XEROX 4045 LASER CP
MODEL 50
USER MANUAL

XEROX

Publication #601E81150-A
Version 3.0
January 1987

Xerox Corporation
701 South Aviation Boulevard
El Segundo, CA 90245

© 1987 by Xerox Corporation. All rights reserved.

Publication #601E81150-A

January 1987

Printed in the United States of America

Xerox®, 4045, and all other Xerox products mentioned in this publication are trademarks of Xerox Corporation.

Centronics and all Centronics products mentioned in this publication are trademarks of Centronics Data Computer Corporation.

Dataproducts and all Dataproducts products mentioned in this publication are trademarks of Dataproducts Corporation.

IBM and all IBM products mentioned in this publication are trademarks of International Business Machines.

Copyright protection claimed includes all forms and matters of copyrightable material and information now allowed by statutory or judicial law or hereinafter granted, including without limitation, material generated from the software programs which are displayed on the screen such as icons, screen displays, etc.

Changes are periodically made to this document. Changes, technical inaccuracies, and typographic errors will be corrected in subsequent editions.

WARNING: This equipment generates, uses and can radiate radio frequency energy and, if not installed in accordance with this manual, may cause interference to radio communications.

This equipment has been tested and found to comply with the limits for a Class B computing device pursuant to Subpart J of Part 15 of the FCC Rules, which are designed to provide reasonable protection against such interference in a residential installation. This equipment generates and uses radio frequency energy, and if not installed and used properly in strict accordance with the manufacturer's instructions, may cause interference to radio and television reception. There is, however no guarantee that interference will not occur in a particular installation.

If equipment certified to meet the Class B limits does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures: (a) Reorient the receiving antenna. (b) Relocate the computer with respect to the receiver. (c) Move the computer away from the receiver. (d) Plug the computer into a different outlet so that the computer and the receiver are on different branch circuits.

If necessary, the user should consult the dealer or an experienced radio/television technician for additional suggestions. In addition, an FCC booklet, *How to Identify and Resolve Radio-TV Interference Problems*, Stock No. 004-000-00345-4, is available from the U.S. Government Printing Office, Washington, D.C., 20402.

In equipment labeled Class B compliant, a shielded and grounded I/O cable is necessary to achieve compliance with the FCC Rules regarding radio emissions from computers. Please consult your dealer or authorized sales representative for further details regarding such a cable.

Laser safety

The Xerox Model 50 complies with appropriate safety standards.

With specific regard to laser, the equipment complies with laser product performance standards set by governmental agencies as a Class I laser product. It does not emit hazardous light; the beam is totally enclosed during all modes of customer operation and maintenance.

When performing various operator functions laser warning labels may be visible. These labels are for the service mode and are placed on or near panels or shields which require a tool for removal. **THESE PANELS ARE NOT TO BE REMOVED.**

CAUTION: Use of controls or adjustments or performance of procedures other than those specified herein may result in hazardous light exposure.

Operation—safety

Your Xerox equipment and supplies have been designed and tested to meet strict safety requirements. These include safety agency examination and approval, and compliance to established environmental standards. Attention to the following notes will ensure the continued safe operation of your equipment.

Always connect equipment to a properly grounded power source receptacle. If in doubt, have the receptacle checked by a qualified electrician.

> **WARNING:** Improper connection of the equipment grounding conductor can result in risk of electrical shock.

Always locate equipment on a solid support surface with adequate strength for the weight of the machine.

Always exercise care in moving or relocating the equipment.

Always use materials and supplies specifically designed for your Xerox equipment. Use of unsuitable materials may result in poor performance, and can possibly create a hazardous situation.

Never use a ground adapter plug to connect equipment to a power source receptacle that lacks a ground connection terminal.

Never attempt any maintenance function that is not specifically described in this Reference Manual.

Never remove any covers or guards that are fastened with screws. There are no operator serviceable areas within these covers.

Never override or "cheat" electrical or mechanical interlock devices.

Never use supplies or cleaning materials for other than their intended purposes. Keep all materials out of the reach of children.

Never operate the equipment if unusual noises or odors are noticed. Disconnect the power cord from the power source receptacle and call Xerox service to correct the problem.

If you need any additional safety information concerning the equipment or Xerox supplied materials, you may call the following toll-free number:

1-800-828-6571

Notice

This publication describes all the features supported by release level 3.0 of 4045 Laser CP Model 50 firmware.

Specifications described in this publication are subject to change without notice. The customer's use of certain features which become available may be limited by the customer's hardware/software configuration. Customers should consult their Xerox sales representative for details.

Related publications

Title	Publication #
4045 Laser CP Model 50 User Manual	601E81150-A
4045 Document Creation Reference Card	610P50749
4045 Laser CP Installation Planning Guide	600P87284
274 Interface Controller User Guide	600P86478
275 Interface Controller User Guide	600P86732
276 Interface Controller User Guide	600P86734
280 Interface Sharing Device User Guide	600P87285

Table of contents

1. The Xerox Model 50: how it works

The Xerox 4045 Laser CP Model 50 is designed to fit easily into a variety of office environments. With both parallel and serial interfaces, the Model 50 can be readily integrated into a variety of network configurations. Some typical setups are diagrammed in Figure 1-1.

In addition, the Model 50 offers both Xerox 2700 and Diablo 630 emulation. The Xerox 2700 is a laser-based electronic printing system, whose command set is supported in the Model 50. The Diablo 630 is an impact-type daisywheel printer; the Model 50 supports a subset of this product's ECS/API configuration. The commands supported by each emulation are provided in Tables A-2 and A-3 in appendix A.

This chapter of the *Reference Manual* provides an overview of the Model 50 and describes how it works internally. This chapter is intended to be read in its entirety as an introduction to the system. Succeeding chapters provide reference information on certain aspects of the system, hints for successful Model 50 operation, or general reference information. At the end of this chapter, under *System management tools*, are brief descriptions of the type of information contained in the other chapters.

Note: If you are a new computer user, or are unfamiliar with the nuts and bolts of data encoding, data transmission, or electronic printing, you may wish to begin your reading of this manual with chapter 6, *Technical tutorial*.

2700 and **630** are registered trademarks of the Xerox Corporation.

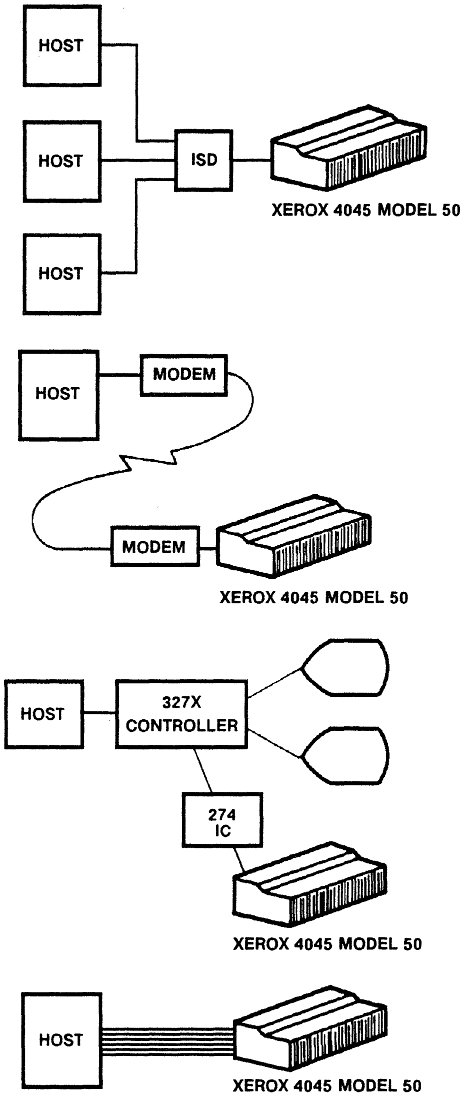


Figure 1-1 **Some possible network configurations**

Printer emulations

Many types of computers, from mainframes to micros, incorporate software that formats and delivers data to printers like the Xerox 2700 and the Diablo 630. Your Model 50 can be configured to plug into any system that can accommodate one of these printers and begin printing. In addition to being compatible with existing systems, the Model 50 offers these capabilities:

- Graphics printing
- Multiple font storage
- Automatic rule drawing
- Multinational character sets and customized code mapping
- Forms merging

All of these features are available whether your Model 50 is set up as a 2700 or a 630. You choose the emulation you wish by a switch position on the configuration cartridge. The configuration cartridge is described in the *User Manual* and in chapter 4 of this manual.

There are, of course, some differences between the way the Model 50 works and the way either the 2700 or the 630 works. Some of the major differences are listed below.

2700 differences

The 2700, like the 4045, is a laser-based electronic printing system. The 2700 is larger and faster than the Model 50 and has larger paper input and output capacities, but the Model 50 is designed to accept 2700 formatting commands and produce a printed page that is similar in appearance to the same page printed on a 2700. The exact set of 2700 commands supported in the Model 50 are described in the *Creating documents* portion of the *User Manual*, and in Table A-2 in appendix A of this manual.

Some of the important differences between the 2700 and the Model 50 are:

- The Model 50 cannot plug directly into a BSC (binary synchronous communications) network.
- If the Model 50 receives an escape code (ISO and ASCII X'1B', EBCDIC X'27') within font data, the entire font load is aborted. Details about codes and font data are provided in chapter 2 of this manual.
- The 2700 scans the long edge of the page, while the Model 50 scans the short edge, as described later in this chapter.
- The Model 50 can accommodate approximately 15,000 characters per page, as compared to 30,000 for the 2700. (See "Imaging errors" in chapter 5 for a discussion of page complexity problems.)
- The Model 50's printed image is smaller than the 2700's by about 1%.
- The 2700 font cartridges are not interchangable with Model 50.

630 differences

The Diablo 630 is an impact-type, daisywheel printer. The Model 50 is designed to accept 630-type commands in the data stream, and produce a printed page whose appearance is similar to the same page produced on the 630. While there are several types of 630 available, the Model 50's emulation is modeled after the 630 ECS/API configuration. The exact set of commands supported are provided in Table A-3 in appendix A of this manual. Some of the important differences between the Model 50 and the 630 are:

- Downloaded translation tables may apply only to character code assignments within available fonts. The process is analogous to printwheel spoke addressing, with significant differences; for example, print hammer energy is not applicable to the Model 50. Downloaded translation tables are described in chapter 2 of this manual.
- HyPlot graphics are not supported.
- Bolding commands emulate shadow-print, not double print.
- Some remote diagnostics are supported, with differences. The (ESC)(SUB)1 command is supported. The exact functions of these commands are described in appendix A.
- The Model 50 does not have an IEEE 488 interface.
- The Model 50 does not offer API cable support.

Other differences

Other differences between the Model 50 and the 2700 or 630 relate to the way in which the Model 50 positions printed data on the page. Positioning of print data is described later in this chapter.

Data encoding

The Model 50 supports the following data encoding schemes:

ISO 6937—This is a code set derived by the International Organization for Standardization, and includes multinational font characters.

ASCII—This encoding supports the US ASCII code set, and allows automatic code mapping to multinational font characters within the ISO set.

EBCDIC—This is the data processing (DP) code set used by many EBCDIC hosts. **The WP (word processing) set is not supported**. The Data Stream Compatibility (DSC) and System Network Architecture (SNA) environments are supported. SNA support includes automatic code mapping to multinational font characters within the ISO set.

IBM PC—This encoding supports the code mapping and character set used by the IBM PC.

Details about each coding scheme are provided in chapter 2.

Interfaces

Your Model 50 comes configured with a Centronics parallel and serial interface or a Dataproducts connectors and supports the pin assignments and signals used by their standard interfaces. The serial interface supports RS-232-C connection and asynchronous protocols. The details about all these are provided in chapter 3.

The parallel interface offers Vertical Format Unit (VFU) emulation. This function, normally used with the Dataproducts configuration, is available with both parallel setups. It allows the Model 50 to accept specially-encoded command bytes that control vertical print positioning on the page. This feature is described in detail in chapter 3.

The serial interface offers several means to ensure data integrity.

- Mark, space, even, and no (ignore) parity are supported.
- Three common data flow controls (protocols): XON-XOFF (DC1-DC3), ETX-ACK, and Printer Ready, may be selected by setting switches on the configuration cartridge. Any or all of the data flow controls can be used. These are described in chapter 3.

The printer: an inside view

This subsection of the manual describes the default state of the printer and then divides the Model 50 into two major components: the processing module and the printing module.

System initial state—defaults

When the Model 50 is first powered up, a set of *default system values* is assumed. The default values are primarily governed by the switch settings on the configuration cartridge. The defaults condition the system to expect data to be formatted in certain ways, as dictated by the switch settings. Switch settings are described in chapter 4.

The defaults also set up a basic format for the printed page, depending on the default orientation chosen by switch setting. The print format defaults are provided in Table A-1 in appendix A.

Many of the defaults can be altered dynamically by inserting commands in the data stream. As a rule, the values specified by the commands take effect immediately and remain in effect either until new commands replace the previous values or until the system is reset.

Setting or resetting defaults

The system may be reset in one of three ways: by command, by turning the printer off and then on again, or by the pressing of switches on the control panel.

Several reset commands are available; their use is governed by the printer emulation chosen. They are described in Tables A-2 and A-3 in appendix A. Typically, reset commands reset those values changed by commands, not those set by the switches.

If you change the switch settings on the configuration cartridge, you must force the Model 50 to reread the switch settings by either cycling power to the printer, or performing a *soft reset.* A soft reset is performed by pressing the Reset switch on the control panel and, while holding this switch down, pressing the Off-line switch. The effect of each action is as follows:

Cycling power to the printer resets the system to switch-selected defaults, *while erasing all data in memory, including downloaded fonts.* Cartridge fonts are not affected.

A soft reset takes the printer off-line for a few seconds, rereads the switch settings, and comes back on-line, configured with a new set of defaults. During the process, downloaded fonts are retained in memory, *but all other data, including unprinted pages, are erased.* Cartridge fonts are not affected.

The processing module

The processing module performs several tasks, diagrammed in Figure 1-2, on the following page. It controls the following:

- Reading the switches on the configuration cartridge and recording their settings.
- Monitoring the receipt of data through the installed interface and maintaining data integrity.
- Determining the purpose of incoming data (text, fonts or graphics) and managing its use in memory.
- Reading font cartridges.
- Arranging the position of all data to be printed on the page. This is called page composition.
- Generating a "video stream" for the composed page. A video stream is simply a series of 1-bits and 0-bits that represent the printed and unprinted dots on the page.

Figure 1-2 illustrates the following discussion.

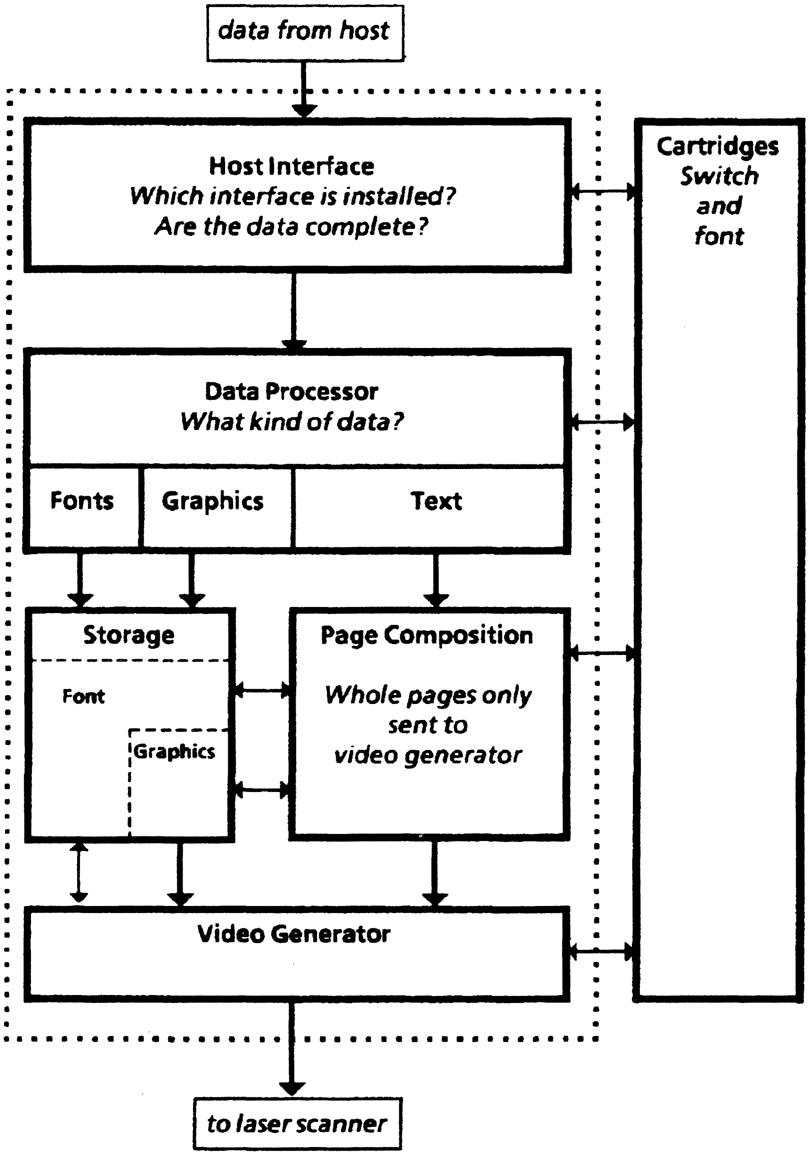


Figure 1-2 **Tasks of the processing module**

Host interface

The host interface function monitors the signals coming from the host through the installed interface. The switches on the configuration cartridge are read at power-up (or whenever the system is reset) to determine the encoding scheme used, whether the printer is to expect 7- or 8-bit data, and, for the serial interface, whether parity or data flow controls are used. Received bytes are then passed on to the data manager.

Data manager

The data manager first determines what kind of data is coming in: text, fonts, or graphics. Details about each type are provided in chapter 2 of this manual.

Textual data is the normally expected type, so incoming data are assumed to be textual data, **unless preceded by a command specifying the data type**. The *Creating documents* portion of the *User Manual* describes these commands. Lists of commands for both 2700 and 630 emulations are provided in Tables A-2 and A-3 in appendix A.

Font data is stored in the font storage area of memory, where it is available for use during the page composition and video generating tasks. Graphic data is stored there as well, until it is printed.

Textual data is passed to the page composition task.

Page composition

Several processes occur during page composition:

1. Each byte of text data is either interpreted as part of a command or mapped to the font character that it represents. Each font character has a code value assigned to it; a received byte of that value causes that character to be printed from the font. Several types of mapping may also occur; that is, a particular value may be translated to different characters, depending on the mapping function invoked. For example, the

language switches on the configuration cartridge allow particular byte values to invoke several different sets of characters. These switches are read either at power-up or when the system is reset, to determine which character set is being used.

2. The position of each printed item (character, graphic, or rule) is calculated. Items which appear to fall in illogical places, such as outside the margins or beyond the page's edge, may or may not be printed, depending on the nature of the problem and the configuration installed. For example, with the 630 emulation, it is possible to set a switch on the configuration cartridge which, when set, causes words extending beyond the margin to be placed on the next line. However, in the 4045 (2700) mode, words extending beyond the margin are printed only if justification is used. Print positioning is described in other parts of this manual.
3. When an entire page is ready to be printed, the page is transferred to the video generator for bit mapping.

Video generation: bit maps and band buffers

The video generator looks at the font data stored both in font memory and in font cartridges. Font data consists essentially of collections of character bit maps. The bit map describes the arrangement of printed and unprinted dots that make up the printed image. A typical bit map is illustrated in the following figure.

The bit maps are designed around a printing resolution of approximately 300 dots to the inch on the printed page.

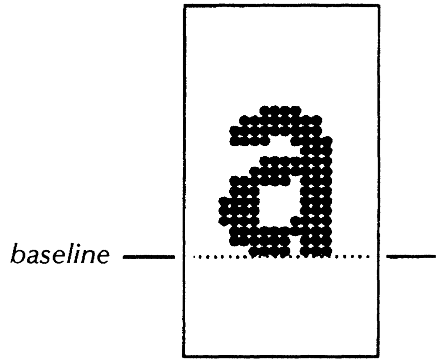


Figure 1-3 **Font character bit map**

Note: The actual number of dots printed per inch varies from machine to machine, and may range from approximately 298.4 to approximately 308.6 per inch. Furthermore, the value along the short edge of the page may differ slightly from the value along the long edge. The values for a particular machine, however, remain constant and do not fluctuate.

The video generator retrieves the bit maps for each item to be printed, along with the positioning information calculated during page composition. Streams of ones and zeros (called "video data"), representing printed and unprinted dots, are then generated and readied for sending to the laser scanner. The ones and zeros are generated in groups called "bands." Bands are illustrated in Figure 1-4.

The laser scans parallel to the short edge of the paper, whether the page is a landscape or portrait page. As it scans, it lays down the patterns of ones and zeros. Each pass of the laser across the width of the short edge constitutes a **scan line**. Each scan line may include up to 2,550 dots, for a maximum image of 8.5 inches (approximately 216 mm). Thirty-two consecutive scan lines constitute a band. Thus, a band-full of data may describe an image extending up to 8.5 inches (216 mm) along the short edge of a page, and 32/300ths inch (0.1067 inch, 2.71 mm) along the long edge.

Page images are created a band-full at a time. As one band of information is sent to the laser scanner, another is prepared for sending. Each band of data is held in a hardware device called a band buffer before it is sent to the scanner.

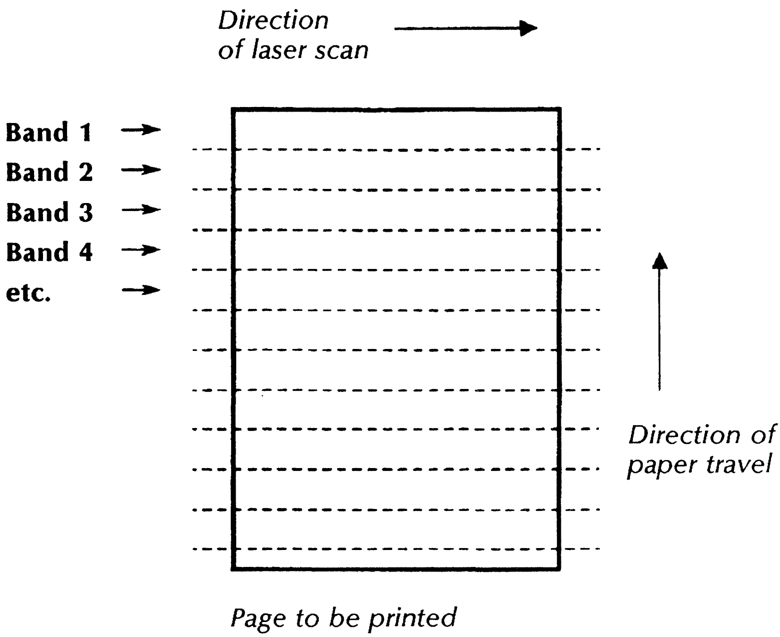


Figure 1-4 **Bands**

The printing module

The printing module performs the following:

- Feeding paper from the paper tray and monitoring its progress through the paper path to the output tray.
- Coordinating the scanning of the imaging laser, the turning of the xerographic drum, and the retrieval of video data from memory.
- Monitoring the density of the printed image.
- Illuminating the indicators on the control panel and monitoring the push buttons for actions from the user.

- Ensuring that the user's actions do not disturb the processing of data in memory. The coordination of the push buttons with data streams from the host is discussed later.
- Controlling all the mechanical functions of the printer.

The printing module also coordinates mechanical functions and user actions with the processing of data by the processing module. For example, the printing module makes sure that a page is completely printed and delivered to the output tray before the processing module can erase that page from its memory space.

Of course, the most important function of the printer module is to print pages. The following steps describe how this is done.

1. First, a band full of video data is transferred from the band buffer to the laser scanner.
2. Each bit in the video data causes the laser to illuminate or not, depending on its value (1 or 0). The beam hits a rotating, multifaceted polygon and bounces off the polygon toward the xerographic drum. The rotation of the polygon moves the beam across the surface of the xerographic drum. This mechanism is diagrammed in Figure 1-5.

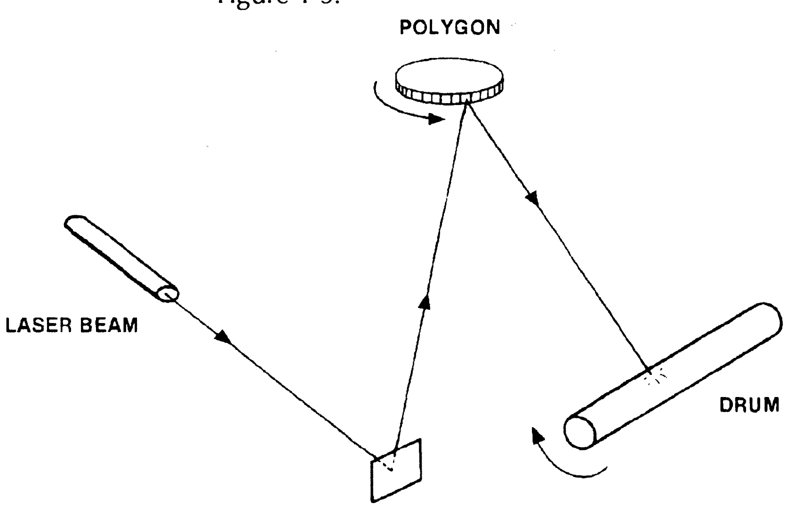


Figure 1-5 **Scanning mechanism**

3. The surface of the drum has electrostatic properties such that, when light hits its surface, the electrical charge on the surface changes. The turning on and off of the laser thus creates an electrostatic pattern of dots on the surface of the drum, corresponding to the pattern of dots that is printed on the page.
4. The drum is then brushed with dry imager, which clings to the pattern of electrostatic charges, making characters, rules, and the other images belonging on the page.
5. As a piece of paper is pulled from the paper tray, it is brought into contact with the surface of the drum. The dry imager is transfered from the drum onto the paper.
6. The paper, carrying the dot-patterns that have been imaged, passes through a fusing mechanism that permanently fixes the imager to the paper.
7. Finally, the finished page is deposited in the output tray. The data for that page is erased from memory.

The second group of tasks that the printer module performs involves the coordination of printing tasks with actions taken at the control panel or copier, if installed, and the illumination of the two-digit codes in the display.

Certain actions taken at the control panel or copier require that the memory module suspend or alter its operations for a time. For example, if the paper tray is empty, the printer module displays a code, may sound the chime (depending on a switch setting), and tells the memory module to hold off sending any more data until the tray is refilled. Similarly, if the Off-line button is pushed, the printing module tells the processing module to stop receiving data until the button is again pushed. Similarly, if a piece of paper is put in the optional copier, the printing module prepares to activate the copier, waiting for the appropriate time to do so.

The effects of specific actions are described under *User controls: how they affect printing,* later in this chapter.

An overview of electronic printing

This manual distinguishes between laser printers like the Model 50 and traditional printers by referring to the latter as "mechanical" printers. Mechanical printers include:

- Daisywheel and thimble printers
- Dot-matrix printers
- Line printers
- Band printers

and any type of printer that depends on the movement of a printhead to produce an image.

The single greatest difference between mechanical and electronic printers is that the electronic printer prints a full page at a time, rather than one character or one line at a time. This section describes some of the methods and advantages of full-page printing.

The data coming in

For textual data, each byte sent from the host represents either a single control function or a single character to be printed. Control functions include either printer controls or communications controls. The precise meanings and formats of each code that the Model 50 recognizes are described in chapter 2 of this manual, but this discussion describes, in general terms, the chief differences between the way the Model 50 looks at data and the way traditional printers do.

Those bytes that represent control functions are called "control codes." A byte representing a printable character, or any value that does not represent a control function, is considered "textual data," for the purposes of this discussion. Thus, although a particular value may not invoke an actual character, as long as it does not represent a control function, it is called textual data in this manual. This means that the escape sequences supported by the Model 50 are also considered textual data. Bytes within graphic data or font data are considered part of a video stream and their particular numeric values are not significant. Figure 1-6 illustrates the differences in the ways basic data types use bit patterns.

Control code:	0001100	=	*Form feed, printer instruction: eject current page*
Textual data:	1000110	=	F (ASCII)
Graphic data: Note: This illustration represents *unencoded* data.	1000110	=	●○○○●●○

Figure 1-6 **Some basic data types**

Control codes

As stated earlier, control codes may represent either communications or printer controls. In the Model 50, communications controls are handled by the host interface task, while printer controls are passed on to the page composition task. Many of these codes are designed to control the operation of mechanical printers; be aware that they may function differently in the Model 50. The precise significance of each code is described in chapter 2.

Textual data

In mechanical printers, those bytes that represent printed characters cause the printer to move its printhead and its associated printwheel or band, or whatever physical device is used to create the set of images, so that the desired character is positioned in front of the paper at the appropriate place on the page. (In the case of dot-matrix printers, there is no separate device carrying the set of characters, but printhead movement is required.) The printhead then uses pressure to impress the image on the paper.

In electronic printers, each byte that represents a character to be printed causes the processing module to find a bit map in whatever font the user has requested and link it to the set of bit maps and other images required for that page. Then, when the entire page of data has been collected and composed in memory, it is turned into video data and sent to the laser scanner.

In the Model 50, a particular byte value may invoke different characters, depending on the language mapping chosen via the switch settings on the configuration cartridge, language commands, and the fonts used. The options are described more fully under *Mapping*, later in this chapter.

Graphic data

The Model 50 treats graphic data as a video stream; therefore, the numeric value of a byte within graphic data has no significance to the Model 50. Graphic data, of course, **must be preceded by a graphic command**. The command includes positioning information that tells the Model 50 where the graphic is to be placed on the page. When the end of the graphic data is reached, the Model 50 reverts to looking at individual bytes as textual data.

Print positioning

Electronic printing allows for dynamic (that is, spur-of-the-moment) positioning of print data. Mechanical printers depend on the movement of the printhead or paper feeding mechanisms to place data on the page. The simplest mechanisms place characters one after the other in a row, then return the carrier to the left margin as the paper is moved up another line. Some allow bidirectional printing, which helps save time; some allow reverse paper feeding, which allows the paper to "back up" to some degree; and some provide for absolute positioning, which allows a particular position on the page to be selected, independent of where the printhead was positioned before. All of them, however, rely on the durability of their movable components for accurate positioning, and all of them are restricted to some degree as to the type of data they can print and their ability to move around the page.

However, with the Model 50, you are free to position print items anywhere on the page, from anywhere on the page.

The Model 50 places characters one after the other in rows, unless it is directed to do otherwise. Each item may be positioned independently of any other on the page. In fact, items may even be printed over each other using text placement commands.

Not all the dynamic positioning options are available with both emulations (2700 and 630). Each has some options that are unique, in emulation of the actual printers. They are described separately, where appropriate, in the following discussion.

Virtual carrier movement

Both emulations make use of the concept of virtual carrier movement. This means that print positions are assigned, in the absence of dynamic positioning commands, as though each character were printed by a mechanical printhead, which moves to the right or left after each character has been printed. The most significant difference in the application of this concept is that, in the Model 50, the virtual carrier movement is not restricted by physical mechanisms. Movement may occur **to** any point on the page, **from** any point on the page, **at any time** (virtual time) before the end of the page. Again, each emulation mode makes use of this concept slightly differently. Differences are described where appropriate.

Default positioning

Normal character placement, in the absence of any type of dynamic positioning, special switch settings, or placement commands, begins at the left and top margins of each page, as illustrated in Figure 1-7A. Margins may be specified, but defaults are always available. Defaults are listed in Table A-1 in appendix A.

Successive characters are then placed adjacent to each other from left to right until a line-end is received. The line-end returns the virtual carrier to the left margin, and one line-height lower on the page.

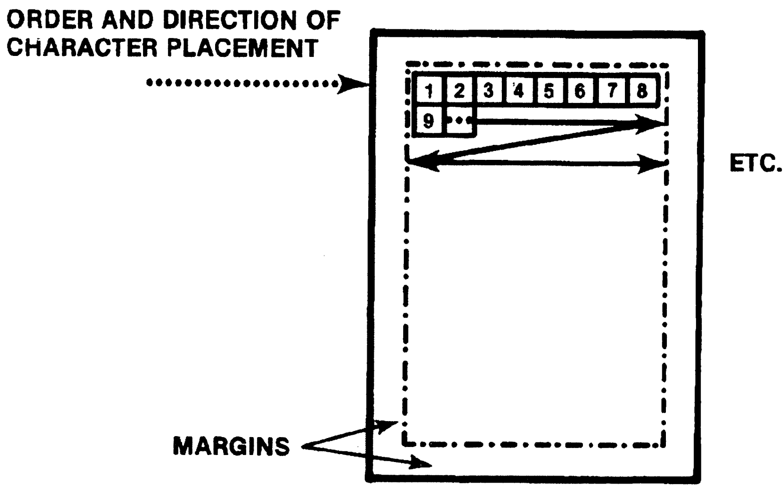


Figure 1-7A **Normal (default) positioning**

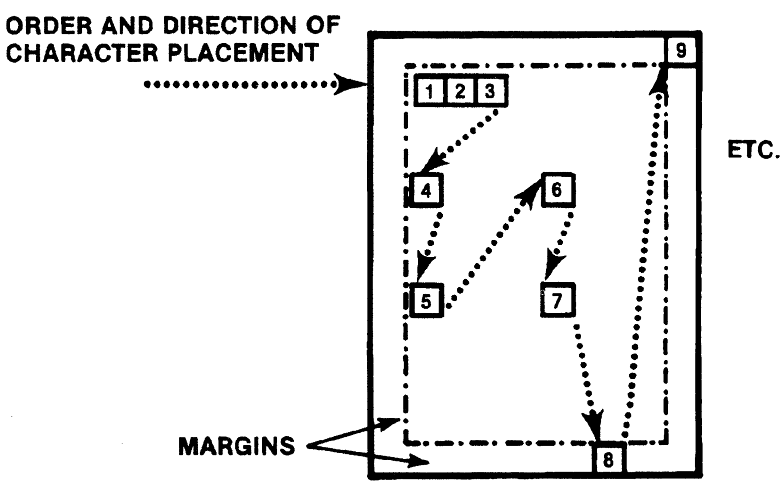


Figure 1-7B **Dynamic positioning**

Lines of data are then stacked from the top margin to the bottom margin, or until a form-feed code is received. As far as the Model 50 is concerned, this constitutes a full page, and the video data for the page is generated and sent to the imaging scanner.

Four variable factors play especially important roles in the placement of print items on the page: the character width, the line-end, the line height, and action at the right margin. Each is discussed separately below.

Character width

Electronic spacing—With electronic spacing, the width of each character is defined within the digitized font. Electronic spacing is the only character spacing option available with the 2700 printer emulation.

Fonts for the Model 50 may be divided into three categories, according to the method of character spacing:

1. Fixed-pitch fonts, in which every character has the same width. Fixed-pitch fonts typically print 10, 12, or 15 characters to the horizontal inch.
2. Proportional spacing does not follow printwheel spacing table for metal/plastic wheels but uses font information.

With PS fonts, the number of characters that will fit in a given line-length varies among the fonts. The best way to estimate the number of characters in a line with these fonts is to divide the margin width (or column width or line length), expressed in dots (1/300th inch), by the average character width of the font.

630 emulation—Besides electronic spacing, the 630 emulation allows switches on the configuration cartridge to set character pitch at 10, 12, or 15 characters per inch, or to use proportional spacing (PS). The spacing used for PS spacing is the same as for 630 PS printwheels. With switch-selected character spacing, default spacing remains the same for all fonts.

The switch settings, in effect, determine the default value for HMI, the "horizontal motion index." In the 630, this value serves as a basis for the calculation of all horizontal printhead motion. HMI may be adjusted by command, to override the switch selections.

Line-end

The normal expected line-end consists of two control codes in succession: a carriage return (CR) followed by a line feed (LF). The carriage return by itself tells the Model 50 to move character placement to the left margin, while the line feed alone moves the virtual carrier one line-height down the page. Together, they move printing to the beginning of the next line. Switch settings are available to activate other line-ending options, as described in the following paragraphs.

Auto carriage return—This option allows a line feed alone to be seen as a line-end. Thus, the LF code alone returns printing to the beginning of the next line and serves as a command terminator. However, when this option is chosen, the line feed always invokes movement to the left margin.

Auto line feed—This option allows a carriage return alone to be seen as a line-end. Thus, the CR code alone returns printing to the beginning of the next line and serves as a command terminator. However, when this option is chosen, the carriage return always invokes movement down the page, as well as to the left margin.

Auto line-end (630 emulation only)—When this option is chosen with the 630 emulation, the Model 50 generates its own line-end when printing reaches the right margin. It does **not**, however, eliminate existing line endings. Thus, if a particular line is, say, one word too long, the Model 50 will place that word on the next line by itself, beginning a new line with the following line ending.

Line-end as command terminator—Besides defining the end of a print line, the line-ending codes also serve as terminators for **some** 2700 and 630 printer commands. Line endings should not be used to terminate commands that do not require them; this causes the printer to advance the print position lower on the page. Note also that, where a line ending is required for a command, the line-feed code alone may be used as a command terminator, regardless of the line-ending option selected.

Line height

The line height determines how far down the page the Model 50 moves the print position with each line-end or line feed. The line height may be influenced by several factors and takes effect differently, depending on the emulation mode you are using.

2700 emulation—With the 2700 emulation, the line height is determined, in the absence of any positioning commands, by the size of the font being used. Note that, while many fonts print six lines to the inch, the line height of a typographic font depends on its point size.

When a line-end is received, the Model 50 moves the print position one line-height down the page, and to the left margin. If fonts of different line heights are mixed on one line, the height of the largest font in the line becomes the line height.

630 emulation—With the 630 emulation, the line height is determined by a value called VMI, for "vertical motion index." The VMI may be changed by command, but the standard VMI in the default mode is six lines per inch.

Right margin

The rule for default character placement is: You have to make sure your lines don't extend past the right margin. Normally, print data extending past the right margin is simply not printed, unless auto line-end is enabled in the 630 mode. The use of justification, however, affects the right margin differently. Justification, and how it affects the placement of text, is described in the next subsection, *Dynamic positioning*.

Backspacing

Backspacing, moving the print position to the left by means of the backspace code, may be used in 630 mode. In 2700 mode, the backspace code takes effect only at the left margin, if it is the first code in the line. Multiple backspaces may be used up to the left edge of the page.

Dynamic positioning

Print position can be influenced or controlled in a variety of ways that give you considerable flexibility in designing the look of the printed page. Some work the same in both emulations, some work differently in one emulation than the other, and there are others that only apply to one emulation. Differences are noted where appropriate.

Page origin

Both emulations view the page as a two-dimensional plane described by a Cartesian coordinate grid. This is illustrated in Figure 1-8. The grid's x- and y-axes correspond to the edges of the page. Each axis is measured in units of 1/300th inch (approximately 0.085 millimeter). The point where the axes meet is called the *origin*; this point always has a value of zero and corresponds to a corner of the page. The origin is located at different corners, depending on the emulation used, as illustrated in the figure. Print positions within the page are identified by their distance from the origin along each axis. Several commands require that the print position be specified by naming both **x** and **y** values.

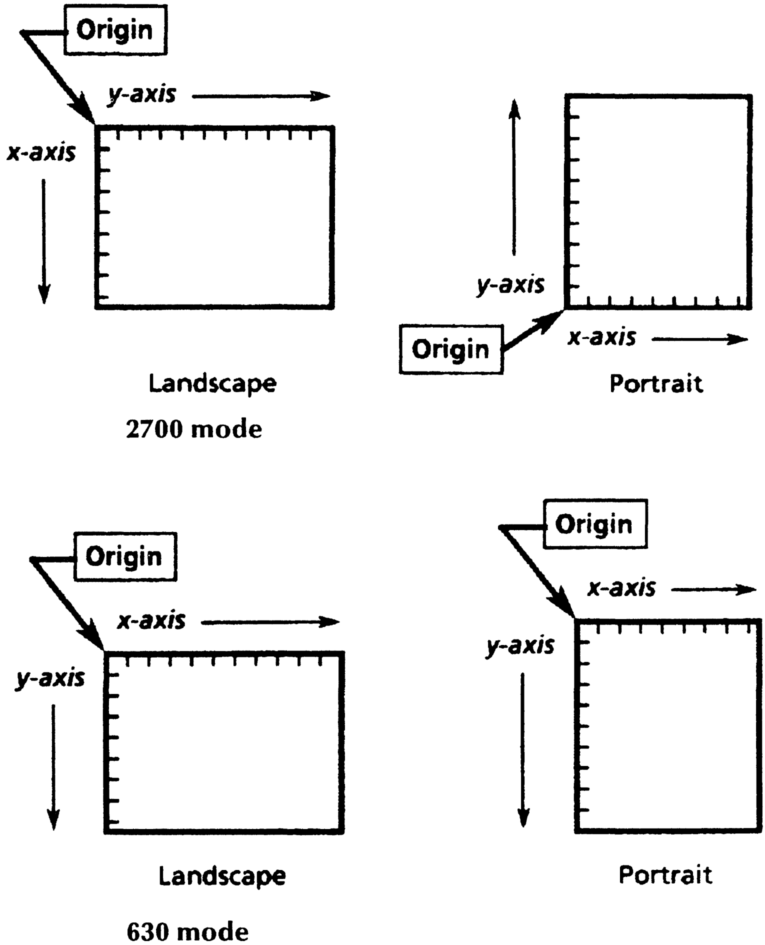


Figure 1-8 **Page origin**

Dynamic text placement

Commands are available for both emulations that allow you to assign the print position at a specific point on the page. This is called absolute text placement. Absolute text placement commands may appear anywhere within the data for the page and they move printing to any other point on the page, independent of the virtual print position. This concept is illustrated in Figure 1-7B. The position is specified by calling out its distance from the origin along each axis.

In the 2700 mode only, the print position may also be moved any distance in any direction **relative to the current virtual print position**. This is called relative text placement. For this type of text placement, distance from the origin is irrelevant.

Centering

Text can be automatically centered between the margins by including a centering command within the line. The command is different for each emulation mode, but works the same in each case, so that the center of the printed line is located exactly halfway between the left and right margins. To center a series of lines, the command must be included in each line.

Justification

The look of a printed page may be enhanced, if desired, by justifying the text. This means that the printed characters are arranged so that they exactly fill the space between the left and right margins. The use of justification affects what happens when text reaches the right margin. Since each emulation justifies text differently, each treats the right margin differently, as described in the following paragraphs.

The **2700 mode** justifies by expanding and contracting the width of the spaces between words. Spaces are contracted as much as 60 percent of their normal width and expanded as much as 300 percent. If a line cannot be justified within these limits, it is printed unjustified.

In the 2700 mode, a line extending past the right margin normally does not print past the margin; however, if the text is justified, characters extending past the right margin are printed, up to the edge of the page.

The **630 mode** justifies somewhat differently. First of all, spaces may be contracted without limit, so that every character in a line, no matter how many, will be printed within the margins, even if it is necessary to print them over each other. However, if the line has to be stretched out in order to fit between the margins, this is done in two steps:

- First, the spaces between words are expanded as much as 150 percent their normal width.
- If there is still extra space left over in the line, it is distributed between the individual characters in the line. For example, if all the spaces in a line are expanded 150 percent and there is still space left over, each character in the line is moved to the right one dot, starting at the left margin. If any are still left over at the right margin, another dot is added between each character, from left to right, until all extra space is distributed within the line. However, the space between characters is never expanded more than 7/120ths of an inch (.0583", 1.48 mm, or approximately 18 dots). Word- and character-spaces are never expanded further. If further expansion is necessary to justify the line, it is printed unjustified.

With the 630 emulation, the action taken at the right margin depends on whether auto line-endings are in effect (by switch setting) **and** whether justification is used.

If the auto line-ending option is in effect, words extending past the right margin are placed on the next line, whether or not justification is used.

If the auto line-ending option is not in effect:

- and justification **is** used, the spaces between characters and words on the line are compressed as much as necessary to fit all characters on that line;
- and justification is **not** used, characters extending past the right margin are printed to the edge of the page.

Electronic spacing in 630 mode

If the Model 50 is operating in 630 mode, a switch on the configuration cartridge may be set to activate electronic spacing. When this option is chosen, the Model 50 calculates horizontal positioning according to the widths of the fonts used.

When electronic spacing is used, the character spacing switches for 630 mode have no effect. However, electronic spacing may be overridden by the use of certain 630 commands, and reinstated by command as well. These commands are identified in Table A-3 in appendix A.

Merging

The Model 50 supports merging of print data in two ways:

– A single page of data may be merged with other pages on command. The page to be merged is called a constant page and the pages that are merged with it are called variable pages. The rules for merging are as follows:

  1. The constant page must be identified to the printer as such. This is done by preceding the data for the page with a special command.
  2. The constant page remains stored in page composition memory until it is erased or replaced by another constant page, or until the system is powered down.
  3. Merging of the constant page may be turned on and off by embedding simple commands in the variable pages to be merged.
  4. The constant page and the variable page may be of different orientations. In fact, this is the only way to print different orientations on a single page.

– Graphic data may be merged with textual data by preceding the graphic data with a special command. The graphic data is stored only until it is printed, unless it is part of a constant page.

The commands used for both types of merging are described in detail in *Creating documents* in the *User Manual*. The format of graphic data is described in chapter 2 of this manual.

Page printing

Mechanical printers, as explained earlier in this chapter, typically print character-by-character or line-by-line. The Model 50, however, prints page-by-page. It collects data, except for font data, and holds it in memory until a page-end is seen. At this point the video data is generated and the laser scanner is activated.

It is important, therefore, that the Model 50 "see" the end of the page. This can be done a number of ways:

- A form feed code, signaling the end of the page, can follow the data.
- The Model 50 prints the page when it has enough print lines to fill the margins.
- In 630 mode, the number of lines per page may be specified by command. When the Model 50 sees that number of line ends, it prints the page.
- A page is composed with one orientation, either landscape or portrait. If a font of opposite orientation is called out before the end of a page, the Model 50 prints the data received to that point, and starts a new page in the new orientation.

  Note: This does not apply to pages that are **merged**.

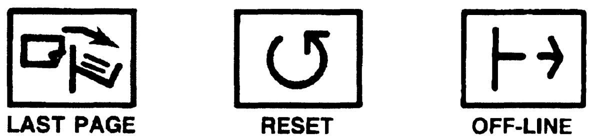
The Model 50 holds all received data in memory until it sees a page-end. If 30 seconds elapses without receiving any data, no page-end is seen, and there is print data in memory, the Model 50 lights the Last Page indicator. Pressing the Last Page switch at this point causes the data in memory to be printed. The use of this switch is described in the next section, *User controls: how they affect printing*.

After a page is composed and the video stream generated, the Model 50 retains all data for the page until it is delivered to the output tray. Thus, if a misfeed should occur, for example, the Model 50 can reprint any pages not yet delivered.

User controls: how they affect printing

This section describes the operation of the Last Page, Reset, and Off-line switches, and the optional copier, and how they affect page composition. The switches are illustrated below.

U.S./Canada version



International version

Figure 1-9 **Control panel switches**

The last page switch

The Last Page indicator is lit whenever there is data in memory that has not been printed because the Model 50 is waiting for an end-of-page condition. (End-of-page conditions are discussed in the previous subsection.) The Last Page indicator is lit when:

- There is data in memory to be printed and no further data is received for 30 seconds, or
- There is data in memory to be printed when the Off-line switch is pressed.

Pressing the Last Page button when the indicator is **not** lit, causes a two-digit code to be displayed, if there is a fault condition in the printer.

The reset switch

The Reset switch serves two main purposes:

– The Reset switch may be held down while the Off-line switch is pressed to perform a "soft reset." A soft reset has the same effect as turning the printer off and on again, with the exception that downloaded fonts are not erased. Among the effects of a soft reset:

  - All data except font data is erased
  - All defaults are reset

– A page in the paper path misfeeds; wait until pages in process are printed before pressing the Reset switch.

– With the serial asynchronous interface, pressing the Reset switch clears a parity error. ("PE" is shown in the display.) Parity errors are discussed further in chapter 3.

The off-line switch

The Off-line switch effectively disconnects the Model 50 from its host; when the Off-line indicator is lit, the Model 50 is unable to receive any data. This switch may be used in three ways:

– **If the Off-line switch alone is pressed**, the Model 50 goes off line (stops accepting data) immediately and lights the off-line indicator, as long as there is no page being printed. If data is being printed, the Model 50 flashes the off-line indicator and waits until all whole pages received up to that point are printed before actually going off line. Then, if there is unprinted data left in memory, the Model 50 lights the Last Page indicator and waits until the Last Page button is pressed to print the data.

– **If the Off-line switch is pressed while the Reset switch is held down**, the Model 50 erases all data from memory, except downloaded fonts, rereads the configuration switches, and restores itself to the start-up (default) state.

— **If the Off-line switch is held while the Model 50 is powering up**, the Model 50 enters the Data Monitor mode. When the Data Monitor mode is entered in this way, the Model 50 returns to normal printing after a soft reset is performed or the power is recycled. The Data Monitor mode is described in chapter 5.

The optional copier

Your Model 50 may include the optional copier. The copier allows you to make xerographic copies of single sheets of paper. Only one copy may be made at a time. Any size paper may be copied, provided that it is not larger than the paper in the input tray.

The chime

The chime sounds to signal that operator attention is required. It can be enabled and disabled by a switch setting on the configuration cartridge. If disabled, the chime never sounds.

When enabled, the chime sounds under the following conditions:

— A BEL code (ISO X'07', EBCDIC X'2F') is received

— The printer is out of paper

— Dry imager is low

— A paper misfeed has occurred

— The cover is open

Fonts and character mapping

Fonts are essentially collections of character bit maps. Typically, all the characters in one font have the same typestyle, the same point size, and share other characteristics. Appendix C provides details about font characteristics.

The character bit map describes the arrangement of printed and unprinted dots that make up the printed image. A typical bit map and several character parts are illustrated in the following figure.

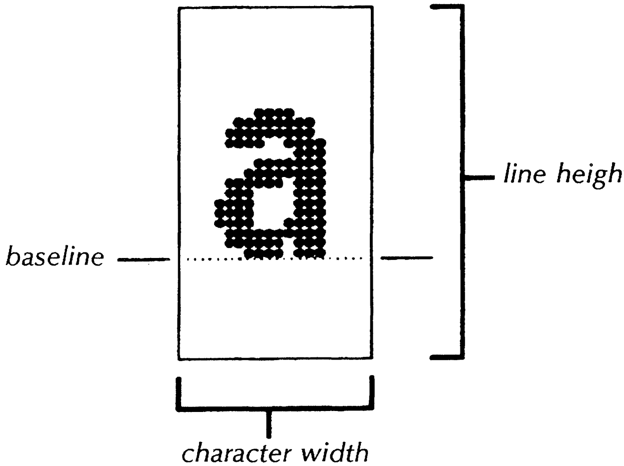


Figure 1-10 **Font character bit map**

Model 50 fonts are *digitized*; that is, each dot is represented either by a 1 or a 0 in the digitized font. The video generator portion of the processing module creates a stream of ones and zeros for the laser scanner by linking individual character bit maps together. This section describes the different types of digitized fonts that are available and how they are used in the Model 50.

Character sets

A great variety of character sets are available for the Model 50. The majority of available fonts support the character set and coding scheme developed by the International Organization for Standardization (ISO), as described in their standard number 6937. IBM PC fonts are also available. Smaller sets, having fewer characters and requiring less storage space, are available as well.

The *User Manual* includes an order card that you may use to obtain more information about the specific character sets available for the Model 50. Use the order card to find out about custom fonts, too. The Xerox Corporate Font Center can digitize any set of characters that you wish, as well as signatures and company logos.

Font storage

For the purposes of this discussion, we shall divide fonts into three types, corresponding to the ways in which they are stored in the machine: resident, cartridge, and downloaded.

Resident

The Model 50 has two resident fonts built into its firmware. One font is landscape and one is portrait. The fonts differ slightly between US/Canada machines and international machines and are described in the following paragraphs.

US/Canada—The US/Canada landscape font is named XCP14iso-L. It prints 13.64 characters to the horizontal inch and 8.57 lines to the vertical inch, such that an 8.5 x 11 inch page can accommodate 66 lines of 132 characters each. The US/Canada portrait font is called Titan10iso-P and prints 10 characters to the horizontal inch and 6 lines to the vertical inch.

International—The international landscape font is named XCP12.5iso-L. It prints 12.5 characters to the horizontal inch and 8.57 lines to the vertical inch, such that an A4 page can accommodate 66 lines of 132 characters each. The international portrait font is called Titan12iso-P and prints 12 characters to the horizontal inch and 6 lines to the vertical inch.

Two fonts are thus always available and one of them may be chosen as the default font; that is, all received textual data is printed in the default font, unless other fonts are called out by command. The default font is selected by a switch setting on the configuration cartridge.

Cartridge

Fonts can be stored in plug-in cartridges that install in cartridge slots, like the configuration cartridge. Each cartridge may hold up to 64 Kbytes of font data, which may be divided among as many as nine fonts, depending on their size. Including the two resident fonts that are always available, this gives you a total of 30 fonts before any fonts have been downloaded from the host.

Downloaded

Downloaded fonts are fonts which have been stored in the host computer and sent to the Model 50. These fonts are then stored in dynamic memory and remain there until they are written over or deleted or until the Model 50 is turned off. Fonts to be downloaded must be preceded by an appropriate command that signals the Model 50 to store the data as font data, rather than print it. (See appendix A for the commands that may be used to identify font data that is being downloaded to the printer.)

The Model 50 has 440 Kbytes of space available for downloading fonts. The total memory currently available for downloading fonts is shown on the status and configuration sheets (described more fully in chapter 4.)

Downloadable fonts are available on magnetic tape and on IBM PC diskettes. Fonts on tape that have been created for the 2700 printer may be downloaded to the Model 50.

Font mapping

Mapping refers to the association of the numeric value of a byte with a particular character in a font. For example, in the ASCII coding scheme, a byte whose numeric value is, say, X'30' normally represents the numeral zero ("0"). Similarly, X'31' represents "1", X'32' represents "2", and so on for the other characters found in common character sets.

The Model 50 supports four different mapping schemes: ISO, EBCDIC, ASCII, and IBM PC, each described later.

7-bit data

7-bit codes can be used to invoke characters in the X'80' to X'FF' range, using the mapping features described under *Multinational language selection and Downloaded translation tables*, later in this chapter.

The 630 mode also allows the use of the Shift In and Shift Out codes with 7-bit data. See Table B-1 in appendix B.

A switch on the configuration cartridge sets the Model 50 for either 7- or 8-bit data.

ISO mapping

The two resident fonts support ISO encoding. This is an international coding scheme derived by the International Organization for Standardization. This mapping is charted in chapter 2, and in Table B-2 in appendix B.

The ISO table serves as a base for the ASCII and EBCDIC coding schemes. When ISO is switch-selected as the default mapping, incoming codes are always mapped directly to ISO font characters. In this mode, the language switches on the configuration cartridge determine the language in which configuration and status sheets are printed.

If ASCII or EBCDIC is switch-selected as the default mapping, the Model 50 looks at the numeric value of each byte and translates that value to its equivalent in the ISO table. ASCII and EBCDIC only support multinational language mapping. This feature is described in greater detail in chapter 2.

The use of IBM PC fonts invokes alternate mapping schemes. The IBM PC mapping and character set are described in chapter 2.

ASCII translation

When ASCII is switch-selected as the default mapping, incoming codes are translated to their equivalent in the ISO table, depending on the switch-selected language or the language commands in the data.

The ASCII multinational mappings are provided in appendix B, Tables B-3 through B-27. Separate mappings are used for the 2700 and 630 modes.

Use of IBM PC fonts invokes alternate mapping schemes. This is described in chapter 2.

EBCDIC translation

When EBCDIC is switch-selected as the default mapping, incoming codes are translated to their equivalent in the ISO table, according to the translation table illustrated in Table B-29, appendix B. This is the basic translation table; further translation may occur, depending on the switch-selected language or the language commands in the data.

The EBCDIC mapping for the DSC environment is provided in appendix B, Table B-30. The EBCDIC multinational mappings for the SNA environment are provided in appendix B, Tables B-31 through B-42.

Use of IBM PC fonts invokes alternate mapping schemes. This is described in chapter 2.

IBM PC mapping

The IBM PC character table is illustrated in chapter 2 and in Table B-44 in appendix B. Control codes are the same as in the ISO table.

The Model 50 supports both the IBM PC character set and IBM PC code mapping. The IBM PC mapping may be selected as the default code mapping by switch setting. IBM PC fonts may be used regardless of the mapping scheme employed (except Danish and Norwegian). However, the use of the IBM PC font and code mapping precludes the use of downloaded translation tables and automatic language mapping. Details are provided in chapter 2.

Floating accents

ISO fonts contain floating accent characters. These are non-spacing characters; that is, they do not cause the virtual print position to move to the next character space. Thus, they can be combined with other, spacing, characters to form a composite, accented character. For example, "~" and "n" can be combined to produce "ñ." Floating accents are described in further detail in chapter 2.

Error character

Every Model 50 font contains an "error" character. This character is printed whenever a code is received which has not been assigned to any printable character. The error character looks like this: ⊞

Multinational language mapping

Many common accented characters and international symbols can be printed using 7-bit codes and any keyboard. Multinational language mapping allows one code to map to different characters depending on the language, or national character set, selected. Various national character sets are supported:

U.S. English	Norwegian
U.K. English	Finnish
French	Swedish
Dutch	German
Spanish	Belgian (EBCDIC only)
Italian	French Canadian
Danish	Portuguese (ASCII 630 only)
	Latin America

A default mapping is selected by switch settings on the configuration cartridge. Any mapping may also be selected dynamically by commands in the data stream. Automatic mapping is only available when ASCII or EBCDIC encoding is chosen by switch setting. ASCII tables include separate sets for 2700 and 630 modes. Tables in appendix B provide the characters associated with each code for each mapping.

Downloaded translation tables

You may download your own translation table specifying a particular mapping that you may wish to use. The specifics of how to create the table and transmit it to the Model 50 are described in chapter 2, but the following paragraphs describe the steps that take place in general terms.

1. You create the table assignment data mapping your codes to a base table in the Model 50.

2. Insert the Table Download command in front of the table data and send the whole package to the Model 50, making sure that the table is ended with the correct terminator (described in chapter 2).
3. The Model 50, upon receipt of the Table Download command, starts mapping the incoming codes to the currently active mapping table. A new table is created and is treated as a new language selection.

To elaborate on step 3:

> Say that German is your switch-selected default language mapping, and that, in the course of a print job, you send a command to select Swedish mapping. Then say that you send down a new translation table that you want to use for some particular application. Since Swedish is in effect when the new table is received, the Model 50 stores the new table *as an alteration of the Swedish table*. The original Swedish table remains intact, however, and you can switch back and forth from your new table to the Swedish table, or any other that you wish.

The downloaded translation table is retained in memory until the Model 50 is turned off, or until a soft reset is performed.

Custom code mappings—cartridge

It may be possible to have your customized assignment table digitized in a font cartridge, along with a customized font, if you wish. A switch on the configuration cartridge must be set to enable this cartridge. This cartridge plugs into a slot in the font compartment and serves as the base table for your Model 50. Customized cartridges may be ordered from the Xerox Corporate Font Center.

Font characteristics

Fonts are categorized and described by many different terms pertinent to their typeface, their size, the thickness of their strokes, etc. appendix C in this manual defines some of the terms by which fonts are described. Included in the *User Manual* is an order card that you can send to the Xerox Corporate Font Center for detailed font information.

System management tools

The remainder of this manual provides more detailed information about the Model 50's functions and features.

Chapter 2: Data formats describes all the coding schemes in detail, including the meanings of those control codes the Model 50 recognizes. It explains how to create and download a customized character translation table. It also describes the creation, formatting, and transmission of font and graphics data.

Chapter 3: Interfaces provides the details you need to hook up and talk to the Model 50. Pin assignments for the signals used with each interface are described. The characteristics of connecting cables or communications equipment are described, as well as the various data flow controls, or protocols.

Chapter 4: Status and configuration explains the switch settings on the configuration cartridge and describes the status and configuration sheets. These sheets provide means for analyzing the more obvious problems that can crop up because of the printer setup or improperly formatted data.

Chapter 5: Problem solving takes a systematic approach to analyzing interface and printing problems. It describes the use of the various analytical tools provided with the printer, including:

- The **status sheet**
- The **configuration sheets**
- The **data monitor**, which allows you to print the hexadecimal value of all the bytes coming through the interface

Chapter 6: Technical tutorial describes, for the new computer user, some basic concepts about data encoding, data transmission, and electronic printing.

The **Appendices** include a series of tables for quick reference, including all the code tables, tables of default format values, switch-setting tables, etc. A special appendix describing font and typographical terms is provided.

A **Glossary** and **Index** are also provided.

2. Data formats

The purpose of this chapter is to acquaint you with the types of data that the Model 50 can handle and how it responds to that data.

The topics covered in this chapter include:

- The coding schemes that the Model 50 supports (ASCII, ISO, EBCDIC,etc.)
- The difference between "control" codes and character codes
- The meanings of all codes and how the Model 50 responds to them
- How font data are encoded
- How to prepare non-textual data, such as graphic data, for transmission to the Model 50

Supported coding schemes

This section describes the meaning of the various control codes supported by the Model 50, provides charts showing all the coding schemes supported by the Model 50, and describes some of the special code-translation, or mapping, features of the Model 50.

Control codes

In the ISO, ASCII, and IBM PC coding schemes, all bytes whose values are less than X'20' represent control functions. In the EBCDIC scheme, all values less than X'40' represent control functions. Control codes designed to control the operation of traditional mechanical printers or other devices may have no meaning to the Model 50, or meanings that differ from their usual purpose. The following table describes the control codes that the Model 50 recognizes; all codes not described in the table are ignored.

Table 2-1A **Meaningful control codes**

Code name	ISO value	EBCDIC value	Description
NUL	00	00	**Null**. Used for filling in time over a communication line or space on storage media.
ETX	03	03	**End of text**. Signals the end of a block of data. ETX prompts a response from the Model 50, if the ETX/ACK protocol is active.
ACK	06	2E	**Positive acknowledgement**. The Model 50 transmits this code to the host upon receipt of an ETX, if the block of data has been successfully received and the ETX/ACK protocol is active.
BEL	07	2F	**Bell**. Causes the chime to sound, if this option has been enabled by switch setting.
BS	08	16	**Backspace**. Causes printer to move the print position one space to the left. In the 2700 mode, BS has this effect only if it is the first code in the print line.
HT	09	05	**Horizontal tab**. Causes the print position to be moved horizontally to a predetermined location on the page.

Table 2-1B **Meaningful control codes (continued)**

Code name	ISO value	EBCDIC value	Description
LF	0A	25	**Line feed**. Causes the print position to be moved down one line-height. This code may be used as a valid line-ending code, performing both the carriage return and line feed functions, and as a command terminator.
VT	0B	0B	**Vertical tab**. Causes the print position to be moved vertically to a predetermined location on the page.
FF	0C	0C	**Form feed**. Causes the printer to eject the current page and begin a new page. In systems where output is printed on continuous-form paper, this code is often inserted between jobs to facilitate the separation of jobs. Redundant (i.e., consecutive) form feeds are ignored by the Model 50.
CR	0D	0D	**Carriage return**. Causes the printer to return to the first print position of the current line, or may be used as a valid line-ending, performing both the carriage return and line feed functions.
SO	0E	0E	**Shift out**. *In the 630 mode*, this code is used with the 7-bit data to print font characters in the X'80' to 'FE' range. *With IBM PC encoding*, this code is used to print font characters in the X'01' to 'IF' range.

Table 2-1C **Meaningful control codes (continued)**

Code name	ISO value	EBCDIC value	Description
SI	0F	0F	**Shift in.** *In the 630 mode,* this code terminates access to the X'80' to 'FE' range. *With IBM PC encoding,* this code terminates access to the X'01' to 'IC' range.
DC1	11	11	One of four device control codes (DC1, DC2, DC3, and DC4). If the XON/XOFF protocol option has been selected with the serial asynchronous interface, the Model 50 transmits a DC1 code to the host to signal its readiness to receive data. If received from the host, this code is ignored. This protocol is described in greater detail in chapter 3.
DC2	12	12	*In the 630 mode only,* this code is recognized as part of certain command sequences.
DC3	13	13	If the XON/XOFF protocol option has been selected with the asynchronous interface, the Model 50 transmits this code to the host when certain fault conditions have occurred, as described in chapter 3.
DC4	14	3C	*In the 630 mode only,* this code is recognized as part of certain command sequences.
NAK	15	3D	**Negative acknowledgement.** The Model 50 transmits this code to the host in response to an ETX, if the ETX/ACK protocol is active, to indicate that the last block of data was not successfully received.

Table 2-1D **Meaningful control codes (continued)**

Code name	ISO value	EBCDIC value	Description
EM	19	19	**End of media**. Originally used to signal the end of a punched card. In the Model 50 *with IBM PC encoding*, this code is used with 7-bit data to print characters in the X'80' to 'FE' range, on a character-by-character basis.
SUB	1A	3F	**Substitute**. *In the 630 mode only*, this code is recognized as part of certain command sequences.
ESC	1B	27	**Escape**. This character gives subsequent codes alternate meanings. Usually initiates word processing command sequences in both 630 and 4045 modes.
GS	1D	1D	**Group separator**. Normally used as a data-block delimiter, in the Model 50 this code is used *with the IBM PC encoding*, to print characters in the X'01' to '1F' range on a character-by-character basis.
RS	1E	35	**Record separator**. Normally used as a data-block delimiter, in the Model 50 this code is recognized as part of certain command sequences.
US	1F	1F	**Unit separator**. *In the 630 mode only*, this code is recognized as part of certain command sequences.

Table 2-1E **Meaningful control codes (continued)**

Code name	ISO value	EBCDIC value	Description
DEL	7F	07	**Delete**. Indicates that subsequent data is to be ignored. In the Model 50, this code prints a font character, if one is assigned to this code in the font used; otherwise, it prints an error character.
Other codes			A number of other control codes are included in the ISO, ASCII, EBCDIC, and IBM PC encoding schemes that are designed to invoke various terminal functions, to control device operation, or to regulate bisynchronous communication. Only the codes described in this table are supported by the Model 50.

The ISO encoding scheme

Table 2-2, pages 2-8 and 2-9, shows the ISO code assignments. The ISO scheme has been derived by the International Organization for Standardization, an international group that develops standards for technological industries throughout the world.

7-bit data The ISO table shows all the code assignments available for 8-bit data. There are three ways that 7-bit hosts may access characters from the right half of the table. Multinational language mapping and custom code translation may be used with both the 2700 and 630 emulations. These are described in detail later in this chapter, under *Alternate mapping features*. In the 630 mode, the shift out and shift in codes may also be used. Their use is described in Table A-3 in appendix A. With IBM PC encoding the end-of-media code may be used. This is described in Tables 2-1 and B-1.

Floating accents Columns 8 and C in the ISO table contain characters called floating accents. These characters are non-spacing. That is, they do not cause the print position to be moved forward; instead, they are printed in the same character-space as the next character that follows.

For example, the code C8 invokes the non-spacing umlaut: ¨. The code 75 invokes the lowercase u. Thus, the code sequence 'C8 75' prints: ü.

The accents in column 8 are printed together with uppercase, or "tall" characters, while the accents in column C are printed with lowercase characters. However, your host need only send a value from 'C0' to 'CF'. The Model 50 automatically prints the correct accent character from column 8 or C, depending on the character with which it is to be printed.

The two points to remember about using floating accents are:

1. The accent code must precede the code for the letter with which it is printed.
2. Only codes from 'C0' to 'CF' should be sent for the accent code. The Model 50 automatically selects the accent from the proper column.

Again, multinational language mapping and custom translation may be used to print accents. See *Alternate mapping features*, later in this chapter.

The ASCII coding scheme

The ASCII coding scheme is a US derivative of the ISO scheme and is most often used with 7-bit data. Fonts are available for the Model 50 that support this smaller character set, thus requiring less storage space in memory. Control codes in the ASCII set are the same as in the ISO set.

ASCII supports multinational code mappings; these are charted in Tables B-3 to B-27 in appendix B.

Table 2-2A **ISO encoding scheme**
ASCII encoding scheme*

Most →

Least ↓	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	NUL	dle	Sp	0	@	P	`	p
1	soh	DC1	!	1	A	Q	a	q
2	stx	DC2	"	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	eot	DC4	$	4	D	T	d	t
5	enq	NAK	%	5	E	U	e	u
6	ACK	syn	&	6	F	V	f	v
7	BEL	etb	'	7	G	W	g	w
8	BS	can	(	8	H	X	h	x
9	HT	EM	)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[	k	{
C	FF	fs	,	<	L	\	l	|
D	CR	GS	–	=	M	]	m	}
E	SO	RS	.	>	N	^	n	~
F	SI	US	/	?	O	_	o	del

*2-0 through 7-E represent the ASCII encoding scheme.

Notes: 1. Code names in lowercase indicate codes that have no significance to the Model 50.
2. Underscored code names indicate codes that have significance **in some modes only**.
3. Font characters marked with an asterisk are present only in the resident fonts, unless specially ordered.

Table 2-2B **ISO encoding scheme (continued)**

Most →

Least ↓	8	9	A	B	C	D	E	F
0				°		―	Ω	ĸ
1	`		¡	±	`	¹	Æ	æ
2	´		¢	²	´	®	Đ	đ
3	^		£	³	^	©	ª	ð
4	~		$	×	~	™	Ħ	ħ
5	¯		¥	µ	¯	♪		ı
6	˘		#	¶	˘		Ĳ	ĳ
7	˙		§	·	˙		Ŀ	ŀ
8	¨	¬*	¤	÷	¨		Ł	ł
9	¨	¦*	‘	’	¨		Ø	ø
A	˚		“	”	˚		Œ	œ
B	¸		«	»	¸		º	ß
C	_		←	¼	_	⅛	Þ	þ
D	˝		↑	½	˝	⅜	Ŧ	ŧ
E	˛		→	¾	˛	⅝	Ŋ	ŋ
F	ˇ		↓	¿	ˇ	⅞	ŉ	

Equivalence table

Hex	Binary	Hex	Binary	Hex	Binary	Hex	Binary
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

The EBCDIC coding scheme

EBCDIC is an 8-bit coding scheme used in many IBM systems, although it is not used in the IBM PC. The basic EBCDIC coding scheme is charted in the next table. The Model 50 uses this table to map non-ISO fonts, and for the U.S. English code mapping in the SNA environment. All multinational mappings for the SNA environment are charted in Tables B-31 through B-42. The EBCDIC mapping for the DSC environment is charted in Table B-30.

The EBCDIC scheme includes many control codes not supported by the Model 50. These include word processing codes and device-control codes designed for IBM equipment and environments. These are represented in the table by null codes (00).

The Model 50 handles EBCDIC codes by translating them to an ISO equivalent. The EBCDIC-to-ISO translation table is provided in Table B-28 in appendix B. An ISO-to-EBCDIC equivalence table is also provided.

The IBM PC coding scheme

The IBM PC coding scheme and its character set are charted following the EBCDIC table.

The PC encoding switches, A:6 and A:7, must be set to ON to use the PC character set. If non-PC fonts are used while the PC encoding is selected, the fonts should be ISO fonts. If you are printing most of your documents with PC font, it is recommended you make it the default font by plugging the cartridge in the first cartridge slot.

IBM PC fonts may be used with other coding schemes. Similarly, other fonts may be used when IBM PC mapping is the default mapping. However, the use of some Model 50 features changes under these circumstances. For example:

- **When an IBM PC font is in use**, the following may not be used:
  - downloaded translation tables

- automatic language mapping except for the Norwegian and Danish table

- **When IBM PC coding is switch-selected:**
  - and ISO fonts are used, characters are printed from the ISO font that most closely resemble their equivalent in the IBM PC set, if possible.
  - and ISO fonts are not used, no code translation occurs.
  - and IBM PC fonts are used, specific control codes must precede some characters in order to print the characters. Specifically, <EM> precedes IBM PC characters in the X '80' to 'FE' range 'and <GS> or '0E' precedes IBM PC characters in the X '01' to '1F' range. These control codes are fully explained in Table 2-1, meaningful control codes.
  - and the 630 mode is selected, characters in the following hex positions are not printed: '07' through '0D' and '1B' through '1F'.

The table following the code charts summarizes the effect of IBM PC font and code use.

Table 2-3A **4045 Model 50 EBCDIC encoding scheme**

Least ↓ / Most →	0	1	2	3	4	5	6	7
0	NUL	dle	00	00	Sp	&	–	♪
1	soh	DC1	00	00	£	"	/	
2	stx	DC2	fs	syn	$	»	˙	
3	ETX	DC3	00	00	¥		¨	
4	00	00	00	00	#	`	¨	
5	HT	ACK	LF	RS	§	½	–	Ω
6	00	BS	etb	00	¤	¼	¹	Đ
7	del	00	ESC	eot	‡	~	®	ª
8	00	00	can	00	"	¯	©	Ħ
9	00	EM	00	00	«	˘	™	`
A	00	00	00	00	¢	!	¦ *	:
B	VT	00	00	00	.	¤	,	#
C	FF	00	00	DC4	<	*	%	@
D	CR	GS	enq	NAK	(	)	_	'
E	SO	RS	ACK	00	+	;	>	=
F	SI	US	BEL	SUB	|	¬ *	?	"

Notes: 1. Control codes are translated to an ASCII equivalent, where possible. Code names in lowercase indicate codes that have no significance to the Model 50. "00" indicates codes that are translated to nulls.

2. Underscored code names indicate codes that have significance **in some modes only**.

3. Font characters marked with an asterisk are only available in the resident fonts.

Table 2-3B **4045 Model 50 EBCDIC encoding scheme (continued)**

Most →
Least ↓

	8	9	A	B	C	D	E	F
0	µ	æ		°	{	}	\	0
1	a	j	˜	±	A	J	Æ	1
2	b	k	s	²	B	K	S	2
3	c	l	t	³	C	L	T	3
4	d	m	u	×	D	M	U	4
5	e	n	v	^	E	N	V	5
6	f	o	w	¶	F	O	W	6
7	g	p	x	•	G	P	X	7
8	h	q	y	÷	H	Q	Y	8
9	i	r	z	'	I	R	Z	9
A	¡	đ	↑	ø	˚		Œ	œ
B	Ĳ	ð	´	^	¸		º	ß
C	Ŀ	ħ	←	'	_	⅛	Þ	þ
D	Ł	ı	[	]	˝	⅜	Ŧ	ŧ
E	Ø	ĳ	→	¾	˛	⅝	Ŋ	ŋ
F	ĸ	ŀ	↓	¿	ˇ	⅞	ŉ	

Equivalence table

Hex	Binary	Hex	Binary	Hex	Binary	Hex	Binary
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

Table 2-4A **4045 Model 50 IBM PC encoding scheme**

HEXA-DECIMAL VALUE	0	1	2	3	4	5	6	7
0	BLANK (NULL)	►	BLANK (SPACE)	0	@	P	‘	p
1	☺	◄	!	1	A	Q	a	q
2	☻	↕	"	2	B	R	b	r
3	♥	‼	#	3	C	S	c	s
4	♦	¶	$	4	D	T	d	t
5	♣	§	%	5	E	U	e	u
6	♠	▬	&	6	F	V	f	v
7	•	↨	’	7	G	W	g	w
8	◘	↑	(	8	H	X	h	x
9	○	↓	)	9	I	Y	i	y
A	◙	→	*	:	J	Z	j	z
B	♂	←	+	;	K	[	k	{
C	♀	∟	,	<	L	\	l	¦
D	♪	↔	–	=	M	]	m	}
E	♫	▲	.	>	N	^	n	~
F	☼	▼	/	?	O	_	o	⌂

Note: The control codes occurring in columns 0 and 1 are the same as in the ISO table; the characters shown here in those two columns are alternate font characters.

Table 2-4B **4045 Model 50 IBM PC encoding scheme (continued)**

HEXA-DECIMAL VALUE	8	9	A	B	C	D	E	F
0	Ç	É	á	░	└	╨	∝	≡
1	ü	æ	í	▒	┴	╤	β	±
2	é	Æ	ó	▓	┬	╥	Γ	≥
3	â	ô	ú	│	├	╙	π	≤
4	ä	ö	ñ	┤	─	╘	Σ	⌠
5	à	ò	Ñ	╡	┼	╒	σ	⌡
6	å	û	ª	╢	╞	╓	µ	÷
7	ç	ù	º	╖	╟	╫	τ	≈
8	ê	ÿ	¿	╕	╚	╪	Φ	°
9	ë	Ö	⌐	╣	╔	┘	θ	•
A	è	Ü	¬	║	╩	┌	Ω	·
B	ï	¢	½	╗	╦	█	δ	√
C	î	£	¼	╝	╠	▄	∞	ⁿ
D	ì	¥	¡	╜	═	▌	φ	²
E	Ä	₧	«	╛	╬	▐	∈	■
F	Å	ƒ	»	┐	╧	▀	∩	BLANK 'FF'

Equivalence table

Hex	Binary	Hex	Binary	Hex	Binary	Hex	Binary
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

Table 2-5 **Effect of IBM PC font and mapping**

When the font used is an IBM PC font	
Switch selection	**Effect**
IBM PC	No translation occurs in the 2700 mode; all incoming codes are mapped to their IBM PC characters. In the 630 mode, all codes except X'07'-'0D' and '1B'-'1F' are mapped to IBM PC characters. X'07'-'0D', '1B' and '1D'-'1F' are not printed. A left arrow is printed at X'1C'. (See *Model 630/630 ECS Printers/Terminals API Interface*.)
ISO	Switch selection is ignored and IBM PC mapping is assumed; all incoming codes are mapped to their IBM PC characters, *for that font.*
ASCII	Switch selection is ignored and IBM PC mapping is assumed; all incoming codes are mapped to their IBM PC characters, *for that font.*
EBCDIC	Switch selection is ignored and IBM PC mapping is assumed; all incoming codes are mapped to their IBM PC characters, *for that font.*

When IBM PC encoding is switch-selected:	
Font in use	**Effect**
IBM PC	No translation occurs in the 2700 mode; all incoming codes are mapped to their IBM PC characters. In the 630 mode, all codes except X'07'-'0D' and '1B'-'1F' are mapped to IBM PC characters. X'07'-'0D', '1B', and '1D'-'1F' are not printed. A left arrow is printed at X'1C'. (See *Model 630/630 ECS Printers/Terminals API Interface*.)
ISO	Incoming codes are mapped to their nearest graphic equivalent in the ISO font; codes for which there is no corresponding character print the error character.
Other	No translation occurs; all incoming codes are mapped to whatever character is assigned to that code in the font.

Alternate mapping features

The coding schemes described on the previous pages present the standard code assignments available with each of the four switch settings available on the configuration cartridge. Each of these may be augmented to suit your system's individual needs or capabilities through the two mapping features described in this subsection: *Multinational language mapping* and *Custom code translation.*

Multinational language mapping

The Model 50 incorporates several code mappings that provide character sets for 14 different national coding schemes, including:

U.S. English	Norwegian
U.K. English	Finnish
French	German
Dutch	Swedish
Spanish	Belgian (EBCDIC only)
Italian	French Canadian
Danish	Portuguese (ASCII 630 only)
	Latin America

Among these mappings, one code may invoke several different characters, depending on the language chosen. One language is chosen as a default language by means of switch settings on the configuration cartridge. The switch settings for the configuration cartridge are described in chapter 4.

Alternate mappings may be chosen dynamically by command. Mappings for each language are provided in the appendix for ASCII and EBCDIC SNA environments. The ISO and IBM PC coding schemes do not support language mapping.

The commands to invoke the various mappings are provided in Tables A-2 and A-3 in appendix A, and, for the 4045 (2700) mode, in the *Creating documents* section of the *User Manual.*

Custom translation tables

You can create your own code assignment table and download it to Model 50 memory, where it remains available as long as the Model 50 is powered on, in much the same manner as a downloaded font. The downloaded table may be invoked by command as an alternate language selection.

You can also have your particular code assignment table digitized and stored in a cartridge, called the "custom cartridge," that can be plugged into a slot in the font compartment. This custom cartridge is selected by a switch setting on the configuration cartridge. When invoked in this way, the custom table becomes the **default** code assignment table.

The remainder of this section describes how to create a custom table for downloading.

Note: The IBM PC code mapping cannot serve as the base table for a downloaded table.

Structure

Figure 2-1, Table Data Structure, shows the major components of the downloaded table. The following text describes the components briefly, and subsequent subsections provide more detail.

Table-Load Initiator—This is a sequence of codes that tell the Model 50 that custom table data is to follow.

Table Data Record or Records—This is the actual code assignment data, which may be sent as one long record, or as a series of records. The format of each record is very precise, as described later.

Table-Load Terminator—This is a code or codes that tell the Model 50 that the custom table data has all been sent; the Model 50 returns to printing when the terminator is received.

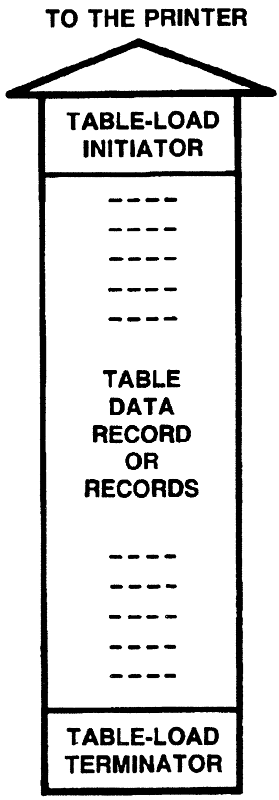


Figure 2-1 **Table data structure**

Table-load initiator The following command signals the Model 50 that table data is to follow:

(Esc) **+T**,comment(line-end)

where

(Esc)	is the escape code (ASCII or ISO X'1B', EBCDIC X'27'), or UDK.
+T	are the plus sign and the uppercase letter "T."

,comment	is an optional comma and comment. If these are included in the command, a status sheet will print at the end of the job, on receipt of the next (Esc)+ command; the comment will be printed on the status sheet.
(line-end)	is a carriage return (X'0D') or line-feed (ASCII or ISO X'0A', EBCDIC X'25'), or a combination of both, depending on the line-ending switch setting on the configuration cartridge.

Unless otherwise specified in the command, the Model 50 assumes that the coding scheme *in effect when the command is received* is to serve as the base table for the new mapping. For example, if the German mapping is in effect when the table-load command is received, the Model 50 treats the German mapping as the basis for the new table. The new table becomes a separate mapping, but all locations not altered by the table-load remain the same as in the German table. Other mappings are not affected. The IBM PC table cannot be used as the base table.

Table data The table data tells the Model 50 how the new table is to translate incoming data. The format of the table data is very specific and must be followed carefully.

The data may be grouped in one long record or in a series of records, but each record must follow the format illustrated in the figure on the next page, Table Data Record. The following paragraphs describe the elements of the data record.

Note: In the following descriptions, hexadecimal values are expressed as two alphanumeric characters. For example, for a field which is to contain a hexadecimal value and whose length is two bytes, the field consists of the two **characters** used to represent that value as typed at the keyboard. Where **hexadecimal values** are expressed, they are enclosed in single quotes: '0B'. Where **alphanumeric characters** are indicated, they are enclosed in double quotes: "0B"

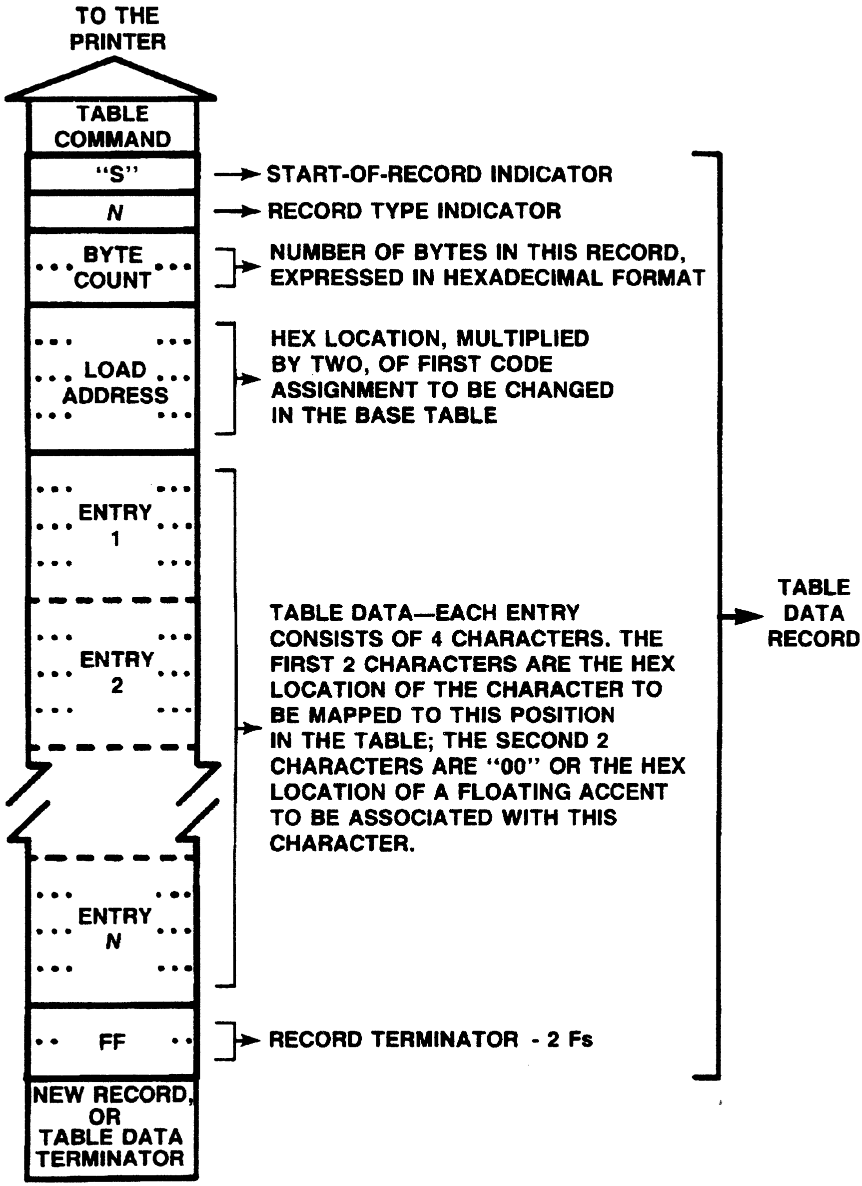


Figure 2-2 **Table data record**

Start-of-record indicator	This is always the letter "S".
Record-type indicator	This is a 2, 3, or 4. Each number has a different significance: **2** indicates that only the U.S. English ASCII table is to be altered. The U.S. English ASCII table serves as a base for many non-ISO fonts. **3** indicates that only the ISO table is to be altered. The ISO table serves as a base for both ASCII and EBCDIC multinational mappings. **4** indicates that both the ISO and the U.S. English ASCII tables are to be altered.
Byte count	is one-half the number of bytes (characters) in the **remainder** of the record, including the load address and the two "F"s at the end. Each character in the record represents a 4-bit hexadecimal value; therefore the byte count is half the actual number of characters in the record. The byte count is always expressed as a hexadecimal value and is always two bytes long.
Load address	is the first hex location to be altered, **multiplied by 2**, and is always four characters long. For example, if the first location to be altered is X'40', the value to be entered as the load address is "0080". Entry 1 of the table data applies to this location. The Model 50 assumes that all subsequent entries apply to successive hex values. For example, if the load address is '5B', entry 1 will be applied to hex location 5B, entry 2 will be applied to hex location 5C, entry 3 will be applied to hex location 5D, and so forth. If there are locations between the first and last entry that are not to be altered, there must be an entry for each such location ascribing its own value or the locations to be altered can be loaded with two separate records. A table of hexadecimal values multiplied by two is included in appendix A.

Table entries

The table data consists of a series of four-character entries, one for each location from the load address to the last location to be altered. Intervening locations which do not need to be altered may be included by simply reassigning their normal hex value; for example, by assigning '7B' to hex location 7B.

The four characters include the following information:

- The first two characters are the new hex value to be assigned to the current location. Suppose, for example, that you wish to print the Japanese yen sign (X'A5') from an ISO font. Suppose also that you do not need to print the backslash (\, X'5C'). You can use custom table downloading to assign the hex value A5 to location 5C. If 5C were your first table location, A5 would be the first two bytes of entry 1 in your table data.

- The second two characters of each entry are the hex location of a floating accent character to be printed along with the character specified by the first two bytes. Only the value from column C should be included. The Model 50 automatically selects an accent from column 8 or C, depending on the height of the character with which it is associated. If no floating accent is to be printed, the value of the null code, 00, should be included here.

  For another example, let's say that you have no math font and you wish to create a symbol to approximate "≤". Suppose also that you're willing to sacrifice the printing of the ampersand (&, X'26'). You might try printing the less-than sign (<, X'3C') along with the floating underscore character (_ , X'CC'). Your four-byte entry for location X'26' (the ampersand) would be "3CCC". Thereafter, when you invoke the downloaded table and send an X'26', the Model 50 will print "<."

	Note: The following locations are assigned to control codes, frequently-used alphanumeric characters, and characters used in printer command language. It is recommended that these locations not be altered: – Any control code location (less than X'20' in ISO or ASCII, or less than X'40' in EBCDIC) – ISO X'30' through '39' (numerals 0 through 9) EBCDIC X'F0' through X'F9' – ISO X'41' through X'5A' (letters A through Z) EBCDIC X'C1' through 'C9', 'D1' through 'D9', and 'E2' through 'E9' – ISO X'61' through '7A' (letters a through z) EBCDIC X'81' through '89', '91' through '99', and 'A2' through 'A9' – ISO X'20', EBCDIC X'40' (space) ISO X'2B', EBCDIC X'4E' (plus sign +) ISO X'2C', EBCDIC X'6B' (comma ,) ISO X'3D', EBCDIC X'7E' (equals sign =)
Record terminator	The last two characters of every record are the characters "FF". These two bytes must be included in the byte count at the beginning of the record.

Table-load terminator The table-load terminator follows the last record in the file. The form of the terminator is:

X(line-end)

where **X** is the capital letter X and (line-end) is (CR)(LF), (CR), or (LF), depending on the line ending option installed for your system.

Table-load example

Let us suppose that you do most of your business with firms in the United States, so that the U.S. English code mapping, illustrated in Table 2-6, is suitable for most of your applications, but that you also need the ability to print the pound sterling sign (£), the yen sign (¥), and the international currency symbol (¤). Let us also suppose that you are willing to sacrifice the printing of the backslash (\, X'5C'), the spacing circumflex (^, X'5E'), and the spacing grave (`, X'60'). You might build your custom translation table as follows.

Table 2-6 **U.S. English code mapping**

Most →

Least ↓	0	1	2	3	4	5	6	7
0			Sp	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	\	l	¦*
D			–	=	M	]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

Building the file First, determine the load address of the first entry to be included in the table. The lowest value you wish to change is '5C', therefore, as shown in the Double Hex Values table in appendix A, the load address for this table is B8. As the load address must always be four characters in length, the value to be included as the load address is "00B8".

Next, build the entries to be included in the table. The first entry, at location '5C', is to contain the pound sign in the altered table. The pound sterling sign is found in the ISO table at location 'A3'. Since it is not to be associated with an accent character, the first entry is "A300".

The second entry falls at location '5D'. Since this location is not to be changed, the entry here is "5D00".

The third entry falls at '5E'. This location is to be changed to the yen sign, which is found in the ISO table at location 'A5'. It is not to be associated with an accent, so the third entry is "A500".

The fourth entry, at '5F' is not to be changed: "5F00".

The fifth and final entry, at '60', is to be changed to the international currency symbol, from location 'A8' in the ISO table. The fifth entry is "A800".

Thus far, including the load address and the five entries, the table record consists of the following characters:

    00B8A3005D00A5005F00A800

Adding two "F"s to the end of the record yields:

    00B8A3005D00A5005F00A800FF

It is now possible to determine the byte count to be included in the record: one load address (four characters) plus five four-character entries, plus two "F"s yields:

$$4 + (5 \times 4) + 2 = 26$$

The byte count is one-half the actual number of characters in the record and must be expressed as a hexadecimal value ($26 \div 2 = 13$). Appendix A provides a decimal-to-hex equivalence table that shows that decimal 13 is equal to hexadecimal 0D. Now the record contains the following characters:

0D00B8A3005D00A5005F00A800FF

The last determination to be made is the record type. Since we are using ISO characters in this table, either "3" or "4" is appropriate. The entire file then becomes:

(Esc)+**T**(line-end)
S40D00B8A3005D00A5005F00A800FF**X**(line-end)

The components of the table record are delineated in Figure 2-2.

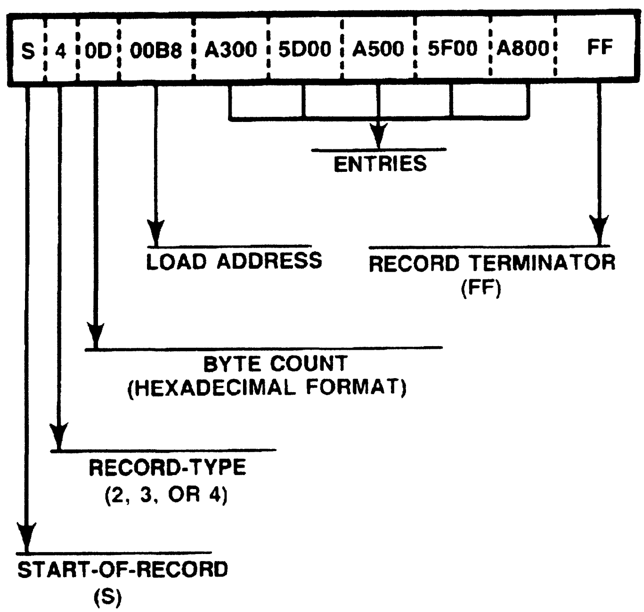


Figure 2-3 **Completed table download record**

Invoking the base table

Before sending the table data file to the Model 50, be sure to invoke the mapping table you wish to alter.

When the Model 50 receives the table downloading command, it creates a new table by laying the new data over the mapping table currently in effect. Locations not specified in the table data remain the same in the new table as in the table over which it was laid, as illustrated in Figure 2-4. In our example, we laid new data over the U.S. English table.

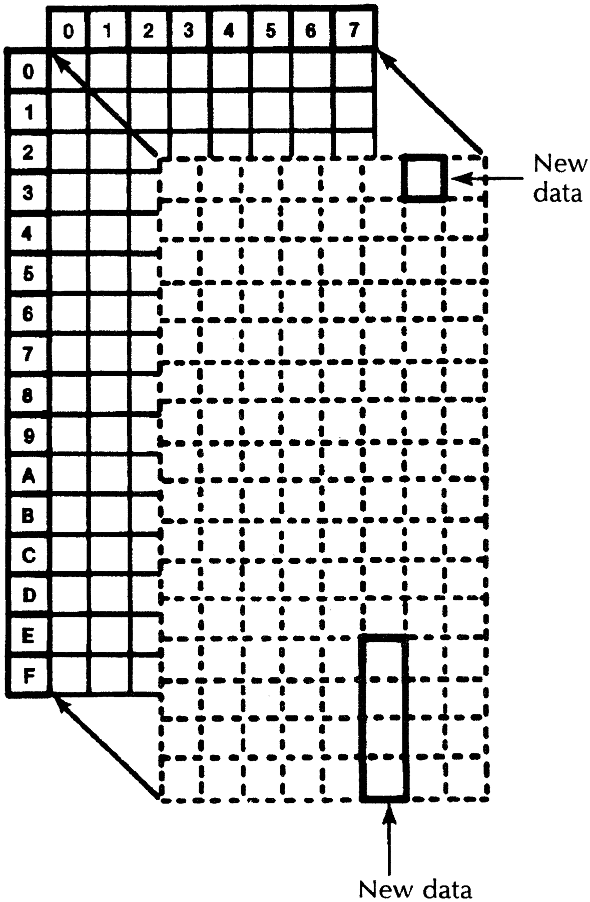


Figure 2-4 **Table overlay**

The resulting new table remains in the system until the printer is turned off, or a new table is created. The table created in this example is illustrated in Table 2-7.

New tables may be invoked by command, or may be selected as the default table by switch setting on the configuration cartridge.

Table 2-7 **Customized mapping table (U.S. English code)**

Most →

Least ↓

	0	1	2	3	4	5	6	7
0			Sp	0	@	P	¤	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	£	l	|
D			-	=	M	]	m	}
E			.	>	N	¥	n	~
F			/	?	O	_	o	

Points to remember about creating tables

1. New tables are created by overlaying other tables; always invoke the base table before transmitting new table data to the Model 50.
2. The table data may consist of one or more records.

3. The record-type indicator must be carefully selected, so as to avoid altering the wrong tables in the system. The IBM PC table may not be altered.
4. The load address is always doubled.
5. Certain codes should not be altered; consult the list under *Table entries*.
6. Each record must be terminated with two "F"s; any other codes in these two bytes cause the entire table-load to be aborted.
7. The proper beginning and ending commands must be chosen according to the printer emulation (2700 or 630) installed in your system.
8. The new table remains in the system as a separate table (that is, all other mappings remain intact) until the Model 50 is turned off or a new table is downloaded.

Formatting non-textual data

Non-textual data includes fonts and graphic data. Fonts are digitized by the Xerox Corporate Font Center and include logos, signatures, and special customized fonts that you may order to suit your particular needs. Graphic data are bit maps you create at your host and send to the Model 50. This section describes how non-textual data may be formatted for printing in the Model 50 and gives you some tips for their successful transmission.

Sixel encoding

Within non-textual data, bit patterns are not intended to represent font characters from a character table. In other words, a hex '47', for example, within graphic data, does not represent the character "G", but is simply a bit pattern, namely: 01000111 for 8-bit hosts or 1000111 for 7-bit hosts. This can be a problem when the bit patterns look like control codes. This is especially true when the data is transmitted through a communications network.

For example, a system that uses space compression might insert the "GS" control code (ASCII X'1D'), followed by a binary number, to indicate a series of space characters; in effect, the host is telling the remote device to "put *n* spaces here." When this code is received by the remote device, the GS code is stripped out and space characters (ASCII X'20's) are inserted. If this should happen within graphic or font data, the bit maps to be imaged would be significantly altered and rendered useless.

To counteract this possibility, Xerox font data (except in cartridges) is encoded using a coding technique called "sixel encoding." The same technique must be used to encode graphic data, to ensure their safe transmission to the Model 50.

Sixel encoding begins with the raw bit pattern images, as generated within the host computer. The stream of 1s and 0s is divided into 6-bit packets, called "sixels." If a binary file of graphic data is not divisible by 24, additional 8-bit bytes of 0s must be added. For example, three 8-bit bytes divide evenly into 24, creating four 6-bit packets, or sixels. But, if a file contains only two 8-bit bytes, after two 6-bit packets are encoded, four bits are left over. A third byte, of 0s, is needed to make an even number of 6-bit packets with no left over bits. Without padding, text data is often lost or graphic images are not printed.

The highest value that six bits can have is 111111, or X'3F' '3F' is then added to each sixel to yield a packet whose value will never be higher than 2 x 3F, or 7E. Values between X'3F' and X'7E' typically represent printable font characters. Thus, sixel-encoded data never contains bit patterns that look like control codes; thus, the chance that they might be altered or removed from the data stream is, for any practical purpose, eliminated.

The Model 50 recognizes sixel-encoded data because of the command preceding it. When the font- or graphic-loading command is received, the Model 50 subtracts '3F' from every succeeding byte it receives and stores the remaining data as bit-mapped data. Figure 2-4A illustrates graphic encoding into sixels and printer decoding. Font data is stored in memory for later use and graphic data is printed on the current page. The Model 50 returns to normal textual printing mode when the end of the font or graphic data is reached. The proper commands are described in the following subsections.

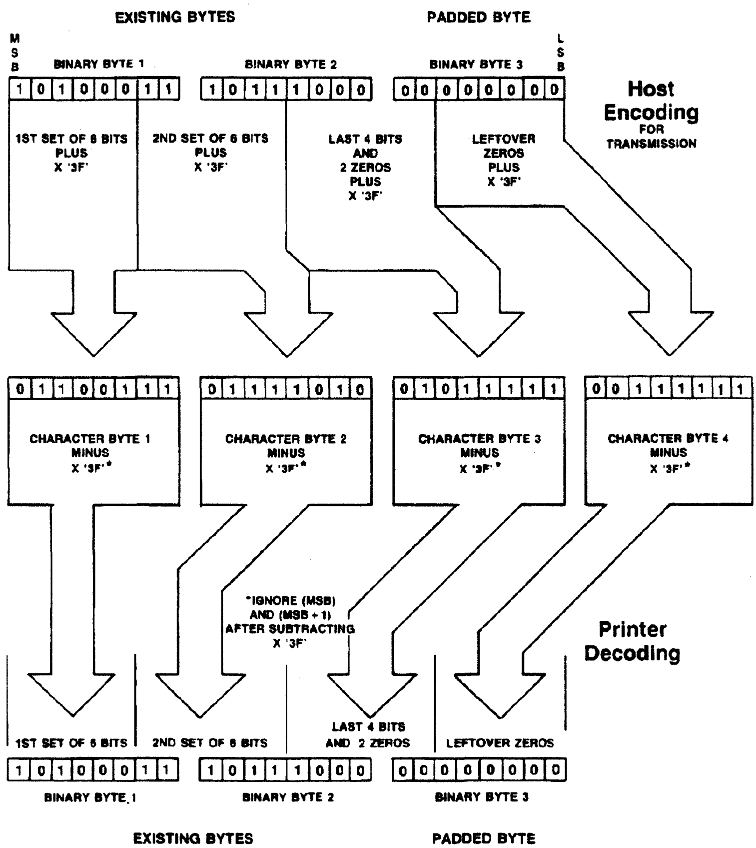


Figure 2-4A **Sixel encoding**

Font data formatting and transmission

Font data for downloading is already sixel-encoded when you receive it. You need to take only three steps to ensure that your fonts will load successfully:

1. Precede the font data with the font-load command. This command has the form:

   (Esc) + **F**,<u>(comment)</u>(line-end)

The components of this command are described in detail in the *Creating documents* section of the *User Manual*. Font data, as encoded by the Xerox Corporate Font Center, contains its own starting and ending delimiters. Thus, more than one font may be loaded following a single command and no special terminator need be sent.

2. If the switch setting for rotating fonts has been selected (see chapter 4), **a font that is landscape for the 2700 will be portrait in the Model 50**. The Model 50 renames the 2700 font when it is received and stores it as a font of opposite orientation. The font is then invoked with its *new* orientation.

3. Ensure that neither your host nor your communications network alter the font data in any way. You may have your fonts blocked in any length when you order them. It is essential that the host not pad the font data, divide it into smaller records, or add codes within the font data.

Remember that these cautions apply only to downloaded fonts. Cartridge fonts require no special handling.

Graphic data formatting and transmission

The Model 50 can accept rasterized data as a series of 1s and 0s (printed and unprinted dots), rather than a series of encoded bytes invoking font characters. The data itself must be preceded by the Graphic Window command, described later under *Creating a graphic window*, and must be sixel-encoded.

You may be using a graphics software package that already has a Model 50 or 2700 printer driver. If so, your package may handle all of the encoding and placement of the graphic data described in this subsection. If not, you will need to read the following paragraphs carefully.

Some tips on using graphic software designed for other printers are provided at the end of this chapter.

Rasterization

"Rasterized data" refers to the ones and zeros contained in an image, laid out in the order in which they will be printed, as diagrammed in Figure 2-5.

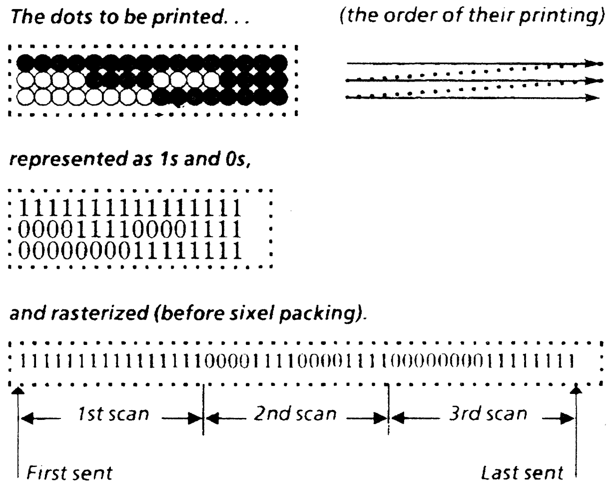


Figure 2-5 **Rasterization**

Graphic data to be printed must be rasterized and sixel-encoded in the host before transmission to the Model 50. **The Model 50 does not recognize vector commands or other types of graphic macro commands.** All the data describing the graphic image must be generated in the host.

Storage

The 4045 Model 50 may be equipped with up to 1.5 MB of memory. This memory is available for storing graphics. The amount of memory that you may use for printing graphic files is listed on the status and configuration sheet as "System Bytes Available." A portion of this memory (up to 440 Kbytes) also could be used for downloading fonts. When fonts are downloaded, the number given for "System Bytes Available" automatically is reduced. The amount of graphic data you can send to the printer, therefore, has two limits:

1. (Total system storage as shown on the status or configuration sheet), **and**
2. 128 graphic images on any given page.

Run-length packing (ASCII encoding only)

The sixel encoding process produces bit patterns whose values range from X'3F' to X'7E'. In addition, the ASCII values X'30' through X'39', representing the numerals 0 through 9, may be used to define the number of consecutive repetitions of a particular sixel value which follow. This is called the "run length." For example, if 100 consecutive values of X'45' were to be transmitted, the data stream would look like this:

31 30 30 45

In this example, spaces have been placed between the ASCII codes for clarity. The transmitted character string would actually take the form X'31303045'.

The repetition count (100 in the above example) is limited to 32,767. If a file contains 32,867 consecutive values of X'45', that is, one hundred more than the maximum repetition count, the data stream would be as follows:

33 32 37 36 37 45 31 30 30 45

The following Basic program illustrates the run-length packing method. This program performs the packing process on a designated file and displays the results on the host's terminal.

```
005   ========A program to pack the sixel-encoded data in
010                  'a given data file and to display the
015                  'results on the CRT screen.
020
025  CLEAR           'Reset all variables
030  CLOSE           'Close any open files
035  PRINT           'Write a blank line on the screen
040  DEFINT I        'Define I as an integer
045                  '
050                  'Prompt the user for the input file name
055                  '
060  INPUT "Enter name of sixel-encoded file";F1$
065                  '
070                  'Open the input file
075                  '
080  OPEN "I",1,F1$
085                  '
090  PRINT           'Write a blank line on the screen
095                  '
100  ICNT=1          'Initialize repetition count to 1
105                  '
110                  'The variables PRESENT$ and PREV$ are used to
120                  'hold the present and previous sixel values
125                  'while searching for consecutive equal values
130                  '
135                  'Read the first sixel into PREV$
140                  '
145  PREV$=INPUT$(1,#1)
150                  '
155                  'Loop to read all the sixels in the file
160                  '
165  WHILE NOT EOF(1)
170                  '
175                  'Read the next sixel value
180                  '
185                  'Compare the present and previous sixels
200                  '
205     PRESENT$=INPUT$(1,#1)
210                  '
215                  'If the sixels are equal, increment the count
220                  'Otherwise, display the sixel on the screen
225                  '
230     IF PREV$=PRESENT$ THEN ICNT=ICNT+1 ELSE GOSUB 360
235                  '
240                  'If the count is at its max value, display
245                  'repetition count and sixel value
250                  '
255     IF ICNT=32767 THEN GOSUB 360
260                  '
265                  'Continue if more sixels to process
270                  '
275  WEND            '
280                  '
285                  'All sixels read from the input file
290                  'Process the last sixels
```

```
295                    '
300 GOSUB 360          '
305                    '
310                    'Close files and end the program
315                    '
320 CLOSE              '
325 END                '
330                    '
335 '========SUBROUTINES: Display the packed sixel data
340                    '
345                    'For repetition counts less than 2, display
350                    'the sixel on the screen and return
355                    '
360 IF ICNT<2 THEN GOSUB 480: RETURN
365                    '
370                    'For repetition counts of 2 or greater, make
375                    'an ASCII string of the count and display it
380                    'on the screen with the sixel
385                    '
390 CNT$=STR$(ICNT)
395                    '
400                    'Get the length of the ASCII string
405                    '
410 ILONG=LEN(CNT$)
415                    '
420                    'Output the string to the screen. Start the
425                    'loop with 2 because the string function
430                    'returns a value which includes the sign
435                    '
440 FOR I=2 TO ILONG
445    PRINT MID$(CNT$,I,1);
450 NEXT I             '
455 GOSUB 480          '
460 RETURN             '
465                    '
470                    'Output the sixel to the screen
475                    '
480 PRINT PREV$;       '
485                    '
490                    'Set previous variable equal to present sixel
495                    '
500 PREV$=PRESENT$
505                    '
510                    'When the repetition counter reaches 32767 it
515                    'must be reset to zero. Otherwise, reset the
520                    'count to 1.
525                    '
530 IF ICNT=32767 THEN ICNT=0 ELSE ICNT=1
535 RETURN             '
```

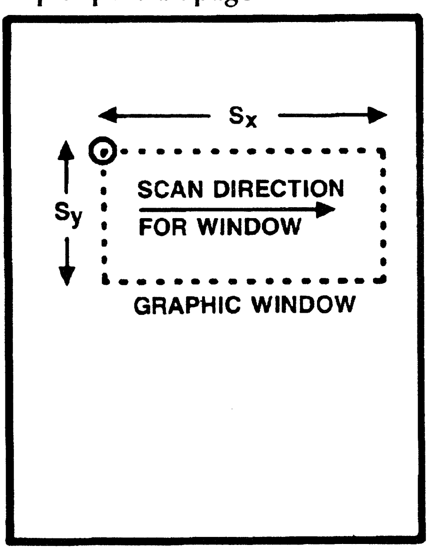
The above example program was written in the Basic-80 language, release 5.0, copyright Microsoft, 1979. It was run on a Xerox 820 II personal computer using the CP/M operating system. CP/M is a registered trademark of Digital Research.

If your host computer, its operating system, or the Basic compiler you use is different, some modifications to the program may be required in order to have it execute properly.

Graphic windows

A graphic window is simply a rectangular area on a page within which the graphic image is to be printed. Figure 2-6 illustrates the concept of a graphic window and calls out some of its elements. The various elements are described in the following paragraphs.

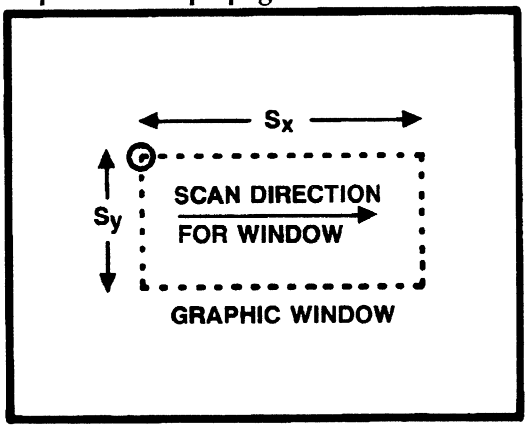
Top of portrait page



O = *graphic window origin*

Figure 2-6A **Graphic window elements**

Top of landscape page



O = *graphic window origin*

Figure 2-6B **Graphic window elements (continued)**

Note that the graphic window is described as a Cartesian coordinate system. However, in the usual Cartesian system, values along the y-axis *below* the x-axis have a negative value. For the graphic window, all y-dimension values are positive. This method of measuring the window's dimensions is similar to that used by most graphics software to measure pixel **locations on the CRT screen**. The creation of a graphic window is, in fact, analogous to the creation of a graphic image on the CRT screen, and then dumping the contents of the screen to the printer.

Origin	The origin of the graphic window is its upper left-hand corner, as the image is viewed in its normal orientation. The origin is often designated as (0,0), indicating that both x- and y-values are zero.
Scan direction	The scan direction of the graphic window is always from left to right, starting at the graphic window origin; successive scan lines are located further down the page and begin at the left edge of the window. In other words, the scan direction of the graphic window is always parallel to the top edge of the page, as viewed in its normal orientation. This is always true, regardless of the printer emulation used, the orientation of the printed page, or the actual scanning direction of the Model 50.
S_x	This is the number of pixels (dots) along the window's x-dimension. The x-dimension of the graphic window is always parallel to the top edge of the printed page.
S_y	This is the number of pixels (dots) along the window's y-dimension, and is always a positive value. The y-dimension of the graphic window is always perpendicular to the top edge of the printed page.

Creating a graphic window

A graphic window is created simply by means of the Model 50's Graphic Window command. This command is sent to the Model 50 in front of the graphic data, to set the printer up for graphic data receipt. Two forms of the command are provided, one for the 2700 mode and one for the 630 mode. The commands are shown below. Some parameters are described in greater detail in following subsections.

2700 mode:

(Esc)**gw**$M;X,Y,S_x,S_y$(line-end)(raster data)

630 mode:

(Esc)**vw**$M;X,Y,S_x,S_y$(line-end)(raster data) where:

(Esc)	is the escape code (ASCII '1B' or EBCDIC '27'), or the UDK
gw	is either ASCII '67 77' or EBCDIC '87 A6' for 2700 mode
vw	is ASCII '76 77' or EBCDIC 'A5 A6' for 630 mode
M	is a numeric field defining the graphic's magnification. Magnification is described in greater detail below.
;	is a required semicolon (ASCII '3B' or EBCDIC '5E'). Absence of this character causes the command to be ignored, and subsequent data to be interpreted as character codes.
X	is an optional, variable-length numeric field defining the x-distance, in pixels, from the page origin to the graphic window's origin. The page origin may vary according to the printer mode (2700 or 630) being used. Positioning is described in greater detail below.
,	is a required comma (ASCII '2C' or EBCDIC '6B'). Absence of this character causes the command to be ignored, and subsequent data to be interpreted as character codes.
Y	is an optional, variable-length numeric field defining the y-distance, in pixels, from the page origin to the graphic window's origin. The page origin may vary according to the printer mode (2700 or 630) being used. Positioning is described in greater detail below.
S_x	is a variable-length numeric field defining the number of pixels in the graphic window's x-direction. **This value must be evenly divisible by eight.**

S_y	is a variable-length numeric field defining the number of scan lines in the graphic window's y-direction.
(line-end)	is a line ending sequence.
(raster data)	is a sequence of data bits to set the pixels in the graphics window to either black or white. This bit stream must be sixel-encoded and may be packed using the run-length packing method described earlier, under *Run-Length Packing*.

Magnification

(*M*) Graphic windows may be printed with 1, 2, or 4X magnification. With 1X (one times) magnification, each bit in the graphic file represents one printed or unprinted dot. With 2X (two times) magnification, each bit in the graphic file represents **two** dots in both X and Y directions. Thus, the number of bits required to cover a given area is reduced by a factor of 4. Similar calculations may be performed for 4X magnification, using a factor of 16. The resulting image is coarser, but requires one-fourth the storage area in memory. This relationship is shown in Figure 2-7 and Table 2-8.

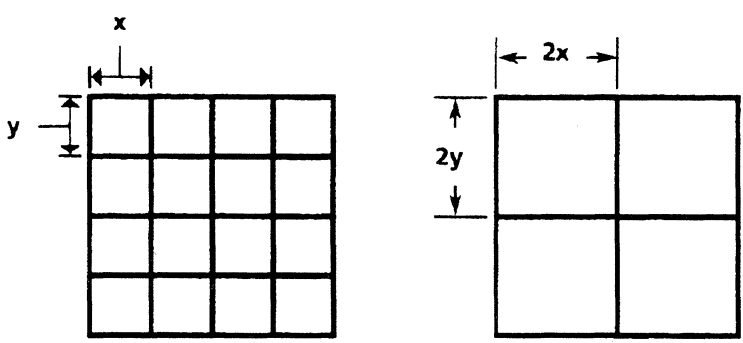


Figure 2-7 **Resolution and pixel size**

Table 2-8 **Effect of magnification on print area**

Units	1X magnification	2X magnification	4X magnification
Sq. in.	Area = $\frac{\text{Memory *}}{11{,}250}$	Area = $\frac{\text{Memory *}}{2{,}813}$	Area = $\frac{\text{Memory *}}{703}$
Sq. cm.	Area = $\frac{\text{Memory *}}{1{,}744}$	Area = $\frac{\text{Memory *}}{436}$	Area = $\frac{\text{Memory *}}{109}$

* Memory is the number of bytes available, as reported on the status and configuration sheets labeled "System Bytes Available."

A value of one (ASCII '31', EBCDIC 'F1') in the magnification field invokes 1X magnification; a value of two invokes 2X magnification; a value of four invokes 4X magnification; any other value, or an empty field, invokes 1X magnification.

Note: The image may fill the 8.5-inch dimension of an 8.5 x 11 or 14-inch page or the 210 mm dimension of an A4 page.

Positioning

(*X* and *Y*) These two parameters control the placement of the graphic window on the page. If they are omitted, the origin of the graphic falls at the current print position, as illustrated in Figure 2-8A.

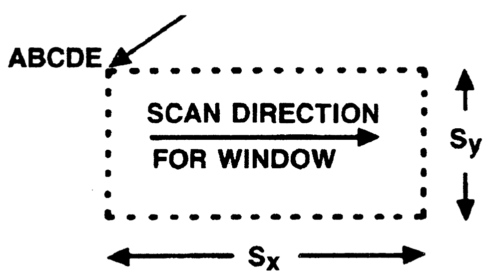


Figure 2-8A **Window placement—current**

The creation of the graphic window does not change the print position placement location. If, after creating the graphic window, the text string 'FGHIJ' were sent to the Model 50, that text string would be printed across the upper edge of the graphic window, as illustrated in Figure 2-8B. The new current placement location would then follow the letter J in the text string.

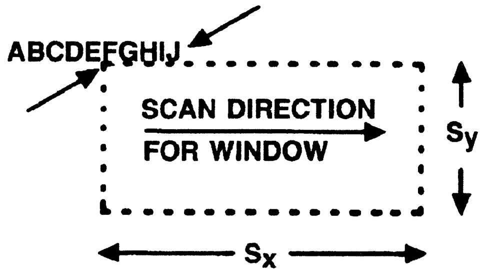


Figure 2-8B **Window placement—current (continued)**

If values for *X* and *Y* are included, they specify the absolute location on the page, relative to the **page origin**, where the graphic window origin is to fall. The page origin may change, depending on the printer emulation used. Figures 2-9 and 2-10 illustrate absolute placement of a graphic window.

If either *X* or *Y* is omitted and the other is present, the omitted field is set to the value of the current print position.

Note that, if values for *X* or *Y* are omitted, **the commas must be included in the command**. For example:

(Esc)**gw**1;,,S_x,S_y(line-end)(raster data)

Omission of the commas causes the command to be ignored and the subsequent raster data to be interpreted as character codes.

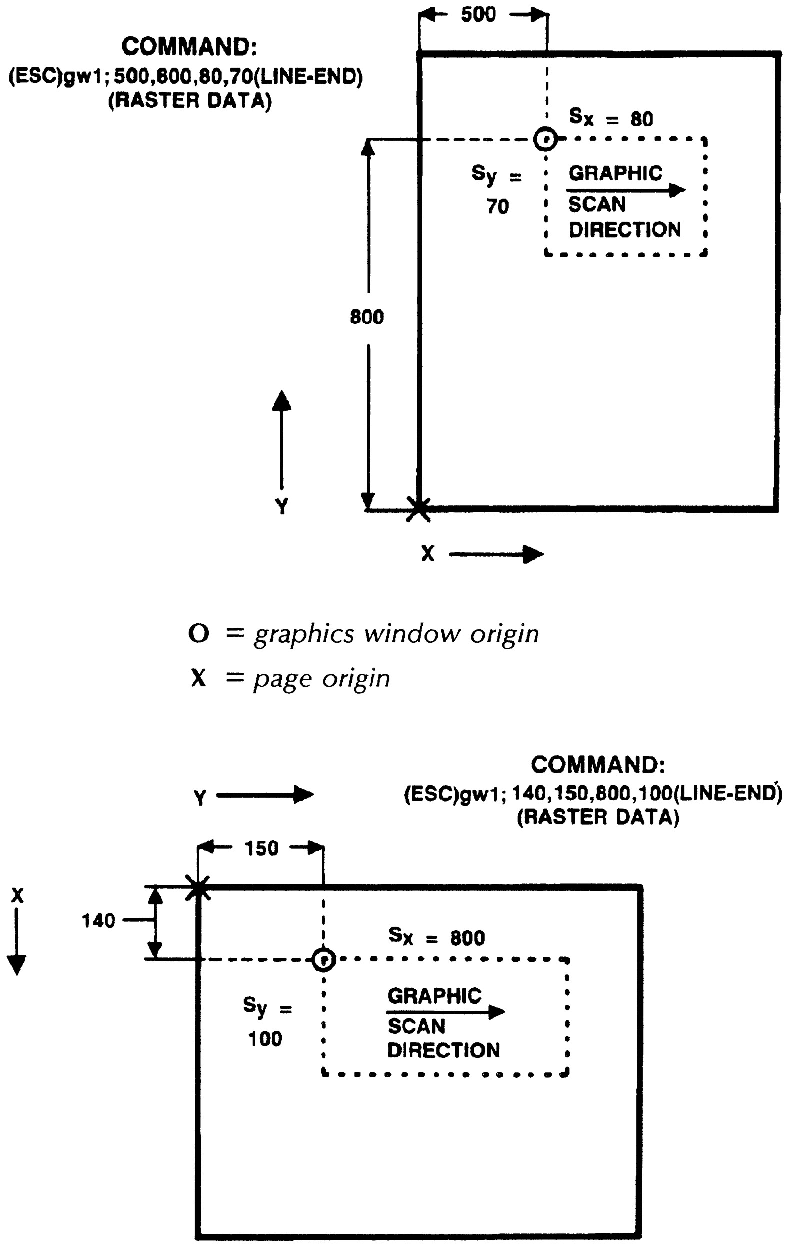
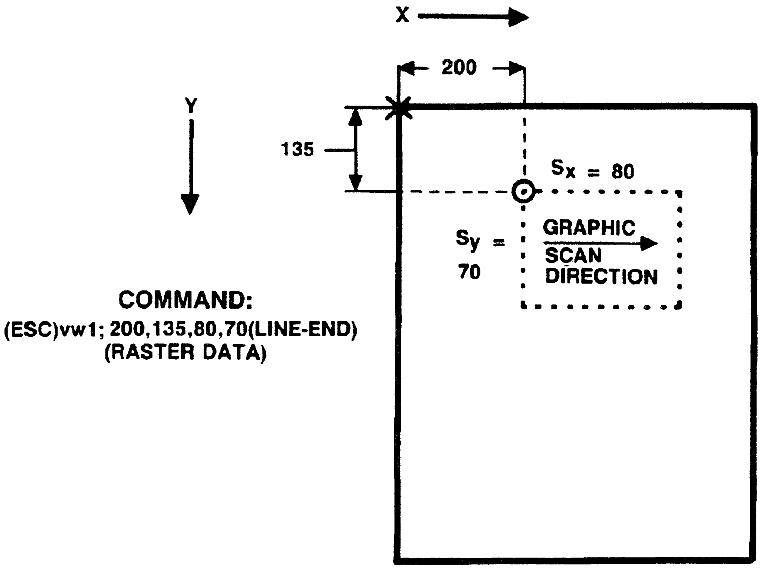


Figure 2-9 **2700 mode graphics placement**



O = *graphics window origin*

X = *page origin*

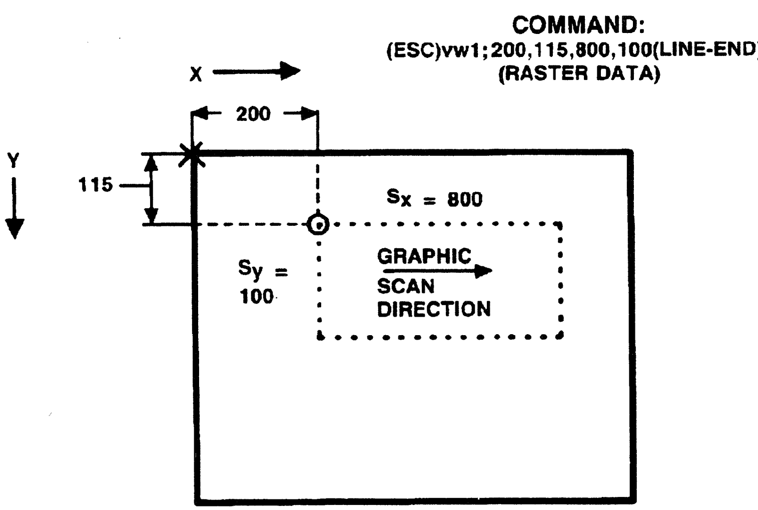


Figure 2-10 **630 mode graphics placement**

Graphic window size

(S_x and S_y) These parameters describe the size of the graphic window. Together, they tell the Model 50 the length of each scan line and the total number of scan lines.

S_x represents the number of pixels (dots) along the top edge of the graphic window and must be divisible by 8. If this value is not an even multiple of 8, the Model 50 truncates this value to the next lower multiple. If the actual number of dots in each scan line is not evenly divisible by 8, each line must be padded with zeros to a multiple of 8. Otherwise, a skewed image will be printed. See *Sixel encoding*, previously discussed in this chapter.

Note that the scan direction of the window may or may not correspond to the scan direction of the Model 50's laser imager. The scan direction of the graphic window is always parallel to the top edge of the image.

S_y represents the number of scan lines in the window; that is, the number of dots along its vertical edge.

S_x and S_y must be separated by commas. The omission of either field causes the command to be ignored and subsequent raster data to be interpreted as character codes.

Raster data

Each dot in the image must be represented by a 1 or a 0 in the data. The first bit received must be the dot to be printed at the graphic window origin. Subsequent bits represent the dots as they are printed from left to right in the first scan line, across the top of the graphic window, and then from left to right in each subsequent scan line. Each scan line begins at the left edge of the window, and the number of dots in each scan line must be evenly divisible by 8. Note again that **the scan direction of the graphic window need not correspond to the scan direction of the Model 50 laser imager**. The Model 50 rotates data as necessary to accommodate the orientation of the page, as chosen by font selection.

Note: Bit patterns whose value is less than X'30' are ignored within graphic data, except X'1B'. An X'1B' within data assumed to be graphic data causes the graphic download to be aborted and all graphic data to be ignored. The remainder of the page is printed. This applies regardless of the coding scheme (ISO, ASCII, EBCDIC, or IBM PC) installed.

Repeating a graphic window

A graphic window that has already been created and stored in the Model 50 can be reprinted by means of the Repeat Window command. The command includes positioning parameters that allow the window to be placed anywhere on the page, as many times as desired. A repeated window does not use extra memory space. Two forms of the command are provided, one for the 2700 mode and one for the 630 mode. The commands are similar to the Graphic Window commands and are shown below.

2700 mode:

(Esc)**gr**M;X_1,Y_1,X_2,Y_2...(line-end)(raster data)

630 mode:

(Esc)**vr**M;X_1,Y_1,X_2,Y_2...(line-end)(raster data)

where:

(Esc)	is the escape code (ASCII '1B' or EBCDIC '27'), or the UDK
gr	is either ASCII '67 72' or EBCDIC '87 99' for 2700 mode
vr	is ASCII '76 72' or EBCDIC 'A5 99' for 630 mode
M	is a numeric field defining the graphic's magnification. This number can be 1, 2, or 4. If the numeric field is 1 (one-times magnification), each bit represents one dot, which indicates no expansion (300 dots/inch). A two-times magnification doubles the graphic field so each bit represents two dots (150 dots/inch). At four-times magnification, a bit represents four dots, creating a graphic with 75 dots/inch.

;	is a required semicolon (ASCII '3B' or EBCDIC '5E'). Absence of this character causes the command to be ignored, and subsequent data to be interpreted as character codes.
X	is a variable-length numeric field defining the x-distance, in pixels, from the page origin to the graphic window's origin. The page origin may vary according to the printer mode (2700 or 630) being used. You may define more than one, depending on the number of windows you want to repeat.
,	is a required comma (ASCII '2C' or EBCDIC '6B'). Absence of this character causes the command to be ignored, and subsequent data to be interpreted as character codes.
Y	is a variable-length numeric field defining the y-distance, in pixels, from the page origin to the graphic window's origin. The page origin may vary according to the printer mode (2700 or 630) being used. You may define more than one, depending on the number of windows you want to repeat.
(line-end)	is a line ending sequence.

Hints for the use of graphic windows

Placement

Graphic windows may be placed anywhere on a page, and may be intermixed with text. The graphic data is stored in memory until a page-end is seen, and then printed with the rest of the data for that page.

A graphic window may be positioned absolutely on the page by the inclusion of values in the Graphic Window command, specifying the distance of its origin from the page origin. Omission of these values causes the window origin to fall at the current print position, depending on where the command is placed within the data for that page.

Orientation

The Model 50 assumes the orientation of the last font specified, or the default font if none is specified, for the page on which the graphic is to be printed. Raster data is rotated as necessary to accommodate the specified orientation. **Be sure to invoke a font of the required orientation before sending the Graphic Window command.**

Magnification

Magnification results in a larger, but coarser, image (150 pixels per inch or approximately 5.9 per millimeter). However, the larger image requires less memory for storage. 2X magnification results in an image covering **four** times the area on the page, and 4X magnification results in an image sixteen times larger.

Maxima

Up to 128 separate graphic windows may be printed on a page, provided that there is enough memory available to store the rasterized data. (The number given for "System Bytes Available" on the status and configuration sheets tells you how much space may be used for graphic windows.) Any window may be repeated as many times as desired; repeated windows do not use extra memory space. The status and configuration sheets provide the amount of memory available.

2x magnification allows an image to cover four times the area, and 4X magnification 16 times the area on the printed page as an image using 1X magnification, for the same amount of memory.

Once a page is printed, all graphic data for that page is erased, and data for up to 128 new graphics may be stored.

Graphics error conditions

A series of error codes are provided, printed on the status sheet, to signal problems in printing graphic images. The following paragraphs describe each code in detail, along with possible sources of error.

Error code **80—Graphic could not be printed beyond the physical limits of the page.**

The Model 50 detected that, due to the graphic size or the positioning parameters called out in the command, or both, that at least a portion of the window fell beyond the physical limits of the page, as illustrated in the following figure.

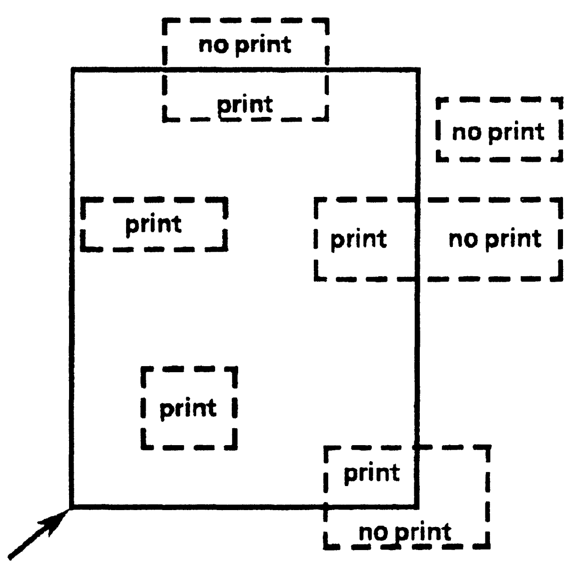


Figure 2-11 **Error code 80**

The Model 50 prints as much of the image as falls within the page limits.

Error code **81—The S_x value was not an even multiple of 8.**

The Model 50 rounds this value down to the next lower multiple of 8. No data is lost, but the printed image is skewed and the data which appears to exceed the window size is interpreted as character code data. This situation is illustrated in Figure 2-12.

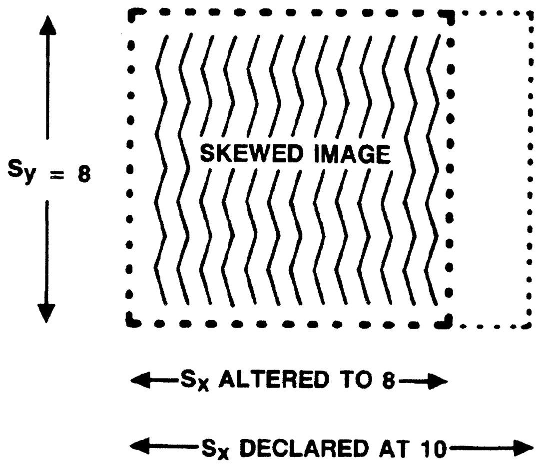


Figure 2-12 **Error code 81**

Error code **83—The sixel-encoded data contained a repetition value in excess of 32,767.**

All the encoded data are ignored; no graphic is printed.

Error code **85—Insufficient memory available for received graphic.**

There was not enough memory available to store the entire graphic. The Model 50 prints as much of the image as it is able to store.

Other conditions **The number of bits of raster data sent is not equal to the product of S_x and S_y.**

The product of these two parameters tells the Model 50 how many bits of subsequent data are raster data. As soon as this number of bits has been received, the Model 50 returns to normal, textual printing. No error code is generated if the graphic file is not of the correct size. Two results are possible, depending on whether the file is too small or too large.

If the file is too small to fill the window, i.e., there are not as many bits in the image as specified in the command, subsequent data is assumed to be raster data and is unpacked and printed as part of the graphic image. Codes whose value is less than X'30' are ignored, except X'1B' (escape). An X'1B' within data assumed to be graphic data causes the entire graphic file to be ignored. This applies regardless of the coding scheme (ISO, ASCII, EBCDIC, or IBM PC) installed.

If the file is too large, and there are more bits in the image than are specified in the command, the excess data is assumed to be character code data and invokes the printing of font characters at the current print position.

Using graphics packages for other printers

Commercial graphics software packages generate raster graphics files to drive the devices they support. It may be necessary to reformat the file, as described in the following paragraphs, in order to print it on the Model 50.

Adapting the graphic file

Once the format and content of the graphic's raster file is known, header and internal control information must be stripped from the file, leaving the raster bit map information only.

The raster bit map information must be formatted so that a 1 will print black and a 0 will print white. Files which contain color information should be modified to print black and white only.

The raster data which now meets the requirements of the Model 50 should then be converted to sixels and packed according to the run-length packing method described earlier.

EBCDIC hosts

If the operating environment of your computer system requires EBCDIC encoded data, the packed sixel data should be translated to EBCDIC according to translation table B-43 shown in appendix B. However, run-length packing in EBCDIC is not supported.

Adjusting the resolution (pixels per inch)

If the graphics application created the raster bit map for a device with fewer pixels per unit length than the Model 50, the image will be printed in reduced size. If the raster bit map was created for a device with more pixels per unit length than the Model 50, the image will be printed in increased size. The amount of reduction or increase of the printed image is the ratio of the two device's pixels per unit length. In other words, a raster bit map created for a device with half the number of pixels per unit length as the Model 50 will print at half the size, if 1X magnification is used. The same image will print at full size if 2X magnification is used.

Transmitting the file

The sixel-encoded and packed raster data can then be appended to a graphics window command with the parameters appropriately set for proper page placement, graphics window size, and magnification. The command and data can then be sent to the Model 50 using the standard print handler in the applications package or by a system command to copy the file to the 4045 Model 50 printer.

3. Interfaces

This chapter describes the configurations and options available with each of the Model 50's supported interfaces: the parallel and the serial asynchronous. Each interface is described in a separate subsection.

Parallel interface

The Model 50 offers two parallel interfaces: a Centronics interface and a Dataproducts interface. Each is described in a separate subsection, following the description of the interface cable.

The parallel interface cable

You must supply the interface cable to connect the Model 50 to your host. The cable must meet the following specifications in order to meet FCC and VDE regulations.

Type:	Twisted pairs, overall foil or braid shield
No. of conductors:	15 pairs
Wire size:	22 AWG stranded
Cable length:	39.3 feet (10 meters) or as specified by the host
Shield connection:	To the connector conductive case at the Model 50; to the frame ground at the data source

Centronics

Connector The 4045 Model 50 Centronics interface uses an Amphenol 36-pin connector, #57-40360. This connector mates with an Amphenol 57-30360 connector and is illustrated in Figure 3-1. The signals associated with each pin are described in the following section.

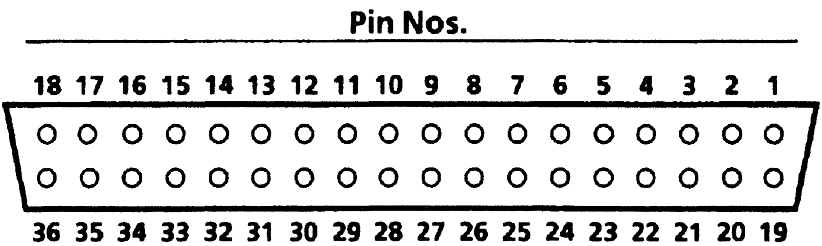


Figure 3-1 **Centronics interface connector**

Signals and pin assignments Table 3-1 describes the signals and pin assignments used with the Centronics 100 interface.

Table 3-1A **Centronics signals and pin assignments**

Signal name	Pins*	Source	Description
Data strobe	1, 19	Host	A 1.0 usec pulse (minimum) used to clock data from the processor to the printer
Data 1	2, 20	Host	Data 1 through Data 8 are input data levels. A high represents one, a low represents zero.
Data 2	3, 21	Host	
Data 3	4, 22	Host	
Data 4	5, 23	Host	
Data 5	6, 24	Host	
Data 6	7, 25	Host	
Data 7	8, 26	Host	
Data 8	9, 27	Host	

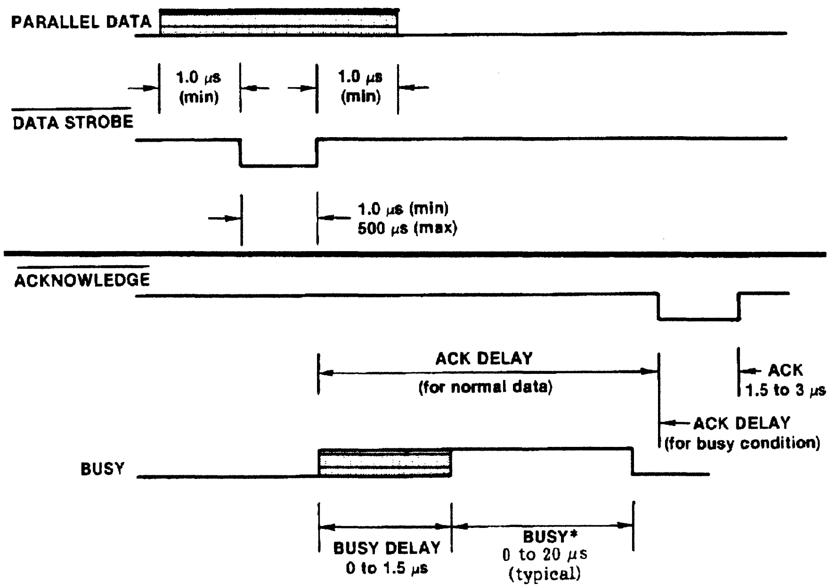
Table 3-1B **Centronics signals and pin assignments**

Signal name	Pins*	Source	Description
Acknowledge	10, 28	4045	The acknowledge pulse indicates that a character has been received.
Busy	11, 29	4045	Indicates that the printer cannot receive data.
PE	12	4045	This signal is always zero.
Select	13	4045	"Printer selected;" remains at one after warm-up indicating on-line and no fault condition.
+0 V	14, 16, 35	4045	Signal ground
Chassis ground	17	4045	Frame ground
+5V	18	4045	+5 volts at 1/2 amp. max.
Fault	32	4045	After warm-up, this signal is one unless there is a fault condition.
The following pins are not used: 15, 30, 31, 33, 34, 36.			
* The second pin of each pair is the twisted signal return (+0 V).			

Figure 3-2 diagrams signal timing and describes voltage levels for this interface.

The two main control signals used by this interface are Select and Fault. At power-up these signals are at logic zero and change to logic one after warm-up. These two signals are always at the same level and remain at logic one as long as no fault condition occurs. Fault conditions are defined as follows:

- Off-line key depressed
- Printer open
- Paper path misfeed (paper jam)
- Dry imager low



*Busy may stay high longer than 20 µs if a fault occurs or the printer buffer is full.

Voltage levels	0 (zero) and +5 VDC (nominal), TTL (SN 74LS00 series)
Logic levels	Positive logic is assumed. A logic one (or high) signal is defined as a voltage in the range of +2.4V to +5V, not to exceed a peak of +5.5V. A logic zero (or low) signal is defined as a voltage in the range of 0.0V to 0.4V, not to exceed a peak negative voltage of -0.5V. However, for received signals a voltage of up to +0.8V is recognized as a logic zero.
Current requirements	The printer interface sources up to -15 ma at +2.4V for a high output signal and sinks up to 14 ma for a low output. 74LS374 ICs are used for data receivers with 1Kohm pull-up resistors, and 74LS244 ICs are used as drivers and for Acknowledge, Busy, Select, etc.
Data rates	The Model 50 supports burst data rates of up to 50 Kbytes per second for at least 560 bytes, or until a line-end or form-feed is detected.

Figure 3-2 **Centronics signal timing**

If a fault condition occurs, the Model 50 continues to receive data until a carriage return, line feed, or form feed is received. On the next following data strobe, if the fault condition still exists, the Model 50 changes the Select and Fault signals to zero. When the fault is cleared, Select and Fault are returned to logic one. If the fault is a paper path misfeed, the signals are not changed until the page that must be reprinted is delivered to the output tray. Synchronization with Data Strobe is required to guarantee that Select is not logic zero when Busy is logic zero.

Dataproducts 2260

The Dataproducts interface must be selected by switch setting on the configuration cartridge.

Connector The Dataproducts interface uses a 50-pin Winchester connector # MRA 50 S D5J, which mates with a Winchester MRAC 50 P JTCH connector, using 50 1020P pins. The connector is illustrated in the following figure. Note that the pin names are provided in the figure for informational purposes—they do not appear on the real connector.

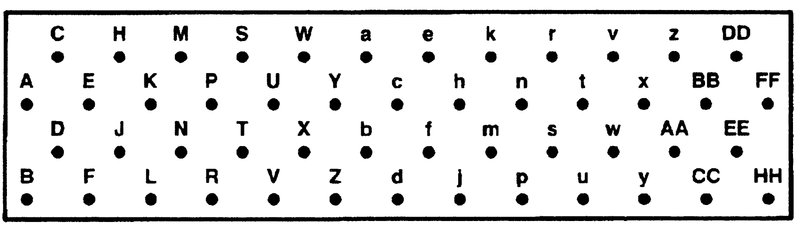


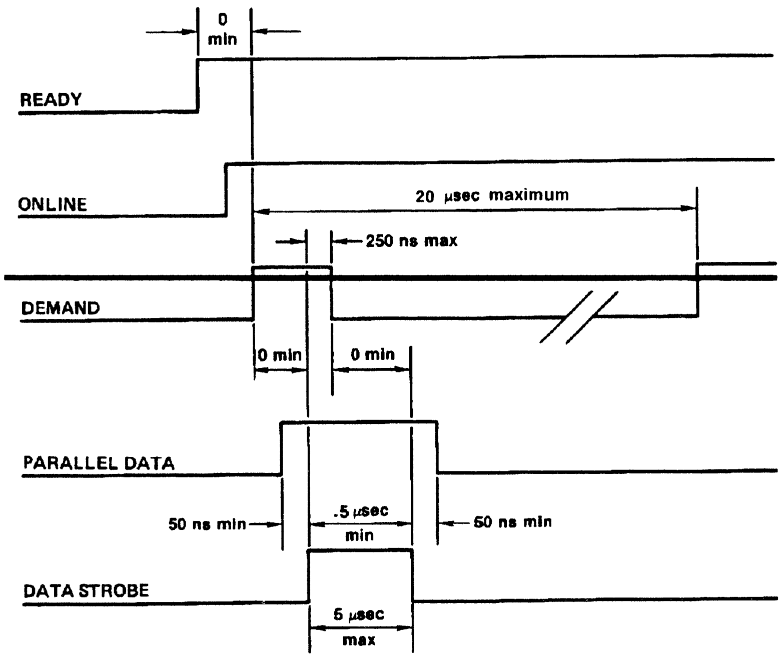
Figure 3-3 **Dataproducts interface connector**

Signals and pin assignments Table 3-2 describes the signals and pin assignments used with the Dataproducts 2260 interface.

Table 3-2 **Dataproducts signals and pin assignments**

Signal name	Pins*	Source	Description
Data 1	B, D	Host	Data 1 through Data 8 are input data levels. A high represents one, a low represents zero.
Data 2	F, J	Host	
Data 3	L, N	Host	
Data 4	R, T	Host	
Data 5	V, X	Host	
Data 6	Z, b	Host	
Data 7	n, k	Host	
Data 8	p, s	Host	
Data Strobe	j, m	Host	A 0.5 u-sec pulse (minimum) used to clock data from the processor to the printer logic.
Demand	E, C	4045	Indicates that the printer is capable of receiving a character.
Ready	CC, EE	4045	Indicates that the printer has powered up successfully.
On-line	y, AA	4045	Indicates that no fault condition exists.
Interface connect	x	4045	Pins x and v provide electrical continuity only.
Verify	v	4045	
The following pins are not used: A, H, K, M, P, S, U, W, Y, BB, DD, FF, HH, a, c, d, e, f, h, r, t, u, w, z.			
* The second pin of each pair is the twisted signal return (+0V).			

Figure 3-4 diagrams signal timing and describes voltage levels for this interface.



Voltage levels	0 (zero) and +5 VDC (nominal), TTL (SN 74LS00 series)
Logic levels	Positive logic is assumed. A logic one (or high) signal is defined as a voltage in the range of +2.4V to +5V, not to exceed a peak of +5.5V. A logic zero (or low) signal is defined as a voltage in the range of 0.0V to 0.4V, not to exceed a peak negative voltage of -0.5V. However, for received signals a voltage of up to +0.8V is recognized as a logic zero.
Current requirements	The printer interface sources up to -15 ma at +2.4V for a high output signal and sinks up to 14 ma for a low output. 74LS374 ICs are used for data receivers with a 220 Ohm pull-up resistor and a 330 Ohm pull-down resistor; 74LS244 ICs are used as drivers and for Ready, On-line, and Demand.
Data rates	The Model 50 supports burst data rates of up to 50 Kbytes per second for at least 560 bytes, or until a line-end or form-feed is detected.

Figure 3-4 **Dataproducts signal timing**

The two main control signals are Ready and On-line. At power-up these signals are at logic zero and change to logic one after warm-up. Ready then remains at logic one until power-down. On-line remains at one as long as no fault condition occurs. Fault conditions are defined as follows:

- Off-line key depressed
- Printer open
- Paper path misfeed (paper jam)
- Dry imager low

If a fault condition occurs, the Model 50 continues to receive data until a carriage return, line feed, or form feed is received. On the next following data strobe, if the fault condition still exists, the Model 50 changes the On-line signal to zero. When the fault is cleared, On-line is returned to logic one. If the fault is a paper path misfeed, the signal is not changed until the page that must be reprinted is delivered to the output tray. Synchronization with Data Strobe is required to guarantee that On-line is logic zero when Demand is logic one.

Special features with the parallel interface

There are two features specific to the parallel interface that are not available with the serial interface: inverted logic (also called "inverted data polarity") and vertical format unit (VFU) emulation.

Inverted data polarity

This option is configuration cartridge switch setting D:2 (see the explanation in chapter 4 of this manual.)

VFU emulation

This feature is available only with the parallel interface in 4045 (2700) mode. It provides a means to select print positions along the vertical dimension of the page. In other words, it is a way of selecting line numbers. This feature imitates the way in which some Dataproducts printers control vertical paper movement by means of a punched tape having 12 "channels." Each channel represents a particular pattern of vertical "stops." One-byte commands are sent to the printer to advance from stop to stop.

Setting the channel stops

Channel stop locations are assigned with the following command:

(Esc)zvn_1,n_2, . . . n_8(line-end)

where:

n_1	is a number from 1 to 12 indicating the channel to which the stops are to be assigned.
n_2 through n_8	are the distances, expressed in units of 1/300th inch, from the top of the page, at which stops for this channel are to be set. The stops become absolute positions on the page and are not affected by margin settings. Up to seven stops may be set for each channel.

For example, the following command:

(Esc)**zv1,150,300,450,600,750,900,1050**(line-end)

sets seven stops for channel 1 at one-half inch intervals from the top of the page. A new command for channel 1 erases these values and replaces them with new ones.

All channel-stop values can be erased with the command:

(Esc)**zw**

Invoking the channel stops

Once the channel stops are set, a single, 8-bit byte may be sent from the host to invoke them. Bit 8 of this byte must be a 1. Note that, if this option is used, it must be enabled by switch setting and 8-bit data must also be selected. Furthermore, font characters whose assigned value exceeds X'7F' must be invoked through automatic language mapping or custom table downloading.

When bit 8 is a 1, the Model 50 reads the remaining bits in the byte to determine the action to be performed. The byte may either select a channel and its associated stops, or specify the number of lines the printer is to skip down the page. Bit 5 determines whether the byte selects a channel or skips a number of lines. The following table lists each possible byte value and its significance. Note that the values of bits 7 and 6 are of no significance.

Table 3-3 **VFU invocation**

			Data lines					Channel
8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**Selected**
1	x	x	0	0	0	0	0	01
1	x	x	0	0	0	0	1	02
1	x	x	0	0	0	1	0	03
1	x	x	0	0	0	1	1	04
1	x	x	0	0	1	0	0	05
1	x	x	0	0	1	0	1	06
1	x	x	0	0	1	1	0	07
1	x	x	0	0	1	1	1	08
1	x	x	0	1	0	0	0	09
1	x	x	0	1	0	0	1	10
1	x	x	0	1	0	1	0	11
1	x	x	0	1	0	1	1	12

			Data lines					Lines to
8	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**Skip**
1	x	x	1	0	0	0	0	00
1	x	x	1	0	0	0	1	01
1	x	x	1	0	0	1	0	02
1	x	x	1	0	0	1	1	03
1	x	x	1	0	1	0	0	04
1	x	x	1	0	1	0	1	05
1	x	x	1	0	1	1	0	06
1	x	x	1	0	1	1	1	07
1	x	x	1	1	0	0	0	08
1	x	x	1	1	0	0	1	09
1	x	x	1	1	0	1	0	10
1	x	x	1	1	0	1	1	11
1	x	x	1	1	1	0	0	12
1	x	x	1	1	1	0	1	13
1	x	x	1	1	1	1	0	14
1	x	x	1	1	1	1	1	15

Notes about VFU use

1. VFU emulation is only available with the parallel interface (switch A:1 off) in 4045 (2700) mode (switch A:2 off).
2. Switch A:5 (7/8-bit data) must be set for 8-bit (on); switch D:3 (VFU) must be on.
3. All codes whose eighth bit is one invoke VFU functions.
4. Channel stops must be set with the VFU Stops Set command before they can be invoked. Each command sets stops for one channel, superceding previous commands for that channel. All stops may be set to zero by the VFU Stops Clear command.
5. When the Model 50 receives a channel-selector command, it moves printing to the next stop set for that channel, below the current print position. If the last stop for that channel has been passed, printing moves to the first stop for that channel on the **next** page, becoming, in effect, a form-feed.
6. When a VFU command specifies a number of lines to be skipped, the actual distance moved down the page is a multiple of the height of the largest font in the line in which the VFU invocation occurs. If, for example, a command specifying eight lines is received and the largest font in the current line is 50 dots high, the actual distance moved is 400 dots. The height of a particular font, as well as other specifics about the font, is provided when the font is ordered.

Serial interface

Connector and pin assignments

The serial interface connector is compatible with RS-232-C standards and is illustrated below. The table following describes the pin assignments for the interface signals.

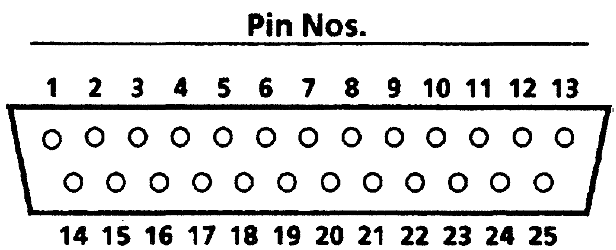


Figure 3-5 **RS-232 (serial) connector**

Table 3-4A **Serial interface pin assignments**

Signal name	Pin #	Tele-phone	CCITT	Source	Description
Protective ground	1	AA	101	--	
Transmitted data	2	BA	103	4045	Data transmitted to host
Received data	3	BB	104	Host	Data received from host
Request to send	4	CA	105	4045	Asserted when Model 50 wishes to send
Clear to send	5	CB	106	Host	Must be asserted to allow Model 50 to transmit
Data set ready	6	CC	107	Host	Must be asserted to allow Model 50 to receive
Signal ground (common return)	7	AB	102	--	
Carrier detect	8	CF	109	Host	Must be asserted to allow Model 50 to receive
Printer ready	11	--	--	4045	Same as Data Terminal Ready, using TTL levels

Table 3-4B **Serial interface pin assignments (continued)**

Signal name	Pin #	Tele-phone	CCITT	Source	Description
Data terminal ready	20	CD	108	4045	Asserted by Model 50 when ready to receive data
Ring indicator	22	CE	125		
The following pins are not used: 9, 10, 12-19, 21, and 23-25.					

Asynchronous data transmission

The serial interface supports asynchronous data transmission only. The input buffer size is 4 Kbytes. The use of the Model 50 as a serial printer requires the setting of several switches on the configuration cartridge that are specific to serial data transmission, including:

- byte length
- baud rate
- parity
- data flow controls (protocols)
- auto-disconnect mode

Byte length

Data may be encoded in either seven- or eight-bit bytes. In addition to these seven or eight bits, the Model 50 expects one start bit, a parity bit if parity has been selected by switch setting, and one stop bit to accompany each byte of data.

Transmission rates

The Model 50 supports the following data rates:

- 300
- 600
- 1200

- 2400
- 4800
- 9600
- 19,200

The rate is set by a switch setting on the configuration cartridge. A modem generating the data clock is not required.

Parity and communication errors

The Model 50 supports even, odd, mark, space, or no parity. Two switches on the configuration cartridge determine 1) whether parity is to be used or not, and 2) if enabled, whether parity is to be odd or even.

If a character or code appears to have incorrect parity, or if a framing error occurs (i.e., no stop bit is detected), the Model 50 substitutes the DEL code (X'7F') for the erroneous code, so that the error character appears in the printout. A font character may or may not be printed when this happens, depending on the language and font in effect when the erroneous code is received. Language tables are provided in appendix B.

Data flow controls (protocols)

Three data flow controls are supported to provide host-to-printer handshaking protocols. Each is described separately below. Each is selected by switch setting, and any combination of controls, or none, may be in effect.

XON/XOFF This is also referred to as "DC1/DC3." Under this protocol, the Model 50 sends an X'13' (DC3) to the host to indicate a "not ready to receive" condition, when:

- The input buffer becomes nearly full, (the host must stop sending within 64 character time frame or data is lost)
- The off-line switch is pressed,
- A paper misfeed or jam occurs,
- The printer is opened, so as to gain access to the paper path,

- The printer is out of paper or dry imager.

A X'11' (DC1) code is sent to the host to indicate a "ready to receive" condition, when:

- After powerup or after a soft reset
- The input buffer is nearly empty,
- The printer is set on-line,
- The printer is closed after having been opened,
- Paper or dry imager is replenished.

ETX/ACK When this protocol is off, the printer ignores ETX codes (X'03') received from the host. When this protocol is on, the printer responds to ETX codes from the host by sending back an ACK code (X'06') if the previous block of data was received without error, or a NAK code (ASCII X'15', EBCDIC X'3D') if parity or communications errors were detected.

Printer ready This is a hardware-based form of handshaking, as opposed to software-based, such as XON/XOFF and ETX/ACK. The Printer Ready protocol uses the Printer Ready and Data Terminal Ready signals, appearing on pins 11 and 20, to toggle their state as a signal to the host of printer readiness.

Both signals perform the same logical functions; however, Data Terminal Ready functions at RS232 voltage levels, whereas Printer Ready functions at TTL voltage levels. If Printer Ready is selected (the switch is ON), Data Terminal Ready and Printer Ready serve as a form of protocol or handshaking similar to that provided by the XON/XOFF protocol. Data Terminal Ready and Printer Ready go false to indicate a "not ready to receive" condition, when:

- The input buffer is nearly full,
- The off-line switch is pressed,
- A paper misfeed or jam occurs,
- The printer is opened, so as to gain access to the paper path,
- The printer is out of paper or dry imager.

The signals are set true again to indicate a "ready to receive" condition, when:

- The input buffer has less than 64 bytes of data,
- The printer is set on-line,
- The printer is closed, after having been opened,
- Paper or dry imager is replenished.

Auto-disconnect mode

When the switch (C:8) is enabled, the Async Interface is monitored periodically. It disconnects from the host if:

- the communication line, which has previously received at least one character, is inactive for more than 30 seconds,
- the modem negates DSR (Data Set Ready) after once asserting it.

During a disconnect, DTR (pin 20) and Printer Ready (pin 11) go false (low) for 30 seconds.

The auto-disconnect mode prevents tying up communication lines if:

- the host is idle,
- the "wrong number" connected momentarily,
- the host relinquishes the communication line.

4. Status and configuration

This chapter covers the following topics:

- How to use the switches on the configuration cartridge. This includes a description of what each switch does and how its setting affects the performance of the machine.
- How to print and interpret a status sheet. The status sheet provides font and storage information, as well as messages describing command errors.
- How to print and read a configuration sheet. The configuration sheet tells you how you've set your switches, as well as provide font storage information.

The configuration cartridge

The configuration cartridge must be installed in the first (leftmost) slot in the font compartment before the Model 50 can be operational. If the cartridge is not installed, the code "P1" is displayed on the control panel.

The cartridge contains four switchbanks, each with eight switches. The switches allow the Model 50 to be installed in a variety of environments, accommodating a variety of data formats. The switches and their functions are described in the following subsections by the order of the name of the switchbank in which they are located: A, B, C or D.

Individual switches are identified by the name of their switchbank, followed by a colon and the number of the switch, as "A:8," which identifies the eighth switch on switchbank A.

Following the description of switchbank D is a set of flowcharts that identify which switches must be set for each interface type (parallel or serial asynchronous), and the order in which they are recommended to be set.

Note: The switchbank illustrations in this chapter are not intended to represent recommended switch settings, unless otherwise noted in the text.

Switchbank A

Switchbank A controls the following functions:

- whether parallel or serial interface is being used
- what will be the format of the line endings and command terminators
- whether the data is 7- or 8-bit
- how the data is encoded
- whether a customized code mapping cartridge is installed

Switch A:1

Parallel or serial interface

Switch A:1 tells the printer whether the interface is parallel or serial:

Switch 1	Effect
OFF	**Parallel interface** is assumed.
ON	**Serial interface** is assumed.

Switch A:2

4045 (2700) or 630 mode

Switch A:2 tells the printer whether the printer mode is 4045 (2700) or 630:

Switch 2	Effect
OFF	**4045 (2700) mode** is assumed.
ON	**630 mode** is assumed.

Switches A:3 and A:4

Line ending decisions

Switches A:3 and A:4 determine the format of the line endings and command terminators that the Model 50 is to expect:

Switch 3	Switch 4	Effect
OFF	OFF	**No line-ending decisions**—The Model 50 expects a carriage return (CR) and a line feed (LF) at the end of every print line and those commands requiring a line ending. Both codes must be included in the data sent from the host.
OFF	ON	**Auto carriage return**—The Model 50 expects only a line feed (LF) for a line ending; it adds a carriage return to every line feed it sees. Only the line-feed code need be included in the data.

Switch 3	Switch 4	Effect
ON	OFF	**Auto line feed**—The Model 50 expects only a carriage return (CR) for a line ending; it adds a line feed to every carriage return it sees. Only the carriage return code need be included in the data.
ON	ON	**Auto line ending**—This setting takes effect in the 630 mode only. When this setting is in effect, the Model 50 automatically moves words that extend past the right margin to the next print line. Received line-ending codes are still honored; that is, a subsequent line-ending occuring in the middle of the line is not moved. At least one space character must be included on every string of 124 characters in the auto line ending mode or a display code indicating buffer overflow appears.

Switch A:5

7- or 8-bit data

Switch A:5 tells the printer whether incoming data is in 7- or 8-bit format:

Switch 5	Effect
OFF	**7-bit data** is assumed.
ON	**8-bit data** is assumed.

Note: See switchbank D, switches 2 and 3, for additional applications of this switch in the serial asynchronous mode.

Switches A:6 and A:7

Data encoding

Switches A:6 and A:7 tell the printer how incoming data is encoded:

Switch 6	Switch 7	Effect
OFF	OFF	**ISO 6937**—All incoming character codes are mapped directly to the ISO 6937 assignment table.
OFF	ON	**EBCDIC**—All incoming character codes are mapped to their EBCDIC equivalent. The EBCDIC tables are provided in appendix B.
ON	OFF	**ASCII**—All incoming character codes are mapped to their ASCII equivalent. The ASCII tables are provided in appendix B.
ON	ON	**IBM PC**—Incoming character codes are mapped to the IBM PC table, provided in appendix B. The effect on non-PC fonts is described in Table 2-5, chapter 2.

Switch A:8

Custom cartridge

Switch A:8 tells the printer whether a customized mapping table in an installed cartridge is to be used by default:

Switch 8	Effect
OFF	Indicates that the custom cartridge table is **not** to be used.
ON	Indicates that the custom cartridge table **is** to be used.

Switchbank B

Switchbank B controls the following functions:

— the national character mapping to be employed

— whether or not status sheets are to be printed

— whether the chime is to sound

— which font is to be the default font

Switches B:1 through B:4

Language (national character mapping)

Switches B:1 through B:4 tell the printer which national character mapping is to be used as the default mapping, and the language in which to print the status and configuration sheets.

Switch 1	Switch 2	Switch 3	Switch 4	Language
ON	ON	ON	ON	U.S. English
ON	ON	ON	OFF	U.K. English
ON	ON	OFF	ON	French
ON	ON	OFF	OFF	Dutch
ON	OFF	ON	ON	Spanish
ON	OFF	ON	OFF	Italian
ON	OFF	OFF	ON	Danish
ON	OFF	OFF	OFF	Norwegian
OFF	ON	ON	ON	Finnish
OFF	ON	ON	OFF	German
OFF	ON	OFF	ON	Swedish
OFF	ON	OFF	OFF	Belgian
OFF	OFF	ON	ON	French Canadian
OFF	OFF	ON	OFF	Portuguese
OFF	OFF	OFF	ON	Latin America

The mapping for each language is provided in appendix B.

If all switches are off, U.S. English is used as the default mapping, but the message "Language Error" is printed on the configuration sheet.

Switch B:5

Status sheet

Switch B:5 tells the printer whether status sheets are to be printed automatically to report job errors.

Switch 5	Effect
OFF	Indicates that status sheets are not to be printed, *unless specifically requested in the job command*. A status sheet is printed, however, at startup, if the Model 50 finds certain types of faults in the system. The fault is indicated by an error code on the status sheet.
ON	Indicates that status sheets are to be printed at job boundaries when errors occur.

Switch B:6

Chime

Switch B:6 tells the printer whether the chime is to sound when a BEL code (ASCII X'07', EBCDIC X'2F') is received, or when a fault condition (out of paper, etc.) is detected:

Switch 6	Effect
OFF	Indicates that the chime is never to sound.
ON	Indicates that the chime is to sound in response to a BEL code, or to signal a fault condition.

Switches B:7 and B:8

Default font

Switches B:7 and B:8 determine which font is to be used as the default font (all data, except status sheets and data monitor output, is printed in the default font unless other fonts are requested by command):

Switch 7	Switch 8	Effect
OFF	OFF	**Resident portrait**—This setting selects the resident portrait font (Titan10iso-P for U.S./Canada machines and Titan12iso-P for international machines).
OFF	ON	**Resident landscape**—This setting selects the resident landscape font (XCP14iso-L for U.S./Canada machines and XCP12.5iso-L for international machines).
ON	OFF	**Cartridge #1**—This setting selects the first font in the cartridge installed in the first font slot (second slot from left).
ON	ON	**Cartridge #2**—This setting selects the first font in the cartridge installed in the second font slot (third slot from left).

Switchbank C

Switchbank C controls various communiations options and is used with the **serial asynchronous interface only**. The following options may be selected:

– The communications protocol, if any, to be employed:

XON/XOFF (DC1/DC3)
ETX/ACK
Printer ready (DTR)

— the baud rate

— the auto-disconnect mode

Communications protocols

Any, or none, of the available communications protocols may be in effect. Details about each protocol are provided in chapter 3 of this manual.

Switch C:1

XON/XOFF

Switch C:1 controls the use of the XON/XOFF protocol:

Switch 1	Effect
OFF	Indicates that XON/XOFF protocol is **not** used.
ON	Indicates that XON/XOFF protocol **is** used.

Switch C:2

ETX/ACK

Switch C:2 controls the use of the ETX/ACK protocol:

Switch 2	Effect
OFF	Indicates that ETX/ACK protocol is **not** used.
ON	Indicates that ETX/ACK protocol **is** used.

Switch C:3

Printer ready (DTR)

Switch C:3 controls the use of the Printer Ready protocol:

Switch 3	Effect
OFF	Indicates that Printer Ready protocol is **not** used.
ON	Indicates that Printer Ready protocol **is** used.

Switches C:4 through C:6

Baud rate

Switches C:4, C:5, and C:6 control the rate at which data is transmitted and received.

	Switch		
4	5	6	Baud rate
OFF	OFF	ON	300
OFF	ON	OFF	600
OFF	ON	ON	1200
ON	OFF	OFF	2400
ON	OFF	ON	4800
ON	ON	OFF	9600
ON	ON	ON	19200

If all switches are set to off, 300 baud is used and an error message is printed on the configuration sheet.

Switch C:7 is unassigned and does not affect printer operation.

Switch C:8

The effect of switch C:8 depends on the type of interface installed. With a parallel interface, the metered data mode can be enabled. With a serial interface, the auto-disconnect mode can be enabled.

Auto-disconnect mode (serial)

In the auto-disconnect mode, the interface is disconnected from the host under the following conditions: 1. The communication line, after a period of activity, becomes inactive for more than 30 seconds. 2. The modem connected to the printer negates a DataSetReady (DSR) it previously affirmed.

Switch 8	Effect (serial)
OFF	Indicates that auto-disconnect mode is not used.
ON	Indicates that auto-disconnect mode is used.

Metered data mode (parallel)

The metered data mode should be selected when the Model 50 is connected to a personal computer. Printer time-out messages do not appear on the screen. Therefore, this mode should not be used if the host needs an immediate report on the state of the printer. Metered data mode also should not be used when the printer is connected to a protocol converter, such as the 274 Interface Controller or the 280 Interface Sharing Device.

Switch 8	Effect (serial)
OFF	Indicates metered data is not used.
ON	Indicates metered data is used.

When switch C:8 is enabled, either "Auto-Disconnect" or "Metered Data" is reported on the configuration sheet.

Switchbank D

The switches on switchbank D control a variety of options; their effect on printer operation depends on some of the choices made on the other switchbanks. Several switches on switchbank D have different meanings, depending on the interface or encoding employed.

Switch D:1

Centronics or Dataproducts (parallel) interface

This switch only has an effect when the parallel interface has been chosen with switch A:1:

Switch 1	Effect
OFF	Indicates that the **Centronics** interface is used.
ON	Indicates that the **Dataproducts** interface is used.

Switches D:2 and D:3 (serial async)

Parity and parity type

These switches work in combination to determine if parity is or is not used, and if used, to select the desired parity type.

Switches D:2	D:3	Effect
OFF	OFF	7-bit/Mark Parity
OFF	ON	7-bit/Space Parity
ON	OFF	7-bit/Odd Parity
ON	ON	7-bit/Even Parity
OFF	OFF	8-bit/No Parity
OFF	ON	8-bit/No Parity
ON	OFF	8-bit/Odd Parity
ON	ON	8-bit/Even Parity

Note: If 7-bit parity is selected, switch A:5 should be set to Off. If 8-bit parity is selected, A:5 should be set to On.

Switch D:2 (parallel)

Inverted data polarity

This switch inverts host data polarity.

Switch D:2	Effect
OFF	Indicates that host data polarity is not inverted.
ON	Indicates that host data polarity is inverted.

Switch D:3 (parallel)

Vertical formatting

Switch D:3	Effect
OFF	Indicates that VFU emulation is not to be used.
ON	Indicates that VFU emulation is to be used.

Note: Switch A:1 sets the Model 50 for parallel or serial interface. (See the explanation on the previous pages of this chapter.)

Switch D:4

Electronic spacing (630 mode) or EBCDIC environment (4045 (2700) mode)

The effect of this switch depends on whether 630 or 4045 (2700) printer mode has been selected by switch A:2. With the 630 mode, this switch controls whether character spacing shall be determined by switch setting (see below) and commands, or by the character widths contained in the fonts used for printing. When the 4045 (2700) mode and EBCDIC encoding (switches A:6 and A:7) are selected, this switch tells the printer whether the environment is a DSC type or an SNA type.

Switch 4	Mode	Effect
OFF	4045	EBCDIC environment is **SNA**.
	630	Character spacing is determined by switch setting and spacing commands.
ON	4045	EBCDIC environment is **DSC**.
	630	Character spacing is determined by parameters stored in digitized fonts. Switches D:5 and D:6 have no effect. HMI commands may be applied to the space character, and VMI and Offset Selection still take effect.

Switches D:5 and D:6

Character spacing (630 mode)

Switches D:5 and D:6 determine the default character spacing under the following conditions:

1. 630 mode is selected (switch A:2 is set to ON), and
2. Electronic spacing is not used (switch D:4 is set to OFF).

These switches have no effect if electronic spacing is set to ON, and the switch settings can be overriden by 630 spacing commands. However, the fonts used have **no** effect on character spacing when these switches are in effect.

Switch 5	Switch 6	Effect
OFF	OFF	**Proportional spacing** is the default character spacing.
OFF	ON	**10 characters per inch** is the default spacing.
ON	OFF	**12 characters per inch** is the default spacing.
ON	ON	**15 characters per inch** is the default spacing.

Switch D:7

Horizontal tabs (630 mode)

This switch only has an effect when the 630 mode has been chosen with switch A:2. It determines whether default horizontal tab settings are used:

Switch 7	Effect
OFF	Indicates no default tabs; all tabs must be set by 630 commands
ON	Indicates default tabs are to be set at 8-character intervals.

Switch D:8

Rotate downloaded fonts

Switch D:8 rotates the orientation of downloaded fonts. For example, a portrait font becomes a landscape font.

Switch 8	Effect
ON	Fonts rotated Portrait loaded as landscape Landscape loaded as portrait
OFF	Fonts not rotated Portrait loaded as portrait Landscape loaded as landscape

Default switch settings

Your Model 50 configuration cartridge comes with its switches preset to standard parallel centronics configurations. The tables on the following pages outline the standard switch settings.

Table 4-1 **Parallel Centronics default switch settings**

Switchbank A		Switchbank C	
Switch	**Setting**	**Switch**	**Setting**
1	OFF	1	OFF
2	OFF	2	OFF
3	OFF	3	OFF
4	OFF	4	OFF
5	OFF	5	OFF
6	**ON**	6	OFF
7	OFF	7	OFF
8	OFF	8	**ON**

Switchbank B		Switchbank D	
Switch	**Setting**	**Switch**	**Setting**
1	**ON**	1	OFF
2	**ON**	2	OFF
3	**ON**	3	OFF
4	**ON**	4	OFF
5	**ON**	5	OFF
6	**ON**	6	OFF
7	OFF	7	OFF
8	OFF	8	OFF

The parallel Centronics default switch settings assume:

- 2700 printer mode
- 7-bit data
- ASCII encoding
- U.S. English character mapping
- Status sheet and chime enabled

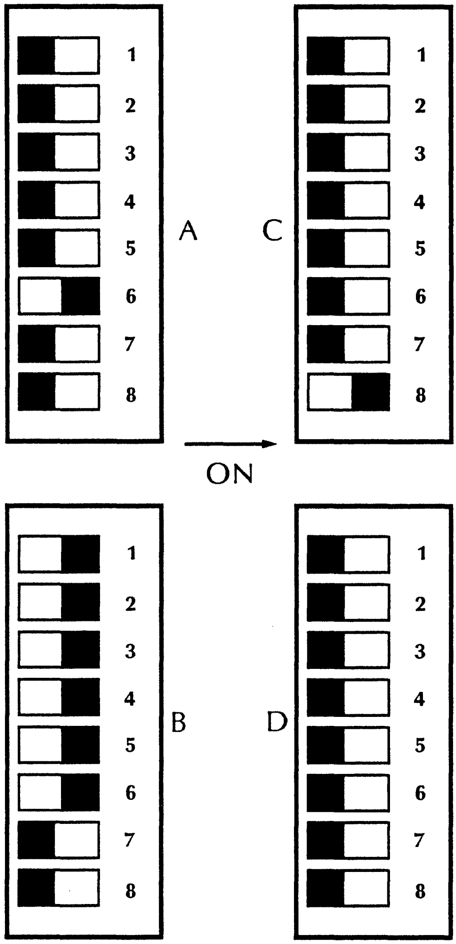


Figure 4-1 **Parallel Centronics default switch settings**

Table 4-2 **Parallel Dataproducts default switch settings**

Switchbank A		Switchbank C	
Switch	**Setting**	**Switch**	**Setting**
1	OFF	1	OFF
2	OFF	2	OFF
3	OFF	3	OFF
4	OFF	4	OFF
5	OFF	5	OFF
6	**ON**	6	OFF
7	OFF	7	OFF
8	OFF	8	OFF

Switchbank B		Switchbank D	
Switch	**Setting**	**Switch**	**Setting**
1	**ON**	1	**ON**
2	**ON**	2	OFF
3	**ON**	3	OFF
4	**ON**	4	OFF
5	**ON**	5	OFF
6	**ON**	6	OFF
7	OFF	7	OFF
8	OFF	8	OFF

The parallel Dataproducts default switch settings assume:

— 2700 printer mode
— 7-bit data
— ASCII encoding
— U.S. English character mapping
— Status sheet and chime enabled

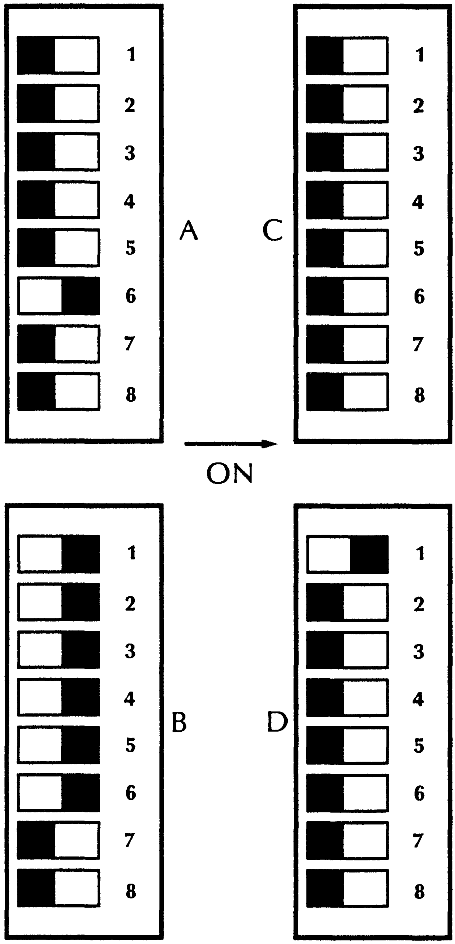


Figure 4-2 **Parallel Dataproducts default switch settings**

Table 4-3 **Serial asynchronous default switch settings**

Switchbank A		Switchbank C	
Switch	Setting	Switch	Setting
1	**ON**	1	**ON**
2	**ON**	2	OFF
3	OFF	3	OFF
4	OFF	4	**ON**
5	OFF	5	**ON**
6	**ON**	6	OFF
7	OFF	7	OFF
8	OFF	8	OFF

Switchbank B		Switchbank D	
Switch	Setting	Switch	Setting
1	**ON**	1	OFF
2	**ON**	2	**ON**
3	**ON**	3	**ON**
4	**ON**	4	OFF
5	**ON**	5	OFF
6	**ON**	6	**ON**
7	OFF	7	OFF
8	OFF	8	OFF

The serial asynchronous default switch settings assume:

— 630 printer mode
— 7-bit data
— ASCII encoding
— U.S. English character mapping
— Status sheet and chime enabled
— XON/XOFF protocol
— 9600 baud
— Even parity
— 10 characters-per-inch default spacing

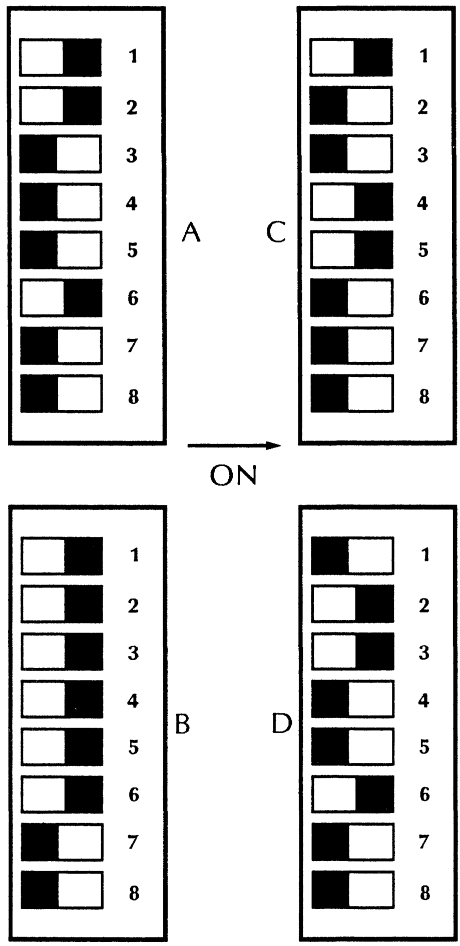


Figure 4-3 **Serial asynchronous default switch settings**

Changing the switch settings

The flowcharts on the following pages illustrate the recommended order in which switches should be set, should you wish to change the default switch settings.

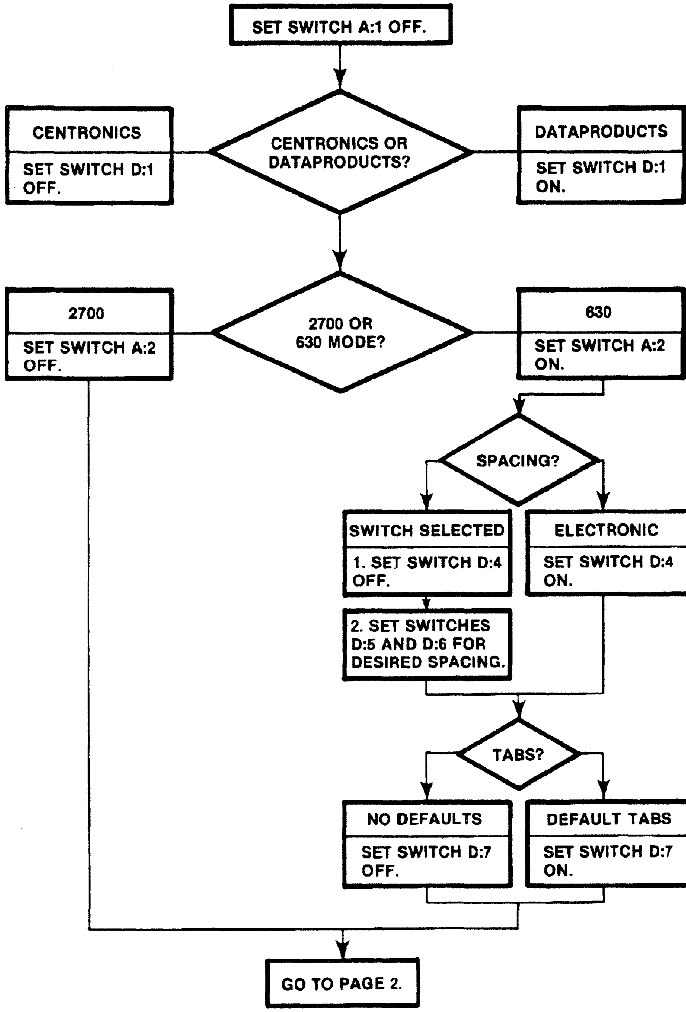


Figure 4-4 **Parallel flowchart, page 1**

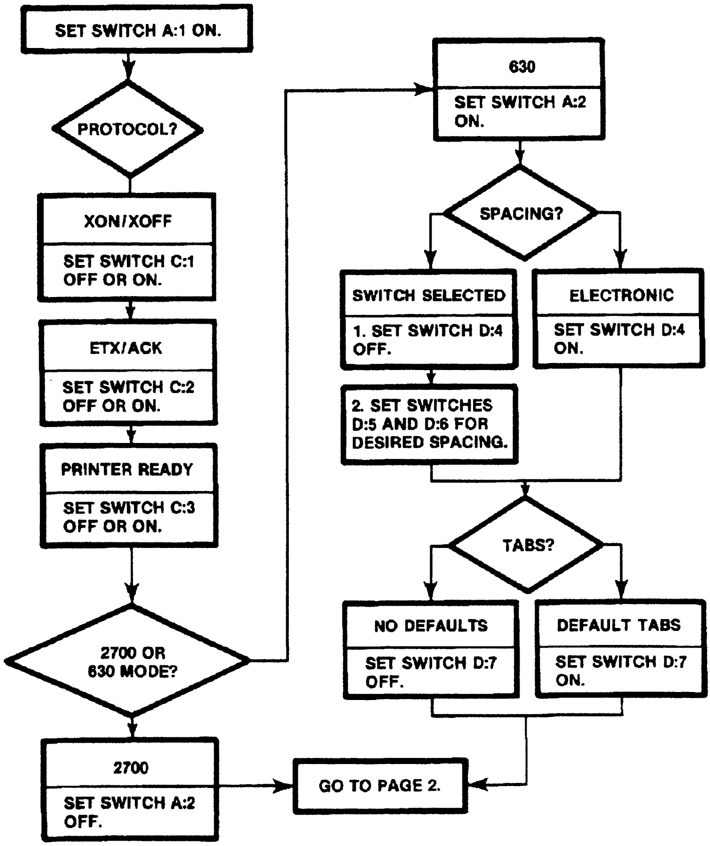


Figure 4-5 **Serial flowchart, page 1**

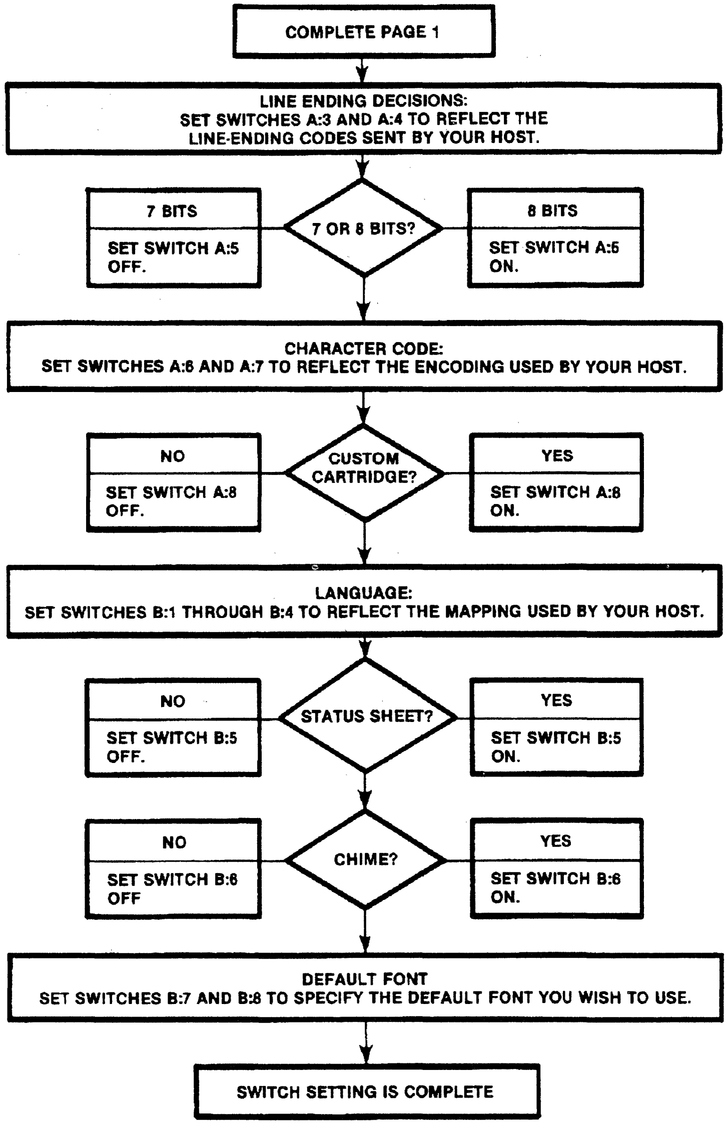


Figure 4-6 **Switch setting flowchart, page 2.**

The status sheet

The status sheet is a one-page printout that provides certain information about the status of the printer and errors. If errors occur in the job, status sheets are printed before the next job, unless switch B:5 on the configuration cartridge is OFF. The status sheet is illustrated in Figure 4-7. The status sheet contains the following information.

Comment—If there are several comments within a job, the comment appearing in the last (Esc)+ command is printed at the end of a job. A comment may contain up to 132 characters.

Font Bytes Available—The number of bytes of RAM currently available for downloaded fonts is printed. (A MAXIMUM of 440 Kbytes may be used for downloaded fonts.)

System Bytes Available—This number gives the total bytes currently available on the Model 50 for either graphics or fonts. Although fonts are limited to a maximum of 440 Kbytes, graphics may use the entire memory space listed for "System Bytes Available." When fonts are downloaded to the Model 50, the numbers listed for both "Font Bytes Available" and "System Bytes Available" change automatically to reflect the current memory available.

 # job errors—The number of errors detected in received commands, and some printing errors. Up to 10 errors are printed in the following format:

Page *xx* : *yy*

where:

xx	is the number of the page in which the error occured.
yy	is a two-digit code describing the error. The codes and their meanings are described in chapter 5 and Table A-9 in appendix A.

Fonts—The first 63 fonts available in the system are listed, in the order in which they are installed, for cartridge fonts, or the order in which they were downloaded. Resident, cartridge, and downloaded fonts are listed separately. For cartridge fonts, the number of the cartridge, as the cartridges are viewed from left to right, is listed. The default font, as selected by switch setting (B:7 and B:8) is printed in bold.

A status sheet is printed automatically on power-up if the Model 50 detects a checksum error in a font cartridge; the error code "49" is printed on the status sheet. Normally this indicates that the cartridge must be replaced.

```
                         STATUS SHEET


 1 Pages    448496 Font Bytes Available
            972799 System Bytes Available

FONTS:

  Resident:
    Titan10iso-P
    XCP14iso-L

  Cartridge:
    Univers10Iiso-P
    Univers10Iiso-L
    Univers10iso-P
    Univers10iso-L
    Univers14Iiso-P
    Univers14Iiso-L
```

Figure 4-7 **Sample status sheet**

Status sheets are printed at job boundaries (the next (ESC)+ command) whenever a comma follows the command; **or** if switch B:5 is enabled, status sheets are printed at job boundaries when an error occurs.

The configuration sheets

A configuration sheet, illustrated in the figure on the next page, is printed automatically on power-up, and when a soft reset is performed. The configuration sheet provides information about the printer as configured by the switch settings on the configuration cartridge, as well as font information.

A configuration sheet may also be printed by performing a soft reset (holding the Reset switch while pressing the Off-line switch). The soft reset configuration sheet is slightly different, and is described separately.

The start-up configuration sheet

The start-up configuration sheet includes the information described in the following paragraphs.

Revision #—The revision number of the installed software.

Hex # and switchbank diagrams—The hex # gives the hexadecimal value of the switch positions in each switchbank, wherein an on-position is a 1 and an off-position is a 0. Thus, four two-character values are given. The first value represents the switch positions in switchbank A, the second, those in switchbank B, and so on. The switch positions are also represented pictorially.

For example, note, in Figure 4-5, that the first hexadecimal value printed is X'0C'. The binary value of X'0C' is 00001100, which here represents switch positions of OFF, OFF, OFF, OFF, ON, ON OFF, and OFF, as illustrated in the switchbank diagram. The hexadecimal values of the switch setting are useful to a remote diagnostician, who may not be able to actually see the switchbanks.

CONFIGURATION SHEET

Revision # 3.0- 1.1-2.0

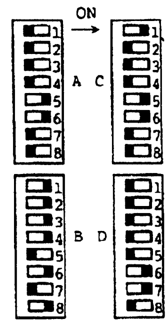
Hex # 0C F5 8C 06

Parallel
Centronics
8 Bit Data

1 = ▭ = ON
0 = ▭ = OFF

4045(2700) Mode

ASCII
U.S. English
Chime



448496 Font Bytes Available
972799 System Bytes Available

FONTS:

Resident:	**Rev.**
Titan10iso-P	B
XCP14iso-L	**4**
Cartridge:	**Rev.**
1 Univers10Iiso-P	1
1 Univers10Iiso-L	1
1 Univers10iso-P	1
1 Univers10iso-L	1
2 Univers14Iiso-P	1
2 Univers14Iiso-L	1

Figure 4-8 **Sample start-up configuration sheet**

Interface Data—The interface options selected by the switch settings are printed below the revision number.

Printer Mode Data—The selected printer mode and its associated options are printed below the interface data.

Other Options—The other options selected by switch setting are printed to the right of the printer mode data.

Font Bytes Available—The number of bytes of RAM currently available for downloaded fonts is printed. (A maximum of 440 Kbytes may be used for downloaded fonts.)

System Bytes Available—This number gives the total bytes currently available on the Model 50 for either fonts or graphics. Although fonts are limited to a maximum of 440 Kbytes, the entire memory space listed for "System Bytes Available" may be taken up by graphics. When fonts are downloaded to the Model 50; the numbers listed for both "Font Bytes Available" and "System Bytes Available" change automatically to reflect the current memory availability.

Fonts—All the fonts available in the system and their revision levels are listed. Resident, downloaded, and cartridge are listed separately. Cartridge fonts are listed in their left-to-right order. The default font, as selected by switch setting (B:7 and B:8), is printed in bold.

The reset configuration sheet

The reset configuration sheet is printed by holding the Reset switch while pressing the Off-line switch. This action also erases all data from memory, except downloaded fonts. In addition to the information printed on the start-up configuration sheet, the reset configuration sheet includes the information described in the following paragraphs.

Two solid bars across the top of the page provide evidence as to the condition of the imaging components of the printing module.

Bytes—The number of bytes of RAM occupied by each downloaded font.

CONFIGURATION SHEET

Revision # 3.0- 1.1-2.0

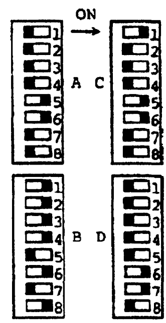
Hex # 0C F5 8C 06

Parallel
Centronics
8 Bit Data

1 = = ON
0 = = OFF

4045(2700) Mode

ASCII
U.S. English
Chime



448496 Font Bytes Available
972799 System Bytes Available

FONTS:

Resident:	**Rev.**
Titan10iso-P	8
XCP14iso-L	4

Cartridge:		**Rev.**
1	Univers10Iiso-P	1
1	Univers10Iiso-L	1
1	Univers10iso-P	1
1	Univers10iso-L	1
2	Univers14Iisc-P	1
2	Universi4Iiso-L	1

Account # ____________ Serial # ____________

Date ____________ Meter ____________

Figure 4-9 **Sample reset configuration sheet**

Fonts—In addition to resident and cartridge fonts, the reset configuration sheet lists all downloaded fonts present in the system, in the order in which they were downloaded. The default font, as selected by switch setting (B:7 and B:8), is printed in bold.

Accounting Information—Blanks are printed to fill in the following information:

Your account number
The serial number of your machine
The date
The meter reading (page count)

This information is useful when a service call is required.

5. Problem solving

The purpose of this chapter is to suggest ways of identifying and solving printing problems. Three sources of error are considered:

— the printer
— the interface or the communication line
— the data

Three steps are proposed to resolving apparent problems:

- Identifying the source of error
- Identifying the nature of the error
- Correcting the error

Finally, a special class of errors is considered, where, although the system may be configured correctly and there are no errors in the data file, the printer may not be able to image the data.

Identifying the source of the error

Step 1: Does the printer print? Turn power to the printer off, then on again.

If a configuration sheet prints, the Model 50 is functioning as it should. Continue with step 2.

If a status sheet also prints, there is a problem with a font cartridge. In this case, the font cartridge must be replaced.

If the configuration sheet does not print, it may be that a Model 50 self-test failed. In this case, a two-digit code is displayed in the display window. Note this code and report it to your service organization. The meanings of display codes are provided in appendix A.

If the printer does not power up, report this to your service organization.

Step 2: Does a file sent from the host print? Send a plain-text file (that is, one without printing commands embedded in it) to the Model 50. (If your input file is less than one page long, you may have to press the Last Page button to print the data. Wait about 20 to 30 seconds for the Last Page indicator to light up before pressing.)

If the file prints correctly, in the default font, with all the characters in the original file, you may be reasonably sure that the data the printer is seeing is the same as the data the host is sending, and that the interface or communication line is functioning correctly. Continue with step 3.

If garbled or no data prints (or if the Last Page indicator does not come on), there are two possible sources of error:

- there is an interface or communications problem, or
- the switches on the configuration cartridge may be set incorrectly.

Means for identifying the nature of such errors are described in the next subsection.

Step 3: Does a printed page look as it should? Send a file containing 2700 or 630 printer commands to the Model 50. (If your input file is less than one page long, you may have to press the Last Page button to print the data. Wait about 20 to 30 seconds for the Last Page indicator to light up before pressing.)

If your page looks as it should, your printer and interface are functioning properly.

If your information prints correctly, but doesn't look right, you may have:

- made a command error
- forgotten to load a font
- mismatched printer options (i.e., switch settings) against your host's formatting

Command errors may be identified by looking at your input file or by using the Data Monitor, which is described in the next subsection. You can determine which fonts are in the system by printing a configuration sheet or a status sheet. Procedures for checking your switch settings are described in the next subsection.

Identifying the nature of the error

If you follow the three steps outlined in the last subsection, you should be able to identify the source of most problems. There are three possibilities:

- the printer itself
- the host-printer interface (this includes several elements)
- the data stream

Various approaches may be taken to identify the nature of problems once the source has been identified. A few approaches to analyze each of the above types of problems are discussed later, after the following description of some analytical tools provided with the Model 50 itself.

Analytical tools

The Model 50 provides three major analytical tools for analyzing printing problems:

— display codes

— error messages on configuration and status sheets

— the Data Monitor

Display codes

The control panel contains a two-digit LED display, controlled by the printer module, to signal a variety of operational states and fault conditions. The following table lists the codes and their meanings.

Table 5-1A **Display codes**

Code	Meaning
The following group of codes report user-remediable conditions. Printing is halted until the condition is corrected:	
1L-9L	Top cover is not seated properly
A1	Paper misfeed in optional copier
C1	Paper pick-up cams misaligned
C3	Paper tray removed
C4	Paper path problem
E2	Paper path problem
E3	Paper path problem
E4	Paper path problem
E5	Cover open
The following group of codes report temporary operational states:	
A2	Page in optional copier waiting to feed
A3	Copy attempted with envelope tray in place. Replace with regular tray.
AA	Printer is in the data monitor mode
Ld	Printer is receiving graphic data
LF	Printer is loading font data or a constant page
PE	Parity error (serial interface). An error character is printed. Pushing the Reset switch clears the code from the display.
The following group of codes report fault conditions that halt printing and may be user-remediable. See Operating the Model 50 for recommended actions:	
L2	Scanning failure
L3	Scanning failure
L5	Laser beam off
P1	Cartridge problem
P5	Overrun error (seral interface)
P6	Overrun error (serial interface)

Table 5-1B **Display codes (continued)**

Code	Meaning
The following group of codes report fault conditions that halt printing and require repair by your service organization:	
Blank	If the display remains blank after power-up and no configuration sheet is printed, there is a fatal failure in the processing portions of memory.
d3-d9	Processing module failure

Error messages

The configuration and status sheets provide a number of messages (configuration sheet) and error codes (status sheet) that point to apparent printer problems or problems in the data stream.

When the additional memory option is installed, the configuration sheet prints at power-up as described in chapter 4. However, if additional memory is not installed, or if the printer is unable to read it, an additional message is printed: "No Optional Memory" next to the "Bytes Available" message. If you have purchased optional memory and this message appears, call your service organization.

If, upon power-up, the printer detects a checksum error in one of the font cartridges, a status sheet is printed in addition to the configuration sheet, with a message identifying the problem.

If the printing of job status sheets has been enabled by the setting of switch B:5, status sheets are printed automatically if any error is detected in a job.

Up to nine specific errors are identified; if more than nine are detected, a tenth code, 26, is also printed. The following table lists the error codes and their meanings. An examination of the input file will usually reveal the cause of these error codes. If not, the Data Monitor is also available, described in the next subsection.

Table 5-2A **Error codes**

Code	Meaning
The following code reports an error associated with a command:	
02	An (Esc)+ was received in the previous job, followed by an invalid character; all codes up to the next line-end were ignored.
The following codes report errors associated with the loading of font data:	
11	A problem was detected within font data. The font load was abandoned.
12	Available font memory was exhausted. Unloadable fonts were ignored (only whole fonts were stored).
18	A Font Unload command was received, which named a font not in storage. The command was ignored.
19	A Font Unload command was received, which named a resident font. The command was ignored.
The following codes report errors associated with print data:	
20	An escape code (or UDK) followed by an invalid character was received. The apparent instruction was ignored.
21	This page could not be printed, due to its complexity. That is, the band buffers were overloaded: either too much data on the page, or too complex images.
22	A print line extended beyond the right margin. Possible data loss.
23	Character-per-page limit exceeded (approximately 15,000); excess data printed on next page.
25	Possible duplicate page.
26	More than nine errors were detected in the previous job.
28	An error was detected in a table download; the download was abandoned.
29	A parameter within a command was illegal or invalid; the command was ignored or may have produced unexpected results.

Table 5-2B **Error codes (continued)**

Code	Meaning
The following codes report errors associated with font usage:	
40	A Font ID Assignment attempted to assign an ID to a font not in storage. The command was ignored.
41	During page composition, a font was invoked whose orientation was opposite that of the page being composed. A new page was started. **Note:** The above does not apply to merge-pages.
42	A Font Change command was received, whose ID number had not been assigned. The command was ignored.
45	A Font ID Assignment contained an ID number greater than 9. The assignment was ignored.
49	A cartridge checksum error was found.
The following codes report errors associated with the use of graphic data:	
80	Graphic could not be printed beyond the physical limits of the page.
81	The S_x value was not an even multiple of 8; S_x was rounded down.
82	The graphic command was incomplete; a partial image may have been printed.
83	The sixel-encoded data contained a repetition value in excess of 32,767. All the data was ignored.
84	The Repeat Graphic Window command was received when no graphic window was defined.
85	Only a portion of the graphic data was loaded, due to insufficient storage area.

The data monitor

The data monitor is a data-analyzing tool that prints the hexadecimal value of all data received, rather than print characters, store data, or execute commands. The data monitor may be used to:

- check for correct command syntax
- check the integrity of data files
- check the integrity of font and graphics data

Figures 5-1A and 5-1B show an example of a data monitor printout and a description of its parts.

```
P-001

0    2A 2B 44 3D 55 44 4B 3D 7E 7E 2B 33 42 6F 6C 64 50 53 69 73
     6F 2D 50 0D 0A 7E 33 7E 7A 6C 31 54 68 69 73 20 69 73 20 61
     6E 20 65 78 61 6D 70 6C 65 20 6F 66 20 61 20 64 61 74 61 20
     6D 6F 6E 69 74 6F 72 20 70 72 69 6E 74 6F 75 74 2E 20 20 49
     6E 63 6C 75 64 65 64 20 69 6E 20 0D 0A 74 68 69 73 20 66 69

100  6C 65 20 61 72 65 3A 0D 0A 0D 0A 09 61 20 55 44 4B 20 73 65
     71 75 65 6E 63 65 0D 0A 09 61 20 46 6F 6E 74 20 49 44 20 41
     73 73 69 67 6E 6D 65 6E 74 0D 0A 09 61 20 46 6F 6E 74 20 43
     68 61 6E 67 65 20 69 6E 73 74 72 75 63 74 69 6F 6E 0D 0A 09
     61 20 4C 61 6E 67 75 61 67 65 20 69 6E 73 74 72 75 63 74 69

200  6F 6E 0D 0A 09 61 20 52 65 73 65 74 20 63 6F 6D 6D 61 6E 64
     0D 0A 0D 0A 45 4E 44 20 4F 46 20 46 49 4C 45 2A 2B 58
```

Figure 5-1A **Data monitor printout**

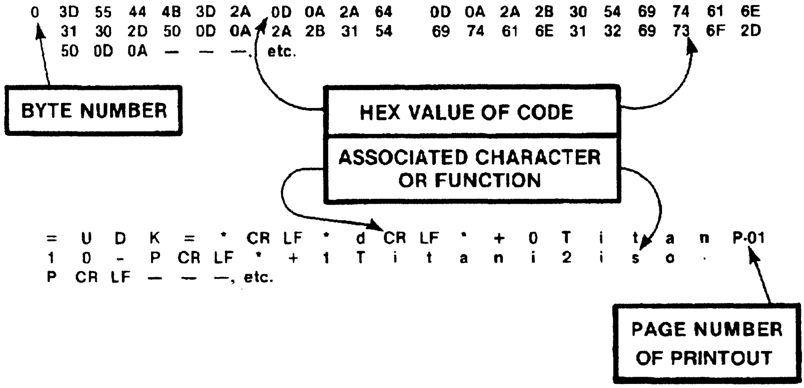


Figure 5-1B **Data monitor printout (continued)**

Note the following about the data monitor printout:

The characters printed on the right half of the printout are either ISO or EBCDIC characters. The data monitor always uses the translation table shown in Table B-28 for EBCDIC data, regardless of the language selected. If ASCII or IBM PC encoding is selected, ISO characters are printed.

The **Page number** refers to the pages of the printout, not of the transmitted job.

Invoking and canceling the data monitor

The data monitor is invoked in one of two ways:

1. The Data Monitor command may precede the data to be analyzed. In this case, the Reset command may be sent to cancel the data monitor; however, if the Data Monitor command is introduced using a surrogate escape character (UDK), the Reset command must be introduced with the same UDK character as the Data Monitor command.
2. The data monitor mode can also be entered by turning the printer off, holding down the Off-line switch, and turning the printer on again. In

this case, the data monitor mode can be canceled by turning the printer off and on again, or by performing a soft reset.

Fonts and graphic data

The data monitor may be used to validate the integrity of downloaded fonts and graphic data. Some hosts may insert unwanted codes into the data before transmission, or perform other transformations on the files that may render them unprintable. The following are items to watch for:

- **Font or graphic files should never contain codes whose value is less than X'30'.** All such files must be sixel-encoded before transmission, and are not unpacked in the data monitor mode.
- **The beginning of a font file can be recognized in a data monitor printout** by looking for:
  - ASCII-encoded fonts: two bytes of '69'
  - EBCDIC-encoded fonts: two bytes of '89'

  The end of a font file can be recognized in a data monitor printout by looking for a series of at least 10 consecutive bytes of the same value, depending on the encoding of the font. The actual number of consecutive bytes varies from font to font. The ending values are:
  - ASCII-encoded fonts: '54'
  - EBCDIC-encoded fonts: 'E3'
- **The beginning of a graphic file can be recognized** by looking for the terminator of the Graphic Window command. The end of a graphic file is harder to determine. *If the graphic file actually contains the number of bytes specified in the Graphic Window command,* the end of the file can be determined by multiplying the graphic width by the number of scan lines, and then using the byte numbers on the printout to locate it. If there is a question as to the length of the file, it is best to send the graphic data by itself with no data following.

If the graphic file includes packed data, the correct length of the file may be determined as follows:

1. Determine the number of bytes in the packed file, using the byte numbers on the printout.
2. Locate the run-length expressions; for example:

   31 30 30 45

   which means "100 bytes of X'45'," and replace the number of bytes in the expression (in this case, 4) with the value they represent (100). The effect of this calculation is to add the difference (96 in this example) to the total number of bytes on the printout.
3. After all run-length expressions have been located and replaced, the total number of bytes in the file should equal the product of S_x and S_y in the Graphic Window command, divided by eight.

Hints for data monitor use

The following steps are recommended as an aid to using the data monitor effectively.

1. ISO, ASCII, IBM PC, and EBCDIC coding schemes contain control codes that produce no effect in the Model 50. A data monitor printout can reveal the presence of these codes that may be designed to invoke functions not supported by the Model 50. The following codes are **not** supported:

ISO, ASCII, or IBM PC*:	01 02 04 05 10 16 17 18 1C
EBCDIC:	01 02 04 06 07 08 09 0A 10 17 18 1A 1B 1C 20 21 22 23 24 26 28 29 2A 2B 2C 2D 30 31 32 33 34 36 37 38 39 3A 3B 3C 3E

*These characters are printable in IBM PC code set if selected by <SO> or <GS>.

1. Identify the line-endings in the printout. Depending on the line-ending switch settings, the possible combinations are:

Codes	ASCII values	EBCDIC values
CR LF	0D 0A	0D 25
CR	0D	0D
LF	0A	25

2. Identify the surrogate escape character. The =UDK= sequence may be identified as follows:

   ASCII: 3D 55 44 4B 3D
   EBCDIC: 7E E4 C4 D2 7E

4. Study the job's status sheet, if one has been printed, for command errors.

5. Observe whether the commands and instructions in the file have been properly specified. The *Creating documents* section of the *User Manual* and the appendix in this manual provide the proper code sequences.

6. Upon identification of possible problems, modify the input file and retransmit.

Identifying printer problems

As described at the beginning of this chapter, if a configuration sheet prints at power-up, the Model 50 is functioning as it should. If the Model 50 detects a problem within itself during warm-up, it may display an appropriate code in the display window. Some conditions reported in the display window are user-remediable; see the *Operating the Model 50* portion of the *User Manual* for recommended actions.

Some areas of the printer that are not user-serviceable may encounter problems that can be corrected by powering the printer off and then on again. Again, these are listed in *Operating the Model 50.* If this action does not clear the problem, a call to your service organization is required.

Some conditions cannot be resolved by user action and a call to your service organization is always required. These conditions are listed in Table 5-1 in this chapter and in Table A-8 in appendix A.

Identifying host-printer interface problems

This is an area that comprises several interrelated components that have to work together to accomplish a single task. Essentially, the task is to get the data from the host to the printer in such a way that the printer sees the same data that the host is sending. This one task may be broken down into several smaller tasks, depending on the configuration and computer environment. Some of the smaller tasks are:

- The host must generate the codes and send them to an output or communication port. The codes must accord with some particular coding scheme.
- The codes must travel as patterns of electrical pulses over a medium of some kind, whether a parallel cable or telephone line or microwave signal. These patterns may require modification or transformation along the way, depending on their medium of travel. Their integrity must be maintained along the way; in other words, they must be in the same form when they arrive at the printer that they were in when they left the host.
- The printer must read the codes and interpret their function. They might constitute font data, graphics, a merge-page, etc., but the printer's primary function is to print; typically, received data is assumed to be print data unless its purpose is otherwise identified.

A partial list of the system elements contributing to the performance of these tasks might include:

- the host's communicating software and supported protocols (serial configuration)
- the host's input/output routines and supported data signals (parallel configuration)
- modems (serial)
- communication lines (serial)
- cables and connectors (all configurations)
- 4045 Model 50 firmware
- the configuration cartridge

The task of identifying host-printer interface problems is complicated by the fact that all these various system elements work in conjunction with one another. Thus, a specific problem may be traceable to more than one component of the system. This section attempts to provide guidelines to follow when tracing these problems.

Notice: Certain types of problems are beyond the scope of this manual to investigate. These include:

> *Peculiarities of host software*—This includes the host's data signals, communicating software, and protocols. All those signals, codes, and protocols supported by the Model 50 are described in chapters 2 and 3.
>
> The data monitor, however, might be used to point to possible modification of application software, especially where the application assumes a non-electronic printer or other output device.
>
> *The physical host-printer interface*—As a rule, other sources of error should be investigated and ruled out before attempting to diagnose the condition of these components.

When to suspect a problem with host-printer interface

You can suspect an interface problem when

1. the Model 50 powers up and prints a configuration sheet, and
2. a plain-text file, that is, one with no 2700 or 630 printing commands, does not produce readable output on the Model 50. "Readable output" means, here, anything that contains legible printed characters, whether or not they print in the desired format.

Recommended actions

The approach to a solution recommended in this manual consists of two parts:

1. **Double-check the configuration cartridge switch settings**. This procedure is outlined in the next subsection, Correcting the error. If following this procedure fails to correct the problem, the next step is:
2. **Resend the problem file, with the printer in the Data Monitor Mode**. There are two possible results:

   Nothing prints, in which case it can be assumed that the host and printer are not "talking the same language." That is, there is a problem with the physical interface, the manner of signal transfer, or the communication protocols.

   A data monitor printout is produced, in which case the printout may be studied for command errors or erroneous codes. Further adjustment of switch settings or modification of host software might be indicated.

Either of the above actions are recommended as the primary troubleshooting tools in the rest of this chapter. If both actions fail to reveal the nature of the error, it may be time to look to the physical interface.

Identifying data problems

The data monitor is the principal analytical tool here. However, it can point to problems with any of the following:

- host application software
- input errors (command syntax, etc.)
- configuration cartridge switch settings

When to suspect a problem with the data

You can suspect an interface problem when

1. the Model 50 powers up and prints a configuration sheet, **and**
2. a plain-text file produces readable output on the Model 50, **but**
3. the printed data is not the same as the data input at the host, or does not *look* as intended, or is garbled.

Recommended actions

1. Resend the file, preceded by the data monitor command. Follow the steps suggested under Hints for data monitor use, earlier in this chapter. Look for command errors, missing commands, etc. If no fault can be found in the data file,
2. Check for indications of incorrect switch settings:
   - *Printer commands are not being honored*—If the command syntax is correct and parameters have been correctly specified, check the following switch settings:

     — printer mode (2700 or 630, switch A:2)
     — line-ending decisions (A:3 and A:4)
   - *Font characters are missing, or unexpected characters are printing*

     — 7/8 bit data (switch A:5)
     — character code (A:6 and A7)
     — language (B:1 through B:4)

- *Garbled or nonsense data is printing*
  - 7/8 bit data (switch A:5)
  - character code (A:6 and A7)
  - for the serial interface, parity (D:3 and D:4)
- *Text is not positioned properly on the page*
  - line-ending decisions (A:3 and A:4)
  - for 630 mode: electronic spacing (D:4), character spacing (D:5 and D:6), and horizontal tabs (D:7)
- *Page is printed in the wrong orientation*
  - default font (B:7 and B:8)

Correcting the error

Printer problems

The Operator guide lists recommended actions for solving printer problems indicated by codes in the display. A number of conditions can be corrected by cycling power to the printer, i.e., turning it off and then on again.

Host-printer interface problems

Setting the configuration switches:

If you have determined that there definitely is a problem in the way that the host and printer talk to each other, probably the wisest first step toward resolving the problem is to double-check your configuration cartridge switch settings. Follow the procedures outlined below.

Step 1: *If you are able to print a page, print a data monitor listing of a plain-text file.* Make sure that the file contains at least one line-ending. If you are unable to print anything, continue with step 2.

Note: At the conclusion of each step, try once again to print a plain-text file, followed by a data monitor printout of the same file.

Step 2: *Reset all switches to the minimum configuration for your interface.* The minimum configurations are listed in chapter 4 in Tables 4-1, 4-2, and 4-3. Then check the following switches and set them according to the configuration of your system:

- 7/8-bit data—switch A:5
- character coding—switches A:6 and A:7

For the serial interface only, also set:

- protocols—switches C:1, 2, and 3 (try to select the minimal protocol for your host), and
- baud rate—switches C:4, 5, and 6, and
- parity—switches D:2 and D:3

Step 3: *Perform a soft reset.* Hold down the Reset switch switch and press the Off-line switch. This must be performed every time the switches are reset, to force the Model 50 to reread the switch settings.

If all switches to this point are set correctly, you should be able to print a plain-text file. Send a plain-text file, containing at least one line-end, to the printer, and then re-send it with the printer in the data monitor mode.

Step 4: *Set the line-ending decisions switches (A:3 and A:4).* Study the data monitor printout to determine the codes your host sends for a line-end. If you haven't produced a data monitor printout, study the printed file.

If your host sends both CR and LF (X'0D' and '0A'), set the switches both to OFF. If you don't have a data monitor printout, you can assume this is the case if the second print line is printed directly below the first, at the left margin.

If your host sends LF only, set switch A:3 to ON. If you don't have a data monitor printout, you can assume this is the case if the second print line begins below and to the right of the first.

If your host sends CR only, set switch A:4 to ON. If you don't have a data monitor printout, you can assume this is the case if the second print line is printed directly over the first, at the left margin.

Perform a soft reset.

Step 5: *If you can't print a plain-text file by this point, your problem lies somewhere else in the system.* The remaining switch settings select options pertinent to character mapping and the placement of print data on the page. The following paragraphs list the remaining switch settings in the order in which they should be set.

Printer mode	Switch A:2—4045 (2700) or 630 mode. The choice of mode determines which command set the Model 50 will respond to for print placement instructions. Many hosts are equipped with 630 printer drivers, so that text files are automatically formatted with 630 printing commands. *If 630 mode is chosen,* you need to select the character placement options you wish to use by setting switches D:4 through D:7. Note that if you choose electronic spacing (D:4 ON), switches D:5 and D:6 are ignored.
Custom cartridge	Switch A:8—If you're using a custom mapping table digitized in a plug-in cartridge, you must set switch A:8 to ON. All mapping will follow the table stored in the cartridge, with the exception that status and configuration sheets will use the mapping selected by the language switches.
Language	Switches B:1 through B:4—The language chosen determines the default character mapping to be used for status and configuration sheets and all print data, unless language commands, custom cartridges, or downloaded translation tables are used.
Status sheet	Switch B:5—If this switch is ON, status sheets are printed at job boundaries if requested in a command, or when errors are detected in the data stream. If this switch is OFF, status sheets are printed only in response to a comma in a job command.
Chime	Switch B:6—This switch activates the audible tone, which sounds under the following conditions: — when the BEL code is received — when a fault condition (out of paper, out of dry imager, paper jam, etc.) is detected — when a communications error has occurred If this switch is OFF, the chime never sounds.

Default font	Switches B:7 and B:8—Your selection here determines which font the printer will use when no fonts are called out in the data. All printed pages, except status sheets, will be printed in this font, unless other fonts are requested by command. Status and configuration sheets are always printed in the resident portrait font.

By this point, you should have all the switches set for your system. To change the configuration, follow this procedure, or the flowcharts in chapter 4.

Data problems

If you have your printer running and know that your host and printer are communicating successfully, but are having problems with the look of the printed page, it is reasonable to assume that the problem lies with the data. However, there are some switch settings that could affect this area as well. Again, your most valuable analytical tool is the data monitor. Resend the job with the printer in the Data Monitor Mode, then follow these steps.

Step 1: *Check the line ending codes sent by the host against your switch settings.* Verify that the codes your host is sending are the same as what the printer is expecting. (Switches A:3 and A:4)

If your host sends both CR and LF (X'0D' and '0A'), set the switches both to OFF. If you don't have a data monitor printout, you can assume this is the case if the second print line is printed directly below the first, at the left margin.

If your host sends LF only, set switch A:3 to ON. If you don't have a data monitor printout, you can assume this is the case if the second print line begins below and to the right of the first.

If your host sends CR only, set switch A:4 to ON. If you don't have a data monitor printout, you can assume this is the case if the second print line is printed directly over the first, at the left margin.

Step 2: *Check that the language mapping you have selected (switches B:1 through B:4) corresponds to the mapping used in the host.* This is also important if you are using a custom mapping cartridge (switch A:8). This should especially be considered when many of your characters are printing correctly, but you are getting some unexpected characters. Each language mapping is charted in appendix B.

Step 3 (630): *If you are using 630 mode, check that you have set switches D:4 through D:7 properly.* These switches affect character placement in 630 mode only, but their effect is significant.

Once you have determined that all your switch settings are correct, study the data monitor, following the hints suggested under *Hints for data monitor use*, earlier in this chapter. Check, where appropriate, for:

— command syntax
— line endings
— font selection
— code mappings

A careful study of the data monitor should reveal any sources of error that exist in the data stream. Modification of either the input file or the host application software may be indicated.

Imaging errors

It may occur that, athough the switch settings are all correct and there are no command syntax errors, the Model 50 is unable to image the page. This can happen when the number of items to be printed in a given area on the page is so great that the imaging software is unable to generate the video data fast enough to transfer it to the scanner. (See the discussion of bit maps and band buffers in chapter 1.)

When this occurs, the page is only partially printed and a status sheet (if enabled by switch B:5) is printed with the error code 21 (this page could not be printed, due to its complexity). Generally, it is extremely difficult and, in effect, impractical to attempt to predict when this will occur. As a rule, there are three circumstances where this can happen:

- There are many characters to be printed, using small fonts, in a relatively small area of the page. This situation is further complicated when the data includes a large number of commands, such as font changes, bolding commands, overstrike commands, etc.
- Positioning commands are used to locate a large number of print items in a small area. Again, bolding commands, etc., complicate the imaging task of the printer.
- Overlapping graphics or the use of the 2X multiplication factor in graphics.

There are two ways to alter the data on the page, such that the page may become printable. One way is to reduce the amount of data to be printed on the given page, and the other is to reduce the amount of electronic printing commands embedded in the data. Both approaches serve to reduce the number of items to be processed by the image generator.

Data may be reduced by:

— Increasing margins

— Using larger fonts

— Increasing line spacing

— Any method that lessens character density on the page; the difference between an unprintable page and a printable one may be miniscule.

Some ways to reduce printing commands:

— Use a bold font instead of bolding commands.

— Reduce the number of font changes; this may only be necessary in a single line.

— Avoid excessive overlays, as in the use of excessive text placement commands over the same area.

— Any method that lessens "print item density" in a relatively restricted area; again, the difference between an unprintable page and a printable one may be miniscule.

6. Technical tutorial

The purpose of this chapter is to introduce some basic concepts about

- data encoding,
- data transmission, and
- electronic printing.

For the new computer user, or for a user who is unfamiliar with one of the above areas, this overview provides a background for better understanding the rest of the manual, and for the effective use of the Model 50.

The phenomenon of computer data

Computer data are the pieces of information that the computer uses to perform the tasks assigned it. The amazing thing about this data is that, in a sense, it doesn't exist! Not by itself, anyway, as the following discussion shows.

"Data," as used to describe information stored in a computer, may refer to:

- a list of names
- a set of facts about one individual
- a series of numbers
- a single number or a single letter
- a collection of numbers and letters

and the list goes on and on. Much of the data is the information you either program into the computer, or feed into it as an input file. However, the computer generates more data, as necessary, to accomplish the task you set for it. Some of this new data you never see, since it disappears when the need for it has passed, and some of it you see as some sort of output file. But in every case, the data within the computer is continually changing, *because the computer itself is changing*.

On-states and off-states

Perhaps the most basic task performed by the computer is the turning off and on of electronic components within itself. A "component" may be nothing more than a junction of two microscopic circuits built into a chip, but it shares one characteristic with every other component in the computer, and that is the ability to be either on or off.

When a component is on, it is said to be in the *on-state*; when it is off, it is said to be in the *off-state*. Usually, the on-state is represented by the number **1** and the off-state is represented by the number **0**. These states can also be described as true and false (with reference to their logical value), or high and low (with reference to their electrical state). Most of the work of the computer consists of generating on-states and off-states in vast numbers and at extremely rapid rates. This activity is made useful by the logic imposed on it, and the logic is described by the program.

Programming: firmware and software

The programs used by the computer give meaning to the on-states and off-states being generated in its components. A program may be built into the machine, because of the design of the circuitry in its chips. This kind of programming is called *firmware*, because it is a permanent part of the machine. Many computers are also designed to run other programs, designed by the personnel operating the computer. These programs are called *software*, because they exist in the machine only when they are run, or executed.

There are many ways a program might use the on-states and off-states of the computer to perform work.

The state of a single component may act as a sort of logical switch. For example, a particular component in an off-state might signal the computer to stop processing at the moment that it "reads" that component. It may be said to turn the program off. Then again, a switch to an on-state may tell the computer to resume processing at that point.

On- and off-states can assume even greater power in groups. A single component by itself can represent two values: on and off (or 1 and 0). But two elements together can have four possible values:

off and off	0 0
off and on	0 1
on and off	1 0
on and on	1 1

The larger the group, the more possible meanings each group can have. In fact, since each element of the group has exactly two possible states, on or off, the number of values that a group can assume is equal to 2^n, *n* being the number of elements in the group. In other words, a group of two elements can have 2^2 possible values, a group of three elements can have 2^3 possible values, and so on.

The number of ways in which these elements might be grouped is limited only by the imagination of the programmer and the size of the computer's memory banks. However, to achieve consistency from machine to machine, and from programmer to programmer, a number of conventions are widely observed that impose a degree of order on the manipulation of these states, and make it possible to refer to them as "data." We say that the data are **encoded**.

Encoding

Data is said to be encoded when it obeys certain conventions of organization, as described in this subsection. The conventions describe the use of data from the level of a single component state, to a byte, to whole sets of character codes.

Bits and binary expressions

The smallest unit of data, i.e., the state (on or off) of a single component at any given moment, is called a "bit," short for "binary digit." A single digit is said to be "binary" because it may exhibit one of two states.

A single bit in the on-state is nearly always represented as a one (1), while a bit in the off-state is represented as a zero (0). This does not mean that bits always represent numbers in the computer; they may represent letters or logical states as well. However, this form of representation quite obviously lends itself to expressing data, on the level of the bit, as numeric values.

The system used on the bit-level is the binary (2-based) numbering system. (On a higher level, a 16-based system is used; more on that later.) Whereas the decimal numbering system uses ten different numerals, 0 through 9, to express values, the binary system uses just two: 0 and 1. The list below illustrates comparable values in both numbering systems:

Decimal	Binary
0	000
1	001
2	010
3	011
4	100

While strings of ones and zeros represent occurrences, or streams, or groups of on- and off-states, it is a convenient and widespread practice to speak of them as having numeric value.

The next level of data organization is the *byte*.

Bytes and hexadecimal expressions

A byte is a standard grouping of bits that work, or are operated on, as a unit. Theoretically, a byte may be any length, but the most common length for a byte is eight bits, with some computers using seven. The

Model 50 can accommodate both seven- and eight-bit lengths.

The number of possible combinations of eight bits, or the number of possible values an eight-bit byte can have, is 2^8, or 256. The number of possible values for a seven-bit byte is 2^7, or 128. Note, however, that the highest numeric value an eight-bit byte can have is 255 (or 127 for seven bits). This is because the first combination in the series is always zero.

Table A-6 in appendix A lists all the possible combinations for eight bits, expressed numbers in the following numbering systems: hexadecimal, decimal, and binary. Hexadecimal notation is the most common way to express the value of a byte and is described next. Note first that the binary expressions given in Table A-6 show eight bits each. To find the expression for a seven-bit byte, simply drop the first digit in each expression.

Hexadecimal notation uses the number 16 as a base and allows the value of any byte to be expressed with only two characters. Decimal notation requires up to three characters for a byte (255), with each digit having ten possible values (0 through 9). Binary notation requires up to eight characters (11111111), with each digit having two possible values.

In hexadecimal notation, each digit represents a grouping of four bits. A group of four bits may have 2^4, or 16 possible values. Therefore, the notation system requires 16 different characters. The convention observed in this case is that the letters A through F are used to represent values greater than 9, as follows:

A = 10
B = 11
C = 12
D = 13
E = 14
F = 15

Thus, the combination "AF" refers to a number whose first digit, A, has a value of 10, and whose second digit, F, has a value of 15. As in decimal notation, the first digit is equivalent to a multiplier of the base, to which the value of the second digit is added. For example, the number 24 in decimal notation is equivalent to two times the base (10) plus four:

$(2 \times 10) + 4 = 24.$

Similarly, AF in hexadecimal notation is equivalent to A times the base (16) plus F. The value of AF in decimal notation is:

$(10 \times 16) + 15 = 175.$

The list below provides equivalent values in hexadecimal, decimal, and binary notation for each possible combination of four bits.

Hexadecimal	**Decimal**	**Binary**
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

In hexadecimal notation, the value of a byte is expressed by two digits. The first digit represents the first four bits of the byte (or the first three, for a 7-bit byte), and the second hexadecimal digit represents the last four bits, as:

Byte	**Hexadecimal expression**
0000 1111	0 F

A common convention for indicating hexadecimal values is to enclose them in single quotes, preceded by the capital letter X, a kind of abbreviation for "hexadecimal." For example, X'0C' means the hexadecimal value 0C, which is equivalent to the binary value 00001100. This convention is observed in order to prevent confusion with other numbering systems, such as decimal. Other common methods include:

H0C
Hex0C
$0C_{16}$

and still others. In this manual, the X'0C' convention is used.

"Most significant" and "least significant" digits

The first digit of a hexadecimal expression is called the "most significant" digit, while the second is called the "least significant." The reason for this is that the first digit in any numerical expression represents the greatest value. For example, the first digit in the expression "1,024" is the most significant of the expression because it represents a larger quantity, 1,000, than any of the other digits. Similarly, the "2" represents 20 and the "4" represents simply 4.

The same is true of a binary expression. In the expression "101" , for example, the first 1 represents 1×2^2, the 0 represents 0×2^1 , and the last 1 represents 1×2^0, or 1. (By convention, any number raised to the zero power is equivalent to one.)

In most computer systems, a byte of data is transmitted by sending the least significant bit first, followed by the next-least significant bit, and so on.

Codes and coding schemes

The pattern of ones and zeros represented by a particular byte-value can have different uses or meanings, depending on the device (host) that generates it and the device (in this case, printer) that receives it. The Model 50 supports a variety of applications.

Most often, each byte is used as a *code*. This is to say that each particular bit pattern represents a commonly agreed-upon function pertinent to computers and computer networks.

In most computer applications, codes are of two types: "control" codes and "character" codes. A control code represents a function, whereas a character code represents a printable character of some sort. For example, X'09', HT or horizontal tab, tells the printer to move the print position a specified distance to the right. However, X'4D' represents the capital letter "M" and tells the printer to print an "M" at the current print position.

Control functions can usually be categorized as either *communications* control functions for use with serial networks, or *device* (in this case, printer) control functions. The horizontal tab example given above is an example of a device control function. X'06', ACK, is an example of a communications control function. In some systems, the ACK code is sent from the printer to the host, to signal that the printer has received the last-sent block of data and is ready to receive further data.

The exact meaning of each code is dependent on the *coding scheme* used. Each scheme assigns different meanings to each code. The Model 50 supports four different coding schemes: ISO, ASCII, EBCDIC, and IBM PC. The ISO coding scheme is an internationally recognized code set derived by the International Organization for Standardization. This is the basic code set in the Model 50 from which the ASCII and EBCDIC schemes are derived. All supported code sets are described in chapter 2.

The ISO code set appears on the next two pages, as well as in chapter 2 and appendix B. It is reproduced here to demonstrate how to read the code tables.

Note first that the "most significant" hexadecimal values are located at the top of the chart, increasing in value from left to right. The "least significant" values are located at the left of the chart, increasing in value from top to bottom. To find the function or character associated with any hexadecimal value, it is necessary to find the first digit in the top row and the second digit in the left column. Read down the most significant column, until you reach the least significant row.

Table 6-1A **ISO encoding scheme**

Most →

Least ↓

	0	1	2	3	4	5	6	7
0	NUL	dle	Sp	0	@	P	`	p
1	soh	DC1	!	1	A	Q	a	q
2	stx	DC2	"	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	eot	DC4	¤	4	D	T	d	t
5	enq	NAK	%	5	E	U	e	u
6	ACK	syn	&	6	F	V	f	v
7	BEL	etb	'	7	G	W	g	w
8	BS	can	(	8	H	X	h	x
9	HT	EM	)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[	k	{
C	FF	fs	,	<	L	\	l	|
D	CR	GS	–	=	M	]	m	}
E	SO	RS	.	>	N	^	n	~
F	SI	US	/	?	O	_	o	del

Notes: 1. Code names in lowercase indicate codes that have no significance to the Model 50.

2. Underscored code names indicate codes that have significance **in some modes only**.

3. Font characters marked with an asterisk are present only in the resident fonts, unless specially ordered.

Table 6-1B **ISO encoding scheme (continued)**

Most →

Least ↓

	8	9	A	B	C	D	E	F
0				°		—	Ω	ĸ
1	`		¡	±	`	¹	Æ	æ
2	´		¢	²	´	®	Đ	đ
3	^		£	³	^	©	ª	ð
4	~		$	×	~	™	Ħ	ħ
5	¯		¥	µ	¯	♪		ı
6	˘		#	¶	˘		Ĳ	ĳ
7	˙		§	·	˙		Ŀ	ŀ
8	¨	¬*	¤	÷	¨		Ł	ł
9	¨	¦*	'	'	¨		Ø	ø
A	˚		"	"	˚		Œ	œ
B	¸		«	»	¸		º	ß
C	_		←	¼	_	⅛	Þ	þ
D	˝		↑	½	˝	⅜	Ŧ	ŧ
E	˛		→	¾	˛	⅝	Ŋ	ŋ
F	ˇ		↓	¿	ˇ	⅞	ŉ	

Equivalence table

Hex	Binary	Hex	Binary	Hex	Binary	Hex	Binary
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

For example, to find the meaning of code X'2C', locate column 2. The most significant value, in this case 2, is found at the top of the chart. Then read down column 2 until you reach row C; C is read from the left end of the row. At this location, you should find the comma (,). This means that, when printing text, the printer will print a comma when it receives the code X'2C', providing that the coding scheme used is ISO.

Note that the first two columns in the table do not contain printable characters, but code-names. These are the control codes supported by the ISO coding scheme. Some of them have significance to the Model 50, and some do not. Those that are meaningful are described in Table 2-1 in chapter 2.

Further details about the Model 50's supported coding schemes are provided in chapter 2. Chapter 2 also provides details about font and graphic data, in which individual bytes do not serve as codes, but as bit maps, that is, patterns of ones and zeros representing printed and unprinted dots.

The host-printer interface

Encoding is the first step in getting data from the host to the printer. The interface comprises the second set of factors that come into play. The interface includes:

1. The physical medium of transmission (cables, connectors, modems, etc., as necessary), i.e., the *path* along which the data travels; and
2. A set of logical conventions shared between the host and printer that allow the transfer to occur in an orderly manner; this is called "handshaking" and may be said to correspond to the *rules of the road*.

Each of the above are implemented differently, depending on the type of interface employed.

Interface types

Interfaces may be divided into two types, according to the manner in which individual bits are transferred across the interface: *parallel* and *serial.*

With the parallel interface, every bit in a single byte is transferred simultaneously via individual wires, as illustrated in Figure 6-1. Note that, in this figure, the least significant bit appears on data line 1, while the most significant bit appears on data line 8.

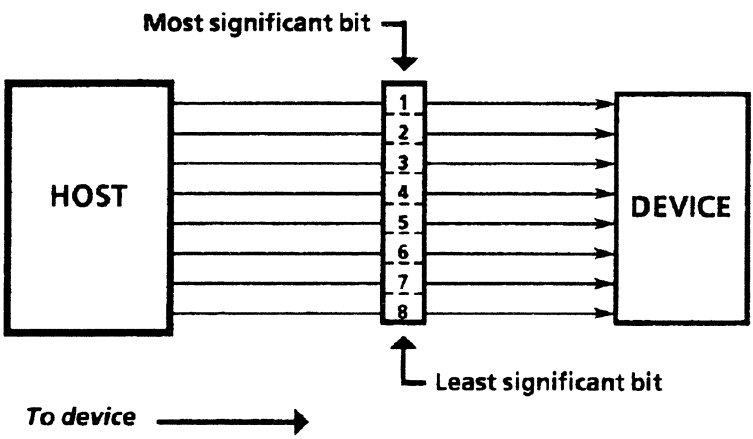


Figure 6-1 **Data transfer—parallel interface**

With the serial interface, each bit is transferred individually, in series, as illustrated in Figure 6-2. Note that the least significant bit is sent to the printer first, and the most significant bit is sent last. The modems (**mo**dulator-**dem**odulators) are necessary to convert the electrical signals from the digital form used in the computer to the analog form necessary for transmission over communication lines, and back to digital form for use in the device. This is described a little further under *The serial interface.*

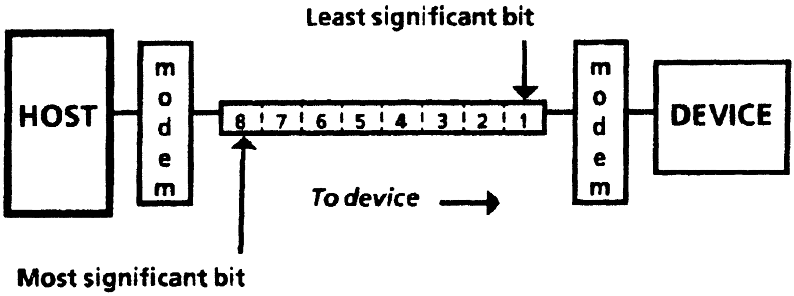


Figure 6-2 **Data transfer—serial interface**

Within each type of interface, different "rules of the road" are used to accomplish the transfer of data. The Model 50 offers two different parallel interfaces in widespread use for local printing (within 10 meters of the host). A variety of switch-selected options for communicating through the serial interface are also available, to accommodate a broad range of remote configurations. Each interface type, methods of handshaking, and pertinent options are described in the following paragraphs.

Parallel interface—Centronics or Dataproducts

The two parallel interfaces are designed to accept data and "talk to" a host in the same way that a Centronics 100-type or Dataproducts 2260-type printer would. The Model 50 uses the same type of cables and connectors as those printers and understands the same signals coming over them.

The parallel interface talks to the host by means of electrical signals coming over the various wires in the interface cable. The purpose and meaning of each signal is dependent on the particular wire on which it appears (i.e., through which it travels).

Chapter 3 provides tables of pertinent signal names for each interface, as well as diagrams that show the timing relationships among the various signals.

Handshaking

Handshaking between the host and the printer is accomplished, for the parallel interface, by means of various signals appearing on different pins, or wires, in the interface connections. However, each interface uses a different system and different signals to accomplish this. The signal timing diagrams in chapter 3 illustrate the interrelationships between signals for each parallel interface. Some of the signals are described in the following paragraphs to provide a general understanding of how the parallel interface transfers data. By no means are these descriptions intended to provide a complete understanding of the intricacies of either interface.

Data strobe, or strobe For both interfaces, this is a signal asserted (raised) by the host to tell the printer that it has a byte of data to send. The printer must read the data while this signal is raised; otherwise the data is lost.

Data 1 through data 8 Each of these eight signals represents one bit in a byte of data. Typically, a high signal represents a 1, while a low signal represents a 0. The printer reads these signals, if it is capable of receiving data, when it sees that the host has raised the strobe signal.

Acknowledge With the **Centronics** interface, the printer raises this signal to tell the host that it has successfully received the last byte of data.

Busy With the **Centronics** interface, the printer raises this signal to tell the host that it <u>cannot</u> receive data. This may occur due to a fault condition, the printer being taken off-line, memory being full, etc. Busy is lowered again when the condition is corrected.

Demand With the **Dataproducts** interface, the printer raises this signal to tell the host that it is <u>ready</u> to receive a byte of data.

On-line With the **Dataproducts** interface, the printer <u>lowers</u> this signal when there is a condition that makes it impossible for the printer to receive data. On-line is raised again when the condition is corrected.

Others Each interface uses a number of other signals, not described here, to keep communication between the host and printer open and to confirm operational status. The above descriptions are supplied to highlight the manner in which the parallel interface transfers data, and are not intended to be used as implemental or diagnostic tools.

Special options for the parallel interface

There are a number of options peculiar to the parallel interface that are selected by means of switch settings on the Model 50's configuration cartridge. The use of the configuration cartridge is described under *The printer configuration.* The following paragraphs explain the special switch settings for the parallel interface.

Switch D:1— Cent. or DP	This switch determines whether the interface installed is Centronics or Dataproducts. The interface can be identified by means of the connector illustrations provided in chapter 3. This switch has no effect with the serial interface.
Switch D:3— VFU emulation	This feature enables a form of vertical format control (control of the print position along the vertical dimension of the page) that is found in some Dataproducts printers. VFU emulation is described in detail in chapter 2.

The serial interface

The serial interface supports *asynchronous* data communication. Asynchronous communication is sometimes referred to as "start-stop" communication, since each byte of data is framed by extra bits known as start and stop bits, as illustrated in Figure 6-3. Furthermore, gaps of time may be inserted between the end of one byte and the beginning of the next.

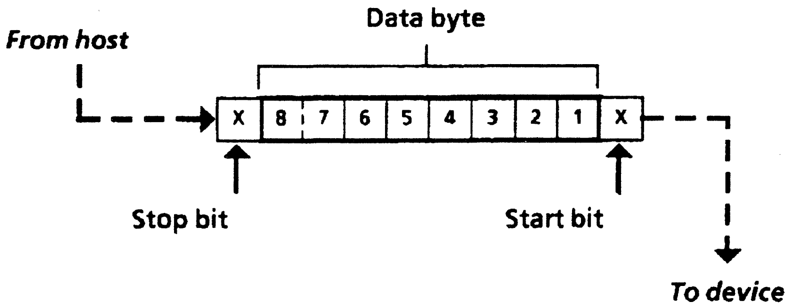


Figure 6-3 **Start and stop bits**

(By contrast, *synchronous* communication requires that no gaps occur between individual bytes and that all data be sent in much longer, and more carefully defined, blocks. A more elaborate dialogue between host and printer is also required.)

The Model 50 asynchronous interface emulates no other printing device in particular, but it follows a widely-accepted set of standards for electronic equipment published by the Electronic Industries Association (EIA). This is the EIA Standard No. RS-232-C, *Interface Between Data Terminal Equipment and Data Communication Equipment Employing Serial Binary Data Interchange.*

Handshaking

With the serial interface, two forms of handshaking are used, one hardware-based and one software-based.

Hardware-based handshaking

The hardware-based form of handshaking relies on electrical signals appearing on specified wires, in a manner similar to that described for the parallel interface. For the serial interface, however, the machine-to-machine dialogue becomes more complex, since modems (**mod**ulator-**dem**odulators) are added at each end of the communication line. In this situation, there are several machine-to-machine dialogues taking place, as illustrated in Figure 6-4.

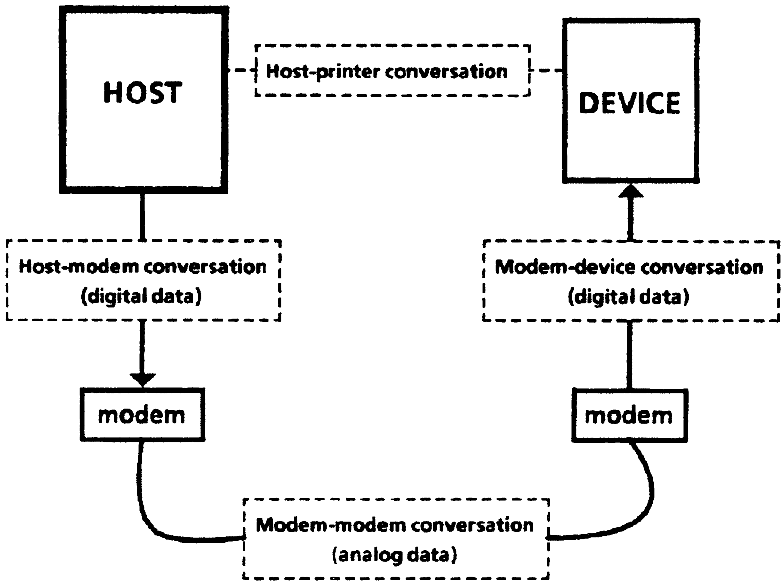


Figure 6-4 **Serial data communication network**

Modems not only transfer signals between the computer and the printer, they transform them as well. Within the computer and within the printer, data exist as on- and off-states, or high and low signals. This data is said to exist in a *digitized*, or *digital*, form. To be transferred over a telephone line, however, the signals must be modulated into a more wave-like, or *analog*, form. At the receiving end of the line, the signals must then be demodulated back into their digital form.

Not all serial networks use modems. Modems are required where data communication occurs over telephone lines that also carry voice signals. Communication lines that carry only data are often called *dedicated* lines.

Chapter 3 provides the names, pin assignments, and meanings of important serial interface signals. The variety of configurations for a serial network precludes the possibility of describing them in detail in this manual.

Software-based handshaking—data flow controls

Handshaking is also performed on a software level by means of data flow controls or, as they are often called, data protocols. This form of handshaking involves the exchange of encoded bytes of data between the host and the printer. Both the host and the printer must be programmed to recognize these codes in context, so that their meaning is understood.

The Model 50 supports two software-based data flow controls, XON/XOFF and ETX/ACK. The XON/XOFF dialogue allows the printer to tell the host when it is ready and when it is not ready to receive data. The ETX/ACK dialogue allows the printer to respond to the host with a positive or negative acknowledgement, indicating whether or not data has been successfully received. These controls are described in detail in chapter 3.

Either or both controls may be active; similarly, none need be used, depending on the requirements of the host.

Data integrity—parity

Due to the complexities of serial networks, the transformations that the data undergoes, and other factors such as electrical noise on the communication line, it can happen that data is distorted during serial transmission. Therefore, many serial networks incorporate means to help guarantee that data arrives at its destination intact. One very common method to ensure the integrity of transmitted data is called *parity*.

In addition to start, stop, and data bits, a host may send an additional bit, called a parity bit, with each byte. The parity bit is either a 1 or a 0, depending on the pattern of bits in the rest of the byte. The value of the parity bit is such that the number of 1-bits in the byte will always be even or always be odd.

For example, the data byte "10100011" contains four 1-bits. To maintain *odd* parity, the host would add an additional 1-bit to guarantee that there is an odd number of 1s in the byte. To maintain *even* parity, the host would add a 0-bit to this byte. The receiving device looks at the additional bit and, if correct parity has been maintained, it assumes that the byte has been successfully transmitted. When running 7-bit, you have a choice of mark, space, even, or odd parity. In mark parity, the parity bit is always 1. In space parity, the parity bit is always 0. When running 8-bit you have a choice of none, odd, or even parity.

If it appears that the data has not been successfully transmitted, the Model 50 prints an error character. The Model 50 may or may not signal the host that an error was detected, depending on the data flow control selected.

Special options for the serial interface

There are a number of options peculiar to the serial interface that are selected by means of switch settings on the Model 50's configuration cartridge. The use of the configuration cartridge is described under *The printer configuration.* The following paragraphs show the special switch settings for the serial interface.

Switch C:1 XON/XOFF	This switch governs whether the XON/XOFF data flow control, or protocol, is used. The host must stop sending within 64 character time frames or data is lost.
Switch C:2 ETX/ACK	This switch governs whether the ETX/ACK data flow control, or protocol, is used.
Switch C:3 Printer Ready	This switch governs whether the Printer ready data flow control, or protocol, is used. This is a hardware-based form of handshaking, designed for use where modems are not required. It is described in chapter 3.
Switches C:4-6 Baud rate	These switches govern the rate of data transmission, called the baud rate, used by the Model 50. This is the speed at which the host sends, and the Model 50 receives, bits of data. **The host, both modems, and the Model 50 must all be set up to use the same baud rate.**

Switch C:8 Auto-disconnect	This switch governs whether or not the interface and host are automatically disconnected under certain conditions.
Switch D:2 Parity	This switch governs whether or not parity is to be used. The choice is determined by your host's communication requirements.
Switch D:3 Parity value	This switch governs parity. If switch D:2 is off, D:3 off = mark D:3 on = space If switch D:2 is on, D:3 off = odd D:3 on = even

The printer

When data is received, the action of the printer is determined by:

1. the printer configuration (setup), and
2. the state of the printer when the data is received

The printer's configuration is dictated both by the requirements of the host for successful data communication and by your printing needs. The state of the printer results both from the initial printer setup and the printing commands received within the data. In addition, the user may control, to some extent, the action of the printer by means of the switches on the control panel.

The printer configuration

Your Model 50 may be configured to accommodate a variety of systems. The switches on the configuration cartridge provide the means to alter the way it talks to the host and the way it looks at data. The choice of interface type (parallel or serial) and device type (Centronics or Dataproducts, if parallel) are dictated by the connector installed at the left rear of the machine. Three types are available:

— Parallel Centronics

— Parallel Dataproducts
— Serial asynchronous

Chapter 3 provides illustrations of each type. Once you have determined the interface type and set switches A:1 and D:1 accordingly, you need to set the rest of the switches to correspond to:

a. your system's requirements, and
b. your printing needs.

Your system's requirements

The following switch settings are dictated by the requirements of your computer system:

Settings	Choices	Switches
Interface type	Parallel or serial	A:1
Line endings	CR LF CR only LF only	A:3-A:4
Auto line-ends (630)		
Byte length	7 or 8 bits	A:5
Encoding	ISO ASCII EBCDIC IBM PC	A:6-A:7
Custom cartridge	Used or not used	A:8
Protocols (serial)	XON/XOFF ETX/ACK Printer ready Baud rate— 300 - 19,200	C:1 C:2 C:3 C:4-C:6
Auto-disconntect mode	Used or not used	C:8
Metered data mode	Used or not used	C:8
Device type (parallel)	Centronics Dataproducts	D:1
Parity (serial), or invert data (par.)	Used or not used	D:2
Parity value (serial), or VFU (parallel)	Odd, even, mark or space Used or not used	D:3
EBCDIC	DSC or SNA	D:4

Your printing needs

The following switch settings are dictated by your printing needs. Note that you may be using an application package that dictates some requirements also. For example, your word processor may require the use of 630 mode.

Settings	Choices	Switches
Printer mode	4045(2700) or 630	A:2
Language	U.S. English U.K. English French Dutch Spanish Italian Danish Norwegian Finnish German Swedish Belgian French Canadian Portuguese Latin Amèrican	B:1-B:4
Status sheets	Printed automatically or on request only	B:5
Chime	Used or not used	B:6
Default font	Resident landscape Resident portrait 1st font cartridge 2nd font cartridge	B:6-B:7
Electronic spacing (630)	Used or not used	D:4
Character spacing (630)	PS, 10, 12, or 15 characters per inch	D:5-D:6
Horizontal tabs (630)	Defaults or command settings	D:7

Chapter 4 provides details about each switch setting.

Printer initial state—the default state

Once the switch settings have been chosen, they don't take effect unless the Model 50 "reads" them. You can make this happen by either turning power off (if it is on) and then on again, or by performing a **soft reset**. A soft reset causes the Model 50 to read the switch settings, reconfigure, and print a configuration sheet. The configuration sheet provides a variety of information about the Model 50's setup and is described in chapter 4. The soft reset is described under *User controls* in chapter 1.

After powering on, or after a soft reset, the Model 50 is in its initial, or default, state. The printer can change state subsequently, in response to commands in the data or user action at the control panel.

Changing printer state

In its initial, or default, state, the printer is prepared to receive data to be printed. The Model 50 prints one character, or performs one printer function, for each byte of data coming through the interface. This is the Model 50's normal, textual printing state.

The printer state can be modified, however, without changing the switch settings. Printing commands can set the printer up to load font data or graphic data, change format parameters, provide diagnostic printouts, manipulate received data in other ways, and, finally, return to its initial state.

Changing printer state by command

As an example, let us assume that we configure a Model 50 as follows:

– Serial interface, ETX/ACK protocol, baud rate 1200, odd parity: These are required by the interface and host communication software.

– 630 mode, electronic spacing, no default horizontal tabs: The word processing software expects 630 mode and sets its own tabs; we decide to take advantage of the Model 50's electronic character spacing.

- No line-ending decisions: The word processing software handles line-ends.
- 8-bit data, ASCII encoding, no custom cartridge: These are required by the host operating system.
- U.S. English: For the printing of status and configuration sheets, and for default character mapping.
- Status sheets and chime enabled: We want the Model 50 to signal us when an error is detected, by printing status sheets with error codes, and by sounding the chime when there is a mechanical fault with the printer.
- Resident portrait default font: All pages will be printed in the internal portrait font (Titan 10 or Titan12), unless we specify other fonts by command.

Thus, when the Model 50 is powered up, the first byte received will cause a character from the Titan font to be assigned to the upper left-hand corner of the first page, since the 630 mode has no default margins. The page will be in portrait orientation. Subsequent characters will be placed from left to right until a line-end is received. Successive lines will move the print position vertically down the page, until the bottom is reached, at which time the entire page will be printed and deposited in the output tray.

However, let us also assume that we require the use of other fonts for our print job. Before we begin to print, we must set the printer up with new fonts. We choose to download font data, instead of replacing font cartridges.

Font Load We send a Font Load command to the Model 50, and the two-digit display shows "LF." *At this point, the Model 50 begins to look for the beginning of a font file. If found, the subsequent font information is stored; nothing is printed.* If a font file is not found, the Model 50 reverts to printing textual data; it also records the occurrence of the erroneous font command for printing on the status sheet.

When the end of the font is reached, the Model 50 looks for the beginning of another font file. It continues to store fonts as long as it sees new font files. If a new file is not found, it reverts to textual printing. In other words, it is back to its default state, with the exception that new fonts are available to the printer. We're now ready to start printing.

Margins and Tabs

It is likely that our word processing software, if we use its printing utility to print our data, will assign some margins and tabs to the page and send 630 commands to the printer. Once received, *the margins and tabs take effect immediately and remain in effect until new commands are received, or until the default state is restored.*

As long as the margins are in effect, all received data must fit within the margins, or it is not printed. Of course, if it should happen that a line extends beyond a margin, the Model 50 records the occurrence for later printing on the status sheet.

Font IDs

Probably, one of the first things we'll want to do is assign ID numbers to our stored fonts, so that we can call them out in the documents we want to print. We therefore send a Font ID Assignment command for each font we have stored. Once received, *the font IDs take effect immediately and remain in effect until new commands assigning the same numbers are received, or until the default state is restored.* Any assigned font may be called out repeatedly in a document, or across a series of documents, as long as the fonts remain in storage and the font IDs remain in effect.

Merge-Page

Let us now assume that we wish to print a series of forms. This might be accomplished by the storing of a merge-page, or constant page, to serve as the form. Later pages will contain the variable data.

We first create the form using Draw Rule commands, text placement commands, special forms fonts, etc., and end it with a page-end code. We then send the Merge Page Load command to the printer, followed by our page-long form.

When the Model 50 sees the Merge Page Load command, it recognizes that a job boundary has been reached. Since the printing of status sheets has been enabled by switch setting, any errors that have been detected and recorded up to this point are printed on a status sheet, which is delivered to the output tray.

The Model 50 then proceeds to store the incoming merge-page in memory; no code is displayed and nothing is printed yet.

Assuming that the system has not been reset, *the merge-page will be constructed in memory using the same margins and font IDs that have already been established, unless the merge-page contains its own commands reassigning those values.*

Once stored, the merge-page can be printed by merging it with subsequent pages, using a Merge Start command. The merge-page remains in memory until it is erased by command, replaced with a new merge-page, or until the system is turned off.

Graphics Let us print a graphic image. Once the graphic file has been created, rasterized, and sixel-encoded (see chapter 2), we send it to the Model 50, preceded by the Graphic Window command. Upon receipt of the command, the Model 50 displays "Ld" in the display window, proceeds to unpack the sixel-encoded data, and store it as bit-map information (printed and unprinted dots). When the end of the graphic is reached, the Model 50 reverts to normal, textual printing again. At the next page-end, the graphic is printed along with the rest of the data for that page. The receipt and storage of graphic data is a temporary state whose length is defined by the graphic command. At the end of the graphic data, the Model 50 returns to its normal, textual printing mode.

Data Monitor The data monitor mode is a special state in which received data is neither stored nor printed. Instead, the value of each byte is printed in a special printout form that enables us to see exactly what codes are being sent from our host to the printer. This is a powerful diagnostic tool that enables us to analyze printing problems. Keys to the use of the data monitor are provided in chapter 5.

The data monitor may be invoked either by command or by holding down the Off-line switch while turning the power on. *Once invoked, the data monitor remains in effect until the system is reset.* If the data monitor is invoked by command, the Model 50 may be reset by command. If it is invoked by means of the Off-line and power switches, the printer state must be reset by switch, as described below.

Changing printer state by switch

Two switches on the control panel allow you to alter printer state manually: the Off-line switch and the Reset switch.

The Off-line switch alone effectively "disconnects" the Model 50 from its host, so that no communication with the host can take place and no data can be received. No data stored in the Model 50 is altered in any way.

The Off-line and Reset switches together can be used to effectively return the printer to its default state. However, this action erases all data in memory, except font data. This function is described in greater detail in chapter 1.

The Off-line and power switches together put the Model 50 into the data monitor mode. This function is described in greater detail in chapter 5. The printer must be returned to its default state, as described below, to exit the data monitor mode.

Restoring the default state

The printer can be forced to assume its default state by switch manipulation, i.e., performing a soft reset or cycling power.

In addition, the printer can be forced to assume its default state, or restore its default values, by commands sent from the host.

Reset commands

A number of commands are available, depending on the printer mode (2700 or 630) selected, that restore the printer's default state, or some portion of it. The 2700 mode includes the Reset command, which restores all print parameters to their default values. The Reset command also cancels the data monitor mode, if the data monitor is invoked by command. See the *Creating documents* section of the *User Manual.*

The 630 mode includes the same Reset command as the 2700 mode, as well as the Reset/Initialization and Reset/Initialization Immediate commands. These are described in Table A-3, in appendix A of this manual.

Formatting commands Any print parameter can be changed by means of the appropriate command. For example, margins can be changed by sending a new margin command. It is not necessary to reset the system.

Similarly, tabs, font IDs, and all other format parameters remain in effect until new values are assigned.

To summarize

Following start-up or reconfiguration, the Model 50 alters its state as specified by command, and retains that state until new commands are received, or until the system is reset.

A. Tables

This appendix provides tables for reference. These are not intended to be a substitute for the expository material contained in the rest of the manual. The following tables are included.

Table A-1A **Default format parameters**

4045 (2700) mode*, 8.5 x 11 inch paper

Landscape: XCP14iso-L

Margins				
Units	**Top**	**Bottom**	**Left**	**Right**
Inches	0.40	0.40	0.66	0.66**
mm	10.2	10.2	16.8	16.8**
dots	120	120	200	200**

Tabs				
	Horizontal		**Vertical**	
Units	**1st tab**	**Interval**	**1st tab**	**Interval**
Inches	1.1	0.44	1.1	0.7
mm	28.0	11.2	28.0	17.8
dots	330	132	330	210

* In 630 mode, all defaults are zero, except vertical tabs, which occur at one-inch (25.4 mm, 300 dots) intervals down the page.

** Use of 14-inch paper results in an extra 3" (76.2 mm or 900 dots) of space at the right margin in landscape, the top margin in portrait.

Table A-1B **Default format parameters (continued)**

4045 (2700) mode*, 8.5 x 11 inch paper

Portrait: Titan10iso-P

Margins				
Units	**Top**	**Bottom**	**Left**	**Right**
Inches	0.66**	0.66	0.40	0.40
mm	16.8**	16.8	10.2	10.2
dots	200**	200	120	120

Tabs				
	Horizontal		**Vertical**	
Units	**1st tab**	**Interval**	**1st tab**	**Interval**
Inches	1.1	0.44	1.1	1.0
mm	28.0	11.2	28.0	17.8
dots	330	132	330	210

* In 630 mode, all defaults are zero, except vertical tabs, which occur at one-inch (25.4 mm, 300 dots) intervals down the page.

** Use of 14-inch paper results in an extra 3" (76.2 mm or 900 dots) of space at the right margin in landscape, the top margin in portrait.

Table A-1C **Default format parameters (continued)**

4045 (2700) mode*, A4 paper (210 x 297 mm, 8.27 x 11.69 inch)

Landscape: XCP12.5iso-L

Margins				
Units	**Top**	**Bottom**	**Left**	**Right**
Inches	0.28	0.28	0.56	0.56
mm	7.2	7.2	14.2	14.2
dots	84	84	168	168

Tabs				
	Horizontal		**Vertical**	
Units	**1st tab**	**Interval**	**1st tab**	**Interval**
Inches	1.04	0.48	0.95	0.7
mm	26.4	12.2	23.1	17.8
dots	313	144	287	210

* **Note:** In 630 mode, all defaults are zero, except vertical tabs, which occur at one-inch (25.4 mm, 300 dots) intervals down the page.

Table A-1D **Default format parameters (continued)**

4045 (2700) mode*, A4 paper (210 x 297 mm, 8.27 x 11.69 inch)

Portrait: Titan12iso-P

Margins				
Units	**Top**	**Bottom**	**Left**	**Right**
Inches	0.51	0.51	0.80	0.80
mm	13.0	13.0	20.3	20.3
dots	153	153	240	240

Tabs				
	Horizontal		**Vertical**	
Units	**1st tab**	**Interval**	**1st tab**	**Interval**
Inches	1.04	0.48	1.0	1.0
mm	26.4	12.2	25.4	25.4
dots	313	144	300	300

* **Note:** In 630 mode, all defaults are zero, except vertical tabs, which occur at one-inch (25.4 mm, 300 dots) intervals down the page.

Table A-2A **2700 command set**

The following conventions are observed in the command descriptions:

Required characters are printed in **bold**.

Abbreviated, required codes are printed in (regular type enclosed in parentheses). Examples are: (Esc), (line-end), (CR), etc.

Required, variable fields are printed in *italics*:

— *C* for an alphabetic character

— *n or* n_1,n_2, etc., for variable numeric fields

Optional fields are printed in regular type, underlined.

"(line-end)" may be the line-ending code or codes selected by switch setting, or a line-feed code alone, regardless of switch setting.

No commands include spaces (ISO X'20', EBCDIC X'40').

Command name	Form / Notes
Surrogate Escape	Form: **=UDK=** *C* Notes: 1. *C* may be any printable character, except: — the characters =, U, D, or K — the space character (ASCII X'20', EBCDIC X'40') — the numerals 0 through 9 2. *C* becomes the effective escape code, for the purpose of sending commands to the printer.
Print	Form: (Esc)**+P**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma).

Table A-2B **2700 command set (continued)**

Command name	Form / Notes
Font Load	Form: (Esc) + **F**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to store font data. 2. Several fonts may be loaded with one command. All previously-stored fonts are erased with this command.
Font Add Selected	Form: (Esc) + **A**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to add a new font to previously-stored font data. 2. Several fonts may be loaded with one command.
Font ID Assignment	Form: (Esc) + n_1*font-name*(line-end) Notes: 1. This command assigns an ID number to a font in storage, so that it may be called out by a Font Change command. 2. n_1 is a numeral from 0 to 9 *font-name* is the name of the font

Table A-2C **2700 command set (continued)**

Command name	Form / Notes
Font Change	Form: (Esc)*n* Notes: 1. This command calls out a font in storage, to which an ID has been assigned with a Font ID Assignment. 2. *n* is the ID of the font, as assigned by the Font ID Assignment; if 0 is unassigned, (Esc)**0** selects the default font, as determined by switch setting. 3. When the command is received, all subsequent data are printed in the requested font until a new Font Change command is received, or until a Reset command is received.
Font Delete Selected	Form: (Esc)+**B**,(comment)(line-end) *list of font-names*(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to delete specified fonts from previously-stored font data. 2. Font names must be specified in *list of font names* exactly as they are specified for other font commands. More than one font may be listed, separated by commas.
Font Unload	Form: (Esc)+**U**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and deletes all previously-stored font data.

Table A-2D **2700 command set (continued)**

Command name	**Form / Notes**
Reset	Form: (Esc)+**X**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and resets all format parameters to their defaults. In effect, this command resets the printer to its start-up state, except that downloaded fonts are not erased. 2. The Model 50 ignores a comment within the Reset command, although its inclusion does not invalidate the command. 3. When used to end a data monitor dump, the same escape character must be used to introduce both the Data Monitor and the Reset commands.
Data Monitor	Form: (Esc)+**D** Notes: 1. This command marks a job boundary and causes the printer to print the hexadecimal value of all bytes received. 2. The data monitor mode can be ended by sending a Reset command, or by performing a soft reset at the control panel (pressing the Off-line and Reset switches simultaneously).
Merge Page Load	Form: (Esc)+**M**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to store a page of data, to be merged with later pages. 2. A previously-stored merge-page is erased with this command.

Table A-2E **2700 command set (continued)**

Command name	Form / Notes
Merge Start	Form: (Esc)**ze** Notes: 1. This command causes the page in which it occurs, and all subsequent pages, to be merged with a previously-stored merge-page. 2. All subsequent pages are automatically merged with the merge-page until the Merge Stop command is received.
Merge Stop	Form: (Esc)**zd** Notes: 1. This command cancels page merging. 2. The page in which this command occurs is merged with the constant page; subsequent pages are not.
Merge Page Unload	Form: (Esc)+**V**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and erases a previously-stored merge-page.

Table A-2F **2700 command set (continued)**

Command name	**Form / Notes**
Character Table	Form: (Esc)**+T**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to store a customized character mapping table. 2. This command and its use are described in chapter 2 of this manual.
Language	Form: (Esc)**zl**(character) Notes: 1. This command selects a national character mapping or a customized mapping table, either downloaded or in a custom cartridge. 2. (character) is a character selecting the mapping: 0 = U.S. English 9 = German 1 = U.K. English A = Swedish 2 = French B = Belgian 3 = Dutch C = French Canadian 4 = Spanish D = Portuguese 5 = Italian E = Latin American 6 = Danish G = Custom cartridge 7 = Norwegian H = Downloaded table 8 = Finnish
Units—1/60th inch	Form: (Esc)**zg** Notes: 1. This command tells the printer that numerical values for margin and tab commands are expressed in 1/60th inch (.423 mm).

Table A-2G **2700 command set (continued)**

Command name	Form / Notes
Units—1/300th inch	Form: (Esc)**zf** Notes: 1. This command tells the printer that numerical values for margin and tab commands are expressed in 1/300th inch (.085 mm). 2. Default value is 1/60th inch (.425 mm).
Margins	Form: (Esc)**m**n_1,n_2,n_3,n_4,n_5(line-end) Notes: 1. This command assigns margins for page layout. 2. Units are in 1/60th inch (.425 mm) or 1/300th inch (.085 mm), depending on previous commands. Default is 1/60th. 3. The printer assumes the orientation of the last font invoked for the margin assignment. 4. n_1 is the page-length, from the top to the bottom edge n_2 is the top margin n_3 is the bottom margin n_4 is the left margin n_5 is the right margin, as measured from the **left** edge of the page
Top Margin	Form: (Esc)**zn***n*(line-end) Notes: 1. Once the margins are assigned with the Margins command above, this command sets a new top margin, independent of the current bottom, right, and left margins. 2. Units are in 1/60th inch (.425 mm) or 1/300th inch (.085 mm), depending on previous commands. Default is in 1/60th units. 3. *n* is the top margin, as measured from top edge of the page.

Table A-2H **2700 command set (continued)**

Command name	Form / Notes
Bottom Margin	Form: (Esc)**zq***n*(line-end) Notes: 1. Once the margins are assigned with the Margins command above, this command sets a new bottom margin, independent of the current top, right, and left margins. 2. Units are in 1/60th inch (.425 mm) or 1/300th inch (.085 mm), depending on previous commands. Default is in 1/60th units. 3. *n* is the bottom margin, as measured from the bottom edge of the page.
Left Margin	Form: (Esc)**zk***n*(line-end) Notes: 1. Once the margins are assigned with the Margins command above, this command sets a new left margin, independent of the current top, bottom, and right margins. 2. Units are in 1/60th inch (.425 mm) or 1/300th inch (.085 mm), depending on previous commands. Default is in 1/60th units. 3. *n* is the left margin, as measured from the left edge of the page.
Right Margin	Form: (Esc)**zm***n*(line-end) Notes: 1. Once the margins are assigned with the Margins command above, this command sets a new right margin, independent of the current top, bottom, and left margins. 2. Units are in 1/60th inch (.425 mm) or 1/300th inch (.085 mm), depending on previous commands. Default is in 1/60th units. 3. *n* is the right margin, as measured from the **left** edge of the page.

Table A-21 **2700 command set (continued)**

Command name	Form / Notes
Horizontal Tab	Form: (Esc)**t**$n_1, n_2, \ldots n_9$(line-end) Notes: 1. This command sets horizontal tab stops across the page. 2. Units are in 1/60th inch (.425 mm) or 1/300th inch (.085 mm), depending on previous commands. Default is 1/60th. 3. Up to 9 tabs may be set within each command. 4. Tab stops do not have to be in numerical order. New tabs are added to those previously set, **including defaults**. If more than 9 tabs are set, the highest-numbered tabs are canceled.
Horizontal Tab Clear	Form: (Esc)**d** Notes: 1. This command clears all horizontal tabs, **including defaults**.
Vertical Tab	Form: (Esc)**v**$n_1, n_2, \ldots n_{125}$(line-end) Notes: 1. This command sets vertical tab stops down the page. 2. Units are in 1/60th inch (.423 mm) or 1/300th inch (.085 mm), depending on previous commands. Default is 1/60th. 3. Up to 125 tabs may be set. 4. Tab stops do not have to be in numerical order. New tabs are added to those previously set, **including defaults**. If more than 125 tabs are set, the highest-numbered tabs are canceled.
Vertical Tab Clear	Form: (Esc)**e** Notes: 1. This command clears all vertical tabs, **including defaults**.

Table A-2J **2700 command set (continued)**

Command name	Form / Notes
VFU Stops Set	Form: (Esc)**zv**n_1,n_2, . . . ,n_8(line-end) Notes: 1. This command assigns channel stops for one of 12 channels. Up to seven stops may be set for each channel. 2. n_1 is a number from 1 to 12 indicating the channel for which stops are being assigned. n_2 through n_8 are the distances from the top of the page, expressed in the current units (1/60th or 1/300th inch), at which stops are to be set for this channel. 3. The use of this command is described in chapter 3 of this manual.
VFU Stops Clear	Form: (Esc)**zw** Notes: 1. This command clears all previously-set VFU stops.
Line Spacing	Form: (Esc)**i***n* Notes: 1. This command multiplies the current font height by a number from 1/2 to 3, to achieve variation in line spacing. 2. *n* is a number from 0 to 4: 0 = single line spacing (font height x 1) 1 = $1\frac{1}{2}$ line spacing (font height x $1\frac{1}{2}$) 2 = double line spacing (font height x 2) 3 = triple line spacing (font height x 3) 4 = half-line spacing (font height x $\frac{1}{2}$)

Table A-2K **2700 command set (continued)**

Command name	Form / Notes
Absolute Text Placement	Form: (Esc)**a**n_1,n_2(line-end) Notes: 1. This command specifies the position on the page *relative to the page origin* where subsequent text is to begin printing. 2. n_1 specifies the x-coordinate of the new position. n_2 specifies the y-coordinate of the new position. 3. The point referenced is the upper left corner of the character cell in landscape orientation, or the left edge of the baseline in portrait orientation.
Relative Text Placement	Form: (Esc)**r**C_1nC_2 Notes: 1. This command specifies a position on the page *relative to the current print position on the page* where subsequent text is to begin printing. 2. C_1 is an alphabetic character indicating the direction of movement: u = up d = down l (lowercase letter L) = left r = right 3. *n* is the distance, expressed in dots, that printing is to be moved. 4. C_2 is any non-numeric character. This character may be a space.

Table A-2L **2700 command set (continued)**

Command name	Form / Notes
Graphic Window	Form: (Esc)**gw**n_1;n_2,n_3... (line-end) graphic data Notes: 1. This command prepares the printer to create a graphic window for printing on the *current page*. 2. n_1 specifies the magnification for this graphic window. 2 specifies 2X magnification; 4 specifies 4X magnification; any other value specifies 1X magnification. n_2 is a variable-length numeric field specifying the distance along the x-axis between the page origin and the graphic origin. n_3 is a variable-length numeric field specifying the distance along the y-axis between the page origin and the graphic origin. 3. The use of this command is described in chapter 2 of this manual.

Table A-2M **2700 command set (continued)**

Command name	Form / Notes
Repeat Window	Form: (Esc)**gr**n_1;n_2,n_3...(line-end) Notes: 1. This command repeats the last-stored graphic window for printing on the current page. 2. n_1 specifies the magnification for this graphic window. 2 specifies 2X magnification; 4 specifies 4X magnification; any other value specifies 1X magnification. n_2 is a variable-length numeric field specifying the distance along the x-axis between the page origin and the graphic origin. n_3 is a variable-length numeric field specifying the distance along the y-axis between the page origin and the graphic origin. 3. The use of this command is described in chapter 2 of this manual.

Table A-2N **2700 command set (continued)**

Command name	Form / Notes
Draw Line	Form: (Esc)Cn_1,n_2,n_3,n_4(line-end) Notes: 1. This command draws a solid black line on the page. 2. C is an **x** or a **y**. An **x** indicates that the line is to be parallel to the x-axis; a **y** indicates that it is to be parallel to the y-axis. 3. n_1 specifies the x-coordinate on the page, expressed in dots, where the line is to begin. n_2 specifies the y-coordinate, expressed in dots, where the line is to begin. n_3 specifies the length of the line, expressed in dots. n_4 specifies the thickness of the line, expressed in dots.
Justification Start	Form: (Esc)**j** Notes: 1. This command initiates text justification. Spaces between words are expanded or compressed so that each text line fits exactly between the left and right margins. Spaces are expanded to 300% their normal width, or compressed to 60% their normal width. A line which cannot be justified within these parameters is not justified. 2. Justification begins exactly at the point within the line where the command falls. For example, if it occurs in the middle of the line, spaces occurring **before** the command will not be adjusted, but spaces **after** the command will be adjusted.

Table A-2O **2700 command set (continued)**

Command name	Form / Notes
Justification Stop	Form: (Esc)**k** Notes: 1. This command stops text justification. Spaces between words are given equal width. 2. Justification stops exactly at the point within the line where the command falls. For example, if it occurs in the middle of the line, spaces occurring **before** the command will be adjusted, but spaces **after** the command will not be adjusted.
Centering	Form: (Esc)**q** Notes: 1. This command centers the line of text in which it occurs between the left and right margins. If the line is too long to fit between the margins, it is printed so that it extends equally beyond the left and right margins. 2. Only one line of text can be centered at a time with this command.
Superscript Start	Form: (Esc)**h** Notes: 1. This command raises the baseline of subsequent characters, so that they may be printed as superscripts. 2. If the superscript is to be printed in a smaller font, the smaller font should be invoked **after** the superscript command is given, since the distance the baseline is raised depends on the font in effect. 3. Superscripts cannot be nested; that is, a superscript command must be canceled before another can take effect.

Table A-2P **2700 command set (continued)**

Command name	Form / Notes
Subscript Start	Form: (Esc)**l** (lowercase letter L) Notes: 1. This command lowers the baseline of subsequent characters, so that they may be printed as subscripts. 2. If the subscript is to be printed in a smaller font, the smaller font should be invoked **after** the subscript command is given, since the distance the baseline is lowered depends on the font in effect. 3. Subscripts cannot be nested; that is, a subscript command must be canceled before another can take effect.
Super/Subscript Stop	Form: (Esc)**s** Notes: 1. This command cancels a super- or subscript command. 2. Super- and subscripting is also canceled by a carriage return, line feed, or form feed.
Underlining Start	Form: (Esc)**u** Notes: 1. This command initiates underlining of all characters between this command and the Underlining Stop command, including spaces and tabs. 2. The thickness of the underline within any line is determined by the size of the last font before the end of the line or before the Underlining Stop command. 3. Underlining always occurs at the baseline in effect when the command is given. If it is used with superscripted or subscripted text, the underlining remains at its initial baseline. To move underlining up or down with the super- or subscripted characters, stop the underlining before the super- or subscript, and renew the underlining after the super- or subscript.

Table A-2Q **2700 command set (continued)**

Command name	Form / Notes
Underlining Stop	Form: (Esc)**w** Notes: 1. This command cancels underlining.
Bolding Start	Form: (Esc)**b** Notes: 1. This command iniatiates bolding (shadow printing). Every character printed between this command and the Bolding Stop command is printed twice, the second character being displaced two dots to the right of the first, thus achieving a **bold** effect. 2. This command requires extra page composition processing and should be used with moderation; bold fonts are preferred where a lot of bold text is needed.
Bolding Stop	Form: (Esc)**p** Notes: 1. This command cancels bolding (shadow printing). 2. This command also cancels overstriking.
Overstriking Start	Form: (Esc)**zo***C* Notes: 1. This command initiates overstriking. Every character occuring between this command and the Overstriking Stop command is overstruck with another character, thus: ~~W~~. 2. *C* is any printable character. 3. This command requires extra page composition processing and should be used with moderation.
Overstriking Stop	Form: (Esc)**zp** Notes: 1. This command cancels overstriking.

Table A-3A **630 command set**

This command set emulates the commands used with the Diablo 630 Extended Character Set, All Purpose Interface (ECS/API) configuration. The following conventions are observed in the command descriptions:

Required characters are printed in **bold**.

Abbreviated, required codes are printed in (regular type enclosed in parentheses). Examples are: (Esc), (line-end), (CR), etc.

Required, variable fields are printed in *italics*:

— *C* for an alphabetic character

— *n* or n_1,n_2, etc., for variable numeric fields

Optional fields are printed in <u>regular type, underlined</u>.

"(line-end)" may be the line-ending code or codes selected by switch setting, or a line-feed code alone, regardless of switch setting.

No commands include spaces (ISO X'20', EBCDIC X'40').

Command name	Form / Notes
Surrogate Escape	Form: **=UDK=***C* Notes: 1. *C* may be any printable character, except: — the characters =, U, D, or K — the space character (ASCII X'20', EBCDIC X'40') — the numerals 0 through 9 2. *C* becomes the effective escape code, for the purpose of sending commands to the printer.

Table A-3B **630 command set (continued)**

Command name	Form / Notes
Send Status Byte, Firmware Revision	Form: (Esc)(SUB)**1** Notes: 1. This command causes the printer to send a status byte back to the host describing its operational state. Each bit descirbes the state of a different part of the printer. A 1-bit, except where stated, indicates that the following conditions are TRUE (low-order bit is no.0): Bit No. — Meaning 0 — Dry imager is low 1 — Switch-selected pitch (1 = 10, 0 = other) 2 — Paper out or misfeed 3 — Auto line feed 4 — Cover open 5 — Input buffer empty 6 — Fault/printer jam/printer check (diagnostic error) 7 — 1 for seven bit, 0 for eight bit 2. This command is only available with the serial interface.
Print Space	Form: (Esc)**Y** Notes: 1. This command allows the space character code (ASCII X'20', EBCDIC X'40') to be sent to the printer.
Print DEL	Form: (Esc)**Z** Notes: 1. This command allows the DEL code (ASCII X'7F', EBCDIC X'07') to be sent to the printer.

Table A-3C **630 command set (continued)**

Command name	Form / Notes
Print*	Form: (Esc) + **P**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma).
Font Load*	Form: (Esc) + **F**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to store font data. 2. Several fonts may be loaded with one command. All previously-stored fonts are erased with this command.
Font Add Selected*	Form: (Esc) + **A**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to add a new font to previously-stored font data. 2. Several fonts may be loaded with one command.
Font ID Assignment*	Form: (Esc) + n_1*font-name*(line-end) Notes: 1. This command assigns an ID number to a font in storage, so that it may be called out by a Font Change command. 2. n_1 is a numeral from 0 to 9 *font-name* is the name of the font

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3D **630 command set (continued)**

Command name	Form / Notes
Font Change*	Form: (Esc)**F***n* Notes: 1. This command calls out a font in storage, to which an ID has been assigned with a Font ID Assignment. 2. *n* is the ID of the font, as assigned by the Font ID Assignment. 3. When the command is received, all subsequent data are printed in the requested font until a new Font Change command is received, or until a Reset command is received.
Font Delete* Selected	Form: (Esc)+**B**,(comment)(line-end) *list of font-names*(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to delete specified fonts from previously-stored font data. 2. Font names must be specified in *list of font names* exactly as they are specified for other font commands. More than one font may be listed, separated by commas.
Font Unload*	Form: (Esc)+**U**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and deletes all previously-stored font data.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3E **630 command set (continued)**

Command name	Form / Notes
Reset*	Form: (Esc)+**X**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and resets all format parameters to their defaults. In effect, this command resets the printer to its start-up state, except that downloaded fonts are not erased. 2. When used to end a data monitor dump, the same escape character must be used to introduce both the Data Monitor and the Reset commands.
Reset/ Initialization	Form: (Esc)(CR)**P** Notes: 1. This command processes all data received up to that point and resets the printer to the following state: Default font as switch selected Lines per page set to 66 All counters set to zero Margins set to zero Bolding, underlining, and super/subscripting off All tabs cleared Print a partially formatted page All other parameters off, or as switch selected
Reset/ Initialization Immediate	Form: (Esc)(SUB)**I** (capital letter I) Notes: 1. This command performs the same function as Reset/Initialization above.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3F **630 command set (continued)**

Command name	Form / Notes
Data Monitor*	Form: (Esc)+**D** Notes: 1. This command marks a job boundary and causes the printer to print the hexadecimal value of all bytes received. 2. The data monitor mode can be ended by sending a Reset command (both commands must use the same UDK character), or by performing a soft reset at the control panel (pressing the Off-line and Reset switches simultaneously).
Merge Page Load*	Form: (Esc)+**M**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to store a page of data, to be merged with later pages. 2. A previously-stored merge-page is erased with this command.
Merge Start*	Form: (Esc)**we** Notes: 1. This command causes the page in which it occurs, and all subsequent pages, to be merged with a previously-stored merge-page. 2. All subsequent pages are automatically merged with the merge-page until the Merge Stop command is received.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3G **630 command set (continued)**

Command name	Form / Notes
Merge Stop*	Form: (Esc)**wd** Notes: 1. This command cancels page merging. 2. The page in which this command occurs is merged with the constant page; subsequent pages are not.
Merge Page* Unload	Form: (Esc)+**V**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma),and erases a previously-stored merge-page.
Character Table*	Form: (Esc)+**T**,(comment)(line-end) Notes: 1. This command marks a job boundary, causes a status sheet to print (if switch is enabled and an error occurs **or** if previous command included a comma), and prepares the printer to store a customized character mapping table. 2. The use of this command is similar to that for the 4045 (2700) mode, as described in chapter 2 of this manual.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3H **630 command set (continued)**

Command name	**Form / Notes**
Language*	Form: (Esc)**w**(code) *table data***X**(line-end) Notes: 1. This command selects a national character mapping or a customized mapping table, either downloaded or in a custom cartridge. 2. (code) is a code selecting the mapping:

Code	Hex	Mapping	Code	Hex	Mapping
SOH	01	Custom cartridge	VT	0B	Norwegian
STX	02	ISO 6937	FF	0C	Danish
ETX	03	U.S. English	FCR	0D	Finnish
EOT	04	U.K. English	SO	0E	Portuguese
ENQ	05	Swedish	SI	0F	French Canadian
ACK	06	French	DC3	13	Custom cartridge
BEL	07	German	DC4	14	Downloaded table
BS	08	Dutch			
HT	09	Italian			
LF	0A	Spanish			

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-31 **630 command set (continued)**

Command name	Form / Notes
Variable HMI	Form: (Esc)(US)(code) Notes: 1. This command allows you to set the horizontal motion index (HMI) at a chosen value. HMI is expressed as a number of 1/120ths of an inch (approximately 0.21 mm) and serves as the unit of measure for many 630 positioning commands that affect horizontal character placement. The default HMI value depends on the switch selection governing character spacing (D:5 and D:6): Switch-Selected Spacing — Standard HMI 10 characters per inch — 12 12 characters per inch — 10 15 characters per inch — 8 2. (code) is a control code or character whose binary value is one greater than the desired number of HMI units, and may range in value from 0 to 125. Example: For 12 characters per inch, you need an HMI of 10; 10 + 1 = 11 = X'0B' (ESC)(US)(VT)
Reset Standard HMI	Form: (Esc)**S** Notes: 1. This command returns HMI to its standard value.

Table A-3J **630 command set (continued)**

Command name	Form / Notes
Variable VMI	Form: (Esc)(RS)(code) Notes: 1. This command allows you to set the vertical motion index (VMI) at a chosen value. VMI is expressed as a number of 1/48ths of an inch (approximately .53 mm) and serves as the unit of measure for many 630 positioning commands that affect vertical character placement. The default VMI value is 8 (1/6th inch, approximately 4.2 mm). 2. (code) is a control code or character whose binary value is one greater than the desired number of VMI units, and may range in value from 0 to 125. Example: For 6 lines per inch, VMI = 8; 8 + 1 = 9 X'09' (ESC)(RS)(HT)
Reset Standard VMI	Form: (Esc)**m** Notes: 1. This command returns VMI to its standard value.

Table A-3K **630 command set (continued)**

Command name	Form / Notes
Set Left Margin*	Form: (Esc)**l***n*(line-end) Notes: 1. This command defines the left margin as a distance from the left edge of the page. 2. **l** is the lowercase letter L. 3. *n* is a character sequence denoting the numeric value by which the current HMI is multiplied to obtain the distance from the left edge. This number is not the binary value of the code, but a character or series of characters, as typed on a keyboard. Example: Given 10 characters per inch HMI = 12 For one inch margin, n = 10 (ESC)l10(line-end)
Set Left Margin Here	Form: (Esc)**9** Notes: 1. This command sets the left margin at the current print position.
Set Right Margin*	Form: (Esc)**r***n*(line-end) Notes: 1. This command defines the right margin as a distance from the *left* edge of the page. 2. *n* is a number by which the current HMI is multiplied to obtain the distance from the left edge. This number is not the binary value of the code, but a character or series of characters, as typed on a keyboard.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3L **630 command set (continued)**

Command name	Form / Notes
Set Right Margin Here	Form: (Esc)**0** (zero) Notes: 1. This command sets the right margin at the current print position.
Set Top Margin*	Form: (Esc)**wt***n*(line-end) Notes: 1. This command defines the distance from the top edge of the page to the baseline of the first print line. 2. *n* is a number by which the current VMI is multiplied to obtain the distance from the top edge. This number is not the binary value of the code, but a character or series of characters, as typed on a keyboard.
Set Top Margin Here	Form: (Esc)**T** Notes: 1. This command sets the top margin at the current print position.
Set Bottom Margin*	Form: (Esc)**wb***n*(line-end) Notes: 1. This command defines the distance from the *top* edge of the page to the baseline of the last line. 2. *n* is a number by which the current VMI is multiplied to obtain the distance from the top edge. This number is not the binary value of the code, but a character or series of characters, as typed on a keyboard.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3M **630 command set (continued)**

Command name	Form / Notes
Set Bottom Margin Here	Form: (Esc)**L** Notes: 1. This command sets the bottom margin at the current print position.
Clear Top and Bottom Margins	Form: (Esc)**C** Notes: 1. This command resets the top and bottom margins to zero.
Lines Per Page	Form: (Esc)(FF)(code) Notes: 1. This command assigns the printable length of the page, or height of the image area. 2. (code) is any control code or printable character. The binary value of this code is multiplied by the *current* VMI to determine the height of the image area. Subsequent changes in VMI do not affect the resultant value. Example VMI = 8 which is 8/48 = 1/6 60x1/6 = 10 inches code = number of lines per page (ESC)(FF)(<)
Electronic Spacing*	Form: (Esc)_(line-end) Notes: 1. This command enables electronic spacing when it is not switch-selected. 2. _ is the underscore character. 3. Electronic spacing is canceled by the Proportional Space Start, Proportional Space Stop, and Cancel WP Features commands.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3N **630 command set (continued)**

Command name	Form / Notes
Proportional Space Start	Form: (Esc)**P** Notes: 1. This command enables proportional spacing, thus overriding switch-selected, fixed-pitch spacing. 2. This command substitutes a character spacing table for the standard pitch. It does not override electronic character spacing when switch-selected, but does so when electronic spacing has been enabled by the Electronic Spacing command.
Proportional Space Stop	Form: (Esc)**Q** Notes: 1. This command resets character spacing to its switch-selected value. It does not override electronic character spacing when switch-selected, but does so when electronic spacing has been enabled by the Electronic Spacing command.

Table A-30 **630 command set (continued)**

Command name	Form / Notes
Offset Selection	Form: (Esc)(DC1)(code) Notes: 1. This command allows you to adjust character spacing when proportional spacing is being used, or to alter HMI. 2. (code) is any control code or printable character. The bit pattern of the code determines the offset (low-order bit is 1): Bit No. Meaning 1 - 6 Binary value is size of offset in 1/120th inch (approximately .21 mm) units (63 units maximum) 7 A 0 here indicates that offset is positive (increases horizontal displacement); a 1 here indicates that offset is negative (decreases horizontal displacement) 3. The specified offset remains in effect until a new Offset Selection command or a Cancel WP Features command is received.

Table A-3P **630 command set (continued)**

Command name	Form / Notes
Horizontal Tab*	Form: (Esc)**i**n_1,n_2, . . . n_{160}(line-end) Notes: 1. This command sets horizontal tab stops across the page. 2. Units are current HMI value. This number is not the binary value of the code, but a character or series of characters, as typed on a keyboard. 3. Up to 160 tabs may be set. 4. Tab stops do not have to be in numerical order. New tabs are added to those previously set, including defaults. If more than 160 tabs are set, the highest-numbered tabs are canceled.
Set Horizontal Tab Here	Form: (Esc)**1** Notes: 1. This command sets a horizontal tab at the current print position.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3Q **630 command set (continued)**

Command name	Form / Notes
Absolute Horizontal Tab	Form: (Esc)(HT)(code) Notes: 1. This command allows you to select any tab in the first 126 print positions without previously setting tabs. 2. (code) is any control code or character other than NUL and DEL. *The decimal value of the code's bit pattern* selects the nth position. For example, since the decimal value of the code for the letter "A" is 49, the command (Esc)(HT)*A* selects the 49th tab position. The leftmost print position is considered to be binary location 1. Absolute Horizontal Tabs make leftward tabbing possible. 3. Although no prior tabs need to be set, Absolute Horizontal Tabs will not affect any previously-set tabs. Absolute Horizontal Tabs are not retained in memory, so each stop must be commanded each time it is used.
Clear Horizontal Tab Here	Form: (Esc)**8** Notes: 1. This command clears a tab that has been set at the current print position.

Table A-3R **630 command set (continued)**

Command name	Form / Notes
Vertical Tab*	Form: (Esc)**j**n_1,n_2, . . . n_{125}(line-end) Notes: 1. This command sets vertical tab stops down the page. 2. Units are current VMI value. This number is not the binary value of the code, but a character or series of characters, as typed on a keyboard. 3. Up to 125 tabs may be set. 4. Tab stops do not have to be in numerical order. New tabs are added to those previously set. If more than 125 tabs are set, the highest-numbered tabs are canceled.
Set Vertical Tab Here	Form: (Esc)**-** (hyphen) Notes: 1. This command sets a vertical tab at the current print position.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3S **630 command set (continued)**

Command name	Form / Notes
Absolute Vertical Tab	Form: (Esc)(VT)(code) Notes: 1. This command allows you to select any of the 126 possible lines of a page regardless of your current line position. 2. (code) is any control code or character other than NUL and DEL. *The decimal value of the code's bit pattern* selects the nth tab position. For example, since the decimal value of the code for the letter "A" is 49, the command (Esc)(VT)*A* selects the 49th print position from the top of the page. (The top of the page is considered to be binary location 1.) 3. You cannot tab beyond the end of a page even if the number of lines per page is less than 126; if you try to tab beyond the limit, the command is simply ignored. Actual paper movement is determined by: (a) paper position before VT command, (b) the code used, and (c) the Vertical Motion Index (VMI).
Tabs Clear	Form: (Esc)**2** Notes: 1. This command clears all horizontal and vertical tabs.

Table A-3T **630 command set (continued)**

Command name	Form / Notes
Backward Printing Start	Form: (Esc)**6** Notes: 1. This command allows character data to be sent in reverse order, on a line-by-line basis; in other words, the first character received in a line is positioned at the right margin, while the last-received is positioned at the left margin. 2. **Internal justification (ESCM) is not implemented with backward printing.**
Backward Printing Stop	Form: (Esc)**5** Notes: 1. This command cancels backward printing.
Line Spacing*	Form: (Esc)**k***n* Notes: 1. This command multiplies the current font height by a number from $\frac{1}{2}$ to 3, to achieve variation in line spacing. 2. *n* is a number from 0 to 2: 0 = single line spacing (font height x 1) 1 = $1\frac{1}{2}$ line spacing (font height x $1\frac{1}{2}$) 2 = double line spacing (font height x 2)
Auto Line Feed On*	Form: (Esc)**"** Notes: 1. This command causes the printer to supply a line-feed code for every carriage return received.
Auto Line Feed Off*	Form: (Esc)**#** Notes: 1. This command cancels the Auto Line Feed On command.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3U **630 command set (continued)**

Command name	Form / Notes
Line Ending Decisions Start	Form: (Esc)**?** Notes: 1. This command initiates auto line-ending decisions, unless they have been selected by switch setting (A:3 and A:4 both ON).
Line Ending Decisions Stop	Form: (Esc)**!** Notes: 1. This command cancels auto line-ending decisions, unless they have been selected by switch setting (A:3 and A:4 both ON).
Half-Line Feed	Form: (Esc)**U** Notes: 1. This command causes the printer to move the print position **down** one-half line-feed. 2. The amount moved up is equal to one-half the current line height, depending on the line-spacing commands and/or electronic spacing being used.
Negative Line Feed	Form: (Esc)(LF) Notes: 1. This command causes the printer to move the print position **up** one line-feed. 2. The amount moved up is equal to the current line height, depending on whether line-spacing commands or electronic spacing is being used.
Negative Half-Line Feed	Form: (Esc)**D** Notes: 1. This command causes the printer to move the print position **up** one-half line-feed. 2. The amount moved up is equal to one-half the current line height.

Table A-3V **630 command set (continued)**

Command name	**Form / Notes**
Absolute Text Placement*	Form: (Esc)**p**n_1,n_2(line-end) Notes: 1. This command specifies the position on the page *relative to the page origin* where subsequent text is to begin printing. 2. n_1 specifies the x-coordinate of the new position. n_2 specifies the y-coordinate of the new position. 3. The point referenced is the upper left corner of the character cell in landscape orientation, or the left edge of the baseline, in portrait orientation.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3W **630 command set (continued)**

Command name	Form / Notes
Graphic Window*	Form: (Esc)**vw**n_1;n_2,n_3, n_4,n_5(line-end) *graphic data* Notes: 1. This command prepares the printer to create a graphic window for printing on the *current page*. 2. n_1 specifies the magnification for this graphic window; 2 specifies 2X magnification; 4 specifies 4X magnification; any other value specifies 1X magnification. n_2 is an optional field specifying the distance along the x-axis between the page origin and the graphic origin. If this field is omitted, the current print position is assumed as the graphic origin. n_3 is an optional field specifying the distance along the y-axis between the page origin and the graphic origin. If this field is omitted, the current print position is assumed as the graphic origin. n_4 specifies the size of the graphic, in dots, along its x-dimension. n_5 specifies the size of the graphic, in dots, along its y-dimension. 3. The use of this command is described in chapter 2 of this manual.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3X **630 command set (continued)**

Command name	Form / Notes
Repeat Window	Form: (Esc)**vr**n_1;n_2,n_3...(line-end) Notes: 1. This command repeats the last-stored graphic window for printing on the current page. 2. n_1 specifies the magnification for this graphic window. 2 specifies 2X magnification; 4 specifies 4X magnification; any other value specifies 1X magnification. n_2 is a variable-length numeric field specifying the distance along the x-axis between the page origin and the graphic origin. n_3 is a variable-length numeric field specifying the distance along the y-axis between the page origin and the graphic origin. 3. The use of this command is described in chapter 2 of this manual.

Table A-3 **630 command set (continued)**

Command name	Form / Notes
Draw Line*	Form: (Esc)Cn_1,n_2,n_3,n_4(line-end) Notes: 1. This command draws a solid black line on the page. 2. C is an **x** or a **y**. An **x** indicates that the line is to be parallel to the x-axis; a **y** indicates that it is to be parallel to the y-axis. 3. n_1 specifies the x-coordinate on the page, expressed in dots, where the line is to begin. n_2 specifies the y-coordinate, expressed in dots, where the line is to begin. n_3 specifies the length of the line, expressed in dots. n_4 specifies the thickness of the line, expressed in dots. 4. Drawing more than 1000 lines in either direction may result in a page too complex to print.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3 **630 command set (continued)**

Command name	Form / Notes
Justification Start	Form: (Esc)**M** Notes: 1. This command initiates text justification. Spaces between words are expanded or compressed so that each text line fits exactly between the left and right margins. Spaces are expanded to 150% of their normal width; there is no limit to how far they are compressed. Lines which cannot be justified within these parameters are not justified. 2. Justification begins on the line **following** the one in which the Justification Start command occurs, unless the command is at the beginning of the line. 3. Justification is canceled by the Cancel WP Features command. 4. This command is not implemented with Backward Printing.
Centering	Form: (Esc)= Notes: 1. This command centers the line of text in which it occurs between the left and right margins. If the line is too long to fit between the margins, it is printed so that it extends equally beyond the left and right margins. 2. This command must occur at the beginning of the line to take effect. 3. Only one line of text can be centered at a time with this command.

Table A-3AA **630 command set (continued)**

Command name	**Form / Notes**
Superscript Start*	Form: (Esc)**t** Notes: 1. This command raises the baseline of subsequent characters, so that they may be printed as superscripts. 2. If the superscript is to be printed in a smaller font, the smaller font should be invoked **after** the superscript command is given, since the distance the baseline is raised depends on the font in effect. 3. Superscripts cannot be nested; that is, a superscript command must be canceled before another can take effect.
Subscript Start*	Form: (Esc)**u** Notes: 1. This command lowers the baseline of subsequent characters, so that they may be printed as subscripts. 2. If the subscript is to be printed in a smaller font, the smaller font should be invoked **after** the subscript command is given, since the distance the baseline is lowered depends on the font in effect. 3. Subscripts cannot be nested; that is, a subscript command must be canceled before another can take effect.
Super/Subscript Stop*	Form: (Esc)**s** Notes: 1. This command cancels a super- or subscript command. 2. Super- and subscripting is also canceled by a carriage return, line feed, or form feed.

* Command(s) are enhanced 630 commands which give you added versatility with your printer. If your 630 driver does not recognize this command, it should be used with a UDK command.

Table A-3AB **630 command set (continued)**

Command name	Form / Notes
Underlining Start	Form: (Esc)**E** Notes: 1. This command initiates underlining. All characters, spaces, and tabs between this command and the Underlining Stop command will be underlined, except for leading and trailing spaces and initial tabs beginning a new line. (The text within the commands will be underlined even if it continues on a new line.) 2. The thickness of the underline within any line is determined by the size of the last font before the end of the line or before the Underlining Stop command. 3. Underlining always occurs at the baseline in effect when the command is given. If it is used with superscripted or subscripted text, the underlining remains at its initial baseline. To move underlining up or down with the super- or subscripted characters, stop the underlining before the super- or subscript, and renew the underlining after the super- or subscript.
Underlining Stop	Form: (Esc)**R** Notes: 1. This command cancels underlining. Underlining is also canceled by a Cancel WP Features command.

Table A-3AC **630 command set (continued)**

Command name	Form / Notes
Bolding Start	Form: (Esc)**W** or (Esc)**O** (uppercase letter O) Notes: 1. These commands iniatiate bolding (shadow printing). Every character printed between these commands and the Bolding Stop command is printed twice, the second character being displaced two dots to the right of the first, thus achieving a **bold** effect. A line-end stops shadow printing. 2. This command requires extra page composition processing and should be used with moderation; bold fonts are preferred where a lot of bold text is needed. 3. A carriage return (CR) turns bolding off.
Bolding Stop	Form: (Esc)**&** Notes: 1. This command cancels bolding (shadow printing). Bolding is also canceled by the Cancel WP Features command and a carriage return (CR).
Cancel WP Features	Form: (Esc)**X** Notes: 1. This command cancels the following functions: justification, underlining, bolding, and offset selection.

Table A-4A **Conversion tables—inches (quarter-inches printed in bold)**

Inches	60ths	300ths	Inches	60ths	300ths
0.10	6	30	3.10	186	930
0.20	12	60	3.20	192	960
0.25	**15**	**75**	**3.25**	**195**	**975**
0.30	18	90	3.30	198	990
0.40	24	120	3.40	204	1020
0.50	**30**	**150**	**3.50**	**210**	**1050**
0.60	36	180	3.60	216	1080
0.70	42	210	3.70	222	1110
0.75	**45**	**225**	**3.75**	**225**	**1125**
0.80	48	240	3.80	228	1140
0.90	54	270	3.90	234	1170
1.00	**60**	**300**	**4.00**	**240**	**1200**
1.10	66	330	4.10	246	1230
1.20	72	360	4.20	252	1260
1.25	**75**	**375**	**4.25**	**255**	**1275**
1.30	78	390	4.30	258	1290
1.40	84	420	4.40	264	1320
1.50	**90**	**450**	**4.50**	**270**	**1350**
1.60	96	480	4.60	276	1380
1.70	102	510	4.70	282	1410
1.75	**105**	**525**	**4.75**	**285**	**1425**
1.80	108	540	4.80	288	1440
1.90	114	570	4.90	294	1470
2.00	**120**	**600**	**5.00**	**300**	**1500**
2.10	126	630	5.10	306	1530
2.20	132	660	5.20	312	1560
2.25	**135**	**675**	**5.25**	**315**	**1575**
2.30	138	690	5.30	318	1590
2.40	144	720	5.40	324	1620
2.50	**150**	**750**	**5.50**	**330**	**1650**
2.60	156	780	5.60	336	1680
2.70	162	810	5.70	342	1710
2.75	**165**	**825**	**5.75**	**345**	**1725**
2.80	168	840	5.80	348	1740
2.90	174	870	5.90	354	1770
3.00	**180**	**900**	**6.00**	**360**	**1800**

Table A-4B **Conversion tables—inches (continued)**

(quarter-inches printed in bold)

Inches	60ths	300ths	Inches	60ths	300ths
6.10	366	1830	9.10	546	2730
6.20	372	1860	9.20	552	2760
6.25	**375**	**1875**	**9.25**	**555**	**2775**
6.30	378	1890	9.30	558	2790
6.40	384	1920	9.40	564	2820
6.50	**390**	**1950**	**9.50**	**570**	**2850**
6.60	396	1980	9.60	576	2880
6.70	402	2010	9.70	582	2910
6.75	**405**	**2025**	**9.75**	**585**	**2925**
6.80	408	2040	9.80	588	2940
6.90	414	2070	9.90	594	2970
7.00	**420**	**2100**	**10.00**	**600**	**3000**
7.10	426	2130	10.10	606	3030
7.20	432	2160	10.20	612	3060
7.25	**435**	**2175**	**10.25**	**615**	**3075**
7.30	438	2190	10.30	618	3090
7.40	444	2220	10.40	624	3120
7.50	**450**	**2250**	**10.50**	**630**	**3150**
7.60	456	2280	10.60	636	3180
7.70	462	2310	10.70	642	3210
7.75	**465**	**2325**	**10.75**	**645**	**3225**
7.80	468	2340	10.80	648	3240
7.90	474	2370	10.90	654	3270
8.00	**480**	**2400**	**11.00**	**660**	**3300**
8.10	486	2430	11.10	666	3330
8.20	492	2460	11.20	672	3360
8.25	**495**	**2475**	**11.25**	**675**	**3375**
8.30	498	2490	11.30	678	3390
8.40	504	2520	11.40	684	3420
8.50	**510**	**2550**	**11.50**	**690**	**3450**
8.60	516	2580	11.60	696	3480
8.70	522	2610	11.70	702	3510
8.75	**525**	**2625**	**11.75**	**705**	**3525**
8.80	528	2640	11.80	708	3540
8.90	534	2670	11.90	714	3570
9.00	**540**	**2700**	**12.00**	**720**	**3600**

Table A-4C **Conversion tables—inches (continued)**

(quarter-inches printed in bold)

Inches	60ths	300ths	Inches	60ths	300ths
12.10	726	3630	15.10	906	4530
12.20	732	3660	15.20	912	4560
12.25	**735**	**3675**	**15.25**	**915**	**4575**
12.30	738	3690	15.30	918	4590
12.40	744	3720	15.40	924	4620
12.50	**750**	**3750**	**15.50**	**930**	**4650**
12.60	756	3780	15.60	936	4680
12.70	762	3810	15.70	942	4710
12.75	**765**	**3825**	**15.75**	**945**	**4725**
12.80	768	3840	15.80	948	4740
12.90	774	3870	15.90	954	4770
13.00	**780**	**3900**	**16.00**	**960**	**4800**
13.10	786	3930	16.10	966	4830
13.20	792	3960	16.20	972	4860
13.25	**795**	**3975**	**16.25**	**975**	**4875**
13.30	798	3990	16.30	978	4890
13.40	804	4020	16.40	984	4920
13.50	**810**	**4050**	**16.50**	**990**	**4950**
13.60	816	4080	16.60	996	4980
13.70	822	4110	16.70	1002	5010
13.75	**825**	**4125**	**16.75**	**1005**	**5025**
13.80	828	4140	16.80	1008	5040
13.90	834	4170	16.90	1014	5070
14.00	**840**	**4200**	**17.00**	**1020**	**5100**
14.10	846	4230			
14.20	852	4260			
14.25	**855**	**4275**			
14.30	858	4290			
14.40	864	4320			
14.50	**870**	**4350**			
14.60	876	4380			
14.70	882	4410			
14.75	**885**	**4425**			
14.80	888	4440			
14.90	894	4470			
15.00	**900**	**4500**			

Table A-5A **Conversion tables—millimeters (centimeters printed in bold)**

mm	60ths	300ths	mm	60ths	300ths
2	5	24	72	170	850
4	9	47	74	175	874
6	14	71	76	180	898
8	19	94	78	184	921
10	**24**	**118**	**80**	**189**	**945**
12	28	142	82	194	969
14	33	165	84	198	992
16	38	189	86	203	1016
18	43	213	88	208	1039
20	**47**	**236**	**90**	**213**	**1063**
22	52	260	92	217	1087
24	57	283	94	222	1110
26	61	307	96	227	1134
28	66	331	98	232	1157
30	**71**	**354**	**100**	**236**	**1181**
32	76	378	102	241	1205
34	80	402	104	246	1228
36	85	425	106	250	1252
38	90	449	108	255	1276
40	**94**	**472**	**110**	**260**	**1299**
42	99	496	112	265	1323
44	104	520	114	269	1346
46	109	543	116	274	1370
48	113	567	118	279	1394
50	**118**	**591**	**120**	**283**	**1417**
52	123	614	122	288	1441
54	128	638	124	293	1465
56	132	661	126	298	1488
58	137	685	128	302	1512
60	**142**	**709**	**130**	**307**	**1535**
62	146	732	132	312	1559
64	151	756	134	317	1583
66	156	780	136	321	1606
68	161	803	138	326	1630
70	**165**	**827**	**140**	**331**	**1654**

Table A-5B **Conversion tables—millimeters (continued)**

(centimeters printed in bold)

mm	60ths	300ths	mm	60ths	300ths
142	335	1677	212	501	2504
144	340	1701	214	506	2528
146	345	1724	216	510	2551
148	350	1748	218	515	2575
150	**354**	**1772**	**220**	**520**	**2598**
152	359	1795	222	524	2622
154	364	1819	224	529	2646
156	369	1843	226	534	2669
158	373	1866	228	539	2693
160	**378**	**1890**	**230**	**543**	**2717**
162	383	1913	232	548	2740
164	387	1937	234	553	2764
166	392	1961	236	557	2787
168	397	1984	238	562	2811
170	**402**	**2008**	**240**	**567**	**2835**
172	406	2031	242	572	2858
174	411	2055	244	576	2882
176	416	2079	246	581	2906
178	420	2102	248	586	2929
180	**425**	**2126**	**250**	**591**	**2953**
182	430	2150	252	595	2976
184	435	2173	254	600	3000
186	439	2197	256	605	3024
188	444	2220	258	609	3047
190	**449**	**2244**	**260**	**614**	**3071**
192	454	2268	262	619	3094
194	458	2291	264	624	3118
196	463	2315	266	628	3142
198	468	2339	268	633	3165
200	**472**	**2362**	**270**	**638**	**3189**
202	477	2386	272	643	3213
204	482	2409	274	647	3236
206	487	2433	276	652	3260
208	491	2457	278	657	3283
210	**496**	**2480**	**280**	**661**	**3307**

Table A-5C **Conversion tables—millimeters (continued)**

(centimeters printed in bold)

mm	60ths	300ths	mm	60ths	300ths
282	666	3331	352	832	4157
284	671	3354	354	836	4181
286	676	3378	356	841	4205
288	680	3402	358	846	4228
290	**685**	**3425**	**360**	**850**	**4252**
292	690	3449	362	855	4276
294	695	3472	364	860	4299
296	699	3496	366	865	4323
298	704	3520	368	869	4346
300	**709**	**3543**	**370**	**874**	**4370**
302	713	3567	372	879	4394
304	718	3591	374	883	4417
306	723	3614	376	888	4441
308	728	3638	378	893	4465
310	**732**	**3661**	**380**	**898**	**4488**
312	737	3685	382	902	4512
314	742	3709	384	907	4535
316	746	3732	386	912	4559
318	751	3756	388	917	4583
320	**756**	**3780**	**390**	**921**	**4606**
322	761	3803	392	926	4630
324	765	3827	394	931	4654
326	770	3850	396	935	4677
328	775	3874	398	940	4701
330	**780**	**3898**	**400**	**945**	**4724**
332	784	3921	402	950	4748
334	789	3945	404	954	4772
336	794	3968	406	959	4795
338	798	3992	408	964	4819
340	**803**	**4016**	**410**	**969**	**4843**
342	808	4039	412	973	4866
344	813	4063	414	978	4890
346	817	4087	416	983	4913
348	822	4110	418	987	4937
350	**827**	**4134**	**420**	**992**	**4961**

Table A-6A **Hexadecimal-Decimal-Binary equivalences**

Hex	Decimal	Binary	Hex	Decimal	Binary
00	0	00000000	20	32	00100000
01	1	00000001	21	33	00100001
02	2	00000010	22	34	00100010
03	3	00000011	23	35	00100011
04	4	00000100	24	36	00100100
05	5	00000101	25	37	00100101
06	6	00000110	26	38	00100110
07	7	00000111	27	39	00100111
08	8	00001000	28	40	00101000
09	9	00001001	29	41	00101001
0A	10	00001010	2A	42	00101010
0B	11	00001011	2B	43	00101011
0C	12	00001100	2C	44	00101100
0D	13	00001101	2D	45	00101101
0E	14	00001110	2E	46	00101110
0F	15	00001111	2F	47	00101111
10	16	00010000	30	48	00110000
11	17	00010001	31	49	00110001
12	18	00010010	32	50	00110010
13	19	00010011	33	51	00110011
14	20	00010100	34	52	00110100
15	21	00010101	35	53	00110101
16	22	00010110	36	54	00110110
17	23	00010111	37	55	00110111
18	24	00011000	38	56	00111000
19	25	00011001	39	57	00111001
1A	26	00011010	3A	58	00111010
1B	27	00011011	3B	59	00111011
1C	28	00011100	3C	60	00111100
1D	29	00011101	3D	61	00111101
1E	30	00011110	3E	62	00111110
1F	31	00011111	3F	63	00111111

Table A-6B **Hexadecimal-Decimal-Binary equivalences (continued)**

Hex	Decimal	Binary	Hex	Decimal	Binary
40	64	01000000	60	96	01100000
41	65	01000001	61	97	01100001
42	66	01000010	62	98	01100010
43	67	01000011	63	99	01100011
44	68	01000100	64	100	01100100
45	69	01000101	65	101	01100101
46	70	01000110	66	102	01100110
47	71	01000111	67	103	01100111
48	72	01001000	68	104	01101000
49	73	01001001	69	105	01101001
4A	74	01001010	6A	106	01101010
4B	75	01001011	6B	107	01101011
4C	76	01001100	6C	108	01101100
4D	77	01001101	6D	109	01101101
4E	78	01001110	6E	110	01101110
4F	79	01001111	6F	111	01101111
50	80	01010000	70	112	01110000
51	81	01010001	71	113	01110001
52	82	01010010	72	114	01110010
53	83	01010011	73	115	01110011
54	84	01010100	74	116	01110100
55	85	01010101	75	117	01110101
56	86	01010110	76	118	01110110
57	87	01010111	77	119	01110111
58	88	01011000	78	120	01111000
59	89	01011001	79	121	01111001
5A	90	01011010	7A	122	01111010
5B	91	01011011	7B	123	01111011
5C	92	01011100	7C	124	01111100
5D	93	01011101	7D	125	01111101
5E	94	01011110	7E	126	01111110
5F	95	01011111	7F	127	01111111

Table A-6C **Hexadecimal-Decimal-Binary equivalences (continued)**

Hex	Decimal	Binary	Hex	Decimal	Binary
80	128	10000000	A0	160	10100000
81	129	10000001	A1	161	10100001
82	130	10000010	A2	162	10100010
83	131	10000011	A3	163	10100011
84	132	10000100	A4	164	10100100
85	133	10000101	A5	165	10100101
86	134	10000110	A6	166	10100110
87	135	10000111	A7	167	10100111
88	136	10001000	A8	168	10101000
89	137	10001001	A9	169	10101001
8A	138	10001010	AA	170	10101010
8B	139	10001011	AB	171	10101011
8C	140	10001100	AC	172	10101100
8D	141	10001101	AD	173	10101101
8E	142	10001110	AE	174	10101110
8F	143	10001111	AF	175	10101111
90	144	10010000	B0	176	10110000
91	145	10010001	B1	177	10110001
92	146	10010010	B2	178	10110010
93	147	10010011	B3	179	10110011
94	148	10010100	B4	180	10110100
95	149	10010101	B5	181	10110101
96	150	10010110	B6	192	10110110
97	151	10010111	B7	183	10110111
98	152	10011000	B8	184	10111000
99	153	10011001	B9	185	10111001
9A	154	10011010	BA	186	10111010
9B	155	10011011	BB	187	10111011
9C	156	10011100	BC	188	10111100
9D	157	10011101	BD	189	10111101
9E	158	10011110	BE	190	10111110
9F	159	10011111	BF	191	10111111

Table A-6D **Hexadecimal-Decimal-Binary equivalences (continued)**

Hex	Decimal	Binary	Hex	Decimal	Binary
C0	192	11000000	E0	224	11100000
C1	193	11000001	E1	225	11100001
C2	194	11000010	E2	226	11100010
C3	195	11000011	E3	227	11100011
C4	196	11000100	E4	228	11100100
C5	197	11000101	E5	229	11100101
C6	198	11000110	E6	230	11100110
C7	199	11000111	E7	231	11100111
C8	200	11001000	E8	232	11101000
C9	201	11001001	E9	233	11101001
CA	202	11001010	EA	234	11101010
CB	203	11001011	EB	235	11101011
CC	204	11001100	EC	236	11101100
CD	205	11001101	ED	237	11101101
CE	206	11001110	EE	238	11101110
CF	207	11001111	EF	239	11101111
D0	208	11010000	F0	240	11110000
D1	209	11010001	F1	241	11110001
D2	210	11010010	F2	242	11110010
D3	211	11010011	F3	243	11110011
D4	212	11010100	F4	244	11110100
D5	213	11010101	F5	245	11110101
D6	214	11010110	F6	246	11110110
D7	215	11010111	F7	247	11110111
D8	216	11011000	F8	248	11111000
D9	217	11011001	F9	249	11111001
DA	218	11011010	FA	250	11111010
DB	219	11011011	FB	251	11111011
DC	220	11011100	FC	252	11111100
DD	221	11011101	FD	253	11111101
DE	222	11011110	FE	254	11111110
DF	223	11011111	FF	255	11111111

Table A-7A **Double hex values**

X	2X	X	2X	X	2X
20	40	40	80	60	C0
21	42	41	82	61	C2
22	44	42	84	62	C4
23	46	43	86	63	C6
24	48	44	88	64	C8
25	4A	45	8A	65	CA
26	4C	46	8C	66	CC
27	4E	47	8E	67	CE
28	50	48	90	68	D0
29	52	49	92	69	D2
2A	54	4A	94	6A	D4
2B	56	4B	96	6B	D6
2C	58	4C	98	6C	D8
2D	5A	4D	9A	6D	DA
2E	5C	4E	9C	6E	DC
2F	5E	4F	9E	6F	DE
30	60	50	A0	70	E0
31	62	51	A2	71	E2
32	64	52	A4	72	E4
33	66	53	A6	73	E6
34	68	54	A8	74	E8
35	6A	55	AA	75	EA
36	6C	56	AC	76	EC
37	6E	57	AE	77	EE
38	70	58	B0	78	F0
39	72	59	B2	79	F2
3A	74	5A	B4	7A	F4
3B	76	5B	B6	7B	F6
3C	78	5C	B8	7C	F8
3D	7A	5D	BA	7D	FA
3E	7C	5E	BC	7E	FC
3F	7E	5F	BE	7F	FE

Table A-7B **Double hex values (continued)**

X	2X	X	2X	X	2X
80	100	A0	140	C0	180
81	102	A1	142	C1	182
82	104	A2	144	C2	184
83	106	A3	146	C3	186
84	108	A4	148	C4	188
85	10A	A5	14A	C5	18A
86	10C	A6	14C	C6	18C
87	10E	A7	14E	C7	18E
88	110	A8	150	C8	190
89	112	A9	152	C9	192
8A	114	AA	154	CA	194
8B	116	AB	156	CB	196
8C	118	AC	158	CC	198
8D	11A	AD	15A	CD	19A
8E	11C	AE	15C	CE	19C
8F	11E	AF	15E	CF	19E
90	120	B0	160	D0	1A0
91	122	B1	162	D1	1A2
92	124	B2	164	D2	1A4
93	126	B3	166	D3	1A6
94	128	B4	168	D4	1A8
95	12A	B5	16A	D5	1AA
96	12C	B6	16C	D6	1AC
97	12E	B7	16E	D7	1AE
98	130	B8	170	D8	1B0
99	132	B9	172	D9	1B2
9A	134	BA	174	DA	1B4
9B	136	BB	176	DB	1B6
9C	138	BC	178	DC	1B8
9D	13A	BD	17A	DD	1BA
9E	13C	BE	17C	DE	1BC
9F	13E	BF	17E	DF	1BE

Table A-7C **Double hex values (continued)**

X	2X	X	2X
E0	220	F0	1E0
E1	1C2	F1	1E2
E2	1C4	F2	1E4
E3	1C6	F3	1E6
E4	1C8	F4	1E8
E5	1CA	F5	1EA
E6	1CC	F6	1EC
E7	1CE	F7	1EE
E8	1D0	F8	1F0
E9	1D2	F9	1F2
EA	1D4	FA	1F4
EB	1D6	FB	1F6
EC	1D8	FC	1F8
ED	1DA	FD	1FA
EE	1DC	FE	1FC
EF	1DE	FF	1FE

Table A-8 **Display codes**

Code	Meaning
The following group of codes report user-remediable conditions. Printing is halted until the condition is corrected:	
1L-9L	Top cover is not closed properly
A1	Paper misfeed in optional copier
C1	Paper pick-up cams misaligned
C3	Paper tray removed
C4	Paper path problem
E2	Paper path problem
E3	Paper path problem
E4	Paper path problem
E5	Cover open
The following group of codes report temporary operational states:	
A2	Page in optional copier waiting to feed
A3	Copy attempted with envelope tray in place. Replace with regular tray
AA	Printer is in the data monitor mode
Ld	Printer is receiving graphic data
LF	Printer is loading font data or a constant page
PE	Parity error (serial interface). An error character is printed. Pushing the Reset switch clears the code from the display.
The following group of codes report fault conditions that halt printing and may be user-remediable. See Operating the Model 50 for recommended actions:	
L2	Scanning failure
L3	Scanning failure
L5	Laser beam off
P1	Cartridge problem
P5	Overrun error (seral interface)
P6	Overrun error (serial interface)
The following group of codes report fault conditions that halt printing and require repair by your service organization:	
Blank	If the display remains blank after power-up and no configuration sheet is printed, there is a fatal failure in the processing portions of memory.
d3-d9	Processing module failure

Table A-9A **Job error codes**

Code	Meaning
The following code reports an error associated with a command:	
02	An (Esc)+ was received in the previous job, followed by an invalid character; all codes up to the next line-end were ignored.
The following codes report errors associated with the loading of font data:	
11	A problem was detected within font data. The font load was abandoned.
12	Available font memory was exhausted. Unloadable fonts were ignored (only whole fonts were stored).
18	A Font Unload command was received, which named a font not in storage. The command was ignored.
19	A Font Unload command was received, which named a resident font. The command was ignored.
The following codes report errors associated with print data:	
20	An escape code (or UDK) followed by an invalid character was received. The apparent instruction was ignored.
21	This page could not be printed, due to its complexity. There was either too much data on the page, or the images were too complex to print.
22	A print line extended beyond the right margin. Possible data loss.
26	More than nine errors were detected in the previous job.
28	An error was detected in a table download; the download was abandoned or may have produced unexpected results.
29	A parameter within a command was illegal or invalid; the command was ignored.

Table A-9B **Job error codes (continued)**

Code	Meaning
The following codes report errors associated with font usage:	
40	A Font ID Assignment attempted to assign an ID to a font not in storage. The command was ignored.
41	During page composition, a font was invoked whose orientation was opposite that of the page being composed. A new page was started. **Note:** The above does not apply to merge-pages.
42	A Font Change command was received, whose ID number had not been assigned. The command was ignored.
45	A Font ID Assignment contained an ID number greater than 9. The assignment was ignored.
49	A cartridge checksum error was found.
The following codes report errors associated with the use of graphic data:	
80	Graphic could not be printed beyond the physical limits of the page.
81	The S_x value was not an even multiple of 8; S_x was rounded down.
82	The graphic command was incomplete; a partial image may have been printed.
83	The sixel-encoded data contained a repetition value in excess of 32,767. All the data was ignored.
84	The Repeat Graphic Window command was received when no graphic window was defined.
85	Only a portion of the graphic data was loaded, due to insufficient storage area.

B. Codes and mapping

This appendix provides the descriptions of the control codes and tables of all the coding schemes supported by the Model 50. The following tables are included.

Table B-1A **Meaningful control codes**

Code name	ISO value	EBCDIC value	Description
NUL	00	00	**Null**. Used for filling in time over a communication line or space on storage media.
ETX	03	03	**End of text**. Signals the end of a block of data. ETX prompts a response from the Model 50, if the ETX/ACK protocol is active.
ACK	06	2E	**Positive acknowledgement**. The Model 50 transmits this code to the host upon receipt of an ETX, if the block of data has been successfully received and the ETX/ACK protocol is active.
BEL	07	2F	**Bell**. Causes the chime to sound, if this option has been enabled by switch setting.
BS	08	16	**Backspace**. Causes printer to move the print position one space to the left. In the 2700 mode, BS has this effect only if it is the first code in the print line.
HT	09	05	**Horizontal tab**. Causes the print position to be moved horizontally to a predetermined location on the page.
LF	0A	25	**Line feed**. Causes the print position to be moved down one line-height. This code may be used as a valid line-ending code, performing both the carriage return and line feed functions, and as a command terminator.
VT	0B	0B	**Vertical tab**. Causes the print position to be moved vertically to a predetermined location on the page.

Table B-1B **Meaningful control codes (continued)**

Code name	ISO value	EBCDIC value	Description
FF	0C	0C	**Form feed**. Causes the printer to eject the current page and begin a new page. In systems where output is printed on continuous-form paper, this code is often inserted between jobs to facilitate the separation of jobs. Redundant form feeds are ignored by the Model 50.
CR	0D	0D	**Carriage return**. Causes the printer to return to the first print position of the current line, or may be used as a valid line-ending, performing both the carriage return and line feed functions.
SO	0E	0E	**Shift out.** *In the 630 mode,* this code is used with the 7-bit data to print font characters in the X'80' to 'FE' range. *With IBM PC encoding,* this code is used to print font characters in the X'01' to '1F' range.
SI	0F	0F	**Shift in.** *In the 630 mode,* this code terminates access to the X'80' to 'FE' range. *With IBM PC encoding,* this code terminates access to the X'01' to '1F' range.
DC1	11	11	One of four device control codes (DC1, DC2, DC3, and DC4). If the XON/XOFF protocol option has been selected with the serial asynchronous interface, the Model 50 transmits a DC1 code to the host to signal its readiness to receive data. If received from the host, this code is ignored. This protocol is described in greater detail in chapter 3.

Table B-1C **Meaningful control codes (continued)**

Code name	ISO value	EBCDIC value	Description
DC2	12	12	*In the 630 mode only,* this code is recognized as part of certain command sequences.
DC3	13	13	If the XON/XOFF protocol option has been selected with the asynchronous interface, the Model 50 transmits this code to the host when certain fault conditions have occurred, as described in chapter 3.
DC4	14	3C	*In the 630 mode only,* this code is recognized as part of certain command sequences.
NAK	15	3D	**Negative acknowledgement.** The Model 50 transmits this code to the host in response to an ETX, if the ETX/ACK protocol is active, to indicate that the last block of data was not successfully received.
EM	19	19	**End of media.** Originally used to signal the end of a punched card. In the Model 50 *with IBM PC encoding,* this code is used with 7-bit data to print characters in the X'80' to 'FE' range, on a character-by-character basis.
SUB	1A	3F	**Substitute.** *In the 630 mode only,* this code is recognized as part of certain command sequences.
ESC	1B	27	**Escape.** This character gives subsequent codes alternate meanings. Usually initiates remote device control.

Table B-1D **Meaningful control codes (continued)**

Code name	ISO value	EBCDIC value	Description
GS	1D	1D	**Group separator**. Normally used as a data-block delimiter, in the Model 50 this code is used *with the IBM PC encoding*, to print characters in the X'01' to '1F' range on a character-by-character basis.
RS	1E	1E	**Record separator**. Normally used as a data-block delimiter, in the Model 50 this code is recognized as part of certain command sequences.
US	1F	1F	**Unit separator.** *In the 630 mode only*, this code is recognized as part of certain command sequences.
DEL	7F	07	**Delete**. Indicates that subsequent data is to be ignored. In the Model 50, this code prints a font character, if one is assigned to this code in the font used; otherwise, it prints an error character in 2700 mode. ('7F' is ignored in the 630 mode.)
Other codes			A number of other control codes are included in the ISO, ASCII, EBCDIC, and IBM PC encoding schemes that are designed to invoke various terminal functions, to control device operation, or to regulate bisynchronous communication. Only the codes described in this table are supported by the Model 50.

The ISO encoding scheme table begins on the next page.

Table B-2A **ISO encoding scheme**

Most →

Least ↓	0	1	2	3	4	5	6	7
0	NUL	dle	Sp	0	@	P	`	p
1	soh	DC1	!	1	A	Q	a	q
2	stx	DC2	"	2	B	R	b	r
3	ETX	DC3	#	3	C	S	c	s
4	eot	DC4	¤	4	D	T	d	t
5	enq	NAK	%	5	E	U	e	u
6	ACK	syn	&	6	F	V	f	v
7	BEL	etb	'	7	G	W	g	w
8	BS	can	(	8	H	X	h	x
9	HT	EM	)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[	k	{
C	FF	fs	,	<	L	\	l	|
D	CR	GS	-	=	M	]	m	}
E	SO	RS	.	>	N	^	n	~
F	SI	US	/	?	O	_	o	del

Notes: 1. Code names in lowercase indicate codes that have no significance to the Model 50.
2. Underscored code names indicate codes that have significance **in some modes only**.
3. Font characters marked with an asterisk are present only in the resident fonts, unless specially ordered.

Table B-2B **ISO encoding scheme (continued)**

Most →

Least ↓	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0				°		–	Ω	ĸ
1	`		¡	±	`	¹	Æ	æ
2	´		¢	²	´	®	Đ	đ
3	^		£	³	^	©	ª	ð
4	~		$	×	~	™	Ħ	ħ
5	¯		¥	µ	¯	♪		ı
6	˘		#	¶	˘		Ĳ	ĳ
7	˙		§	·	˙		Ŀ	ŀ
8	¨	¬*	¤	÷	¨		Ł	ł
9	¨	¦*	'	'	¨		Ø	ø
A	˚		"	"	˚		Œ	œ
B	¸		«	»	¸		º	ß
C	_		←	¼	_	⅛	Þ	þ
D	˝		↑	½	˝	⅜	Ŧ	ŧ
E	˛		→	¾	˛	⅝	Ŋ	ŋ
F	ˇ		↓	¿	ˇ	⅞	ŉ	

Equivalence table

Hex	Binary	Hex	Binary	Hex	Binary	Hex	Binary
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

Table B-3 **ASCII character mapping, 2700 mode**

U.S. English

Most →

Least ↓	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**
0			Sp	0	@	P	`	p				
1			!	1	A	Q	a	q				
2			"	2	B	R	b	r			¢	
3			#	3	C	S	c	s				
4			$	4	D	T	d	t			$*	
5			%	5	E	U	e	u				
6			&	6	F	V	f	v				
7			'	7	G	W	g	w				
8			(	8	H	X	h	x		¬*		
9			)	9	I	Y	i	y		¦*		
A			*	:	J	Z	j	z				
B			+	;	K	[	k	{				
C			,	<	L	\	l	|				¼
D			-	=	M	]	m	}				½
E			.	>	N	^	n	~				
F			/	?	O	_	o					

Note:	Font characters marked with an asterisk are present only in the resident fonts, unless specially ordered.

Table B-4 **ASCII character mapping, 2700 mode**

U.K. English

Most →
Least ↓

	0	1	2	3	4	5	6	7
0			Sp	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	\	l	|
D			-	=	M	]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

Table B-5 **ASCII character mapping, 2700 mode**

French

Least ↓ / Most →	0	1	2	3	4	5	6	7
0			Sp	0	à	P	`	p
1			!	1	A	Q	a	q
2			¢	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	°	k	é
C			,	<	L	ç	l	ù
D			-	=	M	§	m	è
E			.	>	N	^	n	~
F			/	?	O	_	o	

Table B-6 **ASCII character mapping, 2700 mode**

Dutch

Most →

Least ↓	0	1	2	3	4	5	6	7
0			Sp	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	\	l	|
D			–	=	M	]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

Table B-7 **ASCII character mapping, 2700 mode**

Spanish

Most →

Least ↓

	0	1	2	3	4	5	6	7
0			Sp	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Ã	k	ã
C			,	<	L	Ñ	l	ñ
D			-	=	M	Õ	m	õ
E			.	>	N	Ç	n	ç
F			/	?	O	_	o	

Table B-8 **ASCII character mapping, 2700 mode**

Italian

Most →

Least ↓	0	1	2	3	4	5	6	7
0			Sp	0	§	P	ù	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	°	k	à
C			,	<	L	é	l	ò
D			-	=	M	|	m	è
E			.	>	N	^	n	ì
F			/	?	O	_	o	

Table B-9 **ASCII character mapping, 2700 mode**

Danish

Most →

Least ↓	0	1	2	3	4	5	6	7
0			Sp	0	É	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			¤	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Æ	k	æ
C			,	<	L	Ø	l	ø
D			-	=	M	Å	m	å
E			.	>	N	Ü	n	ü
F			/	?	O	_	o	é

Table B-10 **ASCII character mapping, 2700 mode**

Norwegian

Least ↓ / Most →	0	1	2	3	4	5	6	7
0			Sp	0	É	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			¤	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Æ	k	æ
C			,	<	L	Ø	l	ø
D			–	=	M	Å	m	å
E			.	>	N	Ü	n	ü
F			/	?	O	_	o	é

Table B-11 **ASCII character mapping, 2700 mode**

Finnish

Most →
Least ↓

	0	1	2	3	4	5	6	7
0			Sp	0	É	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			¤	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Ä	k	ä
C			,	<	L	Ö	l	ö
D			-	=	M	Å	m	å
E			.	>	N	Ü	n	ü
F			/	?	O	_	o	é

Table B-12 **ASCII character mapping, 2700 mode**

German

Most →

Least ↓	0	1	2	3	4	5	6	7
0			Sp	0	§	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Ä	k	ä
C			,	<	L	Ö	l	ö
D			-	=	M	Ü	m	ü
E			.	>	N	^	n	ß
F			/	?	O	_	o	

Table B-13 **ASCII character mapping, 2700 mode**

Swedish

Most →

Least ↓

	0	1	2	3	4	5	6	7
0			Sp	0	É	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			¤	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Ä	k	ä
C			,	<	L	Ö	l	ö
D			–	=	M	Å	m	å
E			.	>	N	Ü	n	ü
F			/	?	O	_	o	é

The ASCII, 2700 Mode, French Canadian character mapping begins on the next page.

Table B-14A **ASCII character mapping, 2700 mode**

French Canadian

Least ↓ / Most →	0	1	2	3	4	5	6	7
0			Sp	0	à	P	ô	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	â	k	é
C			,	<	L	ç	l	ù
D			-	=	M	ê	m	è
E			.	>	N	î	n	û
F			/	?	O	_	o	

Table B-14B **ASCII character mapping, 2700 mode**

French Canadian (continued)

Most →

Least ↓	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0		[	À	°		–	Ω	ĸ
1	`	\	¡	±	`	¹	Æ	æ
2	´	]	¢	²	´	®	Đ	đ
3	^	^	£	³	^	©	ª	ð
4	~	{	$	Î	~	™	Ħ	ħ
5	¯	|	¥	µ	¯	♪		ı
6	˘	}	#	Ï	˘		Ĳ	ĳ
7	˙	~	§	·	˙		Ŀ	ŀ
8	¨	¬*	¤	÷	¨		Ł	ł
9	¨	¦*	'	'	¨		Ø	ø
A	˚	É	"	"	˚		Œ	Û
B	¸	È	«	»	¸		º	Ü
C	_	Ê	Â	¼	_	⅛	ë	ï
D	˝	Ë	↑	½	˝	⅜	Ô	ŧ
E	˛		→	¾	˛	⅝	Ŋ	ü
F	ˇ		↓	Ç	ˇ	⅞	Ù	

Table B-15 **ASCII character mapping, 2700 mode**

Latin American

Most →

Least ↓	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0			Sp	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Ã	k	ã
C			,	<	L	Ñ	l	ñ
D			-	=	M	Õ	m	õ
E			.	>	N	Ç	n	ç
F			/	?	O	_	o	

Table B-16 **ASCII character mapping, 630 mode**

U.S. English

Least ↓ / Most →	0	1	2	3	4	5	6	7
0			Sp	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	\	l	¦*
D			-	=	M	]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

Note:	Font characters marked with an asterisk are present only in the resident fonts, unless specially ordered.

Table B-17 **ASCII character mapping, 630 mode**

U.K. English

Most →

Least ↓	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0			Sp	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	\	l	|
D			-	=	M	]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

Table B-18 **ASCII character mapping, 630 mode**

French

Most →

Least ↓

	0	1	2	3	4	5	6	7
0			Sp	0	à	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	°	k	é
C			,	<	L	ç	l	ù
D			–	=	M	§	m	è
E			.	>	N	^	n	¨
F			/	?	O	_	o	

Table B-19 **ASCII character mapping, 630 mode**

Dutch

Least ↓ / Most →	0	1	2	3	4	5	6	7
0			Sp	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	[	k	{
C			,	<	L	Ĳ	l	ĳ
D			-	=	M	]	m	}
E			.	>	N	^	n	~
F			/	?	O	_	o	

Table B-20 **ASCII character mapping, 630 mode**

Spanish

Least ↓ / Most →	0	1	2	3	4	5	6	7
0			Sp	0	§	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	¡	k	º
C			,	<	L	Ñ	l	ñ
D			–	=	M	¿	m	ç
E			.	>	N	^	n	~
F			/	?	O	_	o	

Table B-21 **ASCII character mapping, 630 mode**

Italian

Most →

Least ↓

	0	1	2	3	4	5	6	7
0			Sp	0	§	P	ù	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	°	k	à
C			,	<	L	ç	l	ò
D			-	=	M	é	m	è
E			.	>	N	^	n	ì
F			/	?	O	_	o	

Table B-22 **ASCII character mapping, 630 mode**

Danish

Least ↓ / Most →	0	1	2	3	4	5	6	7
0			Sp	0	É	P	é	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Æ	k	æ
C			,	<	L	Ø	l	ø
D			–	=	M	Å	m	å
E			.	>	N	Ü	n	ü
F			/	?	O	_	o	

Table B-23 **ASCII character mapping, 630 mode**

Norwegian

Most →

Least ↓	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0			Sp	0	@	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Æ	k	æ
C			,	<	L	Ø	l	ø
D			-	=	M	Å	m	å
E			.	>	N	^	n	~
F			/	?	O	_	o	

Table B-24 **ASCII character mapping, 630 mode**

Finnish

Most →

Least ↓	0	1	2	3	4	5	6	7
0			Sp	0		P		p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3				3	C	S	c	s
4				4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Ä	k	ä
C			,	<	L	Ö	l	ö
D			-	=	M	Å	m	å
E			.	>	N		n	
F			/	?	O	_	o	

Table B-25 **ASCII character mapping, 630 mode**

German

Most →

Least ↓	0	1	2	3	4	5	6	7
0			Sp	0	§	P	`	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Ä	k	ä
C			,	<	L	Ö	l	ö
D			-	=	M	Ü	m	ü
E			.	>	N	^	n	ß
F			/	?	O	_	o	

Table B-26 **ASCII character mapping, 630 mode**

Swedish

Most →

Least ↓	0	1	2	3	4	5	6	7
0			Sp	0	É	P	é	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			¤	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Ä	k	ä
C			,	<	L	Ö	l	ö
D			-	=	M	Å	m	å
E			.	>	N	Ü	n	ü
F			/	?	O	_	o	

Table B-27 **ASCII character mapping, 630 mode**

French Canadian

Least ↓ / Most →	0	1	2	3	4	5	6	7
0			Sp	0	à	P	ô	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			#	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	â	k	é
C			,	<	L	ç	l	ù
D			-	=	M	ê	m	è
E			.	>	N	î	n	û
F			/	?	O	_	o	

Table B-28 **ASCII character mapping, 630 mode**

Portuguese

Least ↓ / Most →	0	1	2	3	4	5	6	7
0			Sp	0	ª	P	º	p
1			!	1	A	Q	a	q
2			"	2	B	R	b	r
3			£	3	C	S	c	s
4			$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			'	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
A			*	:	J	Z	j	z
B			+	;	K	Ã	k	ã
C			,	<	L	Ç	l	ç
D			-	=	M	Õ	m	õ
E			.	>	N	^	n	¨
F			/	?	O	_	o	

Table B-29A **EBCDIC-to-ISO code translation**

Most →

Least ↓	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	00	10	00	00	20	26	2D	D5
1	01	11	00	00	A3	BA	2F	9A
2	02	12	1C	16	A4	BB	C7	9B
3	03	13	00	00	A5	C0	C8	9C
4	00	00	00	00	A6	C1	C9	9D
5	09	06	0A	1E	A7	BD	D0	E0
6	00	08	17	00	A8	BC	D1	E2
7	7F	00	1B	04	F8	C4	D2	E3
8	00	00	18	00	AA	C5	D3	E4
9	00	19	00	00	AB	C6	D4	60
A	00	00	00	00	A2	21	99	3A
B	0B	00	00	00	2E	24	2C	23
C	0C	00	00	14	3C	2A	25	40
D	0D	1D	05	15	28	29	5F	27
E	0E	1E	06	00	2B	3B	3E	3D
F	0F	1F	07	1A	7C	98	3F	22

Note: The codes **within** the table represent the ISO values which correspond to the EBCDIC values on the **outside** of the grid.

Table B-29B **EBCDIC-to-ISO code translation (continued)**

Most →

Least ↓

	8	9	A	B	C	D	E	F
0	B5	F1	A0	B0	7B	7D	5C	30
1	61	6A	7E	B1	41	4A	E1	31
2	62	6B	73	B2	42	4B	53	32
3	63	6C	74	B3	43	4C	54	33
4	64	6D	75	B4	44	4D	55	34
5	65	6E	76	5E	45	4E	56	35
6	66	6F	77	B6	46	4F	57	36
7	67	70	78	B7	47	50	58	37
8	68	71	79	B8	48	51	59	38
9	69	72	7A	B9	49	52	5A	39
A	A1	F2	AD	F9	CA	9E	EA	FA
B	E6	F3	C2	C3	CB	9F	EB	FB
C	E7	F4	AC	A9	CC	DC	EC	FC
D	E8	F5	5B	5D	CD	DD	ED	FD
E	E9	F6	AE	BE	CE	DE	EE	FE
F	F0	F7	AF	BF	CF	DF	EF	FF

Equivalence table

Hex	Binary	Hex	Binary	Hex	Binary	Hex	Binary
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

Table B-30A **ISO-to-EBCDIC equivalence**

Most →
Least ↓

	0	1	2	3	4	5	6	7
0	00	01	40	F0	7C	D7	79	97
1	01	11	5A	F1	C1	D8	81	98
2	02	12	7F	F2	C2	D9	82	99
3	03	13	7B	F3	C3	E2	83	A2
4	37	3C	5B	F4	C4	E3	84	A3
5	2D	3D	6C	F5	C5	E4	85	A4
6	15/2E	32	50	F6	C6	E5	86	A5
7	2F	26	7D	F7	C7	E6	87	A6
8	16	18	4D	F8	C8	E7	88	A7
9	05	19	5D	F9	C9	E8	89	A8
A	25	3F	5C	7A	D1	E9	91	A9
B	0B	27	4E	5E	D2	AD	92	C0
C	0C	22	6B	4C	D3	E0	93	4F
D	0D	1D	60	7E	D4	BD	94	D0
E	0E	1E/35	4B	6E	D5	B5	95	A1
F	0F	1F	61	6F	D6	6D	96	07

Note: The codes **within** the table represent the EBCDIC values which correspond to the ISO values on the **outside** of the grid.

Table B-30B **ISO-to-EBCDIC equivalence (continued)**

Most →, Least ↓

	8	9	A	B	C	D	E	F
0			A0	B0	53	65	75	8F
1			8A	B1	54	66	E1	90
2			4A	B2	AB	67	76	9A
3			41	B3	BB	68	77	9B
4			42	B4	57	69	78	9C
5			43	80	58	70		9D
6			44	B6	59		8B	9E
7			45	B7	62		8C	9F
8		5F	46	B8	63		8D	47
9		6A	BC	B9	64		8E	BA
A		71	48	51	CA		EA	FA
B		72	49	52	CB		EB	FB
C		73	AC	56	CC	DC	EC	FC
D		74	AA	55	CD	DD	ED	FD
E		DA	AE	BE	CE	DE	EE	FE
F		DB	AF	BF	CF	DF	EF	FF

Equivalence table

Hex	Binary	Hex	Binary	Hex	Binary	Hex	Binary
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

Table B-31A **EBCDIC encoding scheme**

U.S. English

Most →

Least ↓	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
0	NUL	dle	00	00	Sp	&	–	♪
1	soh	DC1	00	00	£	"	/	
2	stx	DC2	fs	syn	$	»	˙	
3	ETX	DC3	00	00	¥		¨	
4	00	00	00	00	#	`	¨	
5	HT	ACK	LF	RS	§	½	–	Ω
6	00	BS	etb	00	¤	¼	¹	Đ
7	del	00	ESC	eot	Ӏ	˜	®	ª
8	00	00	can	00	"	¯	©	Ħ
9	00	EM	00	00	«	˘	™	`
A	00	00	00	00	¢	!	¦ *	:
B	VT	00	00	00	.	¤	,	#
C	FF	00	00	DC4	<	*	%	@
D	CR	GS	enq	NAK	(	)	_	'
E	SO	RS	ACK	00	+	;	>	=
F	SI	US	BEL	SUB	|	¬ *	?	"

Notes: 1. Control codes are translated to an ASCII equivalent, where possible. Code names in **lowercase** indicate codes that have no significance to the Model 50. "00" indicates codes that are translated to nulls.
2. Underscored code names indicate codes that have significance **in some modes only**.
3. Font characters marked with an asterisk are only available in the resident fonts.

Table B-31B **EBCDIC encoding scheme (continued)**

U.S. English

Most →

Least ↓	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
0	µ	æ		°	{	}	\	0
1	a	j	˜	±	A	J	Æ	1
2	b	k	s	²	B	K	S	2
3	c	l	t	³	C	L	T	3
4	d	m	u	×	D	M	U	4
5	e	n	v	^	E	N	V	5
6	f	o	w	¶	F	O	W	6
7	g	p	x	·	G	P	X	7
8	h	q	y	÷	H	Q	Y	8
9	i	r	z	'	I	R	Z	9
A	¡	đ	↑	ø	˚		Œ	œ
B	Ĳ	ð	´	ˆ	¸		º	ß
C	Ŀ	ħ	←	'	_	⅛	Þ	þ
D	Ł	ı	[	]	˝	⅜	Ŧ	ŧ
E	Ø	ĳ	→	¾	˛	⅝	Ŋ	ŋ
F	ĸ	ŀ	↓	¿	ˇ	⅞	ŉ	

Equivalence table

Hex	Binary	Hex	Binary	Hex	Binary	Hex	Binary
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

Table B-32 EBCDIC character mapping

U.K. English

Most →

Least ↓	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	–						{	}	\	0
1	à	À	/		a	j	¯	`	A	J	–	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		`	i	r	z	¨	I	R	Z	9
A	$	!	|	:	¡	¿		˚	$	£	ù	Ù
B	.	£	,	#	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	@	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	|	^	?	"	i̊	I̊		ˇ	ü	Ü	û	Û

Table B-33 **EBCDIC character mapping**

French

Most →

Least ↓	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	-						é	è	ç	0
1	à	À	/		a	j	¨	`	A	J	-	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		`	i	r	z	¨	I	R	Z	9
A	°	§	ù	:	¡	¿		˚	$	£	ù	Ù
B	.	$	,	£	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	à	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	!	^	?	"	i̊	I̊		ˇ	ü	Ü	û	Û

Table B-34 **EBCDIC character mapping**

Dutch

Most →

Least ↓	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	-						{	}	\	0
1	à	À	/		a	j	¯	`	A	J	-	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ij	IJ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		`	i	r	z	¨	I	R	Z	9
A	$	!	|	:	¡	¿		°	$	£	ù	Ù
B	.	£	,	#	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	@	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	|	^	?	"	i̊	I̊		ˇ	ü	Ü	û	Û

Table B-35 **EBCDIC character mapping**

Spanish

Most →

Least ↓

	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	-						{	}	\	0
1	à	À	/		a	j	¨	`	A	J	-	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		`	i	r	z	¨	I	R	Z	9
A	[	]	ñ	:	¡	¿		°	$	£	ù	Ù
B	.	P	,	Ñ	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	@	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	|	^	?	"	i̊	I̊		ˇ	ü	Ü	û	Û

Table B-36 **EBCDIC character mapping**

Italian

Most →

Least ↓	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	-						à	è	ç	0
1	à	À	/		a	j	ì	`	A	J	-	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	ˇ	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		ù	i	r	z	¨	I	R	Z	9
A	°	é	ò	:	¡	¿		˚	$	£	ù	Ù
B	.	$	,	£	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	§	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	!	^	?	"	i̊	I̊		˘	ü	Ü	û	Û

Table B-37 **EBCDIC character mapping**

Danish

Most →

Least ↓	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	-						æ	å	\	0
1	à	À	/		a	j	ü	`	A	J	-	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	˜	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		`	i	r	z	¨	I	R	Z	9
A	#	¤	ø	:	¡	¿		°	$	£	ù	Ù
B	.	Å	,	Æ	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	Ø	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	!	^	?	"	i̊	I̊		ˇ	ü	Ü	û	Û

Table B-38 **EBCDIC character mapping**

Norwegian

Most →

Least ↓

	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	-						æ	å	\	0
1	à	À	/		a	j	ü	`	A	J	–	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		`	i	r	z	¨	I	R	Z	9
A	#	¤	ø	:	¡	¿		°	$	£	ù	Ù
B	.	Å	,	Æ	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	Ø	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	!	^	?	"	i̊	I̊		ˇ	ü	Ü	û	Û

Table B-39 **EBCDIC character mapping**

Finnish

Most →

Least ↓

	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	–						ä	å	É	0
1	à	À	/		a	j	ü	`	A	J	–	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		é	i	r	z	¨	I	R	Z	9
A	§	¤	ö	:	¡	¿		˚	$	£	ù	Ù
B	.	Å	,	Ä	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	Ö	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	!	^	?	"	i̊	I̊		ˇ	ü	Ü	û	Û

Table B-40 **EBCDIC character mapping**

Swedish

Most →

Least ↓	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	–						ä	å	É	0
1	à	À	/		a	j	ü	`	A	J	–	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		é	i	r	z	¨	I	R	Z	9
A	§	¤	ö	:	¡	¿		°	$	£	ù	Ù
B	.	Å	,	Ä	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	Ö	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	!	^	?	"	ı	İ		ˇ	ü	Ü	û	Û

Table B-41 **EBCDIC character mapping**

German

Most →

Least ↓	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	-						ä	ü	Ö	0
1	à	À	/		a	j	ß	`	A	J	-	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		`	i	r	z	¨	I	R	Z	9
A	Ä	Ü	ö	:	¡	¿		˚	$	£	ù	Ù
B	.	$	,	#	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	§	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	!	^	?	"	i̊	I̊		ˇ	ü	Ü	û	Û

Table B-42 **EBCDIC character mapping**

Belgian

Most →

Least ↓

	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	–						é	è	ç	0
1	à	À	/		a	j	¨	`	A	J	–	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	^	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	~	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		`	i	r	z	¨	I	R	Z	9
A	[	]	ù	:	¡	¿		°	$	£	ù	Ù
B	.	$	,	#	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	à	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	!	^	?	"	i̊	I̊		ˇ	ü	Ü	û	Û

Table B-43 **EBCDIC character mapping**

Latin America

Most →

Least ↓

	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	–						{	}	\	0
1	à	À	/		a	j	¨	`	A	J	–	1
2	á	Á	ç	Ç	b	k	s	´	B	K	S	2
3	ä	Ä	ñ	Ñ	c	l	t	ˆ	C	L	T	3
4	â	Â	ĳ	Ĳ	d	m	u	˜	D	M	U	4
5	å	Å	œ	Œ	e	n	v	¯	E	N	V	5
6	è	È	æ	Æ	f	o	w	˘	F	O	W	6
7	é	É	ø	Ø	g	p	x	˙	G	P	X	7
8	ë	Ë	ß	#	h	q	y	¨	H	Q	Y	8
9	ê	Ê		`	i	r	z	¨	I	R	Z	9
A	[	]	ñ	:	¡	¿		˚	$	£	ù	Ù
B	.	P	,	Ñ	ì	Ì	ò	¸	¤	Ò	ú	Ú
C	<	*	%	@	í	Í	ó	_		Ó		½
D	(	)	_	'	ï	Ï	ö	˝	¥	Ö		¼
E	+	;	>	=	î	Î	ô	˛		Ô		¢
F	|	^	?	"	i	İ		ˇ	ü	Ü	û	Û

Table B-44 **EBCDIC character mapping**

French Canadian

Most →

Least ↓

	4	5	6	7	8	9	A	B	C	D	E	F
0		&	-						é	è	¸	0
1			/	É	a	j	¨	`	A	J		1
2	â	ê	Â	Ê	b	k	s	´	B	K	S	2
3		ë		Ë	c	l	t	^	C	L	T	3
4			À	È	d	m	u	¯	D	M	U	4
5		½			e	n	v		E	N	V	5
6		î		Î	f	o	w	ˇ	F	O	W	6
7		ï		Ï	g	p	x	˙	G	P	X	7
8	ç		Ç		h	q	y	¨	H	Q	Y	8
9			\	`	i	r	z	¨		R	Z	9
A	à	´	ù	:				°		£		
B	.	$	,	#				¸	ô	û	Ô	Û
C	<		%	@				_	⊞	ü		Ü
D	(	)	_	'			[	]				ù
E	+	;	>	=				˛	⊞			¢
F	!	^	?	"			|	ˇ				

Table B-45 **EBCDIC character mapping**

DSC environment

Most →

Least ↓

	4	5	6	7	8	9	A	B	C	D	E	F
0	Sp	&	-	ì	ü	é	Ì	E	{	}	\	0
1	Õ	°	/		a	j	˜	E	A	J	æ	1
2	Ê		ò		b	k	s	I	B	K	S	2
3	£	^	ù		c	l	t	O	C	L	T	3
4	¥	¨	ã		d	m	u	U	D	M	U	4
5	P	´	õ	ä	e	n	v	Y	E	N	V	5
6	¤	¸	ÿ	ë	f	o	w	C	F	O	W	6
7	ß	à	à	ï	g	p	x	Ä	G	P	X	7
8	§	è	è	ö	h	q	y	Ë	H	Q	Y	8
9	˜	ì	é	`	i	r	z	Ï	I	R	Z	9
A	¢	!	|	:	â	í	Ò	Ö	Û		ø	Æ
B	.	$	,	#	ê	ó	Ù	Ü	Á	Ú	å	Ø
C	<	*	%	@	î	ú	Ã	Ä	É	Ñ	ç	Å
D	(	)	_	'	ô	ñ	[	]	Í	ü	NUL	Ç
E	+	;	>	=	û	À	Y	Î	ò	ç	NUL	*
F	|	^	?	"	á	È	A	Ô	ù	Ó	;	

Table B-46 **ISO-to-EBCDIC translation for sixel-encoded graphic data**

Least ↓ / Most →	0	1	2	3	4	5	6	7
0					F1	D7	F7	97
1					C1	D8	81	98
2					C2	D9	82	99
3					C3	E2	83	A2
4					C4	E3	84	A3
5					C5	E4	85	A4
6					C6	E5	86	A5
7					C7	E6	87	A6
8					C8	E7	88	A7
9					C9	E8	89	A8
A					D1	E9	91	A9
B					D2	F2	92	F8
C					D3	F3	93	F9
D					D4	F4	94	4E
E					D5	F5	95	7E
F				F0	D6	F6	96	

Note: The codes **within** the table represent the values *to* which packed, sixel-encoded graphic data must be translated.

The IBM PC encoding scheme table begins on the next page.

Table B-47A **IBM PC encoding scheme**

HEXA-DECIMAL VALUE	0	1	2	3	4	5	6	7
0	BLANK (NULL)	►	BLANK (SPACE)	0	@	P	‘	p
1	☺	◄	!	1	A	Q	a	q
2	☻	↕	"	2	B	R	b	r
3	♥	‼	#	3	C	S	c	s
4	♦	¶	$	4	D	T	d	t
5	♣	§	%	5	E	U	e	u
6	♠	▬	&	6	F	V	f	v
7	•	↨	’	7	G	W	g	w
8	◘	↑	(	8	H	X	h	x
9	○	↓	)	9	I	Y	i	y
A	◙	→	*	:	J	Z	j	z
B	♂	←	+	;	K	[	k	{
C	♀	∟	,	<	L	\	l	¦
D	♪	↔	–	=	M	]	m	}
E	♫	▲	.	>	N	^	n	~
F	☼	▼	/	?	O	_	o	△

Notes: 1. The control codes occurring in columns 0 and 1 are the same as in the ISO table; the characters shown here in those two columns are alternate font characters.

Table B-47B **IBM PC encoding scheme (continued)**

HEXA-DECIMAL VALUE	8	9	A	B	C	D	E	F
0	Ç	É	á	░	└	╨	∝	≡
1	ü	æ	í	▒	┴	╤	β	±
2	é	Æ	ó	▓	┬	╥	Γ	≥
3	â	ô	ú	│	├	╙	π	≤
4	ä	ö	ñ	┤	─	╘	Σ	⌠
5	à	ò	Ñ	╡	┼	╒	σ	⌡
6	å	û	ª	╢	╞	╓	µ	÷
7	ç	ù	º	╖	╟	╫	τ	≈
8	ê	ÿ	¿	╕	╚	╪	Φ	°
9	ë	Ö	⌐	╣	╔	┘	θ	•
A	è	Ü	¬	║	╩	┌	Ω	·
B	ï	¢	½	╗	╦	█	δ	√
C	î	£	¼	╝	╠	▄	∞	ⁿ
D	ì	¥	¡	╜	═	▌	φ	²
E	Ä	₧	«	╛	╬	▐	∈	■
F	Å	ƒ	»	┐	╧	▀	∩	BLANK 'FF'

Equivalence table

Hex	Binary	Hex	Binary	Hex	Binary	Hex	Binary
0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

C. Font terminology

This appendix provides explanation of typographic terms used to describe fonts and font characteristics.

Every Model 50 and 2700 font is accompanied by a set of font data sheets which provide detailed information about the font and the characters within it. The figure below illustrates several parts of font characters that are described by the font data sheets.

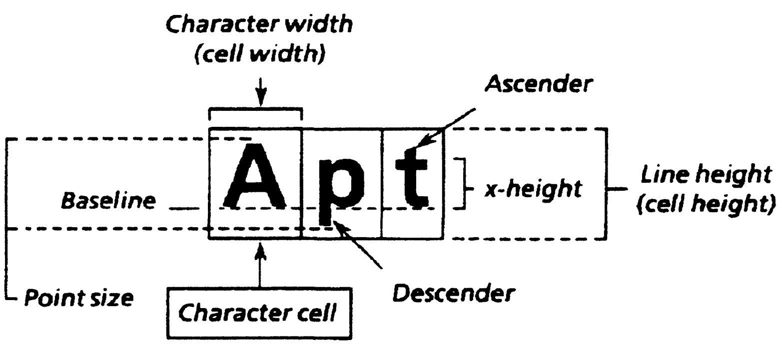


Figure C-1 **Font character components**

Family, weight posture These three attributes together identify the typeface of the font.

– The family may be defined as one or more fonts sharing the same typestyle, but differing in height, weight, and posture. For example, the Titan family includes Titan Regular, Titan Italic, Titan Bold, Titan LS (legal—statistical character set), etc.

- The weight is the thickness of the stroke of the letters, and is denoted by the terms Light, Medium, Bold, etc.
- The posture is the vertical alignment of the letters, either Regular (Roman, i.e., upright) or Italic (slanted).

Point size The height of the alphanumeric characters in the font, from the top of the capitals to the bottom of the descenders, in points.

Revision number The revision level for this font.

Name This is the font-name as it must be specified in the Font Load and Font ID commands. Naming conventions for these fonts are described in the following paragraphs.

1. The font name is never more than 20 characters long.
2. The font name contains no space characters.
3. The first element in the font name is the name of the family to which it belongs (Bold, Helvetica, Titan, etc.). The first character of each word in the family name is capitalized (e.g., Elite, Letter-Gothic), unless the family name is an acronym, in which case all characters are capitalized (e.g., APL, OCR).
4. If the font name includes a version letter or character-set identifier, it follows the family name and is capitalized (e.g., OCRB, SymbolC, TitanLS).
5. The next element in the font name is the pitch of the font, for office printing fonts (10, 12, or PS), or its point size, for typographic fonts (6, 8, 10, etc.).
6. The next element identifies the weight and/or posture, in that order, of the font if they are different from "medium" and "regular." For example, this element may be B for a bold font, I for an italic font, or BI for a font which is both bold and italic.

7. If the font is an ISO font, the lowercase letters "iso" constitute the next element.
8. Finally, the orientation of the font is specified by a hyphen followed by a capital L (landscape) or P (portrait).

Character set encoding name This identifies the character set. The most common entries here are: "2700 I" for 2700 non-ISO fonts, and "iso" for ISO fonts. Other entries indicate unusual character sets such as APL, OCR, scientific, or unique (custom-designed) character sets.

Xerox character set number This contains the octal value of the number of the character set.

Spacing Spacing may be fixed (an equal number of characters per horizontal inch) or proportional (character widths vary according to the shape of the character).

Floating accents These are only found in ISO fonts.

Kerning parameters Kerning is the printing of adjacent characters such that the body of one character overlaps the character space of the next, as illustrated in Figure C-2.

Figure C-2 **Kerning**

Serifs Serifs are short line segments stemming from, and at an angle to, of the character, as illustrated in Figure C-3.

Fonts with no serifs are called "sans-serif" fonts.

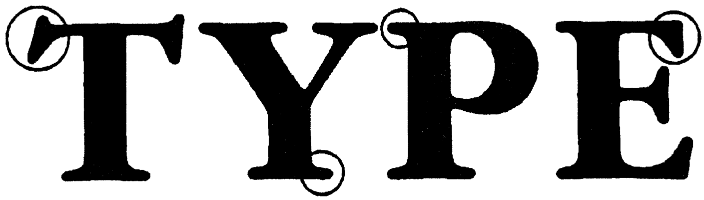


Figure C-3 **Serifs**

Number of characters This is the number of printable characters and the number of space characters (blanks) in the font.

Beginning, ending character This gives the hexadecimal values of the lowest and highest ASCII codes assigned to printable characters in the font.

Nominal line spacing This is the line spacing of the font, expressed in dots ("bits"), when no line spacing command is given. This value is also expressed in its equivalent value in picas (a printer's unit equal to 12 points).

Highest/lowest bit in font (above baseline) These two values describe the distance of the highest and lowest bits in the font from the baseline. If the lowest bit is below the baseline, it is represented as a negative value. The sum of the absolute values of these two parameters is equal to the font height.

Maximum alpha ascender (above baseline) This is the distance from the baseline to the top of the highest alphanumeric ascender.

Maximum alpha descender (above baseline) This is the distance from the baseline to the bottom of the lowest alphanumeric descender. If the lowest bit is below the baseline, this value is negative.

Cap height This is the height of the capital letters in the font.

X-height This is the height of the lowercase letter x of the font. Typically, lowercase characters are designed relative to this value.

Average character width This value is the average number of dots from side to side of the character cell for each character in the font, including the error character, which is 24 dots wide. Therefore, this number may not be an integral number, even for a fixed-pitch font.

Figure width This is the width of the character cell of the numerals (0 through 9) in the font.

Number of characters per inch This value is equivalent to 300 divided by the average character width.

Lowercase alphabet length This is the length of the lowercase letters a through z when printed in a single line, as "abcdefg...," etc.

Uppercase alphabet length This is the length of the uppercase letters A through Z when printed in a single line, as "ABCDEFG...," etc. For fixed-pitch fonts, this value is always the same as that for lowercase alphabet length.

Underline offset (above baseline) This is the distance from the baseline to the top of the underline character; it is always a negative value.

Underline thickness This is the thickness, in dots, of the underline character.

Strikeout offset (above baseline) This is the distance from the baseline to the bottom of the strikeout (hyphen) character.

Strikeout thickness This is the thickness, in dots, of the strikeout character.

Superscript offset This is the number of dots that the baseline is moved upward whenever the Superscript instruction is received, if this font is in effect.

Subscript offset (above baseline) This is the number of dots the character is moved downward whenever the Subscript instruction is received, if this font is in effect. Since it is expressed as a distance above the baseline, it is always a negative value.

File size This is the amount of storage the file occupies on the rigid disk. This line also contains the orientation of the font.

Char code (Hex) This is the assigned binary code, expressed as a hexadecimal value.

Character width The number of dots from one side of the character cell to the other.

Note: If no printable character is assigned to the code on this line, a value of 24 is printed here, the width of the error character.

Left/right side bearing (neg. = kern) The first value is the number of dots from the left side of the character cell to the left edge of the character; the second value is the number of dots from the rightmost dot of the character to the right side of the character cell. A negative value indicates that the character itself extends beyond the side of the character cell.

Note: If no printable character is assigned to the code on this line, a double asterisk (**) appears here.

High/low bit (above baseline) The first value is the distance, in dots, of the highest dot of the character from the baseline; the second value is the distance of the lowest dot from the baseline. A negative value indicates that the dot is below the baseline.

Note: If no printable character is assigned to the code on this line, a double asterisk (**) appears here.

D. Specifications

This appendix describes physical, electrical, and environmental characteristics of the Model 50.

Physical characteristics

Dimensions

Gross dimensions of the Model 50 are as follows:

	Inches	Millimeters
Height:	10.5	270
Width:	27.5	700
Depth:	21.5	550

Weight

Total unpacked weight is: 140 pounds
63.5 kilograms

Electrical characteristics

Power requirements

Power requirements for both the U.S./Canada and international machines are summarized in the following tables.

Table D-1 **Power requirements for U.S./Canada machine**

Nominal voltage: Range:	120V, AC, single phase 103 to 127V
Nominal frequency: Range:	60 Hz 59 to 61 Hz
Circuit requirement:	15 amp
Power consumption:	<150 watts, standby 800 watts, operating

Table D-2 **Power requirements for international machine**

Nominal voltage: Range: Taps are available for a 220V nominal value.	240V, AC, single phase 216 to 264V
Nominal frequency: Range:	50 Hz 49 to 51 Hz
Circuit requirement:	2.2 kVA
Power consumption:	<150 watts, standby 800 watts, operating

Safety and electromagnetic emissions

The Model 50 complies with the following agency requirements for safety.

Table D-3 **Safety requirement compliance**

U.S./Canada	International
UL 478 CSA C 22.2, #154	Certified by UL to IEC 380, IEC 435.
UL 478 U.S. Bureau of Radiological Health requirements for laser safety, class I product CSA 22.2, #154	Certified by UL to IEC 380, IEC 435.
Maximum leakage: 5.0 mA	 3.5 mA

The Model 50 complies with the following requirements for electromagnetic emissions:

U.S./Canada	International
FCC Docket 20780	VDE 0871, Class B, and VDE0875, Level N

Environmental characteristics

The following table summarizes the environmental conditions under which the Model 50 is designed to operate. Degraded performance may be expected under conditions outside the indicated zones.

Table D-4 **Environmental conditions**

Item	Conditions
Temperature range:	15° to 32° C (59° to 90° F)
Humidity range:	15 to 85 %
Altitude tolerance:	Sea level to 2,000 meters (6,561.6 feet)

The following table describes rates of air and heat emission from the Model 50.

Table D-5 **Model 50 emissions—rates**

Heat dissipation:	
Standby condition:	70 watts max.
Operating condition:	800 watts max.
Noise:	<55 db

E. Glossary

The explanations given in this section describe terms as they are used in this guide; they are not necessarily exhaustive descriptions.

alphanumeric Consisting of both letters and numbers, as distinguished from other symbols, such as math symbols, shading symbols, etc.

ascender That portion of an alphabetic character that rises higher than the body of the character (usually the x-height portion).

ASCII American Standard Code for Information Interchange—A digital coding system used to represent characters or control functions electronically, each character being represented by either seven or eight bits.

asynchronous In reference to communications, a protocol in which data bytes are framed by special start- and stop-bits, thus enabling varying rates of transmission.

band A grouping of 32 successive scan lines of video data.

band buffer An area of storage used to transfer a band (32 scan lines) of video data to the laser-based scanner.

binary Characterized by having two states or components. The binary numbering system uses only two values, 0 and 1.

bisynchronous "Binary synchronous"—In reference to data communication, this expression describes a fairly elaborate protocol in which data bytes are blocked in long, well-defined records, and transmitted at a constant rate.

bit A unit of electronic data. Printed dots are represented electronically within the Model 50 by bits.

bit map A pattern of bits representing the dots in a printed image.

byte A set of seven or eight bits used to represent a character or control function.

channel selection A manner of controlling the vertical positioning of characters on a page, in emulation of mechanical printers whose form-feed mechanism is controlled by holes punched in a tape. The tape is divided into channels, each channel having its own set of punched holes that controls a unique pattern of vertical movement. (Same as vertical format unit.)

character cell The rectangular-shaped area on a page containing a single character and the space which separates it from adjacent characters in the same line, as well as from characters in the lines above and below it.

character pitch The "pitch" of a font is equivalent to the number of characters that can be printed in a horizontal inch (25.4 mm). For example, a 10-pitch font prints 10 characters to the inch, while a 12-pitch font prints 12.

character set The collection of all the characters available in a given font.

checksum A numerical value derived by reading a group of data or memory locations and performing some predetermined mathematical operation on the group of values. The result is compared with a predetermined value, to assess the accuracy of the read process.

command A sequence of characters from the host computer which invokes printing options available with the Model 50.

command terminator A character or sequence of characters (such as a line ending) used to mark the end of a command whose length is variable.

configuration cartridge A plug-in cartridge containing four switchbanks, whose switches may be variously set to select alternative system defaults.

configuration sheet A one-page printout, generated by the Model 50 at power-up and after a soft reset, that provides information about the switch settings and fonts available in the system.

constant page A page of data stored in the printer that may be merged with other pages, in the manner of an electronic form or overlay. Also called a "merge-page."

data integrity This expression refers to various means of ensuring that data is not altered during transfer from the host to the printer. Parity checking, for example, is a means of ensuring data integrity in a serial network.

DC1/DC3 An alternate name for the XON/XOFF protocol.

default parameters Format parameters used by the Model 50 when others are not specified from the host.

defaults, or default values A set of values for various encoding and printing parameters, including code set, margins, font selection, language mapping, and others, which are stored in the memory of the Model 50 and are used when alternative values are not selected.

descender That portion of an alphabetic character that extends below the baseline.

digitized Encoded electronically in digital form.

download To load information from the host into Model 50 memory.

dry imager A fine powder (also called toner) used by the Model 50 to print images.

dynamic In reference to memory or data storage, this term is equivalent to "volatile," meaning that data is erased when the device is turned off.

In reference to data streams or data processing routines, this term is usually used to indicate that processes may be invoked at will. For example, in the Model 50, margins may be changed dynamically, that is, the margins for a print job need not be specified in advance of the job and they may be changed at any point within the job.

EBCDIC Extended Binary-Coded Decimal Interchange Code—A digital coding system used to represent characters electronically, each character being represented by eight bits.

ECS/API Extended Character Set / All Purpose Interface—A Diablo 630 configuration which features an IBM PC character set and three interfaces, including RS-232-C, IEEE 488, and Centronics.

escape character A non-graphic (unimaged) code that signals the transmission of control information to the Model 50.

escape sequence A sequence of characters beginning with an escape code.

ETX/ACK A communications protocol for use with the serial interface, in which the printer responds to an ETX code from the host by returning an ACK or NAK code, to signal the successfull or unsuccessfull receipt of data. If the checksum does not equal the predetermined value (a checksum error), a fault in the storage device or the reading process is assumed.

family A set of fonts sharing the same typestyle, but differing in height, weight, and posture.

firmware Programmed data stored in a data processing device as a permanent part of the device.

fixed-pitch In reference to character sets, this term describes a typeface in which all character cells are of equal width.

floating accent A non-spacing accent character that can be combined with spacing characters and printed as a composite.

font A set of graphic characters sharing similar attributes, such as style, size, and orientation.

font data The encoded bit maps that, when stored on the rigid disk, enable the printing of characters in various fonts.

format This term is used in several ways:

With respect to the printed page, a set of parameters for the placement of printed information on a page, including margins, tabs, line spacing, and orientation.

With respect to binary data, a set of rules describing the order and ways in which binary codes may be transmitted to the Model 50 from the host, or among software modules within the Model 50.

May refer to the process of arranging data to correspond with one of the above sets of rules.

formatting Designing the distribution of printed information on a page.

form feed A control code that instructs the printer to process the current page and print it; a page-end.

hardware Any physical device in a computer network.

handshaking An exchange of signals between two devices in a computer network, as a prelude to data exchange. The purpose of handshaking is to determine the readiness of each device to exchange data.

hexadecimal Refers to a numbering system whose base is the value 16; that is, each digit in a hexadecimal expression may have any of 16 different values. Hexadecimal notation, by convention, uses the numerals 0 through 9 and the letters A through F.

host The system which transmits information to the Model 50.

HMI Horizontal Motion Index—A programmable value stored in the Diablo 630 that the printer uses to control the horizontal movement (escapement) of the printhead.

HyPlot graphics A set of plotting routines designed to allow the Diablo 630 to print graphic image.

IEEE 488 interface A parallel interface corresponding to Standard 488 of the Institute of Electrical and Electronics Engineers (IEEE), which has been adopted as an ANSI (American National Standards Institute) standard.

image area The area occupied by all the printed information on a page.

interface The device through which data from the host is received by the Model 50.

internal fonts Font data contained within Model 50 software; fonts which are neither downloaded nor loaded from diskette.

international As used in this manual, refers to a Model 50 designed for use outside the U.S. and Canada.

International Organization for Standardization See "ISO."

ISO International Organization for Standardization—An organization that develops and publishes international standards for a variety of technical applications, including data processing and communications.

justification In the Model 50, the spacing-out of words on a line so that they fit exactly between the right and left margins.

landscape An orientation in which print lines run parallel to the long edge of the paper.

leading A typographical term indicating the amount of white space between lines of printed characters (rhymes with "heading").

least significant With respect to numbering systems, refers to the digit representing the least value. For example, in the expression "1,024," 4 is the least significant digit, representing four ones.

line-ending A code sequence denoting the end of a print line or of a command. Line endings may vary from system to system according to the interface and protocol used.

line-ending decisions A function found in many word processors in which printing is automatically returned to the left margin of the next successive line without a command from the input operator.

line feed A control code that instructs the printer to move the print position vertically down the page, for a distance of one line-height.

load To transmit electronic data from the host for storage in the Model 50.

logic level Either of two voltage levels, corresponding roughly to +5 or 0 volts, that designate the presence of ones and zeros in the data stream. Either may be selected, via the Interface Setup menu, to represent marking (ones); the other automatically represents spacing (zeros).

logotype (logo) A single image which generally contains a symbol, mark, or identifying name.

mapping The process whereby received codes are translated to designated font characters according to menu- or command-selected tables.

merge-page A page of data that, when sent to the Model 50 preceded by the proper command, is stored in memory and may be merged with information contained in subsequent pages.

modem modulator-demodulator—A device that converts electronic signals from the form used in data processing (digital) to the form used in communication (analog), and vice versa.

most significant With respect to numbering systems, refers to the digit representing the greatest value. For example, in the expression "1,024," 1 is the most significant digit, as it represents one thousand.

multinational As used in this manual, refers to Model 50 capabilities designed to reflect printing requirements of various nations.

multinational code mapping A scheme whereby a received character code may invoke the printing of different characters, to accommodate the encoding conventions of different nations.

non-textual data Data in which binary codes are not intended to invoke the printing of font characters.

orientation In reference to fonts or image areas, describes whether printed lines are parallel to the long or short edge of the paper.

origin Zero-point of an X-Y coordinate system. For Model 50 line drawing and text placement commands, refers to a corner of the page.

overstriking Printing characters over each other, as:

~~This is an overstruck sentence~~.

parallel interface A type of interface in which data is transmitted and received in bytes rather than bits. Used for local printing over short distances (10 meters or less).

parity A means of checking data for errors occurring in transmission, by testing each byte for either an odd or even number of 1-bits.

pitch The "pitch" of a font is equivalent to the number of characters that can be printed in a horizontal inch (25.4 mm). For example, a 10-pitch font prints 10 characters to the inch, while a 12-pitch font prints 12.

point size A typographical term describing the height of a character set from the top of its capitals to the bottom of its descenders in units (points) equal to approximately 1/72nd inch. Point size does not always include leading (extra space between lines).

polygon A rotating, many-sided disc whose function is to deflect beams of light at regular intervals across the surface of the xerographic drum.

portrait An orientation in which print lines run parallel to the short edge of the paper.

posture In reference to fonts, the vertical inclination of the font characters. Model 50 fonts are either Roman (vertically erect) or italic (slanted).

printing module Those components in the Model 50 that together drive the laser scanner, create the image on the page, deliver the page to the stacker, and monitor the external controls on the machine.

processing module Those components in the Model 50 that together perform data receipt, data storage, page composition, and video generation for the printer.

proportional spacing Refers to character sets in which characters do not have equal widths.

protocol A set of rules for the exchange of data between data processing devices. In this guide, "protocol" is also used to denote the data flow control options available with the asynchronous interface.

raster data Whereas "video data" is a generic term indicating series of 1-bits and 0-bits, raster data is a pattern of 1s and 0s designed to create a graphic image on a page, and arranged in the order in which they are to be printed.

reset To restore the printer to its default, or start-up, state. In the Model 50, this may be performed by switch operation, or by command.

scan line The line of video data printed in one pass of the laser beam across the length of the xerographic drum.

serial interface A type of interface in which data is transmitted and received one bit at a time over a communication line. Can function over great distances.

space compression A method of packing a series of space characters into a shorter sequence, to save time for transmission. The receiving device then expands the shorter sequence to its original length.

sixel encoding A method of grouping raster data, so that each byte resembles a code for an ASCII character.

soft reset A means of resetting the Model 50 to a default state without erasing downloaded fonts, by the simultaneous operation of the Off-line and Reset switches.

software Programmed data designed to perform some function within a computer network.

status sheet A one-page printout, generated by the Model 50 in response to a parameter within a job command, that presents information about the fonts used, and errors detected in a job.

subscript A printed character whose baseline is below that of the normal print line, as in H_2O.

superscript A printed character whose baseline is higher than that of the normal print line, as in X^2.

surrogate escape character Also called "UDK" for "user-defined key," the surrogate escape character allows the user to substitute a keyboard character for the escape code (ISO X'1B', EBCDIC X'27'), for the purpose of inserting printer commands in a text file.

synchronous In reference to communications, a means of data transfer in which data is transmitted at a continuous rate.

system defaults See "defaults, or default values."

terminator A character or sequence of characters (such as a line ending) used to mark the end of a command whose length is variable.

vertical format control A manner of controlling the vertical positioning of characters on a page, in emulation of mechanical printers whose form-feed mechanism is controlled by holes punched in a tape. The tape is divided into channels, each channel having its own set of punched holes that controls a unique pattern of vertical movement. (Same as channel selection.)

vertical format unit A physical unit designed to implement vertical format control.

VFU See "vertical format unit."

video data The unencoded data used to build images; that is, the pattern of 1s and 0s, representing printed and unprinted dots, that lay the image out on the page, much as a television image is laid out on a screen.

video generator That portion of the Model 50 that takes composed pages of data from the page composition task, converts this data to a video stream, and sends the video stream to the laser scanner.

video stream Video data as it appears over transmission lines.

xerographic drum A rotating metallic drum whose surface may be electrostatically charged in such a way that the charge may be dissipated as a result of the incidence of light.

XON/XOFF A communications protocol for use with the serial interface, in which the DC1 and DC3 codes are sent to the host to signal the printer's readiness, or nonreadiness, to receive data.

Index