieve and maintain character synchronism in synchronous transmission systems. SYN characters can also be used as timefill characters in a message (not stored in the buffer).

DLE Data Link Escape (10)16

The DLE character is used with other characters in a number of 2-character sequences to provide supplementary control in the communication system. These 2-character sequences include ACK 0, ACK 1, WACK and RVI. The DLE character is also used in a number of sequences in order to control operation in the Transparent Monitor mode. See section headed "Modes of Operation".

ACK 0 Even Acknowledgement (10 70)16

ACK 0 is a 2-byte sequence which the terminal sends to the computer in response to an approved selection addressing sequence. The terminal thus indicates that it is ready to receive the message. ACK 0 is also sent by the terminal in response to an even-numbered command sequence (2, 4 etc.).

ACK 0 is sent from the computer when an even-numbered text block has been received and approved.

ACK 1 Odd Acknowledgement (10 61)16

ACK 1 is a 2-byte sequence which the terminal sends in response to each correctly received odd-numbered command sequence (1, 3 etc.).

ACK 1 is sent from the computer when an odd-numbered text block has been received and approved.

Note: ACK out of sequence (ACK 0 is sent by the computer but the terminal expects ACK 1, or vice versa) is interpreted as a NAK by the terminal.

NAK Negative Acknowledge (3D)16

The terminal sends NAK in response to a text transmission from the computer when

- The transmission contained an erroneous CRC character
- The transmission contains a TTD (Temporary Text Delay) sequence, i.e. the computer cannot send the message immediately, but wishes to maintain the connection with the terminal
- The ETX character is missing

- The text message includes ENQ

NAK is also sent by the terminal in response to a single ENQ sent from the computer in response to a text message

When the terminal receives NAK as a response to a transmission, the transmission is repeated.

The terminal counts the received NAKs, and after 15 NAKs the terminal enters the Control mode and sends EOT to the computer.

ENQ Enquiry $(2D)_{16}$

The terminal sends ENQ

- To request a reply from the computer after a 3-second timeout
- To request retransmission of the last reply (WACK or ACK) from the computer
- When WACK is obtained from the computer

When the terminal receives ENQ as a response to a transmission of a data link control character, the transmission of the data link control character is repeated.

When the terminal receives ENQ in response to a transmission of a text message, the terminal sends NAK to the computer.

ENQ must always be included as the last character in a polling or a selection addressing sequence.

WACK Wait before Transmit Positive Acknowledgement $(10\ 6B)_{16}$

WACK is a 2-byte sequence which a terminal uses in order to indicate that it is temporarily not ready for reception. The terminal sends WACK in response to a selection addressing sequence when the printer is busy. Moreover, WACK is sent in response to a Write or Copy command when the Start Printer bit is set in the WCC or CCC.

When WACK is received from the computer, the terminal responds with ENQ.

RVI Reverse Interrupt $(10\ 7C)_{16}$

RVI is a 2-byte sequence which the terminal sends in response to an attempted selection addressing when the terminal has a status/sense message to send.

When RVI is obtained from the computer the terminal responds by sending EOT and resets all waiting status/sense information.

The terminal accepts RVI only when it expects an ACK 0 or ACK 1 from the computer. If RVI is received at the terminal in response to RVI a timeout occurs at the terminal.

STX Start of Text $(02)_{16}$

The STX character is the first character in the text section of the message and starts the message text. When the terminal receives STX as the first character in a command or TTD sequence, a new CRC character accumulation commences.

STX is transmitted to the computer as the first character in the text section except in a status or test-request message.

SOH Start of Heading $(01)_{16}$

SOH is the first character in a 3-character sequence (header) which is used for identifying status request and test request messages from the terminal. For these messages, CRC character accumulation commences after the SOH character.

ETB End of Transmission Block $(26)_{16}$

The ETB character indicates the end of the block of characters that was introduced by SOH or STX. The block check (CRC) character follows immediately after ETB. An ETB character is handled in the same way as ETX, i.e. a CRC is carried out, and ACK 0, ACK 1, WACK or NAK is sent in response.

In addition to STX, address characters and ETB, the first text block from the terminal can contain a maximum of 252 characters. The following text blocks can each contain a maximum of 254 characters due to the fact that the address characters are transmitted only with the first text block. The last text block in the transmission is terminated by ETX.

ETX End of Text $(03)_{16}$

ETX concludes the message that started with STX or SOH, i.e. a read-type, write-type or control command or a status request message or test request message. The check character (CRC) follows immediately after ETX. When the terminal receives ETX, a CRC is initiated, and if no error is found, the terminal responds with ACK 1. WACK can also be encountered as a response to a write-type or copy command. If the CRC character is erroneous, the terminal responds with NAK.

The terminal sends ETX after the last (or only) text block in a message. Then, if the CRC was successful, the computer responds with ACK 0 or ACK 1. If the CRC was unsuccessful, the computer responds with NAK.

EOT End of Transmission
$(37)_{16}$

The EOT character indicates to the slave station that a message has been completed and the transmission has thus been terminated. When the terminal receives EOT it is returned to the Control mode. EOT does not reset status and sense information at the terminal.

EOT is sent to the computer in the following cases

- As the normal end of a read-type operation when the terminal is the master station
- When a General Poll has proceeded through all terminals that are connected to the communication processor
- When the terminal is the slave station and cannot carry out the operation which the computer requests

ITB End of Intermediate Transmission Block
$(1F)_{16}$

The ITB character divides the message into a number of subsections so that error checks can be carried out within the message without changing the direction of transmission. The ITB character is followed immediately by the CRC character.

The terminal does not make error checks as described above, even though it accepts the ITB character. The terminal includes ITB and the associated CRC character in the CRC accumulation, but then removes them from the data stream so that they are not stored in the terminal buffer. No separate CRC is carried out on this occasion. Instead, reception continues until ETB or ETX is received.

ESC Escape
$(27)_{16}$

The ESC character must precede a command sequence and is thus found in read-type, write-type and control commands obtained from the computer. The terminal cannot generate ESC.

TTD $(02\ 2D)_{16}$	Temporary Text Delay	TTD is a 2-byte sequence (STX ENQ) which the computer sends when it wishes to maintain the connection with a selected terminal but does not have the desired message ready for transmission. A TTD sequence is normally transmitted after approximately two seconds if the computer is not, within this interval, able to transmit its message. TTD thus prevents a 3-second receiving timeout from being triggered in the terminal. The terminal always responds to a TTD sequence by sending NAK. The terminal cannot send TTD.

Transmission Sequences

Since all communication on the network is initiated and controlled from the computer, it must find out at certain intervals whether any of the terminals wish to communicate. This check is called polling, and it comprises a request made to ascertain whether a terminal wishes to transmit. If a terminal has a message to send to the computer, it transmits the message. The computer can use a Specific Poll which involves investigation of a certain terminal or a General Poll which involves running through a number of terminals.

The transmission sequences that can be sent between the terminal system and the computer are divided into four categories

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

The sequence diagram prepared for each category of responses presents all terminal responses.

General and Specific Poll Sequences

A Specific Poll addresses a communication processor and a specific terminal in order to determine whether status and sense information or a manually entered message is awaiting transmission to the computer.

In a General Poll, a particular communication processor is addressed, whereupon it investigates all connected terminals, one by one, in order to ascertain whether any of them has status or sense information or an operator message that is awaiting transmission to the computer.

When a General or Specific Poll sequence is issued, one of three situations can arise

- If status/sense information is waiting with or without AIDs, a status/sense message is sent to the computer
- If status/sense information is not waiting and AID is present, a Read Modified command operation is generated into the terminal.
- If there is no status/sense information or no AID waiting, EOT is sent to the computer.

In a General Poll, each connected terminal is investigated in the sequence in which the ENTER key was depressed.

When a message is found, it is sent along with the terminal address in question. When the message has been transmitted (and acknowledged by the computer) the communication processor continues to examine the remaining terminals. If all messages have been transmitted to the computer and all connected terminals have been investigated for pending messages the communication processor concludes the General Poll operation by sending EOT after which it enters the Control mode.

If the computer wishes to conclude the General Poll, an RVI can be sent to the communication processor, thus forcing an EOT response.

If a command is issued instead of ACK (after blocks that end with ETX) the General Poll operation will also be concluded.

A terminal which has sent text (Read Modified or Short Read Message), status or a test request message in response to a poll, can remain selected because the computer wishes to continue with a Write, Erase/Write, Erase/Write Alternate, Copy or Erase All Unprotected command without further addressing. When the terminal has sent the last text block in the message, i.e. when ETX has been sent, the computer responds by sending a command which must pertain to the terminal selected by a Specific Poll or the terminal which sent the most recent response in connection with a General Poll. The command sequence is concluded in the same way as for a normal selection operation, i.e. the terminal sends ACK after which the computer sends EOT. This also causes the polling operation to cease. Any remaining response to a General Poll concluded by a command sequence will be transmitted in connection with the next polling operation.

Fig. 1 presents the General and Specific Poll sequences and the terminal system responses. The Test Request, Read Modified, and Short Read messages are described in the "Read Modified Command" section in the chapter on "Commands and Orders".

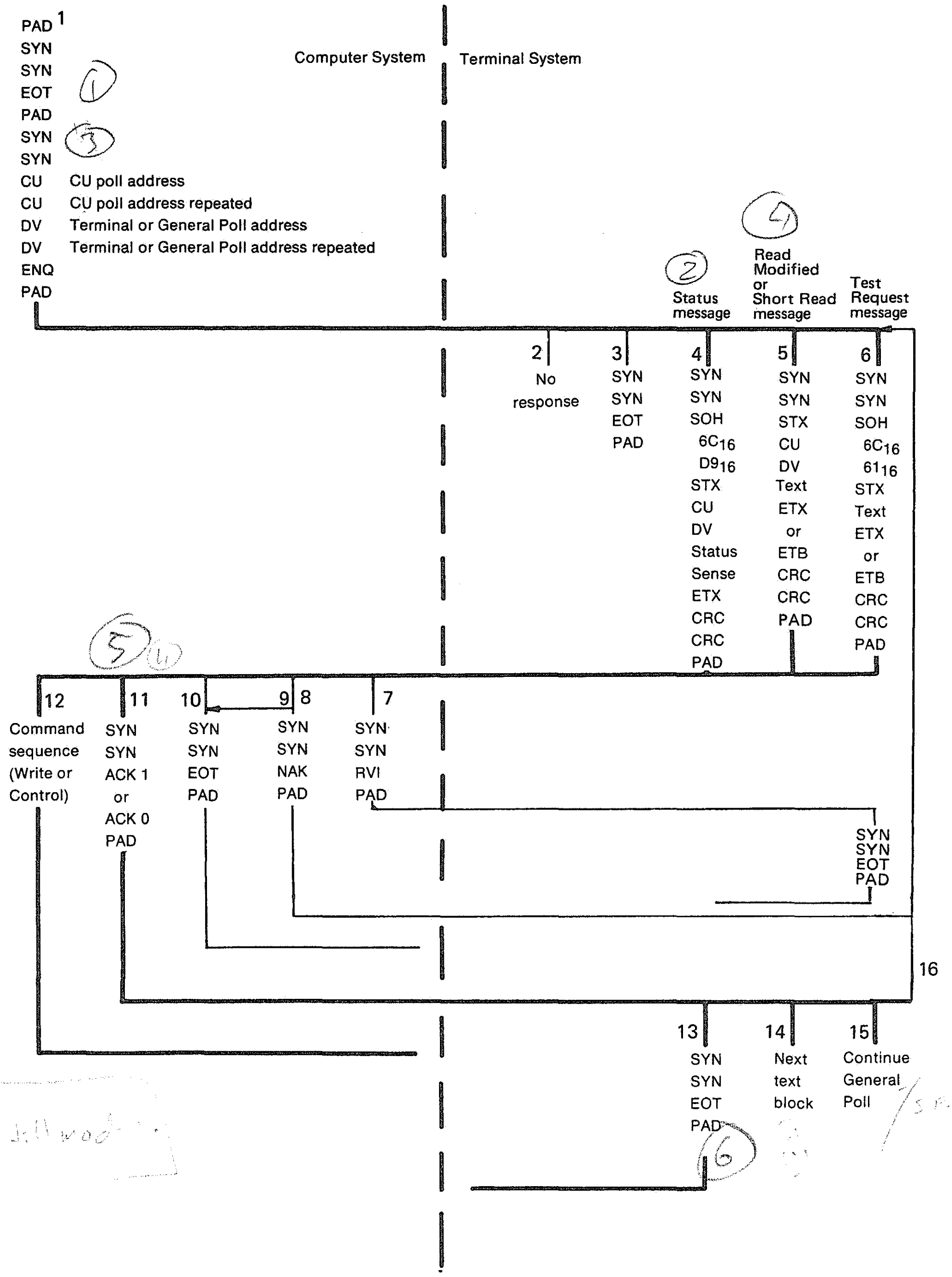


Fig. 1. General and Specific Poll, sequence diagram

Device not polled

Notes on Fig. 1:

1 The computer sends a polling sequence, either a General Poll or a Specific Poll. For information about addressing, see Table 1.

2 The terminal system does not respond for one of the following reasons
- The terminal is not available (power off or not connected)
- Erroneous polling sequence
- The addressed terminal was left selected from the previous transmission (only if EOT is missing in the polling sequence).

3 The terminal responds with EOT to indicate that it does not have any message to send. If a General Poll is being conducted, status information caused by Device Busy or an unavailable device is ignored and polling continues to the next terminal connected to the same communication processor. When all terminals have been examined without any response, EOT is sent.

4 A status message is sent in response to a General or Specific Poll if the terminal has status information that is waiting to be sent. Note that in a General Poll, status information caused by Device Busy or an unavailable device is not sent.

5 Normal response to polling when there is data to send. In such case, this is the first or only text block in the message and thus contains the original terminal address. Note that CU in this case refers to the CU Poll Address. See Table 1. If the message consists of only one text block, it is terminated by ETX. Otherwise it is terminated by ETB. The CU and DV addresses are included only in the first block of a blocked text message.

6 A Test Request message is initiated when the operator depresses the SYREQ key on the keyboard.

7 The computer sends RVI to interrupt the transmission in order to transmit a high priority message. The terminal responds by sending EOT and resets all waiting status/sense information.

8 The computer detects a CRC error and, using NAK, requests the last text block to be retransmitted.

9 If the terminal, in spite of repeated attempts, does not succeed in transmitting the message with a valid CRC character, communication is interrupted by EOT from the computer. Status/sense information is not cleared.

10 The computer has received a message but does not want any more messages from this communication processor thereby terminating the transmission by sending EOT. To retrieve possibly waiting messages on this communication processor, the computer must issue a new General Poll. Note that EOT does not reset status/sense information (except IR + DE) at the terminal.

11 The computer has received and approved a message. ACK 1 is sent in response to the first and all odd-numbered text blocks while ACK 0 is sent in response to all even-numered text blocks.

12 The command, which must be a write-type or control command, is sent either to the terminal that is selected by means of a Specific Poll or to the terminal which sent the last message in connection with the General Poll. In the latter case, the polling of the remaining terminals is interrupted within the communication processor. If they are to be examined, a new polling sequence must be sent from the computer.

13 Normal termination of a General or Specific Poll.

14 The second and all subsequent text blocks in a message do not have any CU and DV addresses, but in other respects are the same as the first text block. See 5.

15 The communication processor continues to send messages from the remaining terminals. See 3, 5 or 6.

16 If ACK is out of sequence, it is interpreted as a NAK by the terminal (note 8).

General and Specific Poll Addresses

The addresses of the communication processors are called the CU addresses. The addresses of the terminals are called DV addresses.

Table 1 presents the communication processor (CU) polling addresses and the terminal (DV) addresses.

Table 1. Communication processor (CU) polling addresses and terminal (DV) addresses

Communication processor or Terminal No.	Address $EBCDIC_{16}$	Communication processor or Terminal No.	Address $EBCDIC_{16}$	Communication processor or Terminal No.	Address $EBCDIC_{16}$
0	40	11	4B	22	D6
1	C1	12	4C	23	D7
2	C2	13	4D	24	D8
3	C3	14	4E	25	D9
4	C4	15	4F	26	5A
5	C5	16	50	27	5B
6	C6	17	D1	28	5C
7	C7	18	D2	29	5D
8	C8	19	D3	30	5E
9	C9	20	D4	31	5F
10	4A	21	D5		

The polling address sequences are arranged as follows

- General Poll
  CU polling address } see Table 1
  CU polling address }
  $7F_{16}$ } used instead of DV address
  $7F_{16}$ }

- Specific Poll
  CU polling address }
  CU polling address } see Table 1
  DV address }
  DV address }

Selection Addressing Sequence

A message that is to be sent from the computer must be preceded by a selection addressing sequence so that the computer can select the terminal associated with the message and ascertain whether or not the terminal can receive the message. The selection addressing sequence is the same as a polling sequence except that the characters used for the CU address differ.

When a valid selection addressing sequence has been received, the terminal answers by sending RVI, WACK or ACK 0. RVI is sent as a reply when the terminal has status and sense information to send. WACK is sent to indicate that the terminal is busy. In both of these cases, the addressing attempt is terminated by having the computer send EOT. ACK 0 is a positive acknowledgement from the terminal indicating that it is not busy and can thus accept the computer message. The fact that the computer accepts ACK 0 in response to a selection addressing sequence indicates that the terminal has been selected, and the computer can then continue by transmitting the command in question.

Fig. 2 presents the selection addressing sequence and the terminal system responses.

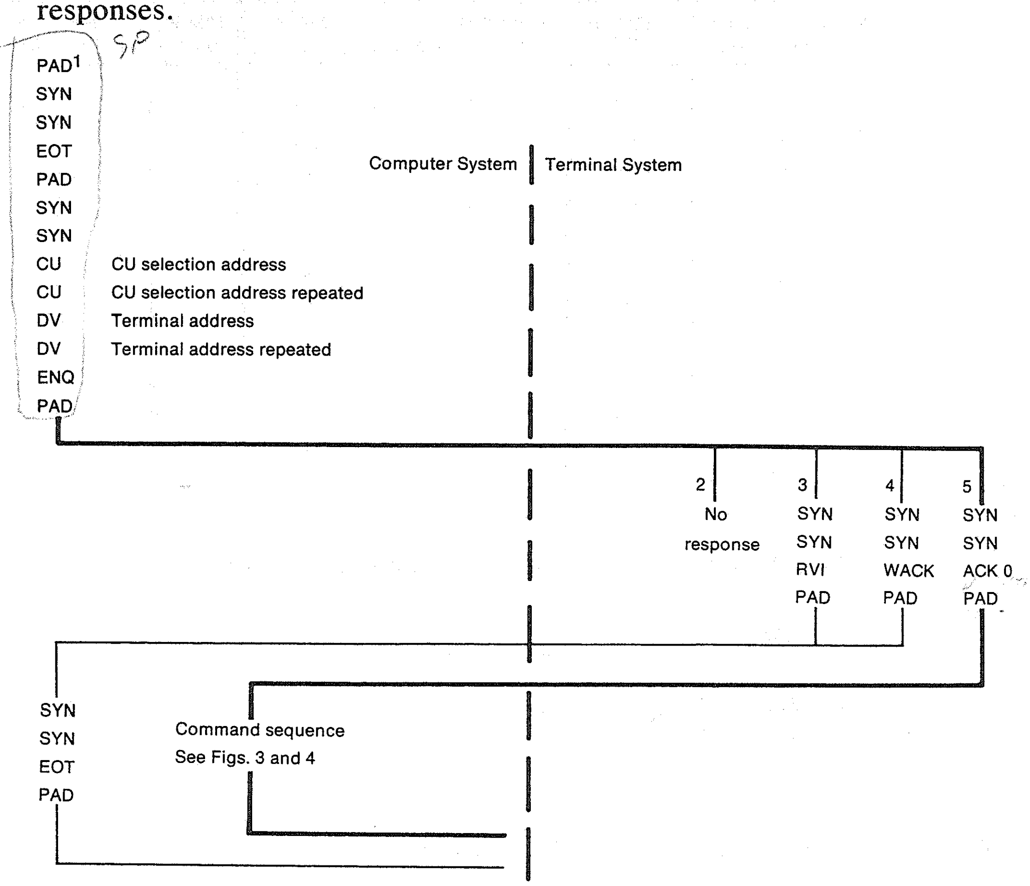


Fig. 2. Selection addressing, sequence diagram

Notes on Fig. 2:

1 The computer sends a selection addressing sequence. Note that the CU address differs from that used in connection with polling. Tables 1 and 2 list addresses.

2 The terminal does not respond because of one of the following reasons
- The communication processor or single display unit is unavailable (power off or not connected)
- Erroneous selection addressing sequence

3 By sending RVI, the terminal informs the computer that status/sense information (other than DE, DB) has been saved and that it must be fetched via a specific poll sequence before any new operation is undertaken. The computer responds by sending EOT in order to return the terminal to the control mode.

4 The terminal responds by sending WACK to indicate that it is busy with a previously called for operation. The computer answers by sending EOT in order to interrupt the addressing attempt.

5 A valid selection addressing sequence has been received and there is no status message.

Selection Addresses

The addresses of the communication processor are called CU addresses. The addresses of the terminals are called DV addresses.

Table 2 presents the communication processor (CU) selection addresses. The terminal addresses are the same in polling address sequences and selection address sequences.

Table 2. Communication processor (CU) selection addresses

Communication processor No.	Address $EBCDIC_{16}$	Communication processor No.	Address $EBCDIC_{16}$	Communication processor No.	Address $EBCDIC_{16}$
0	60	11	6B	22	F6
1	61	12	6C	23	F7
2	E2	13	6D	24	F8
3	E3	14	6E	25	F9
4	E4	15	6F	26	7A
5	E5	16	F0	27	7B
6	E6	17	F1	28	7C
7	E7	18	F2	29	7D
8	E8	19	F3	30	7E
9	E9	20	F4	31	7F
10	6A	21	F5		

The selection address sequence is arranged as follows

CU selection address }
CU selection address } see Table 2
DV address }
DV address } see Table 1

Write-Type and Control Command Sequences

Write-type and control commands can be issued immediately after a selection addressing operation or they can be chained to another command. The Write, Erase/Write, Erase/Write Alternate and Copy commands are always followed by a data sequence (text) consisting of at least one byte. After a positive acknowledgement from the terminal (ACK 0/ACK 1 indicates that the command has been received and executed), the computer can either terminate transmission or, if commands are chained, send another command to the same terminal. Fig. 3 presents a sequence diagram for a write-type or control command.

The remote write-type and control commands are

- Write, code $(F1)_{16}$
- Erase/Write, code $(F5)_{16}$
- Erase/Write Alternate, code $(7E)_{16}$
- Copy, code $(F7)_{16}$
- Erase All Unprotected, code $(6F)_{16}$

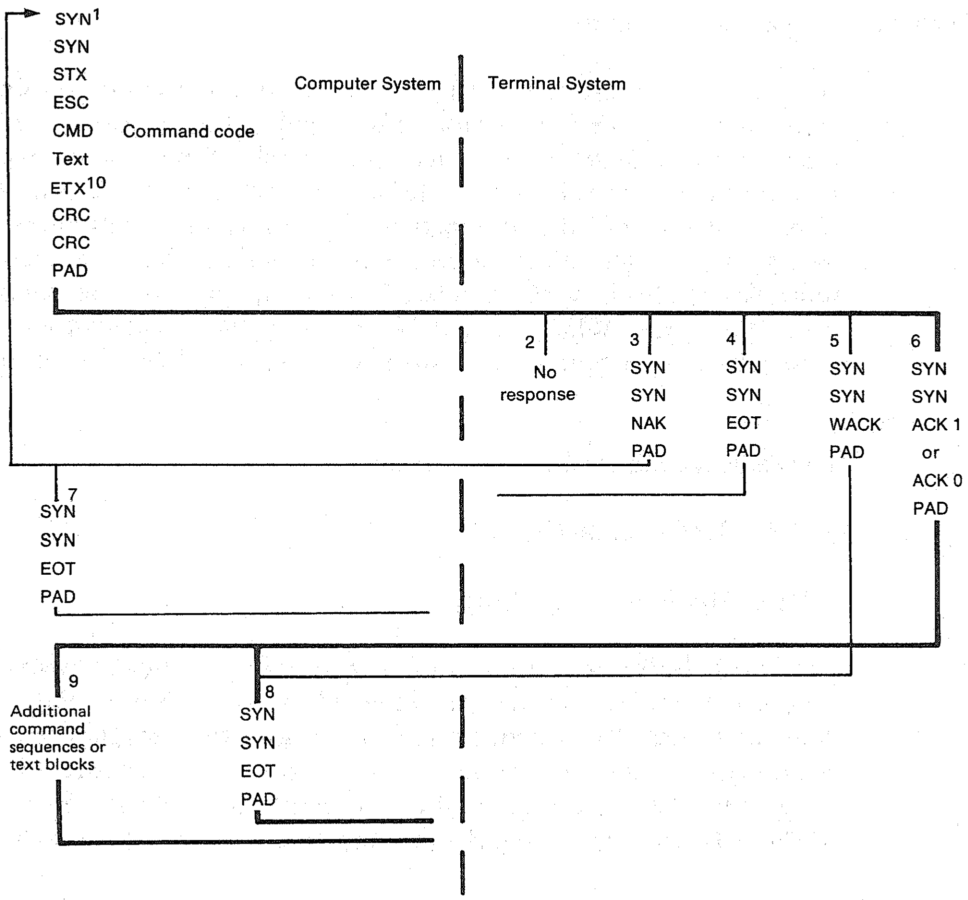


Fig. 3. Write-type and control commands, sequence diagram

Notes on Fig. 3:

1 The command sequence must be preceded by a polling or a selection addressing sequence or a command operation, i.e. the terminal must be selected. If the command is an Erase All unprotected command, no text is included in the message.

2 The command sequence was not received correctly, STX was lacking.

3 The terminal detects a CRC error or a missing ETX and requests, by means of NAK, that the command sequence be retransmitted. In the event of a CRC error, any other transmission error is not reported.

4 The terminal cannot carry out the operation specified by the command because it is busy or unavailable or because of an error condition. EOT indicates that the status and sense information is saved. In order to obtain this information, the computer must send a Specific Poll to the same terminal.

5 WACK indicates that the text has been received correctly and that the printer is now busy with a printout operation (the Start Printer bit is set in WCC or CCC). Consequently, no additional chained command can be accepted.

6 The terminal has received and executed the command. ACK 1 is sent in response to the first and all odd-numbered command sequences or text blocks while ACK 0 is sent in response to all even-numbered command sequences or text blocks.

7 If the computer, in spite of repeated attempts, cannot transmit the message with a valid CRC to the terminal, transmission is interrupted by means of EOT sent from the computer.

8 Normal termination of operation.

9 The operation continues with another write-type or control command (Fig. 3), a read-type command (Fig. 4) or another text block.

10 ETB is sent if the computer system blocks the text. ESC and the command code are not included in the second text block and following.

Read-Type Command Sequences

A read-type operation is carried out by having the computer first send a command sequence to the selected terminal and then read the response. A read-type command can be issued immediately after a selection addressing operation or chained to a write-type or control command. When the terminal has received and approved the command, it responds by transmitting the first text block to the computer which (after a successful cyclic redundancy check, CRC) sends ACK 0 or ACK 1 in order to obtain the next text block. When all text blocks have been transmitted, the communication is terminated by means of EOT sent from the terminal. See Fig. 4.

The remote read-type commands are

- Read Buffer, code $(F2)_{16}$
- Read Modified, code $(F6)_{16}$

The Read Buffer command is used mainly for diagnostic purposes while the Read Modified command is intended for transmissions that are dependent upon measures taken by the operator at the terminal. However, in remote applications, read operations are normally associated with General or Specific Polls (see Fig. 1) which automatically initiate a Read Modified or Short Read operation in the terminal when it has data to transmit.

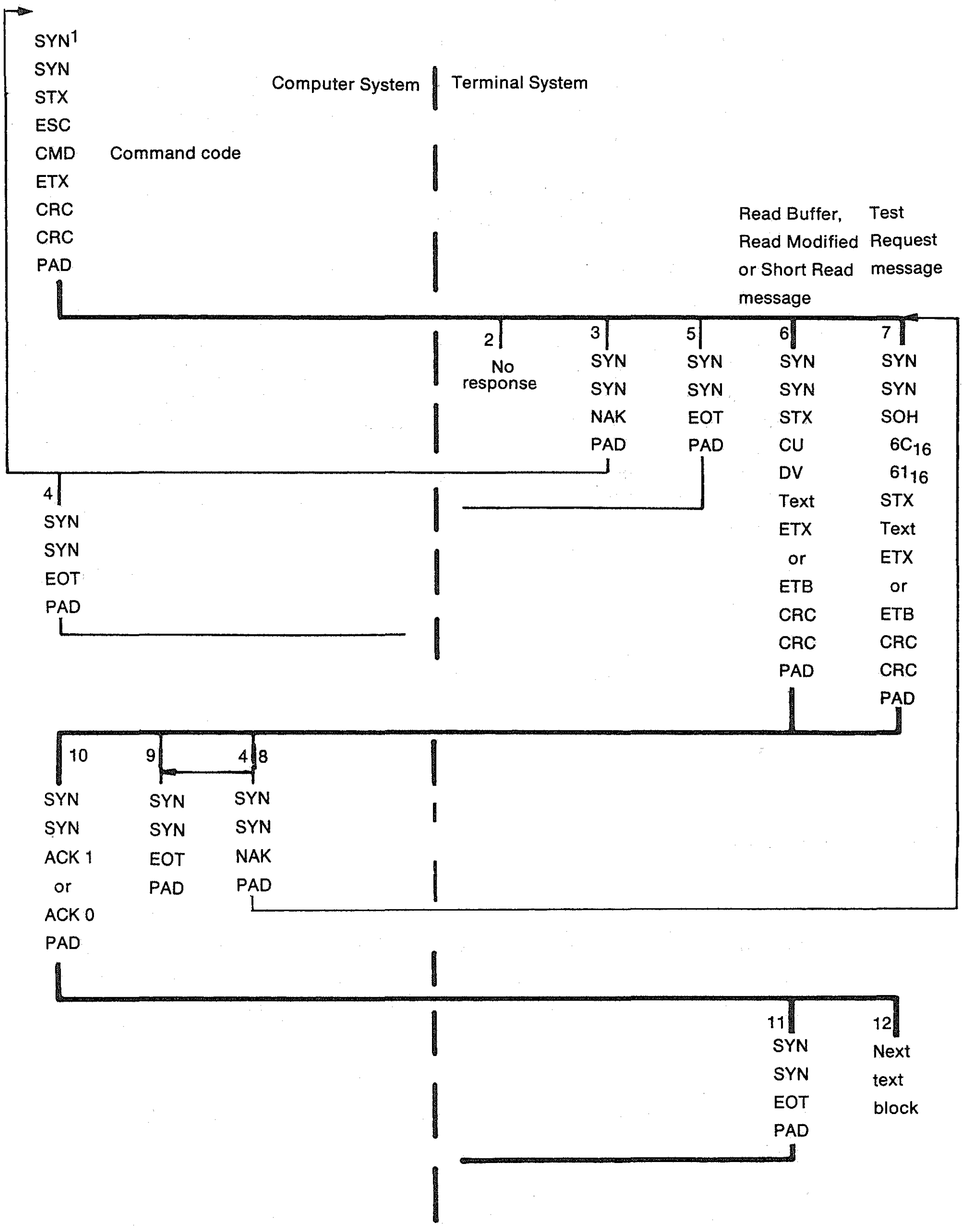


Fig. 4. Read-type command, sequence diagram

Notes on Fig. 4:

1 The command sequence must be preceded by a selection addressing sequence or a command sequence or by a polling sequence which resulted in a positive response, i.e. the terminal must be selected. Text is never included in a read-type command sequence.

2 The command sequence has not been received correctly, STX was lacking.

3 The terminal detects a CRC error or a missing ETX and requests, by means of NAK, that the command sequence be retransmitted. In the event of a CRC error, any other transmission errors are not reported.

4 If, in spite of repeated attempts, the message cannot be transmitted with a valid CRC character, transmission is interrupted by means of EOT sent from the computer.

5 The terminal cannot execute the operation specified by the command because it is busy or unavailable or because of an error condition. EOT indicates that status and sense information is saved. In order to obtain this information, the computer must send a Specific Poll to the same terminal.

6 Normal response to a Read Buffer or Read Modified command when data is available for transmission. In such a case, this is the first or only text block in the message and thus contains the original terminal address. Note that CU in this case refers to the CU poll address. See Table 1. If the message consists of only one text block, it is concluded with ETX. Otherwise it is concluded with ETB. See 12.

7 Response to a program-generated Read Modified command obtained when the operator depresses the SYREQ key on the keyboard.

8 The computer detects a CRC error and, by means of NAK, requests the last message from the terminal to be retransmitted.

9 The computer has received a message but does not want any more messages from this communication processor. It thus terminates the transmission by sending EOT. To retrieve any remaining text blocks, the computer must perform a new selection. Note that EOT does not reset status/sense information (except IR + DE) at the terminal.

10 The computer has received and approved a text block. ACK 1 is sent in response to the first and all odd-numered text blocks, while ACK 0 is sent in response to all even-numbered text blocks.

11 Normal termination of the operation when the last text block has been transmitted to the computer.

12 The second and all subsequent text blocks in a message have no CU/DV address, but are otherwise the same as the first text block. See 6. The last text block in the message is concluded with ETX.

Chaining of Commands

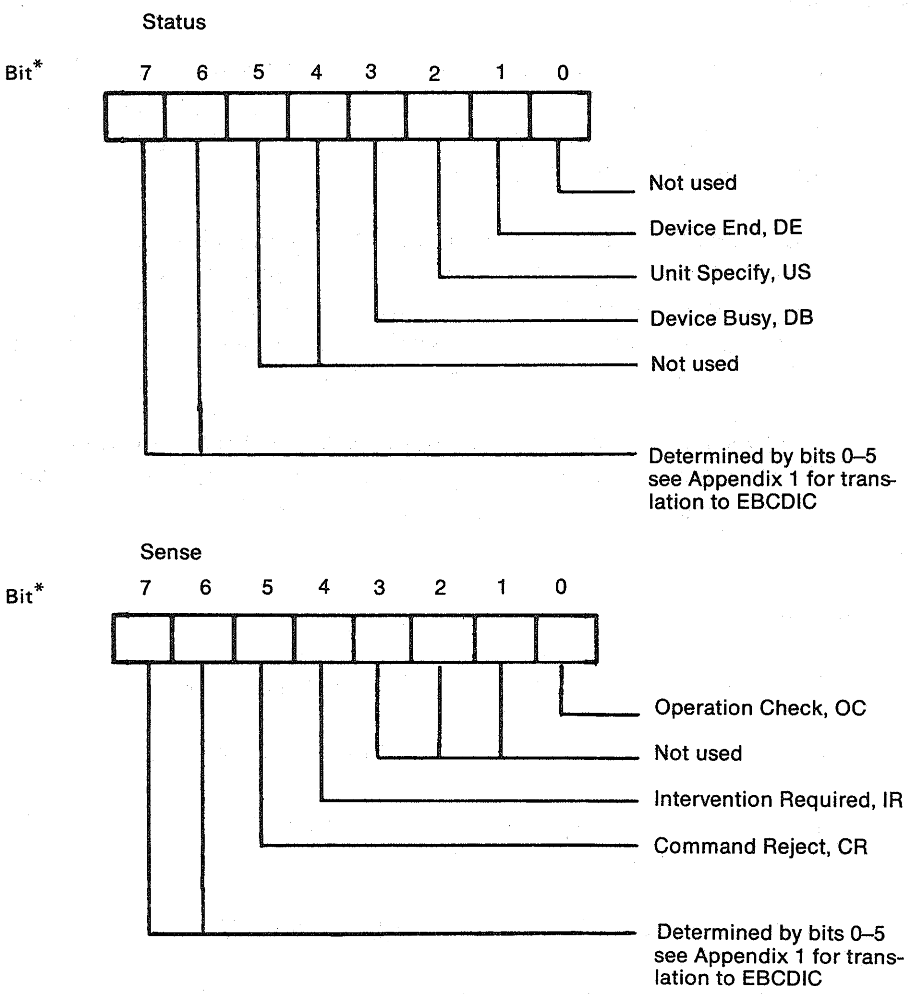
When commands are chained, more than one command sequence is sent to a terminal after a polling sequence or a selection addressing sequence. In a transmission involving chained commands, the computer can, for example send a write-type command followed by another write-type command.

Commands following a selection addressing sequence can be chained provided that the BSC rules valid for the Limited Conversational mode are followed.

Only write-type and control commands can be chained in connection with a polling sequence. All types of commands except Erase All Unprotected can be chained to a Write, Erase/Write, Erase/Write Alternate or Copy command, but if the command causes a print operation to be started, the next command on the chain is inhibited due to the fact that the terminal sends WACK to the computer.

Status and Sense Information

A terminal in which an error condition exists always replies by sending status and sense information in response to the next poll from the computer. This information is cleared when the computer has received and approved the message, but it can also be cleared by means of RVI sent out from the computer. All status and sense conditions are assembled into two bytes called Status and Sense. Both of these bytes are always sent in a status message. The layout of this message appears in the sequence diagram for the General and Specific Polls shown in Fig. 1. Fig. 5 presents the bit assignments for both Status and Sense, and the paragraphs which follow present a detailed description of each status bit.



*Note: IBM designates the most significant bit (bit No. 7) as No. 0.

Fig. 5. Bit assignments for Status and Sense information bytes

Device End, DE

The Device End bit, set alone, indicates that the addressed terminal has changed its status from unavailable to available or from busy to not busy. If it is set, the Device End bit is always included in a response to a General or Specific Poll, but it is not considered waiting status by a selection addressing sequence.

If a selection addressing sequence detects waiting status in the addressed terminal and the terminal is also available or not busy, the Device End bit is set and saved along with the other waiting status while RVI is sent as a response to the computer.

Unit Specify, US

The Unit Specify bit is set

- Together with some status and sense bits to define various states of the terminal, see below
- When a command is addressed to a busy terminal

Device Busy, DB

The Device Busy bit indicates that the addressed terminal is busy with an operation or that it was found busy by a previous command or Specific Poll. The terminal is busy when it is queuing for a local printout operation that is to be performed.

The Device Busy bit is set together with the Unit Specify bit when a command is addressed to a busy device. This can occur by chaining a command to a Write, Erase/Write, Erase/Write Alternate, or Copy command which started a printer or by chaining a command to a specific Poll addressed to a busy device.

The Device Busy bit is set together with the Operation Check bit and the Unit Specify bit when a Copy command is received with the "from" address equal to the "to" address when the device is busy.

Operation Check, OC

The Operation Check bit indicates, when set alone, that

- CCC or a "from" address is lacking in a Copy command
- An invalid command sequence has been received (ESC is lacking in the second character position)
- The terminal buffering capability is exceeded

The Operation Check bit is set together with the Unit Specify bit to indicate that the "from" address on a Copy command specified a device with a locked buffer.

The Operation Check bit is set together with the Intervention Required bit when a Copy command contains an unavailable "from" address.

Command Reject, CR

The Command Reject bit is set when an invalid command is received.

Intervention Required, IR

The Intervention Required bit is set when

- A Copy command contains an unavailable "from" address
- A printer which is not ready receives a start order
- A selection addressing sequence or a Specific Poll sequence is obtained for a terminal which is unavailable or has become not ready during a printout operation
- A command is sent to a terminal which is unavailable or not ready.

Contents

Illustrations

Tables

Abbreviations

A	Address (SDLC)
ACTLU	Activate logical unit
ACTPU	Activate physical unit
BB	Begin bracket
BBI	Begin bracket indicator
BC	Begin chain
BCI	Begin chain indicator
BETB	Between brackets
BIU	Basic information unit
BLU	Basic link unit
BTU	Basic transmission unit
CCN	Cluster controller node
CD	Change direction
CDI	Change direction indicator
CP	Communication processor
CUCN	Communications controller node
CRC	Cyclic redundancy check
DACTLU	Deactivate logical unit
DACTPU	Deactivate physical unit
DAF	Destination address field
DFC	Data flow control
DISC	Disconnect
DLC	Data link control
DM	Disconnected mode
DR	Definite response
DR1	Definite response bit 1
DR2	Definite response bit 2
DSC	3270 data stream compatibility
EAU	Erase all unprotected
EB	End bracket
EBCDIC	Extended binary coded decimal interchange code
EBI	End bracket indicator
EC	End chain
ECI	End chain indicator
EFI	Expedited flow indicator
ER	Exception response
ERI	Exception response indicator
ERP	Error recovery procedure
EU	End user
EUA	Erase unprotected to address

FCS	Frame check sequence (SDLC)
FI	Format indicator
FIC	First in chain
FID	Format identification (field)
FIS	First in segment
FM	Function management
FMD	Function management data
FRMR	Frame reject
HDX	Half duplex
HDX-C	HDX contention
HDX-FF	HDX flip-flop
Hex	Hexadecimal
I	Information
ID	Identifier; identification
INB	In bracket (state)
LIC	Last in chain
LIS	Last in segment
LU	Logical unit
LUSTAT	Logical unit status
MIC	Middle in chain
MIS	Middle in segment
MPF	Mapping field
NAU	Network addressable unit
NC	Network control
NDM	Normal disconnected mode
NRM	Normal response mode
NRZ	Non-return to zero
NRZI	Non-return to zero invert
Nr	Receive number (SDLC)
Ns	Transmit number (SDLC)
OAF	Origin address field
OIC	Only in chain
OIS	Only in segment
PC	Path control; Program check
PEND BB	Pending begin bracket
P/F	Poll/final bit
PI	Pacing indicator
PIU	Path information unit
PLU	Primary logical unit
PS	Presentation services
PU	Physical unit; Printer unit

RA	Repeat to address
RCV	Receive
RH	Request/response header
RNR	Receiver not ready
RQD	Request specifying definite-response
RQE	Request specifying exception-response
RQN	Request specifying no response
RR	Receiver ready
RRI	Request/response indicator
RSP	Response (+RSP = positive response; –RSP = negative response)
RTI	Response type indicator
RTR	Ready to receive
RU	Request/response unit
S41	Alfaskop System 41
SBA	Set buffer address
SC	Session control
SCC	Synchronous communication controller
SCS	SNA character string
SDI	Sense data included indicator
SDLC	Synchronous data link control
SDT	Start data traffic
SHUTC	Shutdown complete
SHUTD	Shutdown
SLU	Secondary logical unit
SNA	System network architecture
SNF	Sequence number field
SNRM	Set normal response mode
SSCP	System services control point
TC	Transmission control
TH	Transmission header
TN	Terminal node
TS	Transmission subsystem
UA	Unnumbered acknowledge
WCC	Write control character

Remote Operation – SNA/SDLC

The Alfaskop System 41 can emulate an IBM 3274 model 1C, that uses the SNA/SDLC protocol.

In order for one unit in a Systems Network Architecture (SNA) communications system to communicate with another, several formally bound pairings called sessions must be established between them. A session is a logical connection which governs the protocol between, and the capabilities of, the session partners (the aforesaid units). These units must be provided with enough resources to support the session or sessions in question. The set of resources provided within a single unit to support a single session is called a half-session.

This chapter provides information about the half-session resources and other SNA functions which are supported by the Alfaskop System 41 terminals. The SNA concept itself is not explained. However, certain general information about SNA is nonetheless provided to make this chapter easier to understand.

Notes on Information Appearing in the Chapter on Operational Characteristics

Generally speaking, operational characteristics do not depend on the communications system. However, the ATTN/SYREQ key has special functions in SNA. The function obtained if this key is depressed while the shift key is up (ATTN) is described in the section entitled Signal, and the function obtained while the shift key is down (SYREQ) is described in the section entitled Session Interaction.

In an SNA environment, most of the messages appearing on the message line refer specifically to SNA. All information appearing on the message line appears in Appendix 4.

Notes on Information Appearing in the Chapter on Commands and Orders

The layouts of the data streams and the use of commands and orders sent within the data stream are as described in the chapter on Commands and Orders, with the following exceptions

- Chaining has a different meaning in an SNA environment. Here, a chain is used to link request units (RUs), and only one command is allowed per chain. Moreover, the command byte must be sent as the first byte in the first RU containing data.

  Command chaining cannot be used in an SNA environment. Consequently, all commands are regarded as unchained in SNA.

- The Read Modified All command, which is used only in SNA, can be sent from the host. This command causes data from all modified fields on the screen to be sent (message layout as shown in Fig. 2, Chapter 4, Commands and Orders). This applies also to those send-initiating keys which otherwise initiate a short read operation.

- The screen sizes called for by the Erase/Write command (default size) and the Erase/Write Alternate command (alternate size) are specified by means of parameters in the SNA command called Bind when a session is established.

- The Copy command cannot be used in SNA. However, a copy function is included. If, in a Write-type command, the start printer bit is set, the text sent from the computer is displayed on the screen of the addressed display unit and also printed out on a printer that is logically attached to the display unit in the local mode or shared mode (via the printer authorization matrix).

- The terms "link header" and "link trailer" appear in the message layout illustrations in the chapter on Commands and Orders. In an SNA environment, the SDLC control headers as well as the transmission and request/response headers have to be considered part of the "link header".

- The SYREQ key does not initiate a test request read operation in SNA. Refer to the section entitled Session Interaction for information about the operation of the SYREQ key.

SNA Overview

Layer Approach

One important aspect of the SNA is that functional responsibilities and capabilities are structured as layers. The three major functional layers are shown in Fig. 1a.

These major layers can be divided into sublayers which have different components tailored to the services provided. Fig. 1b presents a sublayer structure suitable for transferring application data through the communications system.

Application Layer

The application layer deals with the processing of application data. It consists of end users (EU). The EUs comprise the source and destination of information flowing through the communications system. The EUs are thought of as being outside the communications system. They access the communications system via a logical unit (LU). The LUs are described later. An EU can consist of an application program in a host computer, the operator at a terminal, a physical device, a physical medium, etc.

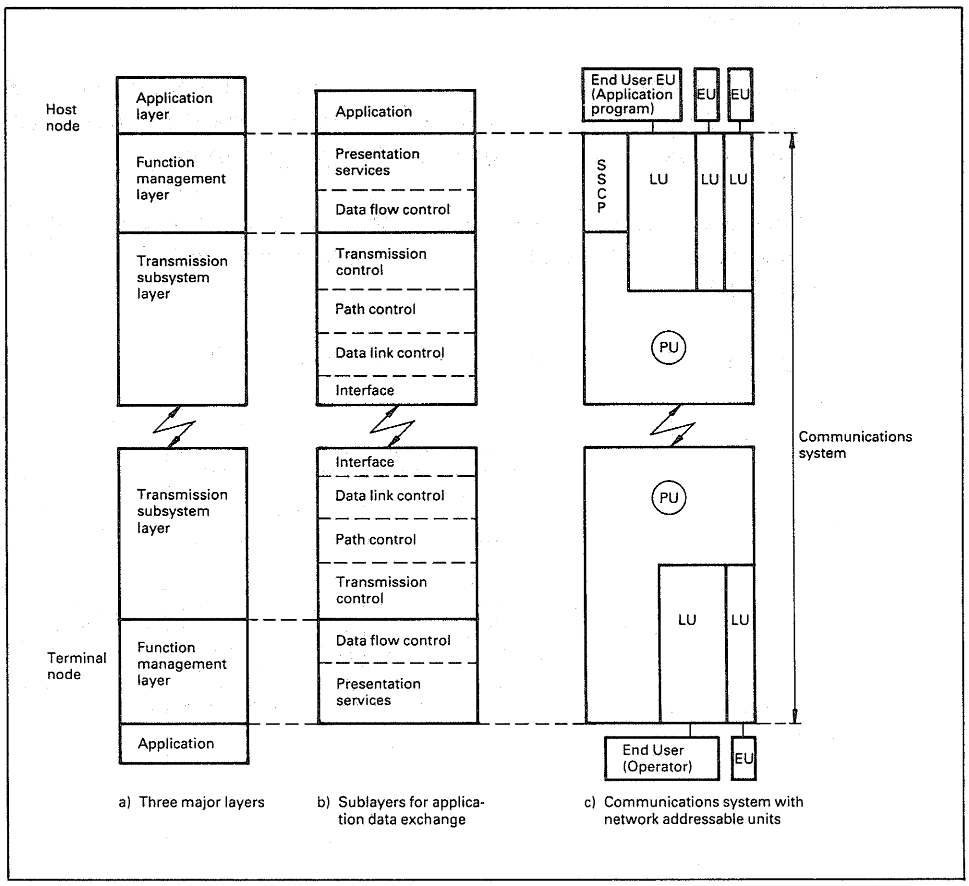


Fig. 1. SNA layering arrangement

Function Management Layer

The function management layer controls the presentation format of information sent to and from the application layer and manages the protocols which support the exchange of EU information. Examples of the services provided by the function management layer include

- Initiation of connection requests between LUs
- Mapping of device-specific characteristics
- Data format conversion
- Controlling the flow of data between LUs

Transmission Subsystem Layer

The transmission subsystem layer controls the movement of data through the network independently of the characteristics of the destination and the content of the data itself. Paths through the network incorporate several nodes and data links shared by multiple applications. Each node has a transmission subsystem layer which manages these shared resources and utilizes the physical connections optimally. One purpose of the transmission subsystem layer is to maintain data integrity between the function management layers of the originating and destination nodes.

SNA Network Nodes

The major components used in the system are called nodes. A node can thus be defined as a point in the network consisting of a physical unit; a node can also comprise some level of programming.

There are four types of nodes: the host node, the communications controller node (CUCN), the cluster controller node (CCN) and the terminal node (TN).

A group of nodes is called a subarea. See Fig. 2.

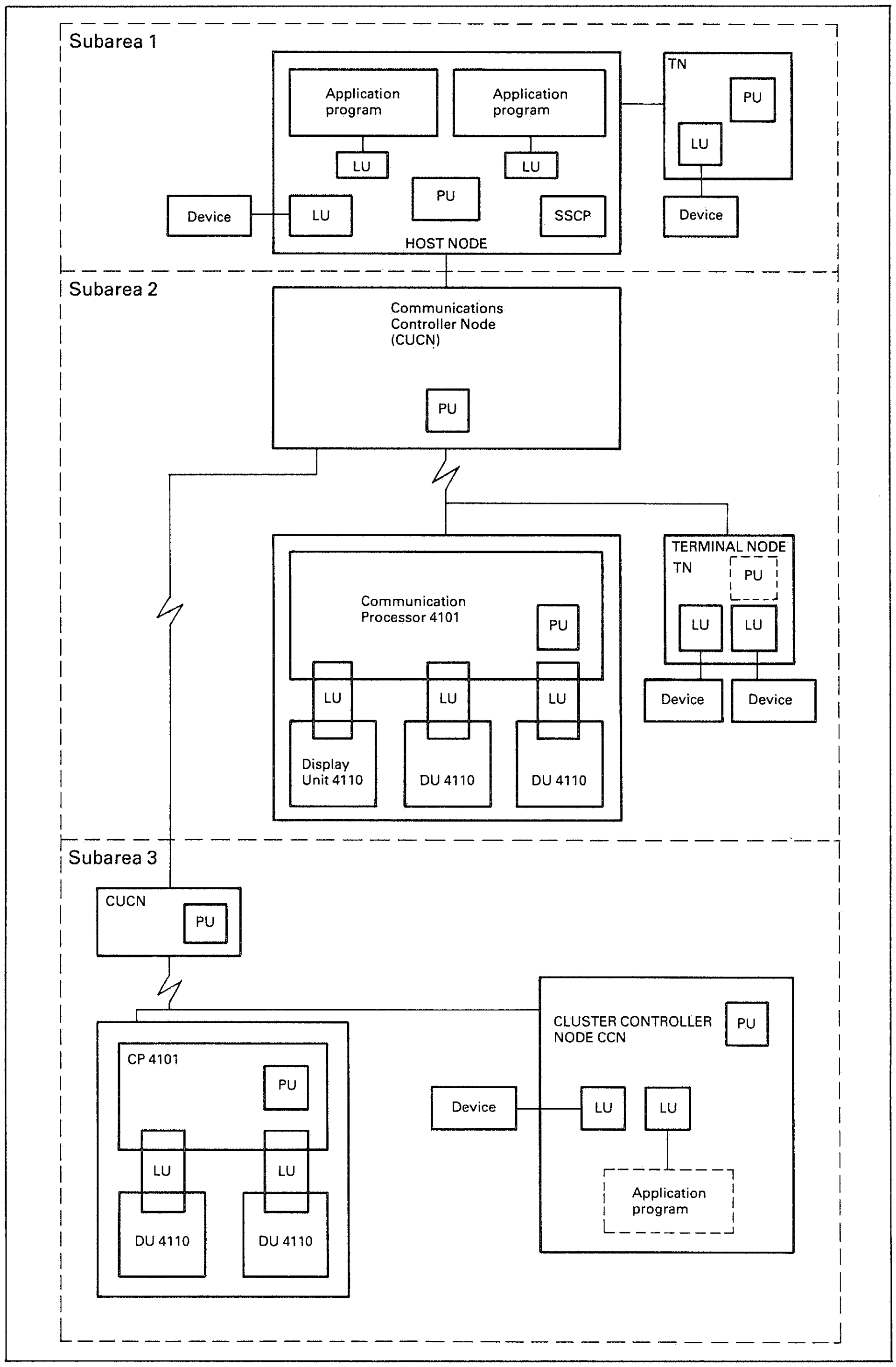


Fig. 2. Relationships between NAUs, nodes and subareas

Host Node

The host node is equivalent to a central processing unit provided with an operating system, a data base and access methods. The host node manages the network, processes data bases, executes application programs and carries out similar tasks.

The host node together with any locally connected cluster controller nodes and terminal nodes form one subarea.

Communications Controller Node (CUCN)

The communications controller node supports end user communication by interconnecting other network nodes. Its responsibilities include

- Controlling communication lines
- Reducing the network management responsibilities of the host node
- Providing data integrity and error recovery within the network

A communications controller node together with the cluster controller nodes and terminal nodes connected to it form one subarea.

Cluster Controller Node (CCN) and Terminal Nodes (TN)

The cluster controller node and the terminal node enable remote locations to access the data base located at the host node. A number of devices can be attached to these nodes. A terminal node has less processing capability than a cluster controller node, which may include application programs for local data processing.

Network Addressable Units (NAU)

Three types of addressable entities which traffic can flow to and from are used in SNA. These are called network addressable units (NAU) and they are associated mainly with the function management layer (see Fig. 1c). They are called

- Logical unit (LU)
- Physical unit (PU)
- System services control point (SSCP)

Logical Unit (LU)

A logical unit is the port through which an end user (EU) accesses the network. It provides the services and protocols which an EU needs in order to communicate with other EUs.

Before two EUs can exchange information, a session must be established between their respective LUs. In the communications system, the EUs are represented by the associated LUs. Each application program must be represented by an LU which is tailored to run with that application program.

SNA defines several types of LU. The following types are encountered for Alfaskop System 41.

- Type 1 – the Alfaskop System 41 component is a printer and the data stream is an SNA character string (SCS)
- Type 2 – the Alfaskop System 41 component is a Display Unit 4110, and the data stream is formatted as set forth in the chapter on Commands and Orders (3270 data stream compatibility (DSC) format)
- Type 3 – the Alfaskop System 41 component is a printer and the data stream is in the DSC format

The two LU half-sessions that are in session must be of the same type. The host half-sessions take the role of primary logical units (PLU) while the terminal half-sessions take the role of secondary logical units (SLU).

Physical Unit (PU)

A physical unit is a set of management services that control the physical resources of the node and its associated devices and/or links. Functions carried out by the physical unit include

- Initialization of a node that is to operate as part of the network
- Activating network links from a communications controller node
- Assistance with recovery in the event of a network communication failure

The physical unit manages physical resources in response to requests from the SSCP.

System Services Control Point (SSCP)

A system services control point can be found only in host nodes. It is responsible for the overall management of the network. The SSCP components perform three types of network services

- Configuration services
- Maintenance services
- Session services

Configuration services include such operations as starting, closing down, restarting, activating and deactivating network links and devices. They are also used to change the status of logical units and physical units wherever necessary. Maintenance services carry out the testing of network facilities and they also collect and record error information. Session services open and close sessions between end users, and support these sessions by undertaking certain responsibilities such as resolving symbolic names to full network addresses.

The SSCP is always involved in establishing EU-EU sessions. Another feature of the SSCP is that it uses sessions between itself and all other logical units and physical units in the network to manage the network elements under its control. Its other point of communication is the network operator who can issue commands to the SSCP in order to direct overall operation of the network and obtain information about its status.

Fig. 2 presents an example showing relationships between NAUs and the different types of nodes.

SNA Data Transfers and Headers

Whenever data is transmitted within an SNA system it is referred to either as a request or a response. A request can consist of application data, and it can also consist of SNA commands used to control the network, the sessions, the data flow and the like. A response is a formal acknowledgement sent in reply to a previously received request. The units of data are referred to as request/response units (RUs).

As RUs pass through the various layered components that lie on the path between the sending and receiving end users, headers (and trailers in some layers) are added by components in the sending node. These headers are used by equivalent layer components in the receiving node where they are stripped off. These headers are the means by which peer layers communicate with each other. Intermediate nodes strip off only enough headers to check the address of the receiving node. They then formulate a new header which is suitable for the next link. Fig. 3 shows how the different headers are added and stripped off, although it does not show any intermediate node.

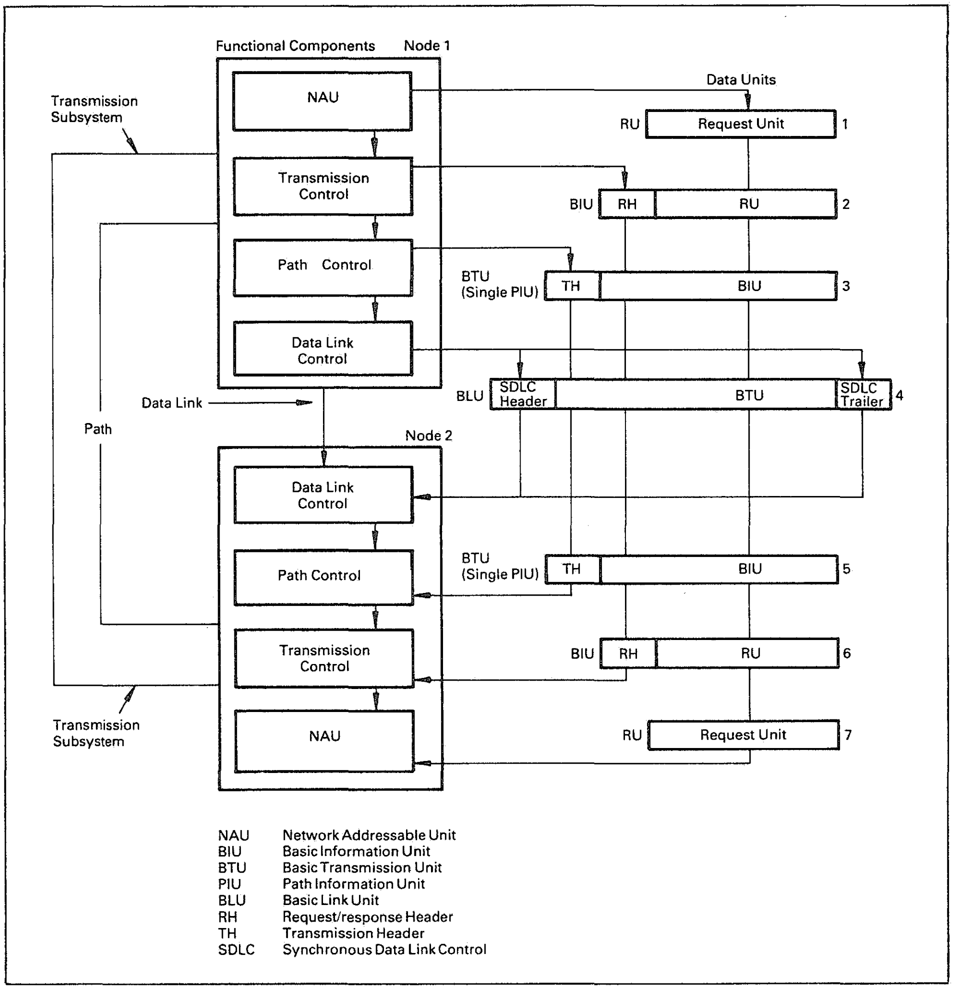


Fig. 3. Diagram showing how headers are added and stripped off

Request/Response Header (RH)

The Request/Response Header consists of three bytes which contain information such as the following

- A request/response indicator showing whether the message is a request or a response
- An indicator showing what form of response is called for
- An indicator showing whether the RU contains end user data or control information
- An indicator that governs the data flow protocols that are used
- An indicator that identifies the chained element (first, middle or last) if an RU is chained. All elements between the first element and last element are considered middle elements.
- A response-type indicator showing whether a response is positive or negative

Transmission Header (TH)

The four most significant bits in the first byte in the TH form a format identification field (FID) which specifies the TH format. The only format encountered in Alfaskop System 41 is format 2 (0010). A 6-byte TH will thus contain the following

- Format identification field
- Mapping field indicating the order of segments in segmented messages (segmenting is carried out by the path control layer when the BIU length is longer than the maximum length specified for the data link). See Fig. 4.
- An indicator that identifies the type of data flow (normal or expedited). Expedited data can bypass normal data at certain points in the network.
- Sequence number of the message. The sequence number of a response must be the same as the sequence number of the associated request.
- Origin address and destination address. The FID2 format and local 8-bit addresses are used within the subarea in which the Alfaskop System 41 is connected. Translation to full (16-bit) network addresses is carried out in the communications controller node to which the terminal node is connected.

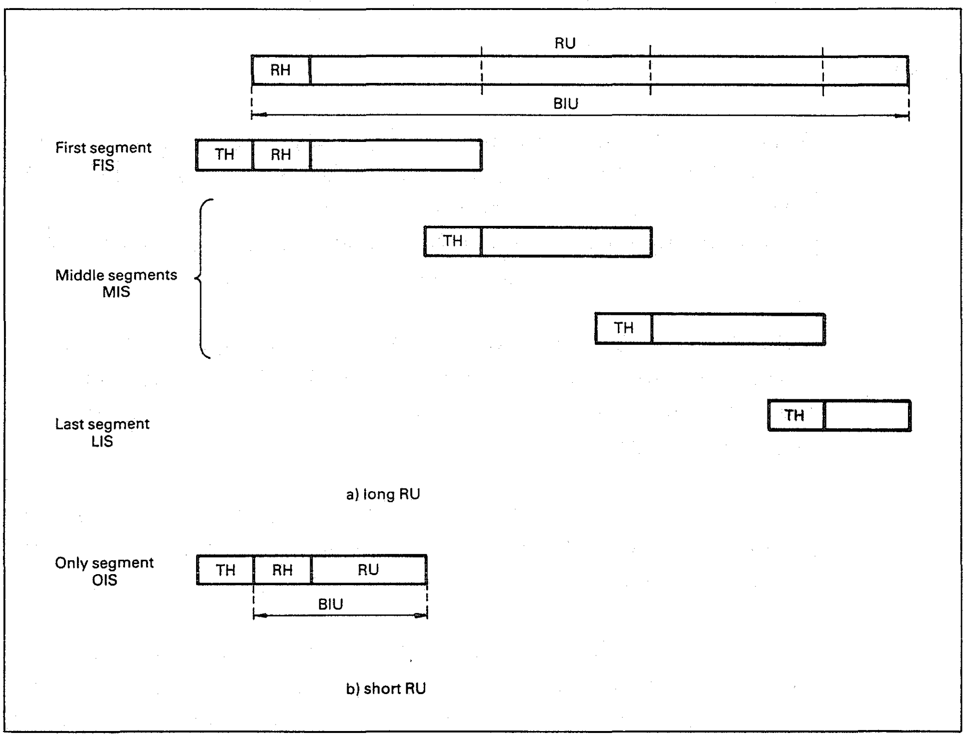


Fig. 4. Segmenting of RUs

Data Flow Types

The term "outbound" is used for transmission from the host to the Alfaskop System 41 terminals. The term "inbound" is used for transmission from the Alfaskop System 41 terminals to the host.

Two types of data flow – normal data flow and expedited data flow – can occur between the host and the terminals

- Normal data flow refers to an orderly flow of requests and responses within the logical path between two LUs. During normal flow, requests and responses must arrive at their destinations in the same order in which they were sent. The sequence number which is located in the TH of the PIU can be monitored to check the order.
- Expedited flow is used in certain situations where delays cannot be tolerated. Requests and responses that are sent using expedited flow can bypass those sent using normal flow. Moreover, expedited requests and responses flowing in one direction can take precedence over normal requests and responses flowing in the other direction. This occurs, for example, when two communicating NAUs use the half-duplex protocol. This protocol allows only one NAU to send using normal flow at any given time. The session partner of the sending NAU must not send concurrently using normal flow. However the receiving NAU can send using expedited flow while its session partner is sending using normal flow.

  Expedited flow permits control requests to be sent in either direction without being restricted by normal flow.

Chains

A chain is a complete unit of data that originates at a single LU. RU chaining provides a method of defining a complete unit of data logically when it consists of a series of consecutive RUs. Each RU can only be associated with one chain. Note, however, that the word "chain" as used here sometimes refers to a single-element chain.

Indicators in the Request Header (RH) specify whether the request is a single-element chain or whether it is the first request in a multi-element chain, the last request in a multi-element chain or somewhere in the middle of a multi-element chain.

Chains can be sent using either normal or expedited flow. However, the only type of chain that can be sent using expedited flow is a single-element chain. Normal flow can be used for both single- and multi-element chains.

It is only possible to send one chain in a given direction at a time. This means that for normal flow, the last request in a chain must be sent before the first request in the next chain can be sent in the same direction.

Responses

Three forms of responses to a request can be asked for by means of indicators in the request header – a definite response, an exception response and a no response

- In response to a request that specifies a ***definite response,*** the receiving network addressable unit (NAU) sends a positive response (+RSP) if the request is accepted and a negative response (–RSP) if the request is rejected.
- In response to a request that specifies an ***exception response,*** the receiving NAU sends a negative response if the request is rejected but sends nothing if the request is accepted.
- In response to a request specifying a ***no response,*** the receiving NAU does not send any response.

The three following types of chains can be sent

- Definite response chain. This is a chain of requests in which the last request specifies a definite response while all other requests specify exception responses.
- Exception response chain. This is a chain of requests in which each request specifies an exception response.
- No response chain. This is a chain of requests in which each request specifies a no response.

Pacing

The purpose of pacing is to control normal data flow between nodes in such a way as to prevent one node from overloading another node. Expedited data flow cannot be paced. The sending node is only allowed to send a limited number of requests before it receives a pacing response. When the pacing response is received, it indicates that the next series of requests (up to the pacing limit) can be sent.

Data Flow Control Modes

One or more of the following modes must be used within a session

- Immediate control mode. Here, outbound (relative to the host) and inbound data flows are independent of each other. A second request cannot be sent until a response has been received for the first, i.e. definite responses must be specified.
- Immediate request mode. Here, a sending network addressable unit (NAU) can have only one definite response outstanding against a request chain at any time. However,one or more exception responses may be outstanding, but the chain specifying an exception response can only be sent if no definite response is outstanding.

- Delayed request mode. Here, the sending NAU can have any number of responses outstanding at any given time.

- Immediate response mode. Here, the receiving NAU must send responses in the same order in which the associated requests were received.

- Delayed response mode. Here, the receiving NAU can send the responses in any order. (The sequence number in TH associates each response with its request.)

The immediate control mode, the immediate request mode or the delayed request mode can be used for normal flow in either direction throughout a session. On the other hand, the immediate control mode must be used for expedited flow.

The receiving NAU must be in the immediate response mode when the sending NAU is in the immediate request mode.

Data Flow Protocols

The half duplex protocols restrict data transmission for the normal flow. They specify that only one LU is to send requests at a given time. This means that the session partner of the sending LU must not send concurrently using normal flow. Two types of half duplex protocols are used: the half duplex flip-flop (HDX-FF) protocol and the half duplex contention (HDX-C) protocol.

- When a session in HDX-FF is initiated, one of the LUs is specified as the first to send. This first sender can then notify the other LU that it (the other LU) can send by setting the change direction (CD) bit in the RH of a request that is sent. The two LUs can then continue to switch back and forth between sending and receiving until the session ends. This means that the CD bit must be set in the last element in a chain.

- When a session in HDX-C starts, both LUs are in contention to send. Either one can send first, thus forcing the other to become the receiving LU. After the last element in the chain has been sent, both LUs return to the contention state. If contention results in a "tie", the winner is the LU that was so specified at session initiation time.

The bracket protocol is also used to restrict data transmission for normal flow. It provides a mechanism that can be used to group a sequence of LU-LU information exchanges into an uninterruptable unit of work. This means that a complete transaction routine will be finished before any request belonging to another transaction can be sent.

Sessions Overview

Note that in this section and the following sections, designation S41 refers to the Alfaskop System 41 complete with all of its components.

The following sessions must be established between the host system and the S41 in order for information to be exchanged:

- SSCP-PU (access method – S41 physical unit (PU))
- SSCP-SLU (access method – S41 SLU)
- PLU-SLU (host program – S41 SLU)

Moreover, a session must be established between SSCP-PLU (access method – host program). The third type of session listed above is frequently referred to as the LU-LU session. PLU means primary logical unit and SLU means secondary logical unit.

The above sessions are discussed in the sections which follow. These sections also present the SNA commands that establish and terminate the sessions and explain how the sessions are established and terminated. The section headed Command Description presents the SNA commands in considerable detail.

SSCP-PU Session

The physical connection to the host must be established before the SSCP-PU session is established. The SSCP-PU session must be the first session that is established between the S41 and the host system. When the S41 is activated by the network operator, the SSCP issues the Activate Physical Unit (ACTPU) command to the communication processor. Moreover, a predefined SSCP start procedure can request the activation of specific communication processors.

The network operator can deactivate the S41 and terminate the SSCP-PU session. Then the SSCP issues the Deactivate Physical Unit (DACTPU) command when all SSCP-LU sessions for the communication processor in question have been terminated. The SSCP-PU session is considered terminated when the S41 sends a positive response to the DACTPU command.

SSCP-SLU Session

After the SSCP-PU session has been established, an activate command can be issued to the SSCP calling for the establishment of the SSCP-SLU session. An Activate Logical Unit (ACTLU) command is issued by the SSCP for the proper SLU(s) in the S41.

The SSCP-SLU session must be established before the LU-LU session is established.

The SSCP-SLU session is terminated when a Deactivate Logical Unit (DACTLU) command is sent to the associated SLU. The SSCP-SLU session is terminated when the S41 sends a positive response to the DACTLU command.

PLU-SLU (LU-LU) Session

Fig. 5 presents the command flow sequence that is needed to establish an LU-LU session. Generalized nomenclature is used in this illustration. In this example, it is assumed that none of the sessions have been active between the S41 and the host. The SSCP issues the ACTPU command to establish the SSCP-PU session (A in illustration). Then, ACTLU commands (B) are sent to establish the SSCP-LU sessions. The host application can establish the SSCP-PLU session at any time prior to logon.

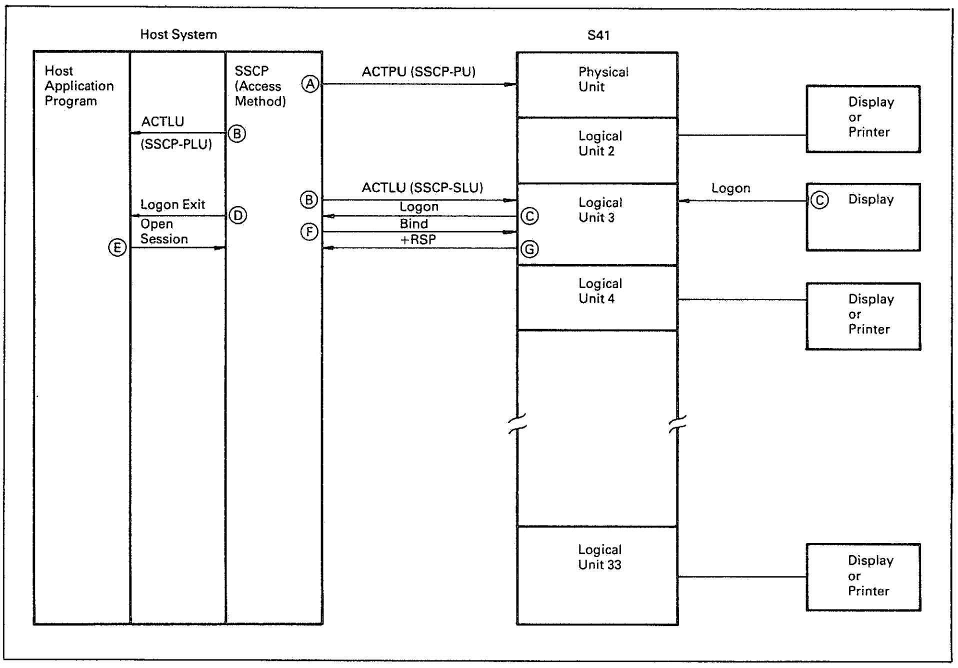


Fig. 5. Establishing a session with an S41

The display terminal operator (C) can initiate the LU-LU session (character coded logon) or the host application program can initiate it (using a simulated logon or by acquiring the terminal for example). If the SSCP receives a character coded logon, it translates the logon request and schedules a logon exit (D) for the PLU. The PLU sends an open session request to the SSCP (E) after the PLU has received control at the logon exit or when the PLU acquires the terminal. As a result of this open session request, an SNA Bind (F) command is sent to the SLU.

The S41 LU checks the session parameters in the Bind command and (if they are approved) permits the session to be established. This is accomplished by sending a positive response (G) to the Bind command. However, in the event that the session parameters cannot be approved, or if power to the device is not turned on, the S41 LU rejects the Bind command. This is accomplished by sending a negative response which indicates the reason for rejection. The table of Bind parameters in Appendix 5 presents the Bind parameters that can be specified for an S41 session. The host program can issue the Start Data Traffic command after the Bind command has been approved. This will permit data traffic to start flowing for the session.

If the LU-LU session is a type 1 session or type 3 session, it must be initiated by the PLU. However, if it is a type 2 session, it can be initiated by either the PLU or the SLU.

Termination of an LU-LU Session

An LU-LU session can be terminated by having the PLU request that the SSCP close the session. After this request has been issued, the SSCP forwards the Unbind command to the secondary LU, thus terminating the LU-LU session.

The terminal operator can terminate a type 2 session in two ways:
a) by notifying the PLU participating in the session that termination is desired, whereupon the PLU terminates the session; or
b) by changing over from the LU-LU session to an SSCP-SLU session by depressing the SYREQ key and then entering a logoff message. If this logoff message is conditional, the logoff request is sent to the PLU by the SSCP. If the logoff message is unconditional, an Unbind command is issued for the PLU.

The session can be closed by having the PLU issue a Shutdown command. When the PLU issues the Shutdown command, the S41 replies with the Shutdown Complete command after any outstanding operation has been completed, providing that there are no unclosed brackets.

Profiles and Usage Fields

Several parameters are sent to the S41 together with the ACTPU, ACTLU and Bind commands. These parameters specify operations within the sessions and they are presented in Appendix 5.

The resources and functional capabilities of a physical or logical unit (e.g. the capability to handle commands) which are to be used in a session are grouped in profiles. The profiles are assigned numbers so that they can easily be specified in the parameters. Other parts of the parameters which are used to further qualify or select functions are called usage fields. The profiles and the usage fields are related to the sublayers where the various functions are supported.

For operation of the presentation services sublayer, the PS profile and usage field specify the following (among other things)

- LU Type
- Screen (or buffer) size

For operation of the data flow control sublayer, the FM (function management) profile and usage field specify the following (among other things)

- Supported DFC commands
- Request and response modes
- Chaining and bracket rules
- Forms of responses
- Data flow protocol

For operation of the transmission control sublayer, the TS (transmission subsystem) profile and usage field specify the following (among other things)

- RU size
- Pacing
- Sequence numbering
- Supported SC commands

Headers

Each segment sent to or from the S41 is provided with a 6-byte TH. Each request and each response sent to or from the S41 is provided with a 3-byte RH, which is sent in the first segment if the RU is segmented. In the following description, some bits define unused fields or are not used. Bits which are not used are handled by the S41 as follows

- They are not checked on reception
- They are sent in a response just as they were received in the associated request
- They are set to zero in a request

Transmission Header (TH)

The format of the TH and an explanation of its contents appear in Fig. 6. The addresses of the displays or printers (each device is represented by an LU) and the address of the PU of the communication processor are listed in Table 1. These addresses appear in DAF for outbound segments and in OAF for inbound segments. The SSCP in the host is identified using address 0 in OAF for outbound segments and in DAF for inbound segments.

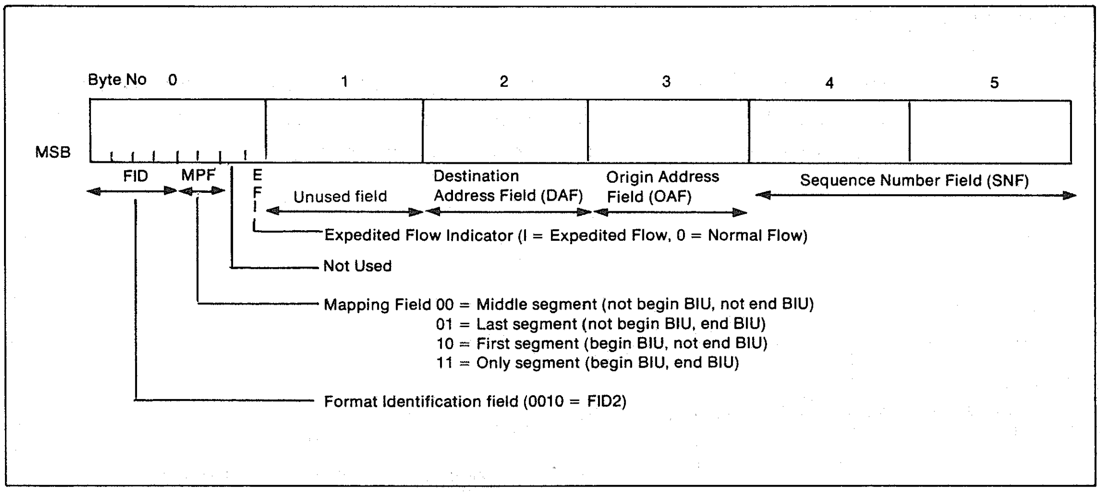


Fig. 6. Transmission header format

Each request sent during normal flow in an LU-LU session must be provided with a sequence number. The sequence number is initialized to zero (after the Start Data Traffic command) and is incremented by one before a new request is sent. Two individual sequence numbers are used: one for outbound flow requests and one for inbound flow requests. All segments of one request have the same sequence number. Each request in a chain has its own sequence number. The sequence number for a response must be the same as the number used for the associated request.

Each request sent during expedited flow or in sessions other than LU-LU sessions is assigned an identifier number rather than a sequence number. Any number can be used as an identifier.

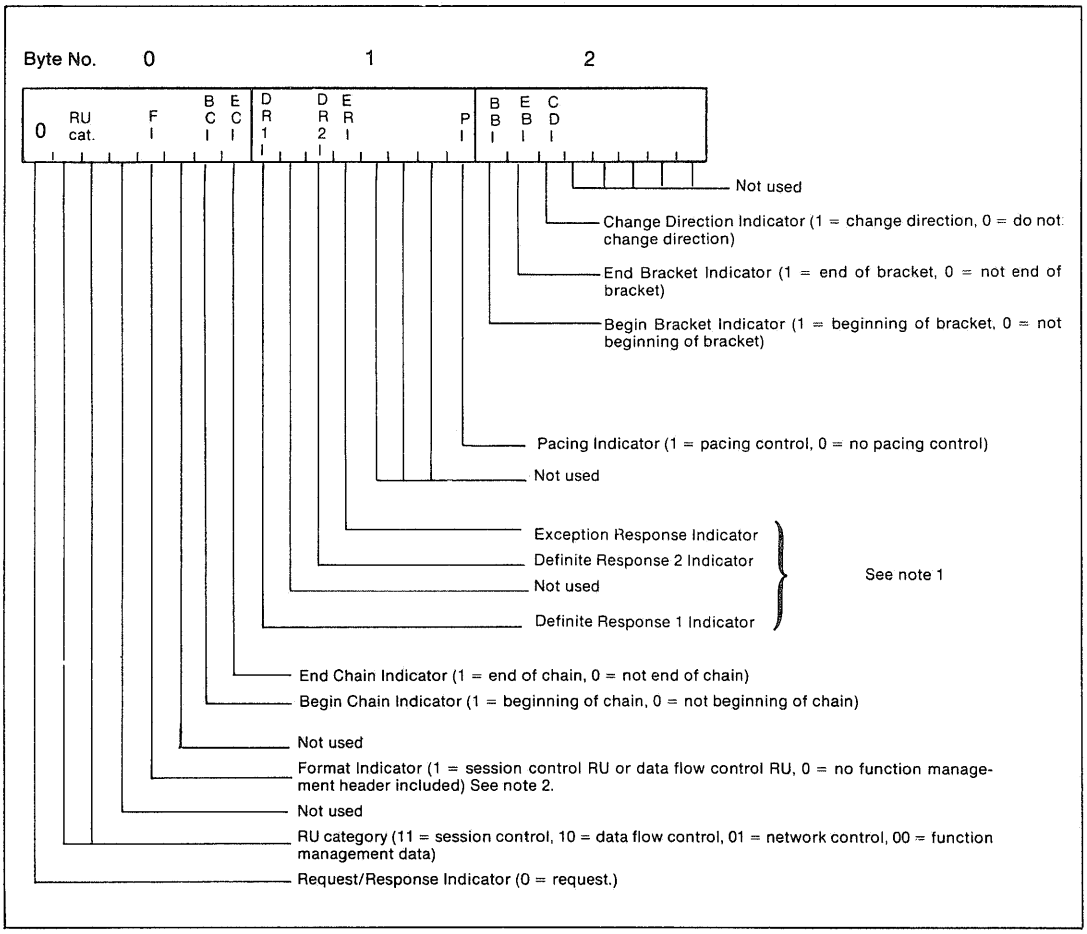
Table 1. Device addressing for Alfaskop System 41 terminals

Device Number	Device Address Field Bits: 7	6	5	4	3	2	1	0 *
PU	0	0	0	0	0	0	0	0
**	0	0	0	0	0	0	0	1
0	0	0	0	0	0	0	1	0
1	0	0	0	0	0	0	1	1
2	0	0	0	0	0	1	0	0
3	0	0	0	0	0	1	0	1
4	0	0	0	0	0	1	1	0
5	0	0	0	0	0	1	1	1
6	0	0	0	0	1	0	0	0
7	0	0	0	0	1	0	0	1
8	0	0	0	0	1	0	1	0
9	0	0	0	0	1	0	1	1
10	0	0	0	0	1	1	0	0
11	0	0	0	0	1	1	0	1
12	0	0	0	0	1	1	1	0
13	0	0	0	0	1	1	1	1
14	0	0	0	1	0	0	0	0
15	0	0	0	1	0	0	0	1
16	0	0	0	1	0	0	1	0
17	0	0	0	1	0	0	1	1
18	0	0	0	1	0	1	0	0
19	0	0	0	1	0	1	0	1
20	0	0	0	1	0	1	1	0
21	0	0	0	1	0	1	1	1
22	0	0	0	1	1	0	0	0
23	0	0	0	1	1	0	0	1
24	0	0	0	1	1	0	1	0
25	0	0	0	1	1	0	1	1
26	0	0	0	1	1	1	0	0
27	0	0	0	1	1	1	0	1
28	0	0	0	1	1	1	1	0
29	0	0	0	1	1	1	1	1
30	0	0	1	0	0	0	0	0
31	0	0	1	0	0	0	0	1

*Note: IBM designates the most significant bit (Bit No. 7) as No. 0.
**Unused address.

Request/response Header (RH)

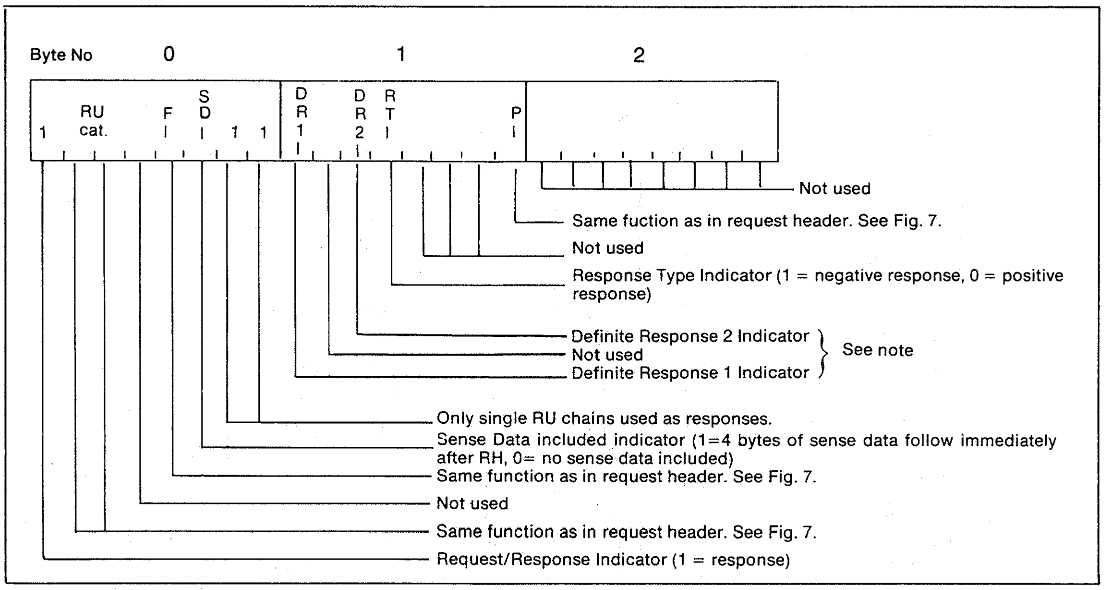
Both the request header and the response header contain these bytes, but the meanings of some bits differ. Figs. 7 and 8 present the meanings of all bits in the request header and response header respectively.



Note 1: Bits DR1I, DR2I and ERI are used as follows: The AS41 interprets 010, 100 and 110 as definite-response requests and 011, 101 and 111 as exception-response requests. The S41 sends 100 to indicate a definite-response request and 101 to indicate an exception-response request (in multiple RU definite-response chains, the S41 sends 101 for all RUs except the last RU and 100 for the last RU).

Note 2: The S41 does not support FM headers. Consequently, FI is always set to 0 in FM data RUs.

Fig. 7. Request header format



Note: The S41 sends the DR1I and DR2I bits as received in the corresponding request.

Fig. 8. Response header format

SNA Command Overview

SNA commands are used to establish and terminate sessions and to manage data flow and sessions (between the host and the S41).

Two types of SNA commands are supported by S41

- Data Flow Control (DFC) commands which control the flow of data during an LU-LU session
- Session Control (SC) commands which are used to establish and terminate network sessions

There are two other types of SNA commands which are not supported by the S41, namely the network control (NC) commands and function management data (FMD) commands.

All commands listed as remote commands in Table 1 in the chapter entitled Commands and Orders (except copy) can be used in connection with data transfers between PLU and SLU.

Table 2 presents the SNA commands supported by the S41.

Table 2. SNA commands supported by Alfaskop System 41.

SNA command				Received by S41		Sent by S41	
Name	Code	Type	Normal or Expedited	From	To	From	To
ACT PU	11	SC	E	SSCP	PU		
DACT PU	12	SC	E	SSCP	PU		
ACT LU	0D	SC	E	SSCP	SLU		
DACT LU	0E	SC	E	SSCP	SLU		
BIND	31	SC	E	PLU	SLU		
UNBIND	32	SC	E	PLU	SLU		
SDT	A0	SC	E	PLU	SLU		
CLEAR	A1	SC	E	PLU	SLU		
LUSTAT	04	DFC	N			SLU	PLU
RTR	05	DFC	N			SLU	PLU
CANCEL	83	DFC	N	PLU	SLU	SLU	PLU
CHASE	84	DFC	N	PLU	SLU		
SHUTD	C0	DFC	E	PLU	SLU		
SHUTC	C1	DFC	E			SLU	PLU
BID	C8	DFC	N	PLU	SLU		
SIGNAL	C9	DFC	E	PLU	SLU	SLU	PLU
*	*	FMD	N	PLU	SLU	SLU	PLU

* For FM data, please refer to chapter on Commands and Orders.

Command Description

Activate Physical Unit (ACTPU)

The ACTPU command is sent by the SSCP in order to establish an SSCP-PU session with the communication processor. This session is established when the communication processor returns a positive response to the ACTPU command. If the ACTPU command is received while SSCP-SLU and LU-LU sessions are active, these sessions are terminated immediately. The communication processor then sends a positive response and re-establishes the SSCP-PU session.

Deactivate Physical Unit (DACTPU)

All LU-LU and SSCP-SLU sessions as well as the SSCP-PU session are terminated when the S41 receives the DACTPU command. The S41 returns a negative response with sense data indicating that PU is not active if some command other than ACTPU is received after a positive response has been sent back for the DACTPU command.

The same terminations also occur if the SDLC command SNRM is received on the communication link.

Activate Logical Unit (ACTLU)

The SSCP establishes an SSCP-SLU session with each LU associated with the S41 by means of an ACTLU command. The SSCP-SLU session is established when the S41 returns a positive response. This occurs if the

SSCP-PU session has been established before the ACTLU command was received. Note that a positive response can be returned even though the display unit or printer in question is not connected or not provided with power.

If an SSCP-SLU session has been previously established and the S41 receives an ACTLU command for the LU in question, any active session between this LU and a PLU is terminated. The S41 returns a positive response to the ACTLU command, whereupon the SSCP-SLU session is re-established.

Deactivate Logical Unit (DACTLU)

When the DACTLU command is received, the SSCP-SLU session is terminated. If, when the DACTLU command is received, an LU-LU session has been established, it is terminated. A negative response is returned with sense data indicating that SLU is not active if the S41 receives a command other than DACTPU, ACTPU or ACTLU after a positive response has been returned for the DACTLU command.

Bind

The Bind command is issued by SSCP to request an LU-LU session between an S41 LU and an application program. The session is established when the S41 returns a positive response. The S41 may establish an LU-LU session only if an SSCP-LU session exists for the LU in question.

The parameters received with the Bind command are checked by the S41. The parameters are used mainly to define the protocols to be used in the session. Appendix 5 presents detailed descriptions of the session parameters that can be sent with the Bind command. A negative response is returned if the operations specified in the parameters cannot be performed or if the device is physically detached from the communication processor or if the power is not turned on for the device.

If the SSCP-SLU session is established and the S41 receives some command other than Bind that normally flows in the LU-LU session, a negative response is returned with sense data indicating that no session has been established.

When one Bind command has been accepted (i.e. an LU-LU session exists) and the S41 receives another Bind command for the same LU, a negative response is returned with sense data indicating that a session has already been established.

Unbind

When the Unbind command is received, the S41 terminates the PLU-SLU session and sends a positive response.

Clear

When the Clear command is received, the S41 imposes the data traffic reset state on the LU-LU session and also initializes all inbound and outbound transmission buffers.

Start Data Traffic (SDT)

After the S41 has received the SDT command and returned a positive response, data traffic is permitted to flow during an LU-LU session. This command must be issued after a Bind command has been used to establish the LU-LU session. Moreover, this command is sent after a Clear command in order to complete a session resynchronization sequence with the S41. The SDT command is only valid when the LU-LU session is in the data traffic reset state.

Cancel

The Cancel command is used when the transmitter wants the receiver to cancel (discard) all elements of the incoming chained transmission. However, the S41 processes data RUs and sends them to the display or printer as they arrive without waiting for the end of the chain. The Cancel command thus only provides a proper termination (without discarding) for a chain which is otherwise incomplete.

If a Cancel command is received between chains, it affects only the S41 state that is governed by the end bracket (EB) and change direction (CD) bits in the RH that is associated with the Cancel command.

If the S41 returns a negative response to an element of a chain, the PLU should terminate the transmission of this chain and issue the Cancel command (unless the last element in the chain has already been sent to the S41).

A Cancel command issued by a type 2 SLU asks the PLU to stop processing a chained transmission and cancel (discard) all elements of the chain that are currently being received. If the S41 fails or if operator action prevents the transmission of all data, the Cancel command serves as a substitute for the end of the chain.

If the PLU returns a negative response for an element of a chain, the entire chain will be transmitted before the PLU response is examined by the S41, and no Cancel command will be sent. Consequently, the PLU should cancel (discard) all elements of the chained transmission after it has sent a negative response.

Chase

The Chase command is used to ascertain whether or not all preceding requests have passed through the network and have been processed. The S41 returns a positive response to this command after all previous chains have been processed. Before issuing the Chase command, the PLU should complete or cancel the current chained transmission. EB or CD can be used with the Chase command.

Bid

The Bid command is sent to the S41 by the PLU to request permission to begin a bracket. The Bid command can be used to prevent long chains of data from taking up transmission time and then being cancelled (discarded) due to the fact that the S41 emerged as the winner of bracket contention. A positive response is returned if the Bid command is accepted by the SLU, whereupon the SLU waits for a request containing BB. The S41 rejects the Bid command if it has won bracket contention. Information about the sense codes used in connection with rejection of a Bid command appears in Appendix 6.

Signal

The PLU can send the Signal command to the S41 to request the setting of the Change Direction (CD) bit. The SLU will then complete any chained transmissions that are in progress and send the CD to the PLU. If the SLU is in the send state but has not yet started transmitting, it will send a request containing CD but no data (that is to say, a null-RU) and change to the receive state. A positive response is sent to the Signal command but no SLU action is taken if the SLU is already in the receive state, the BETB state or the ERP1 state (the states are explained in the section on Session States).

The S41 sends the Signal command when the terminal operator depresses the ATTN key on the keyboard. The Signal command is sent using expedited flow and has no effect on the state of the S41. If the ATTN key on the keyboard is depressed again after the Signal command has been sent, a second Signal command will not be issued unless the S41 has received a response to the first Signal command.

LU Status (LUSTAT)

The LUSTAT command is sent by the S41 to notify the PLU that a change in the operation status of a device has occurred or that an error has been detected. The S41 sends a 4-byte status code which specifies the device status change or the error condition. The LUSTAT command thus provides a means whereby the SLU can report exception conditions or status asynchronously, i.e. even if no request is received from the PLU. However, if the S41 is in the receive state it can only send negative responses and cannot send LUSTAT. Appendix 6 presents the LUSTAT codes and also the conditions that determine which LUSTAT code is to be sent.

Ready to Receive (RTR)

A type 1 or type 3 S41 LU sends the RTR command to indicate when a previously rejected bracket can be initiated by the PLU. An RTR command can only be accepted when the session is ready to receive a new bracket. When the RTR command is sent and a positive response is received from the host program, the printer LU expects the PLU to begin a bracket.

Shutdown

When the Shutdown command is received, the S41 prepares for a session termination sequence. A positive response is returned to the PLU, but data transfer sequences are not inhibited. When the S41 has completed normal flow processing it enters the between (inter) bracket state (BETB) and sends the Shutdown Complete command to the PLU. After that, the S41 cannot send any data, and the keyboard is locked.

The PLU can use the Shutdown command in two situations

- When terminating the session. Here, the PLU issues the Unbind command after the Shutdown Complete command is received.
- To stop traffic. Clear and SDT commands must then be issued in order to start traffic from S41 again.

Shutdown Complete

The Shutdown Complete is sent by the S41 after the Shutdown command has been received from the PLU and the shutdown has been completed. At this point in time, the S41 cannot send any data transmissions to the PLU, but the PLU can continue to send data to the S41.

FM Data Transfers

Function management data is transferred in SSCP-SLU and PLU-SLU sessions. In a PLU-SLU session, the data stream complies with the chapter on Commands and Orders with the exceptions pointed out at the beginning of this chapter. The data may only be sent if data traffic is allowed (SDT has been issued and a positive response has been received) in the LU-LU session. In the SSCP-LU session only write-type commands are used; further information is presented in the section entitled Operation in SSCP-SLU sessions.

The S41 supports the following FM data protocols

Bracket: The bracket protocol is used to delimit a series of related inbound and outbound FM data request units (RUs) – for example all of the RUs needed to complete a transaction.

Chaining: The chaining protocol provides a logical connection between one or more RUs associated with a single LU – for example all of the RUs needed for a complete display image.

Half duplex flip-flop: The half half duplex flip-flop protocol makes use of the change direction (CD) bit to inform the receiving LU that the sending

LU has completed transmission and expects the next transmission to be sent out by the receiving LU.

Bracket Protocol

The bracket protocol, as mentioned previously, is used to delimit a series of related inbound and outbound requests. A bracket can consist of a series of requests flowing in a single direction, many sets of inputs and outputs or just one input and one output. Two bits are used to delimit the bracket, namely the begin bracket (BB) bit and the end bracket (EB) bit. The Bid command and the Ready to Receive (RTR) command are used to facilitate the initiation of a bracket.

Both the PLU and SLU can begin a bracket but only the PLU can end a bracket. The S41 accepts the BB bit only if it is received in an FM data request and accepts the EB bit only if it is received in an FM data request or a DFC request carrying the Cancel or Chase command. The S41 sends the BB bit only in FM data requests. The BB and EB bits are always sent with the first request in a chain.

The S41 has three major states associated with bracket protocol. These are the between bracket (BETB) state, the in bracket (INB) state and the pending begin bracket (PEND.BB) state.

Between Bracket (BETB) State

The situation that exists when the PLU and SLU are in contention to begin a bracket is called the BETB state. This state is entered after the SDT command is accepted. The S41 is the first speaker, which means that it can begin a bracket at any time while it is in the BETB state. The PLU is the bidder, which means that it must ask permission to begin a bracket.

The BETB state ends when the Bid command or an RU containing the BB bit in RH is accepted. However, the S41 remains in the BETB state if it sends a negative response (one example is when the operator has depressed a text key – in which case the "receiver in transmit mode" sense code is returned). The S41 also remains in the BETB state while processing a chain containing both BB and EB bits. If the PLU cancels the chain containing the BB bit, the S41 returns to the BETB state.

The S41 terminates the BETB state and enters the INB state when the first (or only) element of a chain containing the BB bit is ready to be transmitted, i.e. after the operator has performed a send initiating action (depressed the ENTER key for example).

Pending Begin Bracket (PEND.BB) State

When the S41 is in the PEND.BB state, it is waiting for a bracket to be begun by the PLU. This situation arises when the S41 has returned a positive response to a Bid command received from the PLU and also when the S41 has sent an RTR command and received a positive response. In this state, the S41 will accept an RU containing the BB bit from the PLU.

In Bracket (INB) State

The S41 enters the INB state when it receives a chain containing the BB bit but not the EB bit, or when it begins a bracket. After the S41 has entered the INB and send states, any normal outbound requests (Bid command or RU containing the BB bit) received during that session will be discarded, and a negative response containing the "bracket bid reject" sense code will be sent. The S41 bracket state is not changed when it sends a negative response indicating a bracketing error.

The INB state is maintained by the S41 until the bracket is terminated according to bracket termination rule 1 which means that

1. If EB is received and the last element of a chain requires a definite response, the S41 enters the BETB state from the INB state after it has sent a positive response to the chain. However, the S41 remains in the INB state after it has sent a negative response.

2. If EB is received and the last element of a chain requires an exception response, the S41 immediately enters the BETB from the INB state.

Chaining

The following types of RUs can be encountered in a chain

First in chain (FIC) identifies an RU that commences a chained transmission.

Middle in chain (MIC) is transmitted for all RUs except the first in the chain and the last in the chain.

Last in chain (LIC) identifies the RU at the end of a chained transmission.

Only in chain (OIC) identifies a single-element chain.

The four different types of chains described above are identified by means of bits BC and EC.

A chain is in the correct order if the RUs are specified in any of the following sequences

1. FIC, LIC
2. FIC, MIC, . . ., LIC
3. OIC

A chaining error will occur if any other sequences are encountered.

If the S41 receives a chain in which the RUs are in an incorrect sequence such as FIC, MIC, FIC, a negative response that includes sense data indicating a chaining error is returned to the host program. In the above example, the S41 discards the chain and ignores any subsequent elements of the chain until either a Cancel command is received or a data RU specifying LIC is received. If an OIC data RU is received, the discarding of the chain is terminated, and the OIC message is also discarded. If RUs corresponding to FIC, MIC, OIC are sent in that sequence, a chaining error occurs. Here, the chaining error is detected when the S41 receives OIC. As a result, OIC is discarded and a negative response is sent. The S41 is now ready to process the next chain.

Half Duplex Flip-Flop Protocol

This protocol is used for the transfer of normal flow data. Here, only one of the two LUs in the session can send at any given time. Moreover, when one LU is sending, the other must be receiving.

The change direction (CD) bit in the request header is used to keep the two LUs in synchronization. Whenever an LU accepts this CD bit in a request, it means that it is its turn to send. Whenever an LU sends the CD bit in a request, it means that it is its turn to receive. The S41 always sends the CD bit in the last element of a chain.

Pacing and RU Length

The S41 supports outbound but not inbound pacing. Pacing is used only for normal flow.

Pacing Count

The pacing count value (passed as one of the bind parameters) specifies the number of RUs that the S41 may receive before a pacing response is required (so that another block of RUs can be sent). An isolated pacing response (DR and EX bits in RH set to zero) is sent if a response to the outbound request is not required at the time when the pacing response is required.

The S41 will send the pacing response as soon as printer buffer space is available. The S41 will send a negative response if there is insufficient space in the printer buffer. This negative response is sent in response to the RU which caused the overrun.

RU Length Considerations

The maximum RU length is specified in the bind parameters. The S41 will restrict the length of RUs used for inbound transfers as follows

- Bind parameters specifying less than 64 bytes are rejected.

- If 64–256 (inclusive) bytes are specified, the S41 uses the specified value as the maximum RU length.

- If 256–1024 (inclusive) bytes are specified, the S41 will use 256, 512, 768 or 1024 bytes as the maximum RU length as follows: 256–511 incl. becomes 256, 512–767 incl. becomes 512, 768–1023 incl. becomes 768 and 1024 becomes 1024.

- The S41 never sends RUs longer than 1024 bytes.

The length of outbound RUs must be related to the pacing count to make certain that buffer overruns do not occur. However, since editing is usually carried out very quickly, there are no restrictions on LUs of type 2.

Segmenting of RUs

RUs sent to network terminals are often larger than acceptable for optimum transfer of data by the link connecting the terminal to the network. Therefore, a Basic Information Unit (BIU) consisting of RH and RU can be divided into smaller elements, called segments, that are transmitted over the link, see Fig. 4. The S41 supports inbound and outbound segmenting for the LU-LU session.

The segment elements are identified by the transmission header (TH), see Fig. 6. The segment elements must be sent in a correct sequence, i.e. any of the following

- First segment, last segment

- First segment, middle segment, last segment

- Only segment

Segmenting Outbound

The maximum size of segment elements sent to the S41 must not exceed 256 bytes of data plus 6 Transmission Header (TH) bytes and 3 Request/Response Header (RH) bytes for the first (or only) segments. The maximum size of middle or last segments must not exceed 259 bytes of data plus 6 bytes of TH. If more than 259 RU bytes are received, the S41 will neither increment the Nr count nor accept the frame.

Reception of an incorrect segment sequence causes the S41 to enter the normal disconnected mode, which means that it deactivates the PU and all LUs. Moreover, the S41 cannot send any response. The Nr count and normal disconnected mode are explained in the section on Data Link Control Layer (SDLC Operations).

Segmenting Inbound

The S41 supports inbound segmentation if the bind parameters specify an RU length greater than 256. All transmissions containing more than 256 data bytes will then be segmented. Each RU segment (except the first and the last) will contain 259 data bytes. The first RU segment will contain 256 data bytes.

Session Processing States

The processing of SNA commands and responses as well as the data transfers are controlled by a set of session states. These session states are maintained in both Communication Processor 4101 and Display Unit 4110 (only the data traffic state is maintained for printer units). The session states are described in the following sections. The diagrams in Figs. 9 and 10 show the various states and the transitions between them.

Data Traffic States

Data Traffic States for LU-LU Sessions

The data traffic reset state is entered when a Bind or Clear command is received from the PLU. The data traffic active state is entered when an SDT command is received. When the S41 is in the data traffic reset state, it cannot send any data or commands to the PLU; it also rejects any FMD RUs or DFC commands.

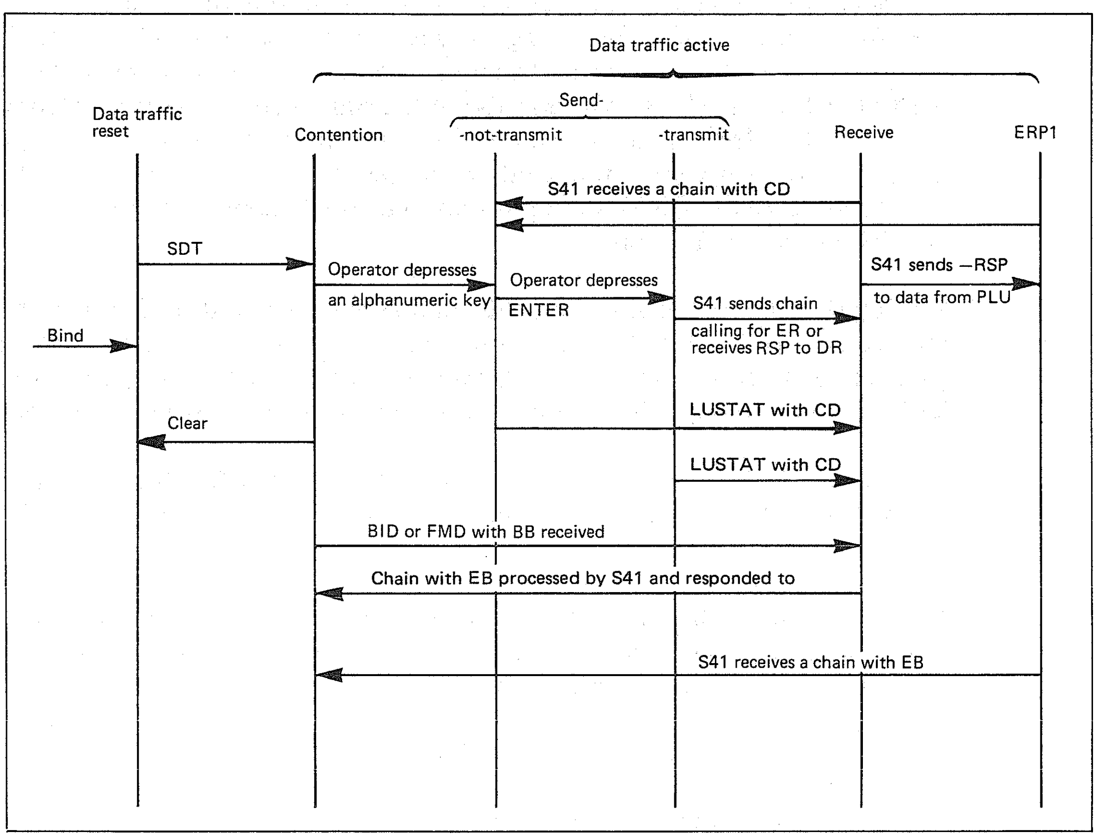


Fig. 9. Session states for LU-LU session

Data Traffic States for SSCP-LU Sessions

The data traffic active state is entered upon the acceptance of the ACTLU command. The data traffic reset state exists before the receipt of the ACTLU command or after the DACTLU command is received.

Contention State

The contention state for the LU-LU session exists only between brackets. The SLU can accept data from either the operator or the host. The first arrival triggers a change to the send or receive state. When data or a Bid command is received from the host, Display Unit 4110 is consulted, and if it is in the send state the PLU request will be rejected.

For the SSCP-SLU session, the contention state exists after the successful completion of the transmission of a chain (only single-element chains are used in an SSCP-SLU session).

Send State

In the send state, the SLU half-session resources are allocated for inbound operation. The send state is subdivided in two states called send-not-transmit and send-transmit. The send-not-transmit state is maintained only by Display Unit 4110 and not by Communication Processor 4101. It exists while the operator is entering data from the keyboard, the magnetic identification device or the selector pen.

The send-not-transmit state is entered from contention in response to the first keystroke capable of changing data on the screen. This state is changed to send-transmit by a send initiating action (usually depression of the ENTER key). The transition to send-transmit also causes transition to the in-bracket (INB) state and causes the keyboard to be locked. When the in-bracket state prevails, the send-not-transmit state is entered from the receive state or the ERP1 state after an outbound chain carrying CD (but not EB) is successfully processed.

The keyboard is unlocked when the send state is entered from either the receive state or the ERP1 state if a previous write control character (WCC) specified keyboard restore. The terminal operator can also depress the RESET key to unlock the keyboard after the send state has been entered.

Any normal outbound requests received in the send-not-transmit state will be discarded and a negative response (receiver in transmit mode) will be sent.

The data is sent from the display unit to the PLU while the send-transmit state prevails. All normal flow chains in the LU-LU session will carry the CD. Transition from the send-transmit state depends upon the response type called for by the request that sends data to the PLU. If a definite response is called for, a transition from send-transmit to receive takes place after the communication processor has received the response to the request. If an exception response is called for, a transition from send to receive takes place as soon as the communication processor has sent the complete chain.

Display Unit 4110 will change its state to receive as soon as the last part of the message is transferred from the display unit to a buffer in the communication processor.

The SSCP-SLU session operates only in the definite response mode. After reception of a positive response, a transition from the send-transmit to the contention state takes place. After a negative response is received a transition to the receive state takes place.

Receive (RCV) State

When the S41 is in the RCV state it cannot send inbound normal flow requests. Only inbound responses and control commands (the latter using expedited flow) can be sent. The keyboard is locked while the display is in the receive state (except the SYREQ key and, in LU-LU sessions, the ATTN key).

In an LU-LU session, the S41 enters the receive state from the contention state when an outbound normal flow message is accepted for processing. It also enters the receive state from the send-transmit state after a message is sent. The transition occurs a) when the S41 receives the response if the inbound request specifies CD and definite response, or b) after the S41 has successfully transferred the chain to the data link if the request specifies CD and exception response.

In an LU-LU session, the receive state is changed to send-transmit after the S41 has successfully processed the last request in a chain if this last request specifies CD. The receive state is changed to the contention state after the S41 has successfully processed and responded to a chain specifying EB, or after the S41 has received a chain specifying EB and exception response. The S41 changes from the receive state to the ERP1 state if a negative response is returned to the outbound request.

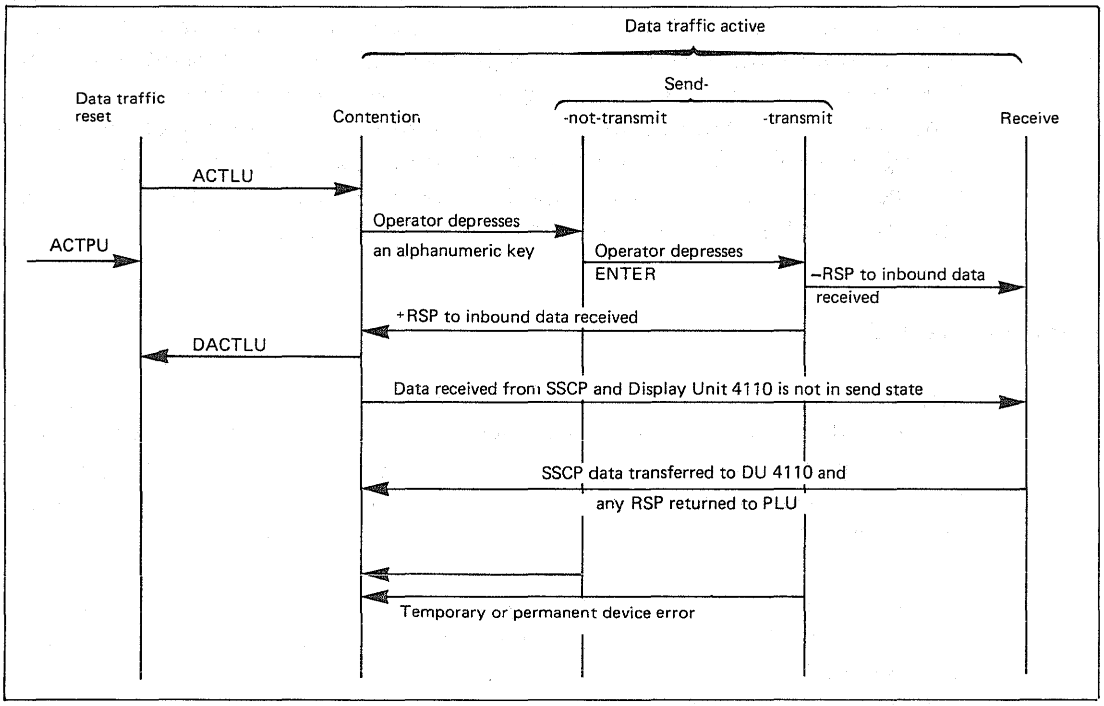


Fig. 10. Session states for SSCP-LU session

In an SSCP-SLU session, the S41 enters the receive state from the contention state when it accepts an outbound normal flow message and returns to the contention state after returning the response to the outbound request. The S41 enters the receive state from the send-transmit state if it receives a negative response to an inbound request.

ERP1 State

ERP1 is a special state used in connection with error recovery. Since the PLU is always responsible for error recovery, the SLU generally awaits an outbound request to correct the error condition. However, there are times when the SLU must first recover and notify the PLU of its recovery by means of a LUSTAT command before the PLU can take action.

The SLU ERP1 state allows a form of contention within brackets. While in this state, the S41 is able to receive any request, but it can only send LUSTATs. The S41 remains in the ERP1 state while LUSTATs are being sent to the PLU. As a result, successive LUSTATs can be sent without requiring the general exchange of CDs between each LUSTAT.

After the S41 has sent a negative response (except for negative responses containing bracket bid reject or receiver in transmit mode), it enters the ERP1 state. If the negative response does not cause a changeover to the between brackets (BETB) state, the transition to ERP1 takes place at end-of-chain.

The ERP1 state is maintained only in the communication processor.

Session Interaction

The three types of sessions (SSCP-PU, SSCP-SLU and PLU-SLU) can exist simultaneously. Moreover, all SLUs can be in session at the same time. A situation can thus arise in which a single display is needed by both the SSCP-SLU and LU-LU sessions. To avoid contention, only one of these sessions is defined as the device owner. The S41 rejects function management data from a non-owner session. Ownership states and transfers between them are maintained entirely in Display Units 4110. The operator is informed about ownership on the message line.

Ownership Transfer

The diagram in Fig. 11 illustrates display unit ownerships and transfers between them. When power is turned on to the S41 and Communication Processor 4101 has received any of commands ACTPU, DACTPU or DACTLU, Display Unit 4110 is in a no-session state. This is indicated by the word ONLINE which appears on the message line.

The operator can use the SYREQ key to transfer ownership from one session to another. Depressing the SYREQ key also clears the screen and unlocks the keyboard when the ownership is transferred to the SSCP-SLU session. The operator can communicate with the SSCP while the DU is in the SSCP-SLU session.

The screen size can be changed when ownership is transferred. The maximum screen size is always selected if SSCP-SLU ownership is established from the unowned state.

Notes on the SSCP-SLU Session

During an SSCP-SLU session, the immediate control and immediate response modes are used together with definite responses. Moreover, the half duplex contention protocol is used, and a normal flow request must be processed and responded to before an opposite-direction normal flow request can be accepted. Upon receipt of a positive response or transmission of a response, the SLU enters the contention state.

An outbound request causes the S41 to enter the receive state. The send-not-transmit state is entered when a text key is depressed, and the send-transmit state is entered when the ENTER key is depressed. The keyboard is locked while the S41 is in the receive or the send-transmit state.

If a negative response is received to an inbound normal flow request, the S41 enters the receive state and awaits an outbound request from the SSCP for recovery. The S41 enters the contention state without informing the SSCP if an error is detected while the S41 is in the send or contention state. The S41 also enters the contention state after having sent a negative response to an outbound normal flow request.

Outbound Messages

The SSCP can send a maximum of 256 bytes in a message string. The only codes that are used are NL (New Line) and graphics (note that IFS and IRS are treated as graphics and displayed as * and ; respectively). Unpredictable symbols may appear on the screen if codes not defined in the code tables are used.

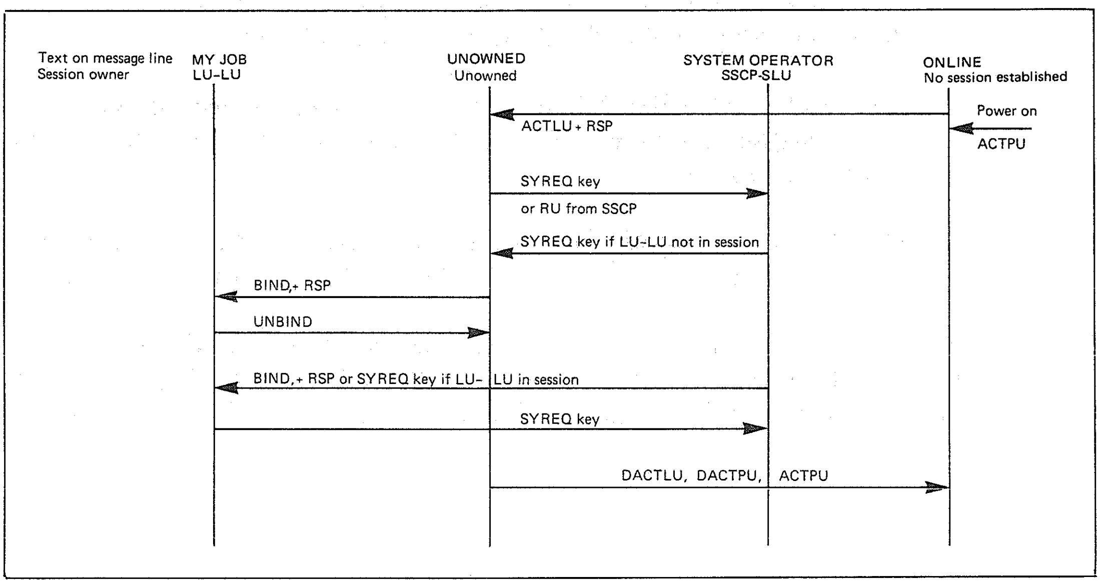


Fig. 11. Session ownership of a Display Unit 4110

Inbound Messages

The operator can enter a message on the screen if the words SYSTEM OPERATOR appear on the message line. Only the ENTER key can be used to initiate sending. The CLEAR key clears the screen, moves the cursor to the first position and switches to maximum screen size, but does not initiate sending.

The S41 will send a chain containing only one RU (maximum of 256 bytes) after the ENTER key is depressed. The data from the "start" position (inclusive) up to the "cursor" position (exclusive) or to end of screen, whichever occurs first, is sent.

The "start" position is defined as follows

- The position of the cursor when the keyboard is released (after a message from the SSCP has been received)
- The first position on the screen (after the CLEAR key or SYREQ key has been depressed)

The "cursor" position is the position of the cursor when the ENTER key is depressed. The operator must remember the "start" position if the cursor movement keys are used to be sure that entered data is actually positioned between the "start" and "cursor" positions.

Logon Procedure for Host Application Program (HAP Logon)

A generalised HAP logon procedure is described below.

1. Check the text on the message line. If MY JOB is displayed, the SLU is already in session with a PLU, i.e. the display unit is connected to an application program. Consequently, no HAP logon is required.
2. If UNOWNED is displayed, the SYREQ key shall be depressed whereupon SYSTEM OPERATOR will be displayed.
3. When SYSTEM OPERATOR is displayed, a HAP logon message defined at installation time shall be entered, after which the ENTER key shall be depressed.
4. MY JOB will then be displayed on the message line, thus indicating that HAP logon was successful and that the LU-LU session is established.

The SSCP can send a prompting message or error message to the S41, which will send a positive response if the text can be displayed or a negative response if the Display Unit 4110 owner is the LU-LU-session.

Logoff Procedure for Host Application Program (HAP Logoff)

A generalised procedure for HAP logoff is described below.

1. Check the text on the message line. If MY JOB is displayed the SYREQ key shall be depressed.

2. When SYSTEM OPERATOR is displayed, a HAP logoff message which was defined at installation time shall be entered, and the ENTER key shall be depressed.

UNOWNED will be displayed if the SYREQ key is depressed after logoff. The SSCP can send a message to the S41 which, in turn, will send a positive response if the text can be displayed or a negative response if the SSCP-LU session no longer owns the display unit.

In some applications it is possible to enter a HAP logoff message in an LU-LU session, i.e. when MY JOB is displayed on the message line.

Data Link Control Layer (SDLC Operations)

Interface Sublayer

The interface used in the S41 complies with the requirements for signal levels, impedances and logic specified for the CCITT V24/28 interface.

The S41 is connected to modem equipment. Half duplex or full duplex modems and 2-wire or 4-wire lines can be used. The transmission is bit and character synchronous and the transfer rates can range up to 9600 bps. NRZ or NRZI transmission coding can be used.

Communication Control

In SDLC communications, two roles – one primary and one secondary – are provided for participating stations.

The communications controller takes the role of the primary station, with responsibility for data link operations. The terminal nodes and the cluster controller nodes take the roles of secondary stations, responsive to commands from the primary station. The primary station takes part in every communication, either as a transmitter or receiver. It polls the other nodes for input messages and selects a particular device for an output operation. Each communication processor connected to a communications controller has a unique address which is used in connection with polling or selection.

Messages to the host are composed on the display screen by an operator using the keyboard. When a message is ready, the operator initiates the transmission. The keyboard is now locked to prevent unintentional changes in the message. The terminal control logic establishes a ready-to-send condition. Then, when the communication processor is polled by the communications controller, it sends a message formatted according to the

SNA/SDLC protocol. This means that the message contains specific control information in a definite pattern together with the actual data message. Messages to the host can also be generated internally within the secondary logical unit half-session (SLU).

Polling is accomplished by using bit 4 (poll bit) in the control field (see section on Control Field)

Frames and Sequences

The SDLC format is based on transmission blocks called frames. Each frame contains a series of 8-bit binary coded bytes which contain data and control information. Each frame is checked for transmission errors.

The frames are transmitted in sequence. Each frame carrying information is numbered at the transmitter. The frame number, designated Ns, is usually transmitted along with the rest of the frame. At the receiver, a counter is incremented for each correctly received frame. The receiver count is designated Nr, and it represents the next expected frame.

The receiver compares each incoming Ns with the expected Nr. If they do not match, incrementation of Nr stops and the frame is not accepted. At frequent intervals, after a maximum of seven frames, the receiver reports its Nr count to the transmitter. If Nr does not match the next Ns that the transmitter expects to send, some of the frames already sent, but obviously not correctly received, must be repeated. Repetition should start at the reported Nr frame.

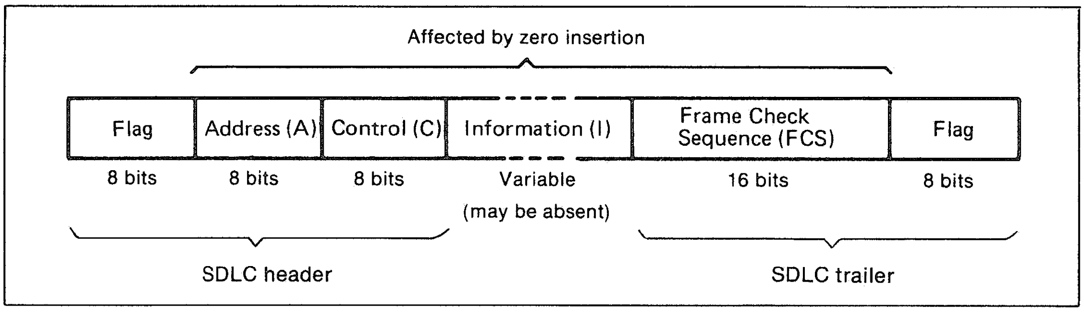
Nr and Ns both wrap around at 7, which means that the sequence becomes 0,1, - - - 6, 7, 0, 1 - - - etc.

Frames containing only control commands and responses etc. do not affect the Ns/Nr count. Such frames are referred to as unnumbered (nonsequenced) frames.

Frame Format

Each frame consists of

- One flag byte
- One communication processor address (A) byte
- One control (C) field byte
- Up to 9 + 256 information (I) field bytes
- Two frame check sequence (FCS) bytes
- One flag byte



Flag

The first and last byte in each frame is called flag and has a fixed binary configuration, namely 0111 1110. The first flag serves as a reference for the position of the A- and C-fields and initiates transmission error checking. The last flag terminates the check for transmission errors. Any flag may be followed by a frame, by another flag or by an idle condition. The last flag can also be the first flag in the next frame.

The purpose of the flags is to provide byte synchronization. Identification of a flag is possible since SDLC procedures do not permit more than five continuous binary 1s within a frame. Therefore, six contiguous 1s are always interpreted as a flag.

Within a frame, zero insertion is used to "get around" the restriction imposed by the aforesaid rule calling for a maximum of five contiguous 1s within a frame. Within the frame, the transmitter inserts an extra binary 0 after any succession of five contiguous 1s. After testing for flag recognition, the receiver removes any 0 that follows a received succession of five contiguous 1s, thus restoring the information content.

Address Field (A)

The address field always contains the address (one byte) of the communication processor receiving or transmitting the frame. Any bit configuration contained in the address byte can be used as an address except 00 and FF.

The address of the specific device (display unit, printer) is given in the transmission header included in the information (I) field.

Control Field

The control field is used to encode the commands and responses used for data link control.

A control field bit configuration sent from the communications controller to Communication Processor 4101 is defined as a command. A control field bit configuration is defined as a response when sent from Communication Processor 4101 to the communications controller. A response can be expected to a given command, but Communication Processor 4101 must receive a command byte in which the poll bit is set to one before the response can be transmitted.

Bit 4 in the control field is the poll/final (P/F) bit. A poll bit is sent to Communication Processor 4101 to request transmission. Communication Processor 4101 is not allowed to send any frame until it receives the poll bit. A final bit is sent by Communication Processor 4101 in the last (or only) frame it sends. When it has sent the final bit it must receive a new poll bit before it can send a new frame.

Three formats are used, depending on the type of command/response. See Fig. 12.

- Supervisory format
- Unnumbered format
- Information transfer format

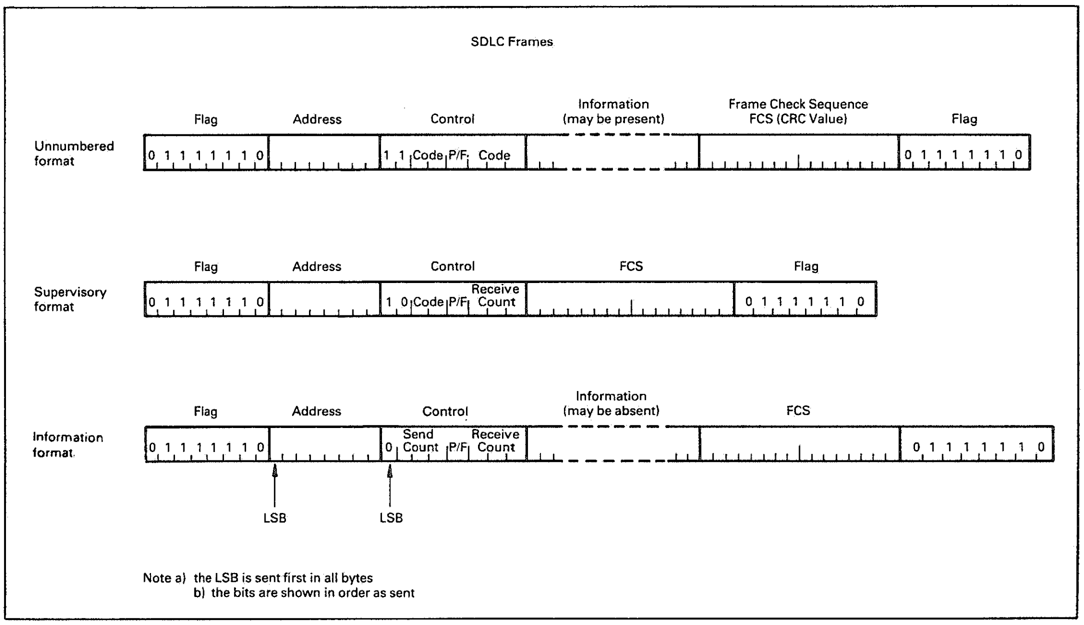


Fig. 12. SDLC frame formats.

Supervisory Format

The purpose of the supervisory format commands/responses is to initiate and control information transfer. No information field may be included in a supervisory format frame. Two supervisory commands/responses are used: Receive Ready (RR) and Receive Not Ready (RNR).

When RR and RNR are sent as responses from the communication processor to the communications controller, the final bit is always set to 1.

- Receive Ready (RR)

Pos* (MSB)	7	6	5	4	3	2	1	0	
	– Nr –			P/F	0	0	0	1	Command or response

Receive Ready confirms sequenced frames up to and including Nr–1 and indicates readiness to receive the frame Nr.

- Receive Not Ready (RNR)

Pos* (MSB)	7	6	5	4	3	2	1	0	
	– Nr –			P/F	0	1	0	1	Command or response

*Note: IBM designates the most significant bit (Bit No. 7) as bit No. 0.

Receive Not Ready indicates a temporarily busy condition. No frames that require buffer space can be accepted. RNR also confirms sequenced frames up to and including Nr–1 and indicates that the frame Nr is expected next, when the busy condition ceases. A Receive Ready response or an information frame, but no unnumbered response, can be used to indicate that the busy condition has ceased.

Unnumbered Format

The unnumbered format is used to transfer commands/responses for data link management. This includes activating communication processors, controlling their response modes, and reporting procedural errors other than those recoverable by retransmission.

A communication processor can be put into two different modes using commands sent in the nonsequenced format. These two modes are the Normal Response Mode (NRM) and the Normal Disconnected Mode (NDM).

A communication processor in NRM does not initiate any unsolicited transmissions. It transmits only in response to a frame in which the poll bit is set to 1.

When it is in NDM, the communication processor must receive a mode setting command in order to change over to NRM. Other commands with the poll bit set to 1 cause a communication processor in NDM to respond with Disconnected Mode. As long as the communication processor is in NDM, it cannot receive or transmit information frames or supervisory frames. The communication processor enters NDM after having received a Disconnect command or as a result of a temporary power off condition.

Two unnumbered commands are used, Set Normal Response Mode (SNRM) and Disconnect (DISC):

- Set Normal Response Mode (SNRM)

Pos* (MSB)	7	6	5	4	3	2	1	0	
	1	0	0	P	0	0	1	1	Command

This command causes the communication processor to enter the normal response mode. The expected response is UA (see below). The Nr and Ns counts are reset to 0. The communication processor remains in the normal response mode until it receives a DISC command.

- Disconnect (DISC)

Pos* (MSB)	7	6	5	4	3	2	1	0	
	0	1	0	P	0	0	1	1	Command

*Note: IBM designates the most significant bit (Bit No. 7) as bit No. 0.

This command causes the communication processor to enter the Normal Disconnected Mode. The expected response is UA (see below). The communication processor remains disconnected until it receives an SNRM command.

Three unnumbered responses are used: Unnumbered Acknowledgement (UA), Disconnected Mode (DM), and Frame Reject (FRMR).

- Unnumbered Acknowledgement (UA)

Pos* (MSB)	7	6	5	4	3	2	1	0	
	0	1	1	F	0	0	1	1	Response

This is the affirmative response to an SNRM or DISC command. Further transmissions are at the option of the communications controller.

- Disconnected Mode (DM)

Pos* (MSB)	7	6	5	4	3	2	1	0	
	0	0	0	F	1	1	1	1	Response

This response is transmitted by a communication processor to indicate that it is disconnected. DM is sent in response to any command (including non-valid commands) in which the poll bit is set, except TEST. It is also sent in response to the SNRM command if the communication processor cannot enter the normal response mode.

- Frame Reject (FRMR)

Pos* (MSB)	7	6	5	4	3	2	1	0	
	1	0	0	F	0	1	1	1	Response

This response is transmitted by a communication processor in the normal response mode when it receives a non-valid command provided that the poll bit is set. Otherwise, the FRMR response is sent to the next command with the poll bit set.

After a communication processor has received a non-valid command, it repeats FRMR whenever it responds, except to mode-setting commands SNRM and DISC.

Status information is sent in the I-field as part of the FRMR response frame from Communication Processor 4101. See Fig. 13. This I-field provides the communications controller with the status data that it needs to select appropriate recovery action.

*Note: IBM designates the most significant bit (Bit No. 7) as bit No. 0.

Pos*	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
									0				0				I	P	N	R	0	0	0	0

- Byte 1: The rejected command, as received from the communications controller.
- Ns / Nr: The current Ns and Nr counts in the communication processor.
- 1)
- Padding

1)
I = Invalid or non-implemented command
P = Prohibited I-field
N = Not used
R = Received Nr is incongruent

Fig. 13. Information field in the FRMR response frame

A special test command/response called Link Test (TEST) is also used.

- Link Test (TEST)

Pos* (MSB)	7	6	5	4	3	2	1	0	
	1	1	1	P/F	0	0	1	1	Command or response

This is a basic test of the data link between the communications controller and Communication Processor 4101. When the Link Test command is transmitted from the communications controller, Communication Processor 4101 checks that the FCS field is valid and that the control field poll bit is set to 1. If the command is received correctly, Communication Processor 4101 sends the Link Test response to the communications controller, regardless of whether it is in normal response mode or normal disconnected mode.

Data in an I-field (see below) can accompany a Link Test command, and a Link Test response.

Information Transfer Format

An information field is required when text is transmitted in either direction between the communications controller and Communication Processor 4101. The control field, which indicates that an information field is being sent, is formatted as follows

Pos* (MSB)	7	6	5	4	3	2	1	0	
	– Nr –			P/F	– Ns –			0	Command or response

*Note: IBM designates the most significant bit (Bit No. 7) as bit No. 0.

Information Field (I)

In an information transfer frame, the information field is transmitted as a series of 8-bit bytes. Up to 256 bytes of message data, preceded by a transmission header (6 bytes), and possibly a request/response header (3 bytes), can be sent in one frame.

Frame Check Sequence (FCS) Field

The frame check sequence field contains 16 binary digits. These 16 digits result from a mathematical computation involving the binary value of all bits within the frame. The purpose of the frame check is to validate transmission accuracy.

The transmitter performs the calculation and sends the resulting FCS value. The receiver performs a similar computation and checks its results. A frame that is found to be in error is not accepted and the receiver does not increment its Nr count.

The frame check is a form of cyclic redundancy checking (CRC).

In the SDLC application of CRC, the binary value of the transmission to be checked is divided by the generating polynomial $X^{16} + X^{12} + X^5 + 1$. Integer quotient digits are ignored and the complement of the resulting remainder value is transmitted as the FCS field.

Note that an ending flag byte follows the FCS field.

Link Error Situations

No Response

Communication Processor 4101 discards frames which are incorrect in any of the following ways

- Frame containing an FCS error
- Frame containing less than 32 bits
- Frame in which the number of bits is not a multiple of 8
- Frame containing more than 259 RU bytes

In these situations the CP 4101 does not send anything.

Sequence Error

An information frame with an incorrect Ns count is discarded by the CP 4101 as well as any information frames which follow. The sequence error condition is cleared when an information frame with a correct Ns is received.

Note: 1) If an acknowledgement is pending, the Nr count in an information frame that contains an incorrect Ns count is accepted.

2) If there is an information frame waiting to be sent from the CP 4101, this frame is sent as a response to the poll bit even if the Ns count in the received frame is incorrect.

Frame Reject Response

When an invalid command is received, the CP 4101 sends the Frame Reject response (FRMR). See the description of frame reject in the section entitled Unnumbered Format.

Communication Processor 4101 Busy

The CP 4101 sends an RNR (Receive Not Ready) response to indicate a busy condition. Received frames will then be discarded (answered by sending RNR) until the busy condition ceases.

Disconnect Mode Response

When a command with the poll bit set to 1 is received by a CP 4101 that is in the normal disconnected mode, the CP 4101 answers with Disconnect Mode. See section entitled Unnumbered Format. Note that frame reject conditions do not exist in the normal disconnected mode.

Frame Received without Poll Bit

The CP 4101 never responds to a frame unless the poll bit is set to 1. Instead, the following will occur.

- If the command is correct it will be executed, but no response will be generated (i.e. the command will not be responded to in any subsequent frame unless the first frame was an information frame).
- If the command is invalid it will cause a frame reject condition.
- If the CP 4101 is in the normal disconnected mode, the frame will be ignored.

Abort Conditions

Premature termination of a data link communication is called an abort. This function is reserved for the transmitting station, which aborts by sending eight consecutive 1s. The abort pattern interrupts the frame without an FCS field or an ending flag. In addition, the continuity of the data link is terminated.

Abort can be initiated from either the communications controller or from Communication Processor 4101. This occurs, for example, if there is underflow in a transmitter buffer. If followed by at least seven additional 1s, the abort pattern becomes an idle pattern which idles the data link.

The abort can also be followed by one or more flags to keep the data link in the active state.

Contents

Illustrations

Local Operation

Communication Processor 4102 is designed for local connection to an IBM host computer system. It can be connected to a selector channel, a byte multiplexer channel or a block multiplexer channel in IBM System/360 or System/370 or to an integrated channel used with a 3031, 3032 or 3033 processor. An Alfaskop System 41 connected locally via Communication Processor 4102 emulates the IBM 3274 Control Unit models 1B and, in a later version, 1D. (When model 1D is emulated, the communication processor cannot be connected to a System/360.) The IBM 3272 Control Unit is not emulated, but most of its operations are the same as those of the IBM 3274 1B.

General

The channel program controls the various operations in the terminal system by transmitting information across the I/O interface. This information consists of

- An address byte that selects one communication processor and one of the connected terminals
- Command bytes that specify which operation is to be carried out
- Data bytes which consist of the text that is to be displayed or printed out on the selected terminal or orders calling for the formatting of the buffer in the selected terminal
- Various control signals

In order to inform the channel program of general conditions in the terminal system, the terminal system generates status bytes on the following occasions

- In response to a command
- When an operation that has been called for has been executed
- When a terminal wishes to communicate with the computer

Communication Sequences

Channel Program

When the computer program wishes to execute a terminal operation, it must prepare a channel program and initiate it by issuing a Start I/O (SIO) instruction.

The Start I/O instruction (SIO) includes the address of the terminal that is to carry out the operation. SIO causes this address to be sent to all units connected to the channel. If a communication processor recognizes the address of one of its connected terminals, it responds by returning the address to the channel.

SIO also causes the channel to fetch the first Channel Command Word (CCW). The channel program comprises one or more Channel Command Words that are stored in main storage. A Channel Command Word contains a total of eight bytes. Its format is illustrated in Fig. 1. It contains the following information

- Command byte – specifies the operation which the terminal shall carry out
- Address bytes – specify the start address of the data area in main storage
- Flag byte – the first six bits in this byte are listed in sequence below
  - CD Chain Data
  - CC Chain Command
  - SLI Suppress Length Indication
  - SKIP Suppresses transfer to storage of data that was read in
  - PCI Program-controlled I/O interrupt
  - IDA Indirect Data Address. Causes bytes 1, 2 and 3 of the CCW to specify the location of the first Indirect Data Address Word (IDAW).
- Length byte – specifies the number of bytes that are to be transferred to/from the data area in main storage.

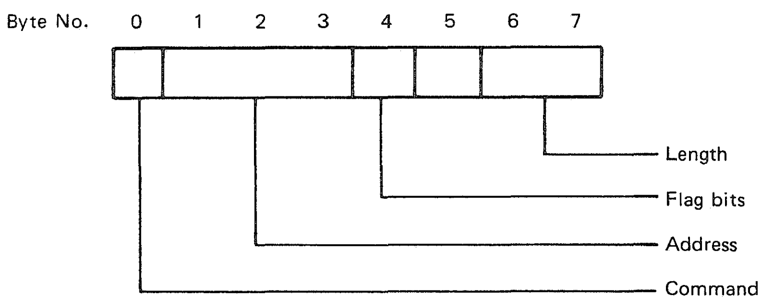


Fig. 1. Channel Command Word format

The command byte is sent by the channel to the selected communication processor, all other information in the CCW is used to control channel operations.

Address Format

When the computer wishes to select a particular communication processor and a particular terminal, it sends out a unique address byte via the channel by means of instruction SIO. This address byte is divided into a communication processor address (CU) and a terminal address (DV) and, depending upon the configuration of the terminal system, these addresses can have the formats shown below. See also Tables 1–2 showing the standard addressing formats for local applications.

- If a 4-bit CU address is used, 16 terminals can be addressed, see Table 1. The address of, for example, communication processor No. 2 and terminal No. 7 would then appear as follows

0 0 1 0	0 1 1 1
CU	DV

$= (27)_{16}$

- If a 3-bit CU address is used, 32 terminals can be addressed, see Table 2. The address of, for example, communication processor No. 2 and terminal No. 3 would then appear as follows

0 1 0	0 0 0 1 1
CU	DV

$= (43)_{16}$

Table 1. Standard addressing format for max. 16 terminals connected to one communication processor (CU)

Device No.	CU0	CU1	CU2	CU3	CU4	CU5	CU6	CU7	CU8	CU9	CU10	CU11	CU12	CU13	CU14	CU15
0	00	10	20	30	40	50	60	70	80	90	A0	B0	C0	D0	E0	F0
1	01	11	21	31	41	51	61	71	81	91	A1	B1	C1	D1	E1	F1
2	02	12	22	32	42	52	62	72	82	92	A2	B2	C2	D2	E2	F2
3	03	13	23	33	43	53	63	73	83	93	A3	B3	C3	D3	E3	F3
4	04	14	24	34	44	54	64	74	84	94	A4	B4	C4	D4	E4	F4
5	05	15	25	35	45	55	65	75	85	95	A5	B5	C5	D5	E5	F5
6	06	16	26	36	46	56	66	76	86	96	A6	B6	C6	D6	E6	F6
7	07	17	27	37	47	57	67	77	87	97	A7	B7	C7	D7	E7	F7
8	08	18	28	38	48	58	68	78	88	98	A8	B8	C8	D8	E8	F8
9	09	19	29	39	49	59	69	79	89	99	A9	B9	C9	D9	E9	F9
10	0A	1A	2A	3A	4A	5A	6A	7A	8A	9A	AA	BA	CA	DA	EA	FA
11	0B	1B	2B	3B	4B	5B	6B	7B	8B	9B	AB	BB	CB	DB	EB	FB
12	0C	1C	2C	3C	4C	5C	6C	7C	8C	9C	AC	BC	CC	DC	EC	FC
13	0D	1D	2D	3D	4D	5D	6D	7D	8D	9D	AD	BD	CD	DD	ED	FD
14	0E	1E	2E	3E	4E	5E	6E	7E	8E	9E	AE	BE	CE	DE	EE	FE
15	0F	1F	2F	3F	4F	5F	6F	7F	8F	9F	AF	BF	CF	DF	EF	FF

Table 2. Standard addressing format for max. 32 terminals conncected to one communication processor (CU)

Device No.	CU0	CU1	CU2	CU3	CU4	CU5	CU6	CU7
0	00	20	40	60	80	A0	C0	E0
1	01	21	41	61	81	A1	C1	E1
2	02	22	42	62	82	A2	C2	E2
3	03	23	43	63	83	A3	C3	E3
4	04	24	44	64	84	A4	C4	E4
5	05	25	45	65	85	A5	C5	E5
6	06	26	46	66	86	A6	C6	E6
7	07	27	47	67	87	A7	C7	E7
8	08	28	48	68	88	A8	C8	E8
9	09	29	49	69	89	A9	C9	E9
10	0A	2A	4a	6A	8A	AA	CA	EA
11	0B	2B	4B	6B	8B	AB	CB	EB
12	0C	2C	4C	6C	8C	AC	CC	EC
13	0D	2D	4D	6D	8D	AD	CD	ED
14	0E	2E	4E	6E	8E	AE	CE	EE
15	0F	2F	4F	6F	8F	AF	CF	EF
16	10	30	50	70	90	B0	D0	F0
17	11	31	51	71	91	B1	D1	F1
18	12	32	52	72	92	B2	D2	F2
19	13	33	53	73	93	B3	D3	F3
20	14	34	54	74	94	B4	D4	F4
21	15	35	55	75	95	B5	D5	F5
22	16	36	56	76	96	B6	D6	F6
23	17	37	57	77	97	B7	D7	F7
24	18	38	58	78	98	B8	D8	F8
25	19	39	59	79	99	B9	D9	F9
26	1A	3A	5A	7A	9A	BA	DA	FA
27	1B	3B	5B	7B	9B	BB	DB	FB
28	1C	3C	5C	7C	9C	BC	DC	FC
29	1D	3D	5D	7D	9D	BD	DD	FD
30	1E	3E	5E	7E	9E	BE	DE	FE
31	1F	3F	5F	7F	9F	BF	DF	FF

Command Initiation

In order to carry out different operations such as writing, reading or erasing the data in a selected terminal's buffer, the computer uses a number of different commands. For local operations, these commands consist of the following

- Read-type commands for the transfer of data from the terminal to the computer. There are two read-type commands
  - Read Buffer, code $(02)_{16}$
  - Read Modified, code $(06)_{16}$

- Write-type commands for transferring data and orders from the computer to the terminal system. There are three write-type commands
  - Write, code $(01)_{16}$
  - Erase/Write, code $(05)_{16}$
  - Erase/Write Alternate, code $(0D)_{16}$

- Control commands which carry out specific operations in the terminal system. The control commands are listed below
  - Select, code $(0B)_{16}$ (in a 3274 model 1B emulation)
  - Select RM, code $(0B)_{16}$ (in a 3274 model 1D emulation)
  - Select RB, code $(1B)_{16}$ (in a 3274 model 1D emulation)
  - Select RMP, code $(2B)_{16}$ (in a 3274 model 1D emulation)
  - Select RBP, code $(3B)_{16}$ (in a 3274 model 1D emulation)
  - Select WRT, code $(4B)_{16}$ (in a 3274 model 1D emulation)
  - Erase All Unprotected, code $(0F)_{16}$
  - No Operation, code $(03)_{16}$
  - Test I/O, code $(00)_{16}$ (The Test I/O command is hardware generated and cannot be initiated by the channel program.)

- Sense-type commands, which transfer data from the terminal system to the computer. There are two sense-type commands
  - Sense, code $(04)_{16}$, transfers one sense byte
  - Sense ID, code $(E4)_{16}$ transfers four bytes (FF, 32, 74 and 1B or 1D) which indicate the type of emulation.

When a terminal has been selected, i.e. the channel has received a response to its addressing operation, the command byte is sent to the communication processor. If the command byte calls for a data transfer operation, the communication processor responds by sending back a status byte (initial status) which indicates whether or not the command can be carried out. A positive response causes the operation to continue. When the operation is terminated, the communication processor sends a status byte indicating whether or not the operation was carried out normally (ending status).

When a command calls for an operation other than data transfer, it is called an immediate command. An immediate command can be terminated in two different ways

- No Operation is concluded by combining the initial status and ending status. Test I/O is concluded by providing initial status only.

- Other immediate operations provide an initial status that concludes the operation on the channel, after which the communication processor and the terminal carry out operations internally. When these internal operations are concluded an indication is provided by sending ending status.

Chaining of Commands

When the channel and the terminal have concluded the operation specified by a Channel Command Word (CCW), the channel can fetch and execute a new CCW automatically. This procedure is called chaining and involves the execution of a number of operations in sequence at the same terminal, and it can be initiated using a single SIO instruction. Certain limitations apply with regard to which commands can be chained. This is described in greater detail below and in the chapter headed Commands and Orders.

Local Commands

The commands are described in the chapter headed Commands and Orders, with the exception of the following select-type commands and the Sense ID command which are specific for local operation.

Select-type commands

The select-type commands are immediate commands used to transfer the content of the selected terminal's buffer to the communication processor, thus preparing the communication processor for a chained Read Modified, Read Buffer or Write command. If no select-type command is used, this buffer transmission operation is performed as the first stage of an unchained Read Modified, Read Buffer or Write command.

When the communication processor receives a select-type command, it sends Channel End status as initial status, freeing the channel for other operations. After the data is transferred, the communication processor sends Device End status asynchronously. Note that in a communication processor emulating a 3274-1B no transfer (only a logical connection) is made before sending Device End status. Also note that in a communication processor emulating a 3274-1D, no transfer is made for the Select RMP and Select RBP commands before sending Device End status. Please see the following detailed description for the latter two commands.

When the channel receives the Device End status, it can send the chained command (except for Select RMP and Select RBP commands). When this command has been executed, it can be followed by one or more additional chained commands of the same type without the select-type command having to be repeated. This is possible because the terminal buffer content remains in the communication processor (valid only for 3274-1D emulation).

There are six different select-type commands, but only five coded characters are used. Thus 0B is interpreted as Select in a communication processor emulating a 3274-1B but as Select RM in a communication processor emulating a 3274-1D. The former communication processor can handle only a Select command, the latter can handle the other five select-type commands.

Select Command

Select is used only in a communication processor emulating a 3274-1B. The general description set forth above applies here.

Select RM Command

Select RM is used when a Read Modified operation is to be performed, and can be executed only in a communication processor emulating a 3274-1D. When Select RM is executed the transfer from the terminal buffer embraces only the information that is included in a Read Modified operation. The content of the communication processor buffer will be arranged in the same way as the message layout for a Read Modified operation as set forth in the chapter headed Commands and Orders.

If any command other than Read Modified is chained to the Select RM command, the execution of this command is carried out as if the command had been received unchained.

Select RB Command

Select RB is used when a Read Buffer operation is to be performed, and can be executed only in a communication processor emulating a 3274-1D. The content of the communication processor buffer will, after the transfer from the selected device, be arranged in the same way as the message layout for a Read Buffer operation as set forth in the chapter headed Commands and Orders.

A Read Buffer command must be chained to the Select RB command. Any other chained command is regarded as illegal, and will thus not be accepted.

Select RMP Command

Select RMP is used when a Read Modified operation is to be performed from a certain position, and it can be executed only in a communication processor emulating a 3274-1D. The sequence of chained commands must be Select RMP, Write and Read Modified. Any other command sequence is regarded as illegal, and these commands will thus not be accepted.

In response to the Select RMP command, the communication processor only records the RMP condition and sends Device End status. The chained Write command should contain only four bytes (WCC, SBA, b1, b2). When the Write command is received, the selected device's buffer is transferred starting at the position specified in the Write command. The content of the communication processor buffer will be arranged in the same way as the message layout for a Read Modified operation as set forth in the chapter headed Commands and Orders.

Select RBP Command

Select RBP is used when a Read Buffer operation is to be performed from a certain position, and it can be executed only in a communication processor emulating a 3274-1D. The sequence of chained commands must be Select RBP, Write and Read Buffer. Any other command sequence is regarded as illegal, and these commands will thus not be accepted.

In response to the Select RBP command, the communication processor only records the RBP condition and sends Device End status. The chained Write command should contain only four bytes (WCC, SBA, b1, b2). When the Write command is received, the selected device's buffer is transferred starting at the position specified in the Write command. The content of the communication processor buffer will be arranged in the same way as the message layout for a Read Buffer operation as set forth in the chapter headed Commands and Orders.

Select WRT Command

Select WRT is accepted only by a communication processor emulating a 3274-1D. No operation is performed in the communication processor. Only Device End status is sent. A Write command must be chained to the Select WRT command. Any other chained command is regarded as illegal and will thus not be accepted. The Select WRT command is not required before a Write command.

Status and Sense Information

General

The communication processor sends out a status byte to inform the channel of the terminal system status. Status can be sent out synchronously, i.e. while a terminal is selected, or asynchronously.

Synchronous status can be an initial status, i.e. the communication processor's response to the command received from the channel, or an ending status which is sent when all channel functions have been concluded for a non-immediate operation. Asynchronous status can be

- Ending status for an immediate operation other than No Operation and Test I/O

- A second ending status for a Write, Erase/Write or Erase/Write Alternate command

- An operator action or equipment condition not associated with command execution (attention).

The communication processor resets the status byte as soon as it has been fetched by the channel. Status which has not yet been read by the channel is called waiting status.

Status Byte Bit Assignments

The significance of the bits in the status byte is illustrated in Fig. 2 and described in the paragraphs which follow.

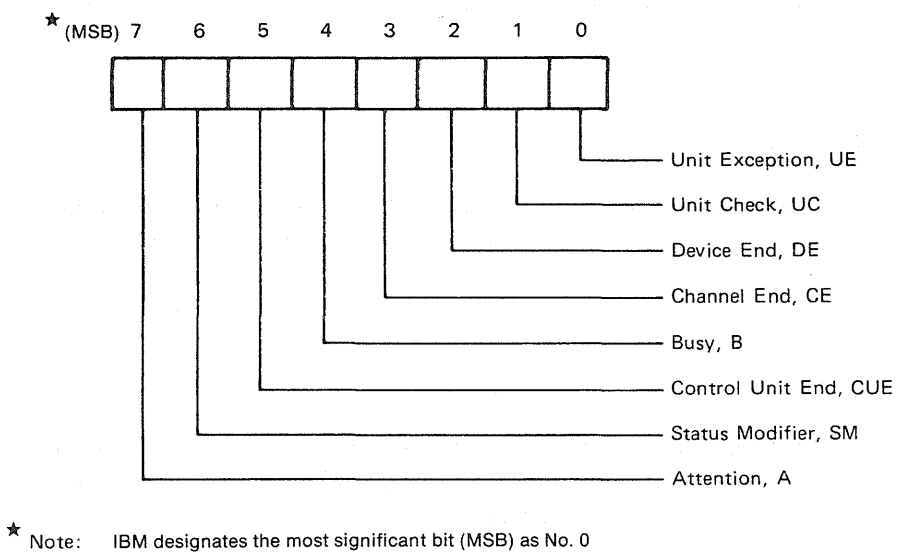


Fig. 2. Status byte bit assignments

Attention, A

The Attention bit indicates that a terminal connected to the communication processor has issued a request for service.

Status Modifier, SM

The Status Modifier bit is set together with the Busy bit in an initial status byte in order to indicate that the communication processor is busy (see Busy below).

Control Unit End, CUE

If the Busy and Status Modifier bits have been sent, the Control Unit End bit is set when the communication processor is no longer busy and thus can receive a new command.

Busy, B

The Busy bit is set alone in an initial status byte when the addressed terminal is busy carrying out a printout operation. The Busy bit is set together with the Status Modifier bit when the addressed communication processor is busy or if a terminal other than the addressed terminal has a waiting status. The Busy bit is also set together with waiting status if the addressed terminal has such status.

Channel End, CE

The Channel End bit indicates that the transfer of data between the channel and the communication processor is concluded.

Device End, DE

The Device End bit indicates that the communication processor and the terminal have concluded the command operation that was called for and can receive a new command.

Unit Check, UC

The Unit Check bit is set when an abnormal condition is detected in the channel program or in the equipment. It indicates that further information can be obtained using a Sense command.

Unit Exception, UE

The Unit Exception bit is set in synchronous or asynchronous ending status bytes when the communication processor has attempted to execute a command, but found the unit busy after the initial status byte has been sent.

Initial Status

Bit combinations in the initial status byte that can be encountered are illustrated in Table 3. If the initial status byte contains Unit Check or Busy the command has not been accepted by the communication processor.

Table 3. Initial status and sense conditions

Status	Sense	Display	Printer	Condition
All zeros		x	x	Normal status for all commands except No Operation, Select and Erase All Unprotected
CE		x	x	Normal status for a Select-type or Erase All Unprotected command
CE, DE		x	x	Normal status for a No Operation command
UC	BOC	x	x	Parity Error in a command byte
UC	IR	x	x	A command other than Sense was sent to a terminal which the communication processor considers not connected or not ready (out of paper or power off)
UC	CR	x	x	The communication processor has received an invalid command
B			x	The addressed terminal is busy with a previous command
B, SM [1])		x	x	The communication processor is busy

Note: [1]) If the communication processor is busy due to a waiting status byte for the addressed terminal, the waiting status byte is sent with the Busy bit set, and the command is not accepted. In response to a Test I/O command (which cannot be issued by the channel program) the status is sent without the Busy bit set.

Synchronous Ending Status

The bit combinations that can be encountered in an ending status byte are illustrated in Table 4. An ending status byte indicates that the channel-dependent part of the operation is concluded. Activity in the terminal can be concluded or can still be in progress. In the latter case, the Device End bit is set and a second asynchronous ending status byte is sent when the terminal has concluded the operation. When the ending status byte has been sent, an I/O interrupt occurs unless chaining is called for.

Table 4. Synchronous ending status and sense conditions

Status	Sense	Display	Printer	Condition
CE		x	x	Sent at the end of the data stream in connection with a Write, Erase/Write or Erase/Write Alternate command.
CE, DE		x	x	Sent at the end of the data stream in connection with a Read Buffer, Read Modified or Sense command or when the channel has filled the designated data area in connection with a Read Buffer or Read Modified command.
CE, DE, UC	BOC	x	x	Parity error in the data stream in connection with the Write, Erase/Write or Erase/Write Alternate command.
CE, DE, UC	CC	x	x	The addressed terminal has not responded within a specified time interval in connection with an Erase/Write or Erase/Write Alternate command or an unchained read Buffer, Read Modified or Write command.
CE, DE, UC	OC	x	x	Can be sent only from a communication processor emulating a 3274-1D. Indicates that an illegal command chaining has been carried out after a Select RB, Select RMP, Select RBP or Select WRT command
CE, DE, UE		x	x	The communication processor has attempted to execute a read-type or write-type operation, but after having sent the initial status byte, it found that the terminal was busy with a local printout.

Asynchronous Status

The bit combinations that can be encountered in an asynchronous status byte appear in Table 5. When a condition occurs which causes status to be generated asynchronously, the communication processor attempts to be connected (logically) to the channel, and when this has happened, the status byte is sent.

If asynchronous status is generated for a terminal while the communication processor is carrying out an operation in another terminal, the operation in progress is first concluded by sending ending status in the normal way, whereupon the communication processor changes over from busy to not busy. The communication processor then fetches the waiting status from the terminal and attempts to present it to the channel as described above.

Table 5. Asynchronous status and sense conditions

Status	Sense	Display	Printer	Condition
A		x		Operator has requested transmission by, for example, depressing a PA key.
DE		x	x	Transmission from communication processor to terminal is concluded in connection with a Write, Erase/Write or Erase/Write Alternate command which did not start a printer.
				The terminal has performed an Erase All Unprotected command. A printer becomes not busy after printout is completed.
				Transfer operation or preparation is concluded in connection with a select-type command.
				A terminal changes from unavailable to available or from not ready to ready. A terminal becomes not busy after having previously sent UE due to the fact that it was busy.
				The communication processor Enable/Display switch is set to Enable. Each terminal that is available sends DE.
A, DE, UC	IR		x	The addressed printer becomes not ready before a printout operation is concluded.
DE, UC	CC	x	x	The addressed terminal has not responded within a specified time interval in connection with a select-type, write-type or Erase All Unprotected command.
				The printer was not ready (paper out for example) when it received a command that involved starting a printer.
CUE		x	x	The communication processor had been addressed while busy, but it is now not busy and can accept a command.

Sense Information

When the communication processor detects an abnormal condition in the channel program or the equipment, the Unit Check bit in the status byte is set. In order to obtain more information, the computer issues a Sense command, whereupon the communication processor transfers a sense byte that further defines the error. The sense byte is reset by all commands except No Operation, Test I/O and Sense. The bit assignments for the sense byte are illustrated in Fig. 3 and described in the paragraphs which follow.

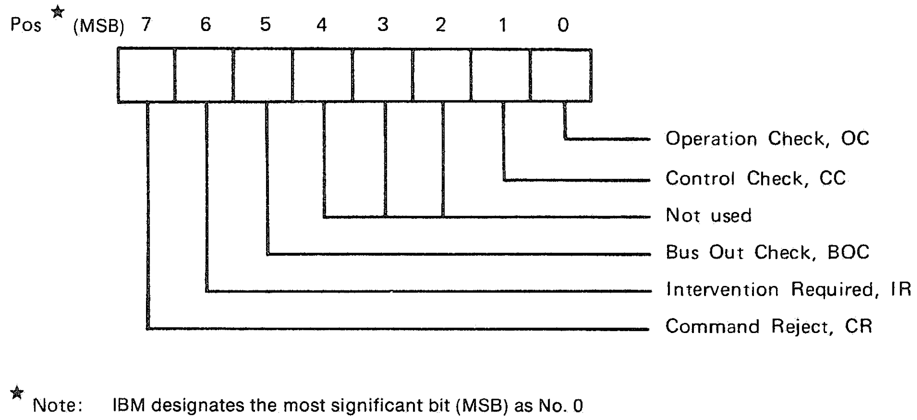


Fig. 3. Sense byte bit assignments

Command Reject, CR

The Command Reject bit is set when the communication processor has received an invalid command, provided that no parity error was obtained in the command byte.

Intervention Required, IR

The Intervention Required bit indicates that some command other than Sense has been sent to a terminal that is not connected or is in the not ready condition (power off or out of paper).

Bus Out Check, BOC

The Bus Out Check bit indicates that the communication processor has detected a parity error in a command or data byte obtained from the channel.

Control Check, CC

The Control Check bit is set when the communication processor detects a timeout condition, i.e. the addressed terminal does not carry out the specified operation or respond within the specified time.

Operation Check, OC

The Operation Check bit can be set only in a communication processor emulating a 3274-1D. It indicates that an illegal command chaining has been carried out after a Select RB, Select RMP, Select RBP or Select WRT command.

Contents

Appendix 1

Code Conversion Table

Bits 5–0	Graphic (SE/FI ALT.)	EBCDIC	Bits 5–0	Graphic (SE/FI ALT.)	EBCDIC
00 0000	SP	40	10 0000	–	60
00 0001	A	C1	10 0001	/	61
00 0010	B	C2	10 0010	S	E2
00 0011	C	C3	10 0011	T	E3
00 0100	D	C4	10 0100	U	E4
00 0101	E	C5	10 0101	V	E5
00 0110	F	C6	10 0110	W	E6
00 0111	G	C7	10 0111	X	E7
00 1000	H	C8	10 1000	Y	E8
00 1001	I	C9	10 1001	Z	E9
00 1010	ö	4A	10 1010	¦	6A
00 1011	.	4B	10 1011	,	6B
00 1100	<	4C	10 1100	%	6C
00 1101	(	4D	10 1101	_	6D
00 1110	+	4E	10 1110	>	6E
00 1111	|	4F	10 1111	?	6F
01 0000	&	50	11 0000	0	F0
01 0001	J	D1	11 0001	1	F1
01 0010	K	D2	11 0010	2	F2
01 0011	L	D3	11 0011	3	F3
01 0100	M	D4	11 0100	4	F4
01 0101	N	D5	11 0101	5	F5
01 0110	O	D6	11 0110	6	F6
01 0111	P	D7	11 0111	7	F7
01 1000	Q	D8	11 1000	8	F8
01 1001	R	D9	11 1001	9	F9
01 1010	å	5A	11 1010	:	7A
01 1011	Å	5B	11 1011	Ä	7B
01 1100	*	5C	11 1100	Ö	7C
01 1101	)	5D	11 1101	'	7D
01 1110	;	5E	11 1110	=	7E
01 1111	¬	5F	11 1111	ä	7F

Note:The following are handled internally as 6-bit structured data

- Attribute characters, ATB
- Write and copy control characters, WCC and CCC
- Status and sense information
- Attention identification, AID
- Buffer and cursor addresses
- Communication processor and terminal addresses, CU and DV

When any of the above is received by the terminal system, only bits 5–0 are used. When any of these characters is transmitted to the computer, the terminal system assigns the appropriate EBCDIC code. As an example, the buffer address 5D7F$_{16}$ in EBCDIC corresponds to 0111,01,11,1111$_2$ internally, i.e. (0)77F$_{16}$ or 1919$_{10}$ (Line 24, pos 80).

Appendix 2

Line Codes and Keyboard Layouts

The Alfaskop System 41 includes the following national versions of keyboard layouts as standard:

The actual layouts of the various national versions of the keyboard are similar to the IBM keyboards, but the Datasaab keyboards have different names (e.g. Swedish/Finnish instead of Swedish).

Non-standard keyboard layouts can be ordered on an RPQ basis.

Swedish/Finnish Line Codes

The characters to the left are valid for SE/FI, those to the right for SE/FI alternate.

Bits	7	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
	6	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	5	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	4	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
3210	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	/DLE			SP	&	—						ä ¦	å ¦	É ¦	0
0001	1	/SOH	SBA					/		a	j	ü ¦		A	J		1
0010	2	/STX	EUA		/SYN					b	k	s		B	K	S	2
0011	3	/ETX	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			/ETB						f	o	w		F	O	W	6
0111	7			/ESC	/EOT					g	p	x		G	P	X	7
1000	8					¦ #	¦ $	¦ ″	¦ !	h	q	y		H	Q	Y	8
1001	9		EM			¦ é	¦ É	¦ ü	é ¦ Ü	i	r	z		I	R	Z	9
1010	A					§ ¦ ō	¤ ¦ å	ö ¦ ¦	:								
1011	B	VT				•	Å	,	Ä								
1100	C	FF	DUP		RA	<	*	%	Ö								
1101	D		SF	/ENQ	/NAK	(	)	_	'								
1110	E		FM			+	;	>	=								
1111	F		/ITB	USM		! ¦ |	^ ¦ ¬	?	″ ¦ ā								/PAD

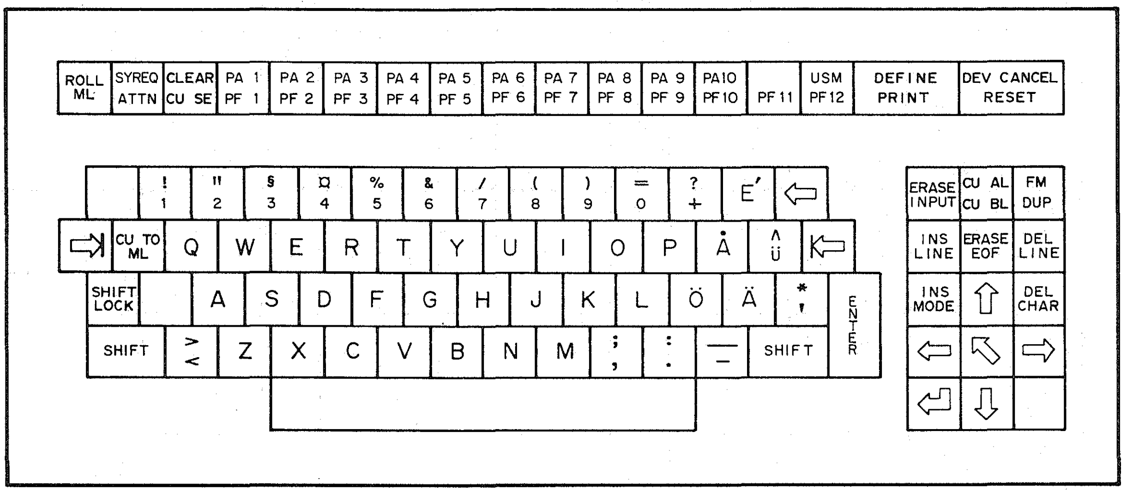
Notes: 1. Character code assignments not specified in the above table are undefined. If an undefined character code is used, the character that will be displayed is an overbar (‾). 60_{16} will be returned on a subsequent read operation.

2. NUL, NL, EM and FF are not displayed.

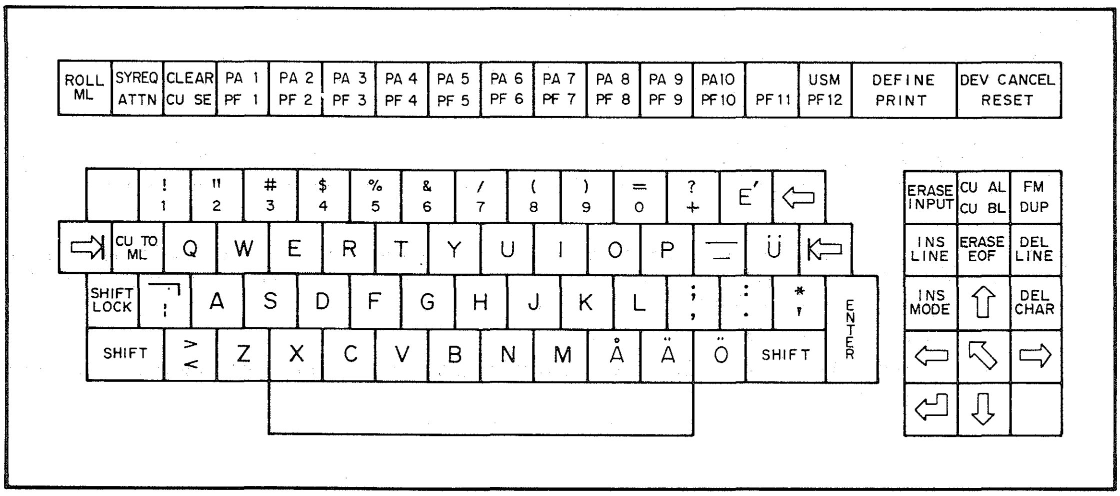
3. The DUP and FM control characters are displayed as * and ; respectively and printed as * and ; respectively.

4. / denotes codes used for remote operation – BSC only.

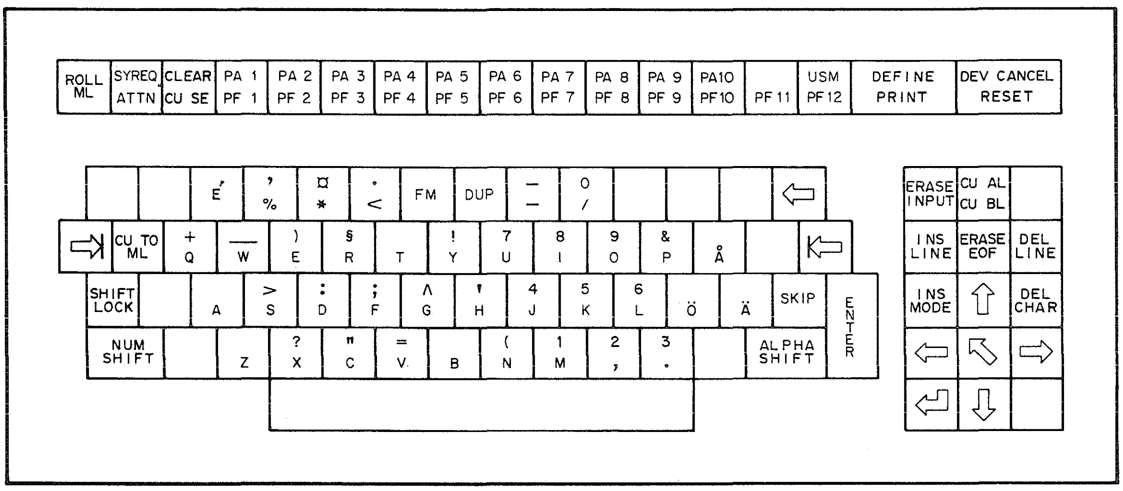
Keyboard Layout for Swedish/Finnish Typewriter Keyboard Unit



Keyboard Layout for Swedish/Finnish Alternate Typewriter Keyboard Unit



Keyboard Layout for Swedish/Finnish Data Entry Keyboard Unit



Danish/Norwegian Line Codes

The characters to the left are valid for DK/NO, those to the right for DK/NO alternate

Bits	7	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
	6	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	5	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	4	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
3210	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	/DLE			SP	&	—						æ	å	\`	0
0001	1	/SOH	SBA					/		a	j	ü		A	J		1
0010	2	/STX	EUA		/SYN					b	k	s		B	K	S	2
0011	3	/ETX	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			/ETB						f	o	w		F	O	W	6
0111	7			/ESC	/EOT					g	p	x		G	P	X	7
1000	8					§	¤	"		h	q	y		H	Q	Y	8
1001	9		EM					ü	\` Ü	i	r	z		I	R	Z	9
1010	A					# ø	¤ å	ø ¦	:								
1011	B	VT				.	Å	,	Æ								
1100	C	FF	DUP		RA	<	*	%	Ø								
1101	D		SF	/ENQ	/NAK	(	)	—	'								
1110	E		FM			+	;	>	=								
1111	F		/ITB	USM		! |	^ ¬	?	" æ								/PAD

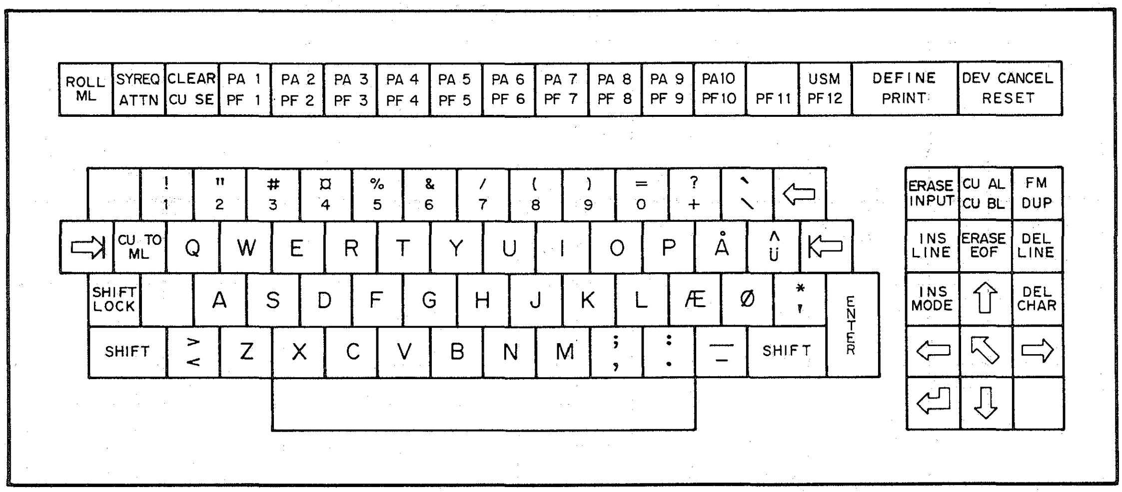
Notes: 1. Character code assignments not specified in the above table are undefined. If an undefined character code is used, the character that will be displayed is an overbar (—). 60_{16} will be returned on a subsequent read operation.

2. NUL, NL, EM and FF are not displayed.

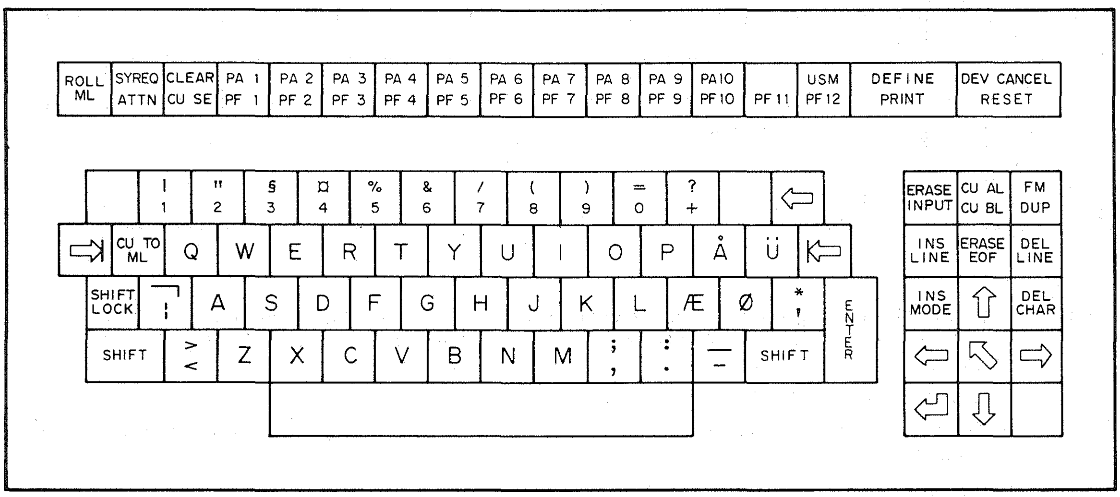
3. The DUP and FM control characters are displayed as $\bar{*}$ and $\bar{;}$ respectively and printed as * and ; respectively.

4. / denotes codes used for remote operation – BSC only.

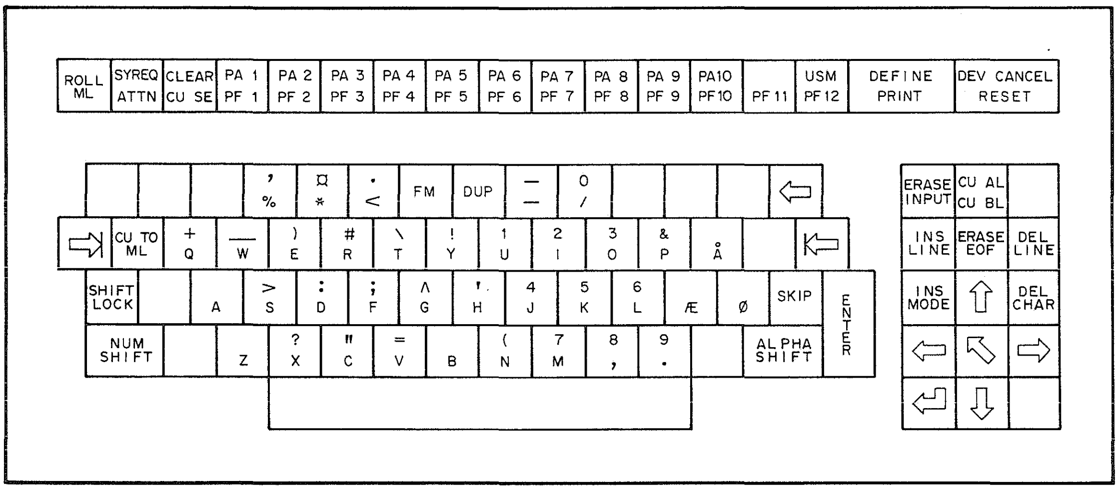
Keyboard Layout for Danish Typewriter Keyboard Unit



Keyboard Layout for Danish/Norwegian Alternate Typewriter Keyboard Unit



Keyboard Layout for Danish Data Entry Keyboard Unit



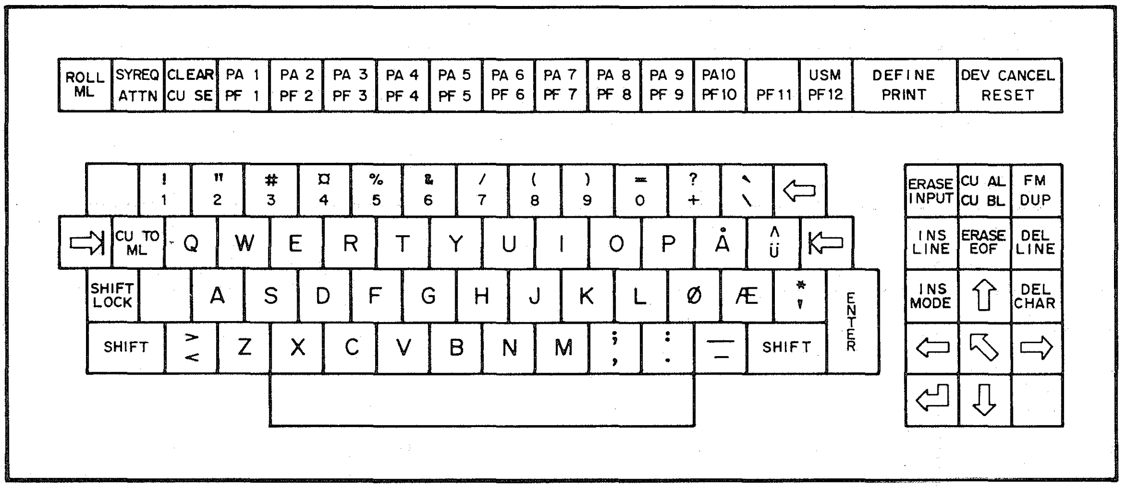
Danish/Norwegian Line Codes

The characters to the left are valid for DK/NO, those to the right for DK/NO alternate

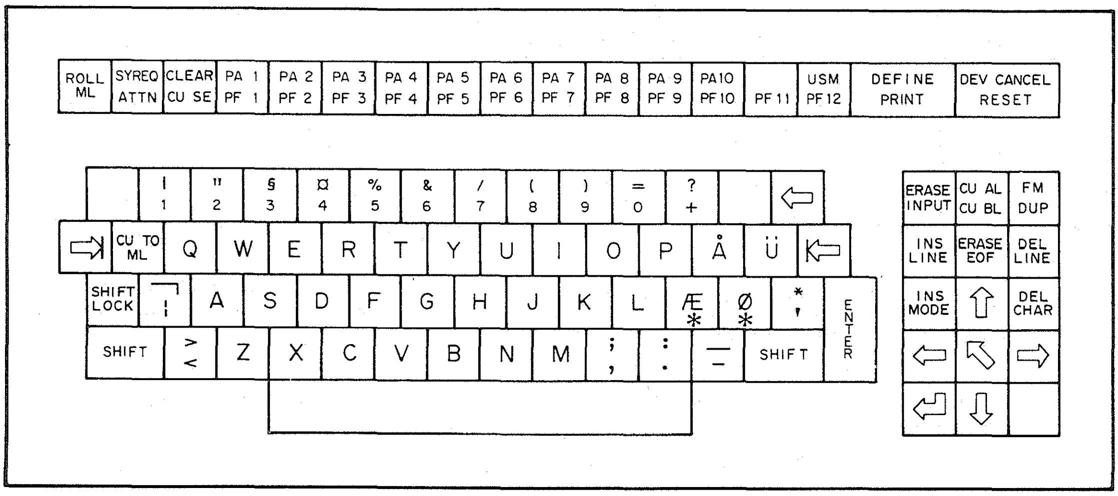
Bits 7 6 5 4		0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
3210	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	/DLE			SP	&	—						æ ‖	å ‖	\ ‖	0
0001	1	/SOH	SBA					/		a	j	ü ‖		A	J		1
0010	2	/STX	EUA		/SYN					b	k	s		B	K	S	2
0011	3	/ETX	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			/ETB						f	o	w		F	O	W	6
0111	7			/ESC	/EOT					g	p	x		G	P	X	7
1000	8					‖ §	‖ ¤	‖ ″		h	q	y		H	Q	Y	8
1001	9		EM					‖ ü	` ‖ Ü	i	r	z		I	R	Z	9
1010	A					# ‖ ø	¤ ‖ å	ø ‖ ¦	:								
1011	B	VT				.	Å	,	Æ								
1100	C	FF	DUP		RA	<	*	%	Ø								
1101	D		SF	/ENQ	/NAK	(	)	—	'								
1110	E		FM			+	;	>	=								
1111	F		/ITB	USM		! ‖ |	^ ‖ ¬	?	″ ‖ æ								/PAD

Notes:
1. Character code assignments not specified in the above table are undefined. If an undefined character code is used, the character that will be displayed is an overbar (—). 60_{16} will be returned on a subsequent read operation.
2. NUL, NL, EM and FF are not displayed.
3. The DUP and FM control characters are displayed as $\overline{*}$ and $\overline{;}$ respectively and printed as * and ; respectively.
4. / denotes codes used for remote operation – BSC only.

Keyboard Layout for Norwegian Typewriter Keyboard Unit

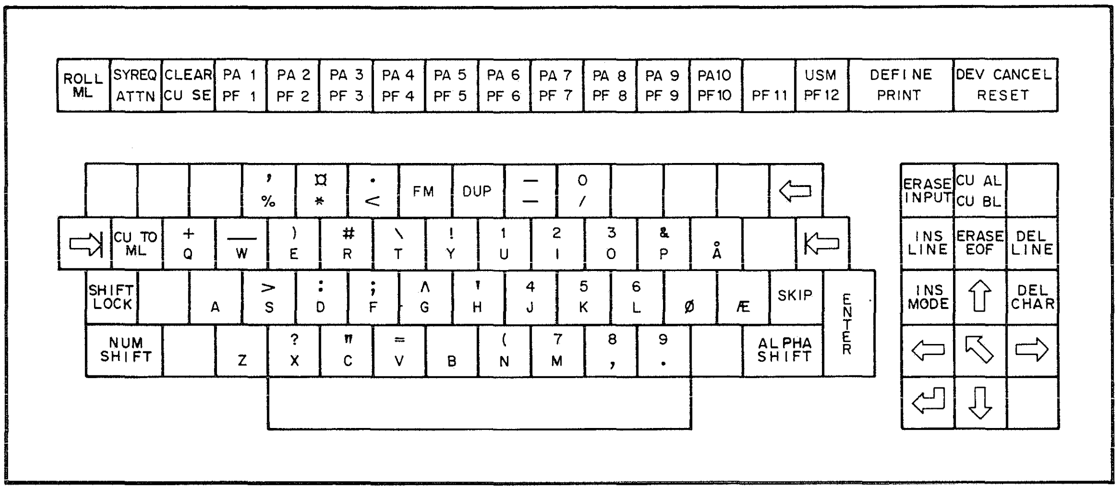


Keyboard Layout for Danish/Norwegian Alternate Typewriter Keyboard Unit



*May be customized differently if desired

Keyboard Layout for Norwegian Data Entry Keyboard Unit



Austrian/German Line Codes

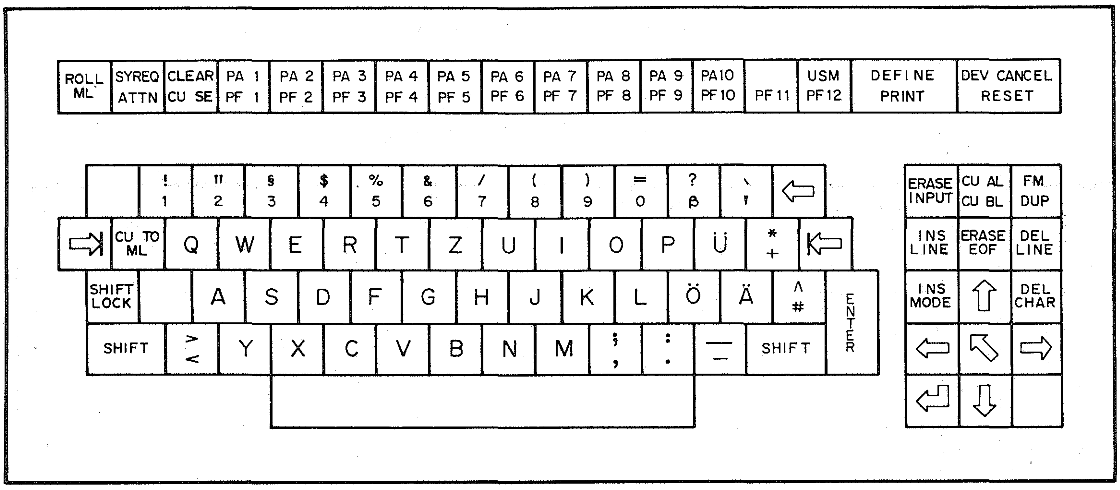
The characters to the left are valid for DE/AT, those to the right for DE/AT alternate.

Bits	7 6 5 4	0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
3210	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	/DLE			SP	&	—						ä	ü	ö	0
0001	1	/SOH	SBA					/		a	j	ß		A	J		1
0010	2	/STX	EUA		/SYN					b	k	s		B	K	S	2
0011	3	/ETX	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			/ETB						f	o	w		F	O	W	6
0111	7			/ESC	/EOT					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					Ä ō	Ü ü	Ö ß	:								
1011	B	VT				.	$ Ü	,	# Ā								
1100	C	FF	DUP		RA	<	*	%	§ Ō								
1101	D		SF	/ENQ	/NAK	(	)	___	'								
1110	E		FM			+	;	>	=								
1111	F		/ITB	USM		! |	^ ¬	?	" ā								/PAD

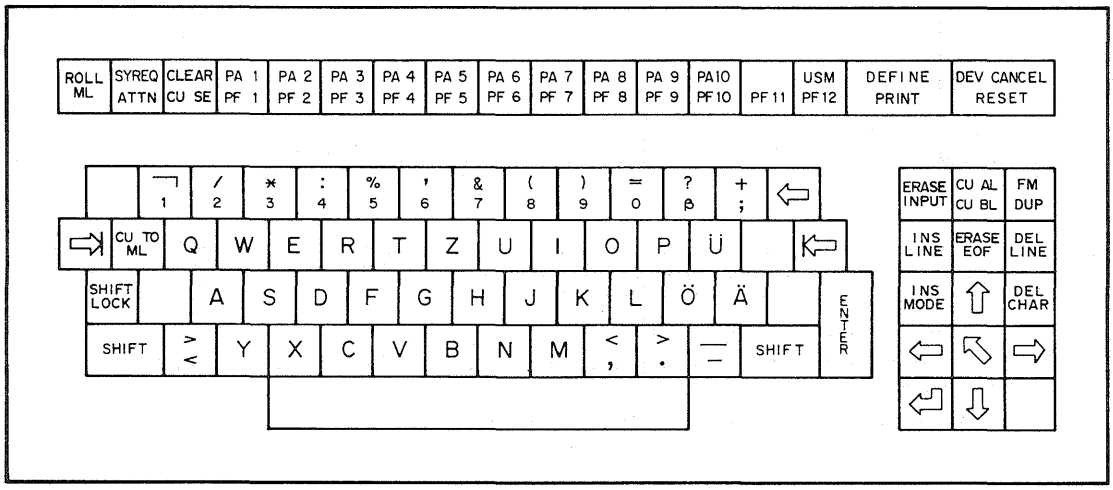
Notes:

1. Character code assignments not specified in the above table are undefined. If an undefined character code is used, the character that will be displayed is an overbar (—). 60_{16} will be returned on a subsequent read operation.
2. NUL, NL, EM and FF are not displayed.
3. The DUP and FM control characters are displayed as $\bar{*}$ and $\bar{;}$ respectively and printed as * and ; respectively.
4. / denotes codes used for remote operation – BSC only.

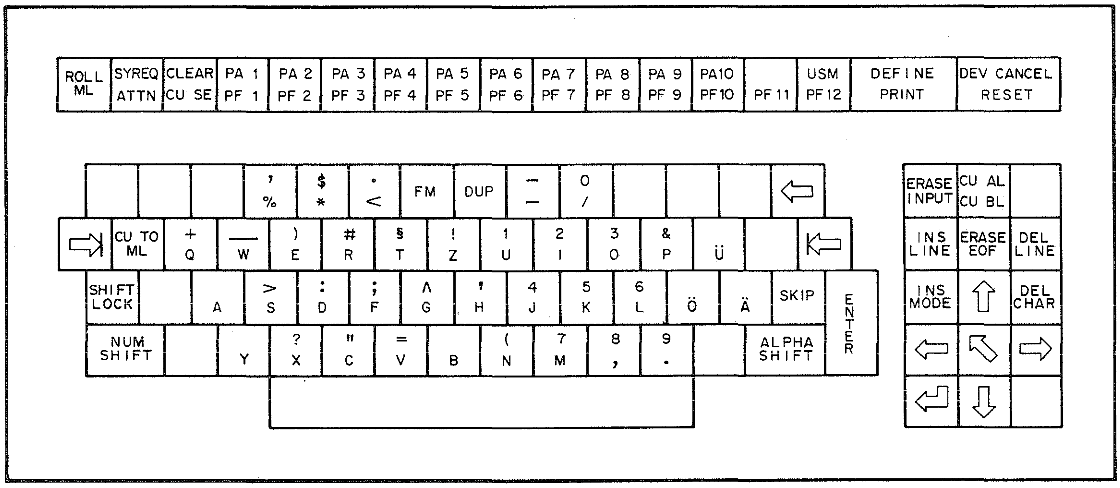
Keyboard Layout for Austrian/German Typewriter Keyboard Unit



Keyboard Layout for Austrian/German Alternate Typewriter Keyboard Unit



Keyboard Layout for Austrian/German Data Entry Keyboard Unit

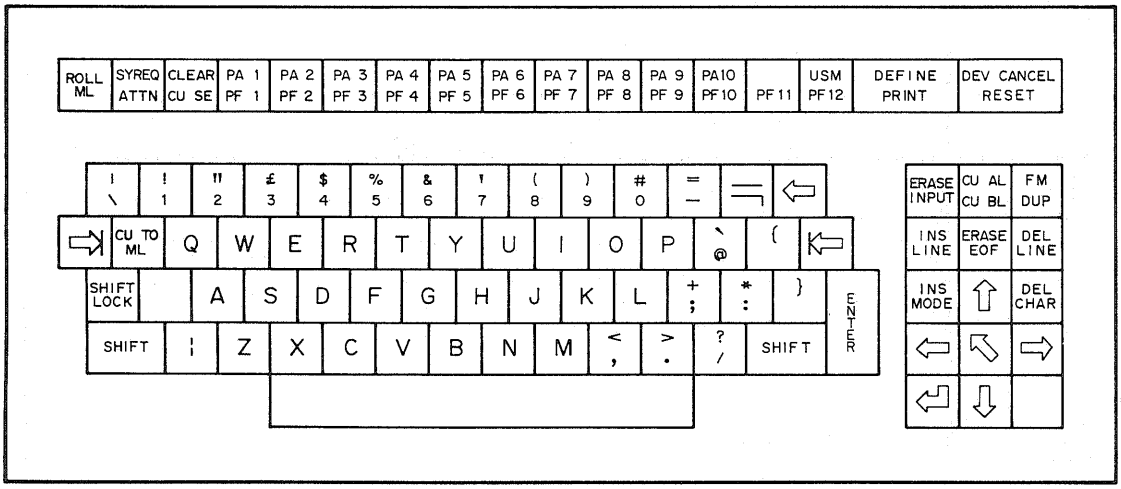


English (GB) Line Codes

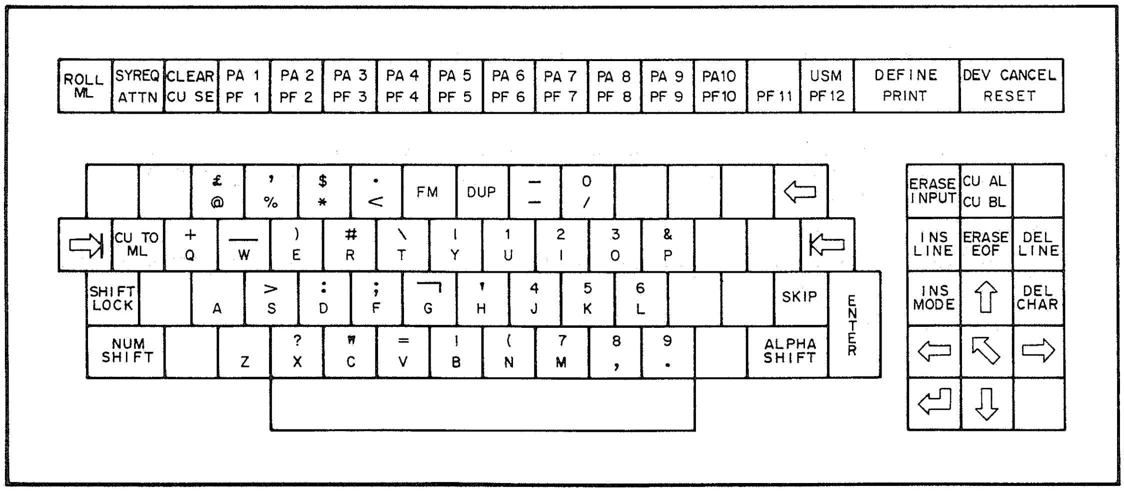
Bits	7 6 5 4	0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
3210	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	/DLE			SP	&	–						{	}	\	0
0001	1	/SOH	SBA					/		a	j	—		A	J		1
0010	2	/STX	EUA		/SYN					b	k	s		B	K	S	2
0011	3	/ETX	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			/ETB						f	o	w		F	O	W	6
0111	7			/ESC	/EOT					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	VT				.	£	,	#								
1100	C	FF	DUP		RA	<	*	%	@								
1101	D		SF	/ENQ	/NAK	(	)	_	'								
1110	E		FM			+	;	>	=								
1111	F		/ITB	USM		|	¬	?	"								/PAD

Notes:
1. Character code assignments not specified in the above table are undefined. If an undefined character code is used, the character that will be displayed is a hyphen (–). 60_{16} will be sent to the computer on a subsequent read operation.
2. NUL, NL, EM and FF are not displayed.
3. The DUP and FM control characters are displayed as $\bar{*}$ and $\bar{;}$ respectively and printed as * and ; respectively.
4. / denotes codes used for remote operation – BSC only.

Keyboard Layout for English (GB) Typewriter Keyboard Unit



Keyboard Layout for English (GB) Data Entry Keyboard Unit



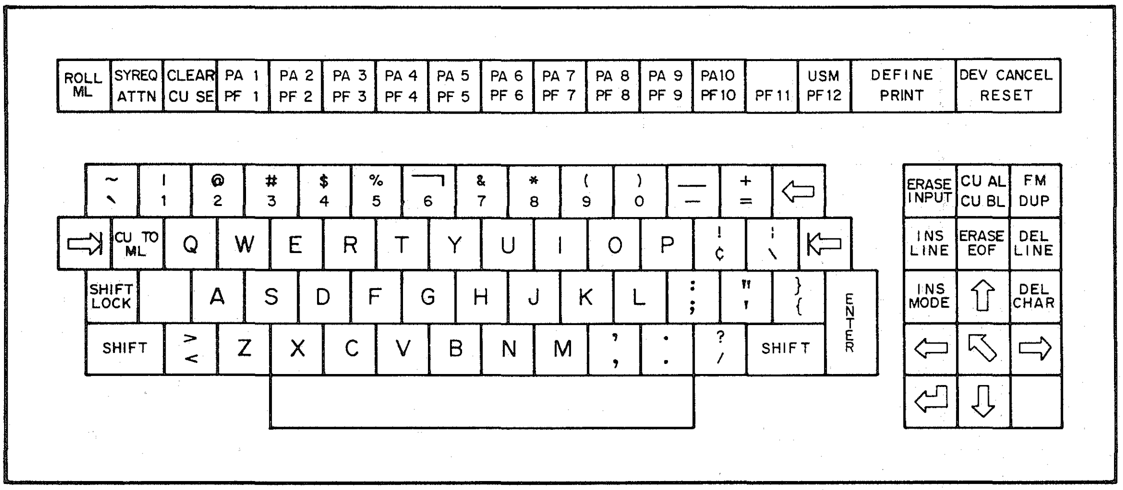
North American (US) Line Codes

Bits	7 6 5 4	0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
3 2 1 0	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0 0 0 0	0	NUL	/DLE			SP	&	–						{	}	\	0
0 0 0 1	1	/SOH	SBA					/		a	j	~		A	J		1
0 0 1 0	2	/STX	EUA		/SYN					b	k	s		B	K	S	2
0 0 1 1	3	/ETX	IC							c	l	t		C	L	T	3
0 1 0 0	4									d	m	u		D	M	U	4
0 1 0 1	5	PT	NL							e	n	v		E	N	V	5
0 1 1 0	6			/ETB						f	o	w		F	O	W	6
0 1 1 1	7			/ESC	/EOT					g	p	x		G	P	X	7
1 0 0 0	8									h	q	y		H	Q	Y	8
1 0 0 1	9		EM						`	i	r	z		I	R	Z	9
1 0 1 0	A					¢	!	¦	:								
1 0 1 1	B	VT				.	$	,	#								
1 1 0 0	C	FF	DUP		RA	<	*	%	@								
1 1 0 1	D		SF	/ENQ	/NAK	(	)	_	'								
1 1 1 0	E		FM			+	;	>	=								
1 1 1 1	F		/ITB	USM		|	¬	?	"								/PAD

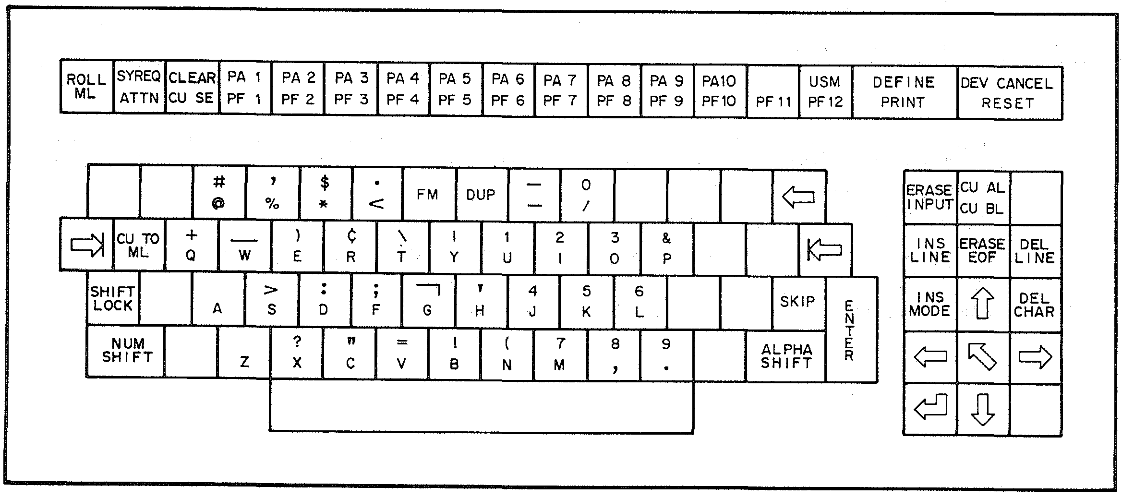
Notes:

1. Character code assignments not specified in the above table are undefined. If an undefined character code is used, the character that will be displayed is an overbar (‾). 60_{16} will be returned on a subsequent read operation.
2. NUL, NL, EM and FF are not displayed.
3. The DUP and FM control characters are displayed as $\bar{*}$ and $\bar{;}$ respectively and printed as * and ; respectively.
4. / denotes codes used for remote operation – BSC only.

Keyboard Layout for North American (US) Typewriter Keyboard Unit



Keyboard Layout for North American (US) Data Entry Keyboard Unit



Spanish Line Codes

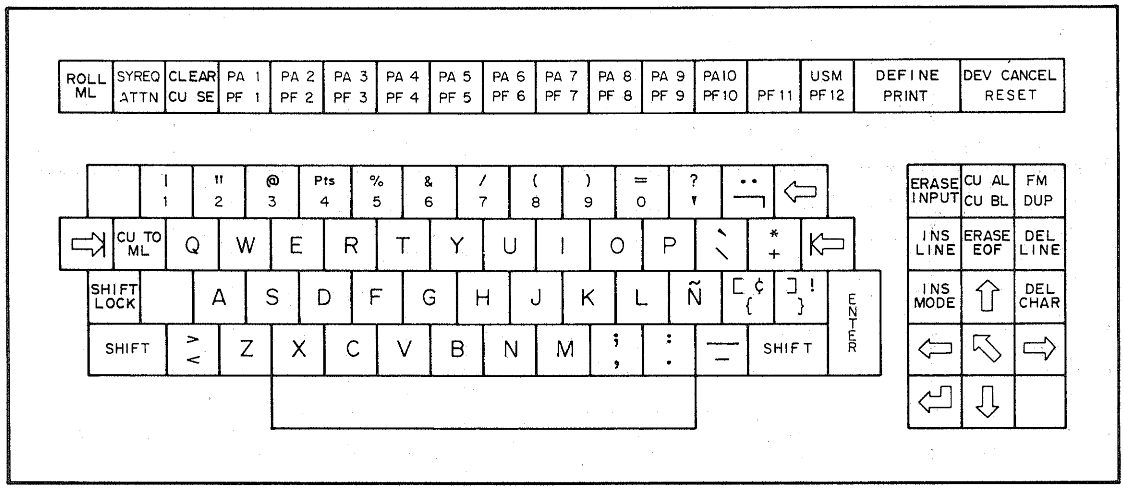
The characters to the left are valid for ES, those to the right for ES alternate

Bits	7 6 5 4	0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
3210	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	/DLE			SP	&	—						{	}	\	0
0001	1	/SOH	SBA					/		a	j	..		A	J		1
0010	2	/STX	EUA		/SYN					b	k	s		B	K	S	2
0011	3	/ETX	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			/ETB						f	o	w		F	O	W	6
0111	7			/ESC	/EOT					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					[¢	] !	ñ ¦	:								
1011	B	VT				∘	Pts	,	Ñ								
1100	C	FF	DUP		RA	<	*	%	@								
1101	D		SF	/ENQ	/NAK	(	)	___	'								
1110	E		FM			+	;	>	=								
1111	F		/ITB	USM		|	¬	?	" ñ								/PAD

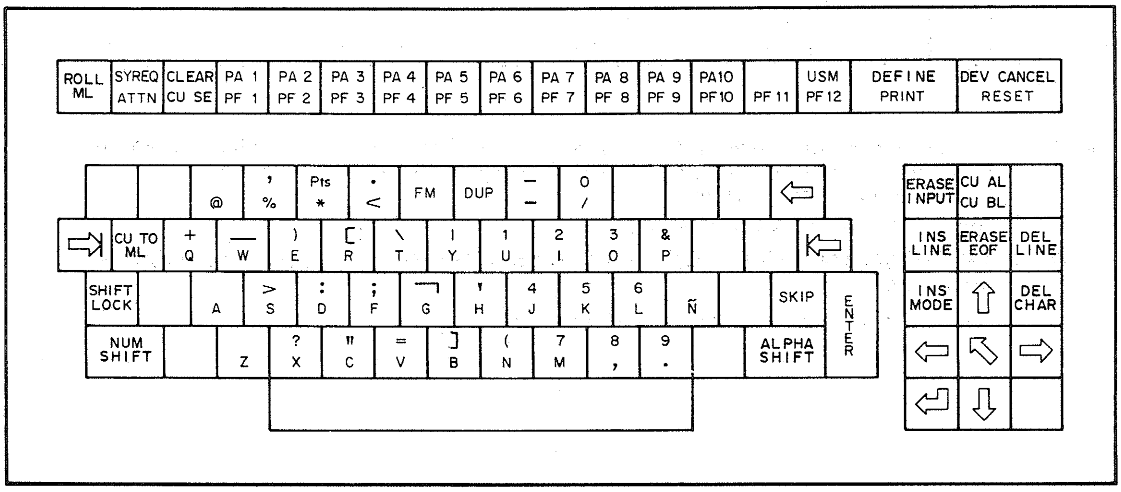
Notes:

1. Character code assignments not specified in the above table are undefined. If an undefined character code is used, the character that will be displayed is an overbar (‾). 60_{16} will be returned on a subsequent read operation.
2. NUL, NL, EM and FF are not displayed.
3. The DUP and FM control characters are displayed as $\bar{*}$ and $\bar{;}$ respectively and printed as * and ; respectively.
4. / denotes codes used for remote operation – BSC only.

Keyboard Layout for Spanish Typewriter Keyboard Unit



Keyboard Layout for Spanish Data Entry Keyboard Unit



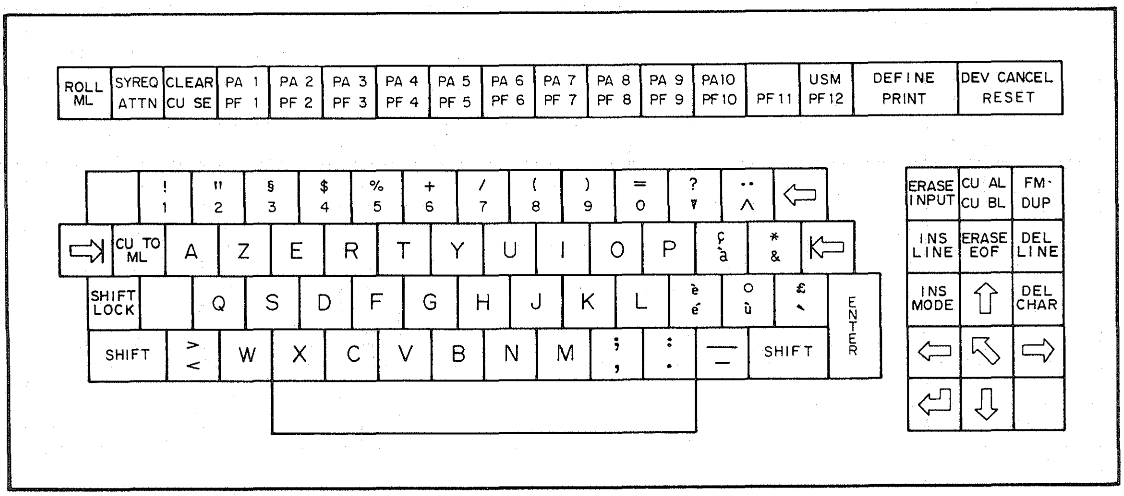
French Line Codes

Bits 7 6 5 4		0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
3 2 1 0	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0 0 0 0	0	NUL	/DLE			SP	&	—						é	è	ç	0
0 0 0 1	1	/SOH	SBA					/		a	j	¨		A	J		1
0 0 1 0	2	/STX	EUA		/SYN					b	k	s		B	K	S	2
0 0 1 1	3	/ETX	IC							c	l	t		C	L	T	3
0 1 0 0	4									d	m	u		D	M	U	4
0 1 0 1	5	PT	NL							e	n	v		E	N	V	5
0 1 1 0	6			/ETB						f	o	w		F	O	W	6
0 1 1 1	7			/ESC	/EOT					g	p	x		G	P	X	7
1 0 0 0	8									h	q	y		H	Q	Y	8
1 0 0 1	9		EM						`	i	r	z		I	R	Z	9
1 0 1 0	A					°	§	ù	:								
1 0 1 1	B	VT				.	$	,	£								
1 1 0 0	C	FF	DUP		RA	<	*	%	à								
1 1 0 1	D		SF	/ENQ	/NAK	(	)	_	'								
1 1 1 0	E		FM			+	;	>	=								
1 1 1 1	F		/ITB	USM		!	^	?	"								/PAD

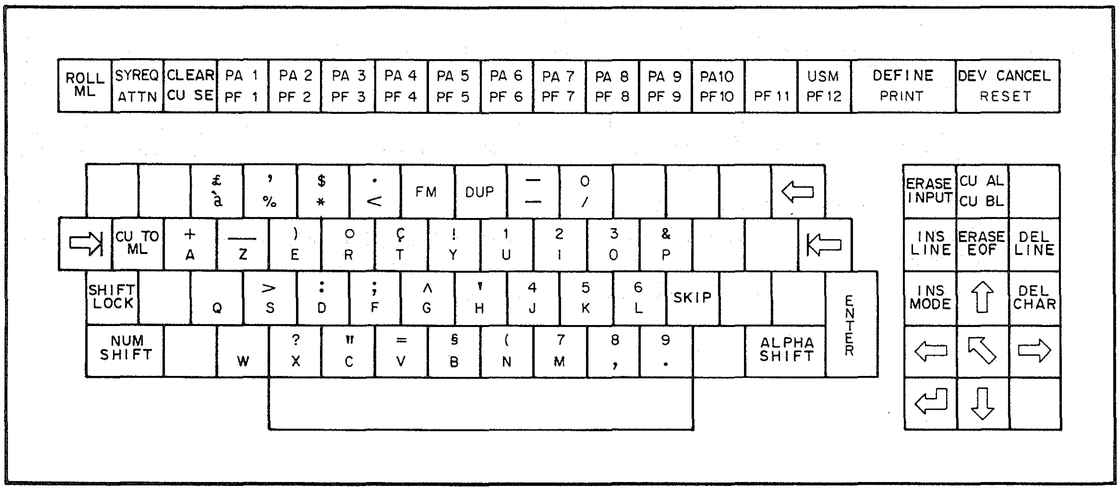
Notes:

1. Character code assignments not specified in the above table are undefined. If an undefined character code is used, the character that will be displayed is an overbar (‾). 60_{16} will be returned on a subsequent read operation.
2. NUL, NL, EM and FF are not displayed.
3. The DUP and FM control characters are displayed as $\bar{*}$ and $\bar{;}$ respectively and printed as * and ; respectively.
4. / denotes codes used for remote operation – BSC only.

Keyboard Layout for French Typewriter Keyboard Unit



Keyboard Layout for French Data Entry Keyboard Unit

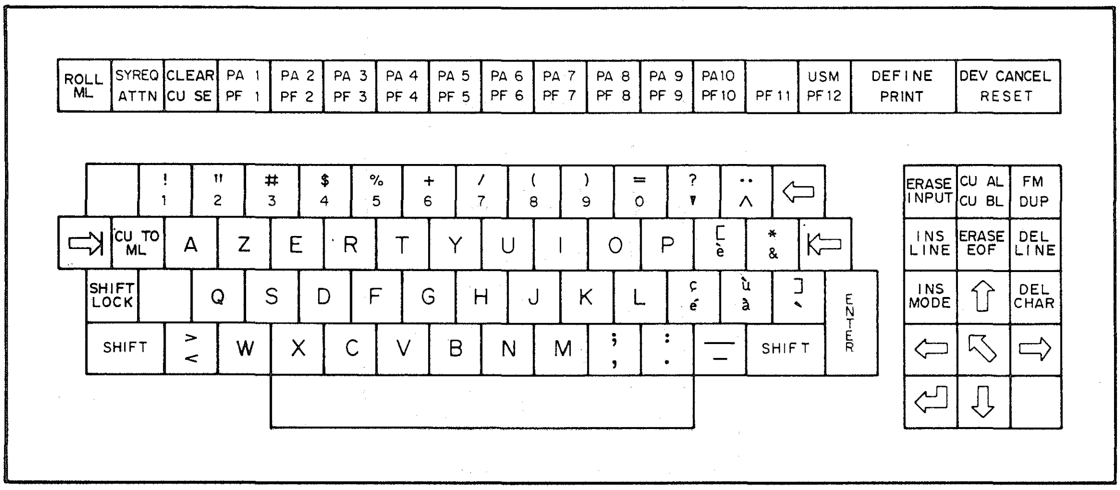


Belgian Line Codes

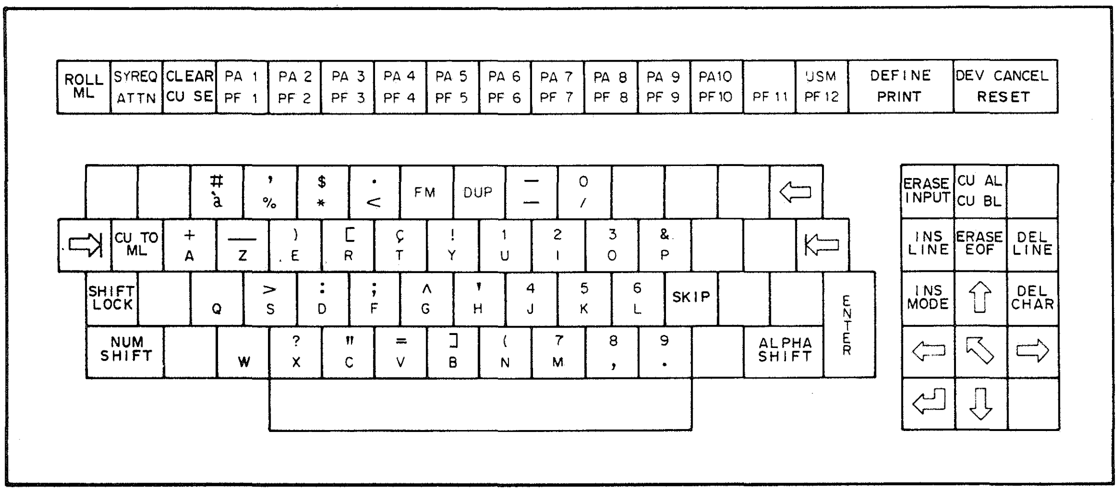
Bits																	
	7	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
	6	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	5	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	4	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
3210	Hex	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0	NUL	/DLE			SP	&	—						é	è	ç	0
0001	1	/SOH	SBA					/		a	j	..		A	J		1
0010	2	/STX	EUA		/SYN					b	k	s		B	K	S	2
0011	3	/ETX	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			/ETB						f	o	w		F	O	W	6
0111	7			/ESC	/EOT					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					[	]	ù	:								
1011	B	VT				.	$	,	#								
1100	C	FF	DUP		RA	<	*	%	à								
1101	D		SF	/ENQ	/NAK	(	)	_	'								
1110	E		FM			+	;	>	=								
1111	F		/ITB	USM		!	^	?	"								/PAD

Notes:
1. Character code assignments not specified in the above table are undefined. If an undefined character code is used, the character that will be displayed is an overbar (—). 60_{16} will be returned on a subsequent read operation.
2. NUL, NL, EM and FF are not displayed.
3. The DUP and FM control characters are displayed as $\bar{*}$ and $\bar{;}$ respectively and printed as * and ; respectively.
4. / denotes codes used for remote operation – BSC only.

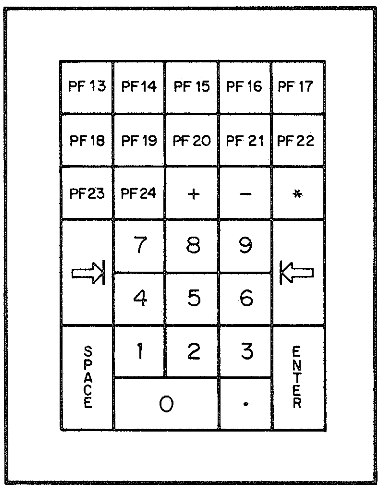
Keyboard Layout for Belgian Typewriter Keyboard Unit



Keyboard Layout for Belgian Data Entry Keyboard Unit



Key Layout for Keyboard Expansion Unit



Appendix 3

Buffer Address Line Codes for 80-character-line formats

2nd Address Byte, b_2 EBCDIC$_{16}$ — Character Position 1–40

Line No	1st Address Byte, b_1 EBCDIC$_{16}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
1	40 C1	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
2	C1 C2	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
3	C2 C3	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
4	C3 C4	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7
5	C5 C6	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
6	C6 C7	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
7	C7 C8	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
8	C8 C9	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7
9	4A 4B	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
10	4B 4C	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
11	4C 4D	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
12	4D 4E	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7
13	4F 50	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
14	50 D1	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
15	D1 D2	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
16	D2 D3	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7
17	D4 D5	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
18	D5 D6	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
19	D6 D7	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
20	D7 D8	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7
21	D9 5A	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
22	5A 5B	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
23	5B 5C	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
24	5C 5D	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7

2nd Address Byte, b_2 EBCDIC$_{16}$ — Character Position 41–80

Line No	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
1	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
2	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
3	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
4	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F
5	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
6	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
8	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F
9	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
10	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
11	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
12	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F
13	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
14	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
15	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
16	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F
17	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
18	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
19	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
20	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F
21	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
22	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
23	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
24	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F

(continued)

Buffer Address Line Codes for 80-character-line formats (continued)

Line No	1:st Address Byte, b_1 $EBCDIC_{16}$	2:nd Address Byte, b_2 $EBCDIC_{16}$ Character Position																																							
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
25	5E 5F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
26	5F 60	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
27	60 61	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
28	61 E2	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7
29	E3 E4	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
30	E4 E5	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
31	E5 E6	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
32	E6 E7	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7
33	E8 E9	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
34	E9 6A	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
35	6A 6B	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
36	6B 6C	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7
37	6D 6E	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
38	6E 6F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
39	6F F0	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7
40	F0 F1	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7
41	F2 F3	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7
42	F3 F4	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7
43	F4 F5	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7

Line No	1:st Address Byte, b_1 $EBCDIC_{16}$	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
25	5E 5F	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
26	5F 60	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
27	60 61	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
28	61 E2	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F
29	E3 E4	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
30	E4 E5	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
31	E5 E6	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
32	E6 E7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F
33	E8 E9	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
34	E9 6A	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
35	6A 6B	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
36	6B 6C	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F
37	6D 6E	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
38	6E 6F	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
39	6F F0	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F
40	F0 F1	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F
41	F2 F3	E8	E9	6A	6B	6C	6D	6E	6F	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F
42	F3 F4	F8	F9	7A	7B	7C	7D	7E	7F	40	C1	C2	C3	C4	C5	C6	C7	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F
43	F4 F5	C8	C9	4A	4B	4C	4D	4E	4F	50	D1	D2	D3	D4	D5	D6	D7	D8	D9	5A	5B	5C	5D	5E	5F	60	61	E2	E3	E4	E5	E6	E7	E8	E9	6A	6B	6C	6D	6E	6F

Appendix 4

Message Line

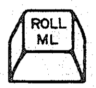
The bottom line on the screen is used to present more detailed information about the status of the terminal system than can be provided by the indicator lamps on the keyboard. Moreover, the operator can, to some extent, determine the operating mode of the terminal system by making entries on the message line. The first two (leftmost) letters on the message line define the message source as follows

OS Status messages from the operating system

EM Status messages from the IBM 3270 emulation software

AP Status messages from the Alfaform software or other system software. See the appropriate reference manuals for descriptions of these messages. This line is only available when such system software as Alfaform is included.

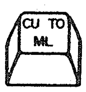
Two of the keyboard keys are used *only* in connection with the message line.



Used to obtain the three different message sources: *OS*, *EM* and *AP*.

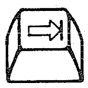
When the system has new information from a source not presently displayed, an automatic roll message line function will be performed, so that the new information appears. The old source can of course be displayed again by help of the ROLL ML key.

If the operator depresses the ROLL ML key repeatedly, the different sources will be displayed in sequence (with wraparound).

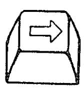


Used to move the cursor back and forth between the regular display area and the first unprotected position on the message line.

Two of the other edit keys carry out functions on the message line.



Moves the cursor to the next unprotected field on the message line or, if there is only one unprotected field on the message line, to the first position in this field.

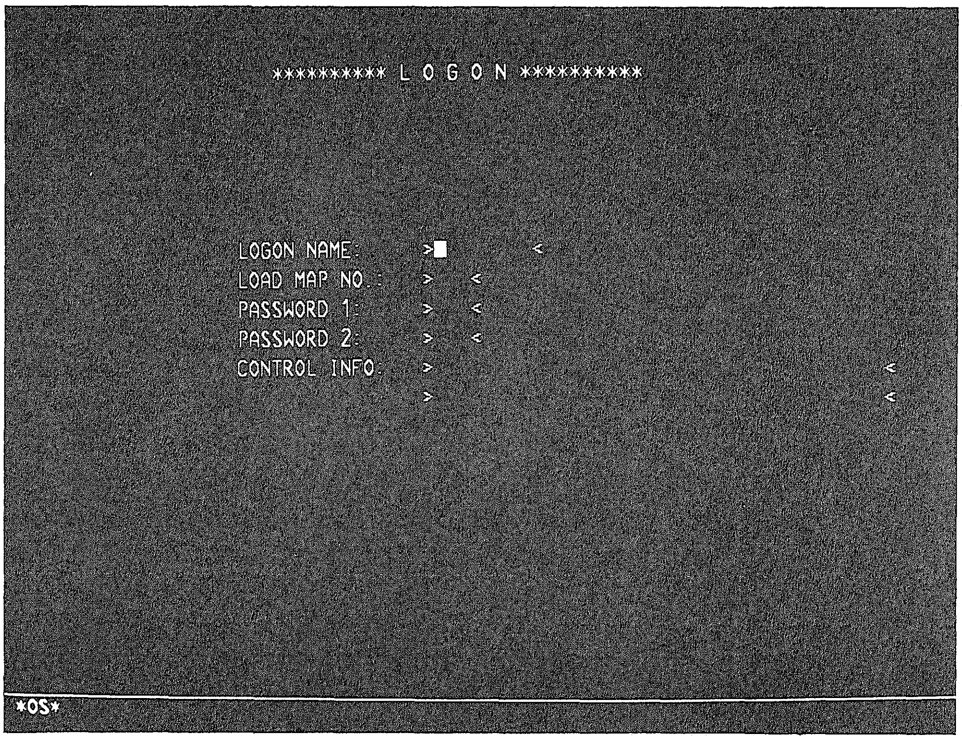


Repetitive or continuous depression of this key moves the cursor (with wraparound) through the unprotected field on the message line in which the cursor is currently located.

Program Logon and Logoff

When power to a display unit is turned on, the following text is presented on the message line

LOAD
After part of the operating system has been loaded, *OS* will replace the word LOAD. When the entire operating system has been loaded, the following LOGON menu will appear, in case automatic logon has not been specified.



The operator then enters the desired name (em 3274 for example), the load map number (001 for example) and, if required, other items of information. The ENTER key shall then be depressed. When the logon operation is completed, the menu will disappear and be replaced by an empty screen or another form, depending upon which software has been loaded. *OS* will still appear on the message line unless the emulation software has been loaded, in which case *EM* will appear on the message line.

If automatic logon was specified for the display unit when the system diskette was customized, no menu will be displayed and, naturally, no entries will have to be made. Instead, during the automatic logon, the text
xxxxxxxxxx A U T O L O G O N xxxxxxxxxx
is presented on the screen.

IPL and Logon Message Line Messages

During initial program loading (IPL) and logon of system software, the error or status indications listed below can appear on the message line.

Text on message line	Error or status
RE	RWM error in basic 32 Kbyte RWM area
PE	PROM error (IPL PROM)
LOAD	DU in IPL phase
LOAD P	DU still in IPL phase, polled by communication processor
LOAD I	DU still in IPL phase, interacting with flexible disk unit

The following error or status indications can appear after the first part of the program has been loaded. Note that each of the indications in the left-hand column is now preceded by *OS*.

Text on message line	Error or status
RESET – DUMP IN PROGRESS	This message, intended for the system engineer, appears briefly after the RESET pushbutton has been pressed (not an error).
DUMPFILE FOR RESET IS FULL	This message, intended for the system engineer, can appear after the RESET pushbutton has been pressed. OS loading proceeds.
DISK FAILURE.FILE $IPLDUMP.F STATUS = XX TYPE = XX	Same as above.
MRW ERROR	This message indicates that an error was detected during the testing of the R/W memory board (call service engineer).
VERSIONS: CP AXXX-XX DU AXXX-XX PU AXXX-XX FD AXXX-XX	This information, which is intended for the system engineer and specifies the operating system versions being used, appears after the operating system has been loaded and remains on the screen until logon is carried out.

When the operating system has been loaded and logon has been ordered, the following error or status indications can appear (preceded by *OS*).

Text on message line	Error or status
ACCESS CONFLICT. DEPRESS ENTER FOR RE-TRY.	Another program uses the same file or library on the flexible disk.
FD TEMPORARILY OVERLOADED. WAIT.	The flexible disk unit job queue is full.
DISK FAILURE. ERROR NUMBER = XX. DEPRESS ENTER FOR RE-TRY.	Hardware error in flexible disk unit, damaged or removed diskette, broken connection etc.

Text on message line	Error on status
INVALID LOGON NAME	No such program found in the logon file (list stored in the display unit)
INVALID LOAD MAP NO.	Same as above
INVALID PASSWORD 1	Invalid password
INVALID PASSWORD 2	Invalid password
INSUFFICIENT AUTHORIZATION	Operator not authorized to log on this program

If any of these error indications should appear, an empty logon menu will once again be displayed together with the error message, and the operator can repeat the logon operation (if the correct diskette is being used).

Furthermore, the message *EM* DISK FAILURE. ERROR NUMBER = XX can be displayed on the message line in the following situations

- Before the LOGON menu is to appear or
- When the filled-in LOGON menu is accepted

and program loading is unsuccessful. In case re-tries are made automatically by the program, the text . W A I T . will also be displayed.

Logoff

The emulation software (or other software) is logged off when the operator moves the cursor to the message line (*OS*) by depressing the CU TO ML key, enters the word LOGOFF and depresses the ENTER key. This will cause the LOGON menu to be displayed again.

The terminal-console-functions software is logged off as the operator depresses the PF 12 key when the first console mode picture is displayed.

Message Line Messages that Start with *EM*

These messages are formatted as follows in BSC and local applications.

EM STATUS PUXX

Message field
Positions 10–30

Field used to specify printer number or class for local printouts.

In SNA/SDLC applications, the *EM* message line is formatted as follows.

EM	MY JOB	CC 02		ASSIGN A PRINTER PU 03
I	II	III	IV	V
	This field is used (among other things) in connection with the selection of the program in the host computer.	This field is used for coded error messages and SNA/SDLC information. This code can be cited when reporting a malfunction.	Field used for error messages issued in connection with (for example) local printout or definition of printer to be used for local printout.	This field is used to assign and define the printer that is to be used for local printout. See subsection which follows.

The following messages can be presented in the message field (field IV for SNA/SDLC applications).

XXXXX XXXX MXXX-XX This information, which is intended for the system engineer and specifies the emulation software being used, appears after logon of the emulation program and remains on the screen until a key is depressed on the keyboard.

PUXX IS NOT IN CONFIG — This message is displayed when the operator enters a printer matrix containing a printer that is not included in the configuration. (The printer matrix is sent from the host computer and then entered into the communication processor by the operator.)

WRONG PRINTER MATRIX — This message is displayed when there is a format error in a printer matrix sent from the host computer. It appears when the operator tries to enter the matrix into the communication processor.

PRINTER IS BUSY — This message is displayed when the operator depresses the PRINT key and the previously assigned printer is busy with another printout. The printout request will be queued and executed later unless the operator depresses the DEV CANCEL key.

PRINTER NOT OPERABLE — This message is displayed when the operator depresses the PRINT key and the assigned printer does not function properly (power off, paper out, etc.). This message is also obtained if no printer within an assigned class is functioning. The printout request will be queued and executed when the printer is operable again, unless the operator depresses the DEV CANCEL key.

HOST LINE IS DOWN	This message is displayed when the operator depresses a send-initiating key and the communication line running to the host computer is not functioning.
ASSIGN A PRINTER	When the DEFINE key is depressed, this message is displayed and the letters PU appear immediately to the left of the cursor on the message line. The default printer No. will appear in positions XX. The operator can now enter the number of the desired printer or class of printers for local printout by overwriting the default No. (This message is displayed in field V in SNA/SDLC application.)
WRONG PRINT ID	This message is displayed if the operator enters a printer number or printer class number other than 0–31 or 70–85.
OPERATOR UNAUTHORIZED	This message is displayed when the operator attempts to assign a local-printout printer for which he has not been authorized.
SYSTEM/CP RESET	This message is displayed when the channel has made a reset operation and if the communications processor has been reset. Applies to local operation only.
SELECTIVE RESET	This message is displayed when the channel has made a selective reset operation. Applies to local operation only.

The following messages can be presented in field II in SNA/SDLC applications.

ONLINE	Appears after execution of the ACTPU command.
MY JOB	Appears when the host application program is logged on.
SYSTEM OPERATOR	Appears when the SSCP-SLU session gains ownership of the screen.
UNOWNED	Indicates that the screen is in the unowned state.

Field III is used in SNA/SDLC applications to present two types of error messages, program check (PC) errors or communication check (CC) errors. Different errors are identified by numbers according to following table.

PC 01	Invalid command received.
PC 02	Invalid (out of range) address was received following an SBA, RA or EUA order.
PC 03	Data stream was received following a Read, Read Mod. or EAU command.
PC 11	LU1 RU longer than specified in Bind parameters.
PC 13	Attempted function is not supported.
PC 20	LIC called for exception response but Bind parameters specified definite response.
PC 21	LIC called for definite response but Bind parameters specified exception response.
PC 22	No-response called for but not allowed.
PC 23	Format indicator (FI) bit is not allowed (for FM data RUs).
PC 30	Sequence number error in TH.
PC 31	Chaining error.
PC 32	Bracket error.
PC 33	Data received while display unit was in data traffic reset state.
PC 34	Direction error.
PC 42	Request not executable.
PC 43	Change direction bit required but not received.
PC 50–55	Unacceptable Bind parameters
PC 50	Profile error (FM or TS)
PC 51	Primary protocol error.
PC 52	Secondary protocol error.
PC 53	Common protocol error.

PC 54	Screen size specification error.
PC 55	SLU type error.
PC 98	Negative response received.
CC 02	No ready for sending signal received from modem.
CC 05	Disconnect command.
CC 18	Incorrect segment sequence was received.
CC 28	Frame reject

Appendix 5

SNA Command Parameters

The parameters sent together with the ACTLU, ACTPU, Bind, DACTPU and Unbind commands are presented here. The S41 does not check all the parameters.

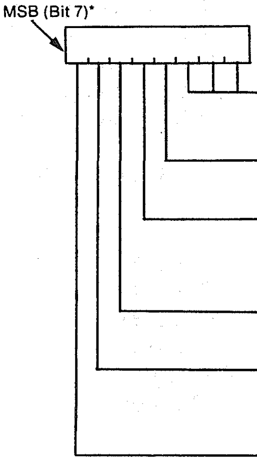
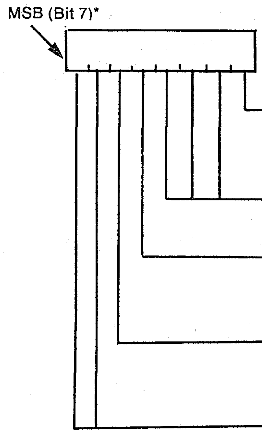
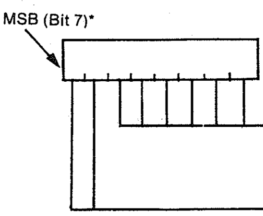
- Bytes or bits which are not used are never checked.
- Bytes or bits which *should* be set to a certain value are not checked and the indicated operation is obtained even if the bits are received with other values.
- Bytes or bits which *must* be set to a certain value are checked and the command is rejected if any other value is received.
- Bytes or bits for which more than one value are specified are checked and the S41 operation is obtained as indicated by the received value.

Bind parameters

Byte No.	Content	Explanation
0	31	Bind command
1	01	Bind type. Must be 01 (non-negotiable)
2	03	Function Management (FM) profile. Must be 03. FM profile 3 specifies immediate response mode
3	03	Transmission Subsystem (TS) profile. Must be 03. TS profile 3 specifies that pacing and sequence numbers are to be used for normal flow and that Clear and Start Data Traffic commands control transmissions

Byte No.	Content	Explanation
4	MSB (Bit 7)*	Primary LU protocols (Bytes 4–7: FM usage)
	Must be 1	PLU can send EB
	Must be 0	PLU cannot send compressed data
	Not used	
	Responses	00 Will be rejected by S41 01 The PLU can specify only exception RSP 10 The PLU can specify only definite RSP 11 The PLU can specify exception or definite RSP
	Must be 0	Immediate request mode used
	Chaining	0 The PLU can send only single-element chains 1 The PLU can send single- or multiple-element chains
5	MSB (Bit 7)*	Secondary LU protocols
	(Should be 0)	The S41 cannot send EB
	(Should be 0)	The S41 cannot send compressed data
	Not used	
	Responses	00 Will be rejected by S41 01 The S41 can specify only exception RSP 10 The S41 can specify only definite RSP 11 The S41 can specify exception or definite RSP
	(Should be 0)	The S41 uses immediate request mode
	Chaining	Must be 1 for LU type 2 indicating that the S41 can send single- or multiple-element chains (the bit is not checked for LU type 1 or 3)

*Note: IBM designates the most significant bit (Bit No. 7) as bit No. 0.

Byte No.	Content	Explanation
6	 MSB (Bit 7)*	Common protocols
	Not used	
	Must be 0	EBCDIC code must be used
	Must be 1	Bracket termination rule 1 (conditional termination) is used
	Must be 1	Bracket protocols are used
	Must be 0	The S41 does not support FM header
	Not used	
7	 MSB (Bit 7)*	Common protocols
	(Should be 0)	The S41 sends normal flow first (after session is activated)
	Not used	
	Must be 0	The S41 is contention winner and brackets first speaker
	Must be 0	The PLU is responsible for error recovery
	Must be 10	HDX-FF transmission is used
8		Not supported by S41. Not checked. (Bytes 8–13: TS usage.)
9	 MSB (Bit 7)*	
	PLU-S41 pacing count. (Zero if pacing is not used.)	
	Not used	

*Note: IBM designates the most significant bit (Bit No. 7) as bit No. 0.

Byte No.	Content	Explanation
10		Maximum RU size to be sent by S41. Value must be ⩾ 64. The S41 never sends RUs longer than 1024. The value is expressed as mantissa (the four MSB; only values 8 through F to be used) and exponent (the four LSB) Example: C6 means $12 \times 2^6 = 768$. Other examples. 83 means 64 bytes 85 means 256 bytes 86 means 512 bytes 87 means 1024 bytes A6 means 640 bytes F5 means 480 bytes
11		Maximum RU size to be sent by PLU. The value is specified in the same way as for byte 10. For LU type 1 or 3 RU size is checked against pacing count value. Please refer to section on RU Length Considerations.
12, 13		Not used
14	01, 02 or 03	Presentation Services (PS) profile. Must be 01 for LU type 1, 02 for LU type 2 and 03 for LU type 3
15–19		Not used (Bytes 15–25: PS usage)
20–24		Screen size specifications (presented below)
25–26		Not used (Byte 26: Crypto option)

Default values for the bind parameters: 31, 01, 03, 03, B1, 90, 30, 80, 00, 00, 87, 87, 00, 00, 02, 00, 00, 00, 00, 00, 00, 00, 00, 00, 02, 00, 00

Screen size: Bytes 20–23 are checked only if byte 24 is not 02

If byte 24 is 02, a 24×80 format is obtained.

If byte 24 is 7E, one size is specified in bytes 20 and 21 (see below)

If byte 24 is 7F, two sizes are specified: default screen size in bytes 20 and 21 and an alternate screen size in bytes 22 and 23

- Number of lines is specified in hexadecimal value in byte 20 or 22:

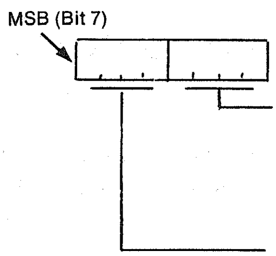
  0C provides 12 lines
  18 provides 24 lines
  20 provides 32 lines
  2B provides 43 lines

- Number of positions per line is specified in hexadecimal value in byte 21 or 23:

  28 provides 40 positions
  50 provides 80 positions

ACTLU Parameters

Byte No.	Content	Explanation
0	0D	ACTLU command
1	01	Should be 01, cold activating
2		



TS profile. Must be 1. A TS profile of 1 specifies no pacing, identifiers rather than sequence numbers, SDT and Clear not supported and maximum RU size of 256 bytes.

FM profile. Must be 0. An FM profile of 0 specifies immediate request and immediate response mode, only single RU chains, definite responses, no DFC RUs, no brackets, SLU may send LUSTAT, HDX-C operation (SLU is contention winner) and SSCP responsible for error recovery

ACTPU Parameters

Byte No.	Content	Explanation
0	11	ACTPU command
1	01	Should be 01, cold activating
2	01	FM and TS profiles (same as for ACTLU command)
3	05	Should be 05, SSCP contained in host node (PU type 5)
4–8		Not used

DACTPU Parameters

Byte No.	Content	Explanation
0	12	DACTPU command
1	02	Should be 02, physical connection is not broken after deactivation

Unbind Parameters

Byte No.	Content	Explanation
0	32	Unbind command
1	01	Should be 01, normal end of session

SNA Response Formats

A one-byte RU (containing the command byte) is returned in a positive response to SC and DFC requests. A null RU is returned in a positive response to FM data requests.

However, the positive response to ACTLU and ACTPU return additional data. The meaning of this additional data is described below.

Four bytes of sense data plus one byte (the command byte) are returned in a negative response to SC and DFC requests. (Sense data is explained in Appendix 6.)

Positive Response to ACTLU

Byte No.	Content	Explanation
0	0D	ACTLU command
1	01	Response format 0 and cold activating
2	01	FM and TS profiles. Returned as 01 if received as 01 in the corresponding request

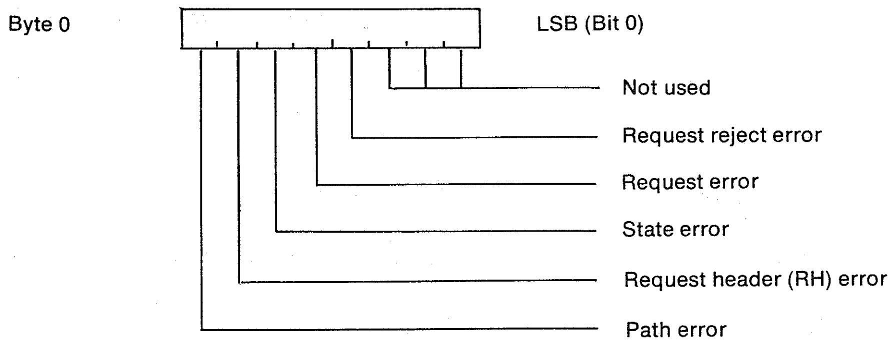
Positive Response to ACTPU

0	11	ACTPU command
1	01	Cold activating
2–9	40	Eight bytes hex 40, (contents ID)

Appendix 6

SNA Sense Codes

The sense data sent from S41 in a negative response includes four bytes. The first byte is a category code and the second a modifier code which, together, are referred to as the "sense code". The two last bytes are all zeros.



Path Errors

Sense bytes	Explanation
8004	DAF not recognized. No LU (display or printer) assigned.
8005	No session established. • The S41 has not accepted Bind command for LU-LU, ACTLU for SSCP-LU or ACTPU for SSCP-PU. • The S41 receives a request, other than bind, with OAF neither 00 or OAF of the PLU in session with addressed DAF.
8008	PU not active. The S41 has not received ACTPU or has deactivated PU internally.
8009	LU not active. The S41 has not received ACTLU or has deactivated LU internally.
800F	Invalid address combination. Example: request to PU (DAF=00) and OAF was not 00 (SSCP).

Request Header (RH) Errors

Sense bytes	Explanation
4006	Exception response called for in LIC but Bind command specified definite response.
4007	Definite response called for in LIC but Bind command specified exception response.
400A	A chain element called for no response.
400F	An FM request indicated header.

State Errors

Sense bytes	Explanation
2001	Sequence number error.
2002	Chaining error.
2003	Bracket state error.
2004	Direction error. The S41 was in send state and received a normal flow request without BB bit.
2005	Data traffic reset state. No SDT command had been accepted when FM data or DFC request was received.

Request Errors

Sense bytes	Explanation
1003	Function not supported • Network control request was received • Unsupported session control request was received. • Unsupported data flow request was received. • Signal code was not 00 01 00 00 • Invalid 3270 data stream. – Data followed a Read, Read Modified, Read Modified All, or Erase All unprotected command. – Any read type command was sent to an LU type 3.

Sense bytes	Explanation
1005	Parameter error. Invalid address followed SBA, RA or EUA order or SCS parameter error.
1007	Category not supported. • An FM data request from SSCP was received by an SLU without keyboard. • An unsupported network service message was received. • An unsupported FM data command was received.

Request Reject Errors

Sense bytes	Explanation
0801	Resource not available. • The printer authorization matrix connects no printer to addressed LU type 2. • Bind command was addressed to a printer authorized for local copies only.
0802	Intervention required at addressed device because of printer out of paper, cover up, paper jam etc.
0805	Session limit exceeded. A bind was received but OAF indicates other PLU from the PLU already in session.
0813	Bracket bid reject – No RTR will be sent. • Display had won contention and started a bracket when Bid command or BB bit was received. • SLU already in in-bracket state when Bid command or BB bit was received.
0814	Bracket bid reject because the printer is doing local copy from a display. RTR will be sent when the printer becomes not busy.
0815	Function active. The SLU is already in session with the PLU that sends Bind command.

Sense bytes	Explanation
081B	Receiver in transmit mode. • The SLU is between bracket but an alphanumeric key has been depressed. • An FM message received from SSCP but the display was owned by PLU-SLU-session. • SSCP-SLU session owned the display, but local copy was in progress when an SSCP FM message was received.
081C	Request not executable since S41 has a non recoverable error.
0821	Invalid session parameters. • Bind parameters were not as set fourth in Appendix 5. • FM or TS profile bytes were not 01 in ACTLU or ACTPU.
0829	Change direction required. A read type command was received without CD bit or with an EB bit.
082B	Presentation space integrity lost. An operator has switched to SSCP-SLU session and back to PLU-SLU session thereby cleaning the screen.
082D	SLU busy. • The display is owned by SSCP-SLU session. • The display is performing a local copy operation.
082E	Intervention required at subsidiary device. An error (same as for 0802 and also power off or not connected to Communication Processor 4101) has occured at a printer involved in host-initiated copy operation.
082F	Request not executable at subsidiary device. A non-recoverable error has occured at a printer involved in host-initiated copy operation.
0831	LU component disconnected. The Communication Processor 4101 cannot contact the device because of power off, connection broken etc.

Sense bytes	Explanation
0843	Required function management synchronization not supplied. A chain sent to LU type 2 och 3 and having the start printer bit set in the WCC must be a definite response chain or an exception response chain carrying CD bit.
0845	Permission rejected. Power is off in display or printer. The SSCP will be informed when power is turned on.

Lustat

A type 2 LU (display unit) causes LUSTAT to be sent

- If a Bind command has been rejected because no response was obtained when the display unit was polled by the communication processor (sense 0845), LUSTAT 082B0000 is sent when the connection between the communication processor and the display unit is established.

- If the connection between the communication processor and display unit is broken, LUSTAT 08310000 is sent for the associated LU. When the connection is re-established, LUSTAT 082B0000 is sent.

- If an FM data request has been rejected in a LU-LU session because a device was busy (sense 082D; device engaged in local copy or owned by an SSCP-LU session), LUSTAT 082B0000 is sent when the device becomes not busy.

- If the operator has changed device ownership to an LU-LU session (and thus cleared the screen), LUSTAT 082B0000 is sent.