EK-LN01S-RM-002

LN01 Programmer Reference Manual

digital

EK-LN01S-RM-002

LN01 Programmer Reference Manual

Prepared by Educational Services
of
Digital Equipment Corporation

1st Edition, September, 1983
2nd Edition, November, 1983

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

The LNØ1 Electronic Printer is a standalone non-impact page printer that uses laser imaging and xerograhic printing techniques. The LNØ1 is plug-compatible with the LP25 and the LP26 line printers and offers significant enhancements to typical line printer operation. The printer prints letter quality output on cut sheet plain paper at a maximum rate of 12 pages per minute.

The LNØ1 is capable of printing text in two different orientations; portrait and landscape (Figure 1-1). In the landscape orientation, characters print parallel to the long edges of the paper. In portrait orientation, characters print parallel to the short edges of the paper.

The LNØ1 is supplied with two DIGITAL Multinational fonts built into read-only memory (ROM). These fonts are always present in the LNØ1. One ROM font prints in landscape orientation; the other in portrait. Other fonts can be loaded from the host computer. Up to 22 additional fonts can be stored in the LNØ1.

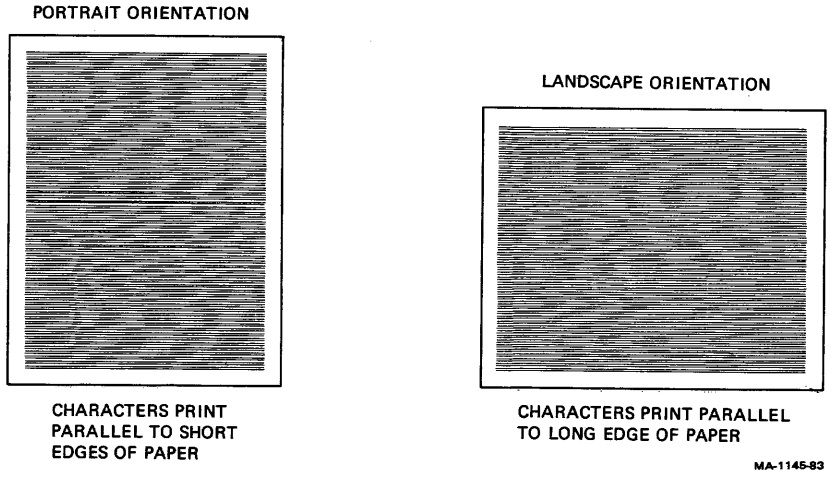


Figure 1-1 Landscape and Portrait Orientation

A complete set of control functions allow the printer to have adjustable tabs, margins, and line spacing, draw horizontal and vertical lines, absolute text placement commands, multi-font page printing, text justification underlining, superscripting and subscripting and access to many font styles and sizes. Other commands allow for selecting either or both input paper trays, and offset stack the printer output.

Other features include:

- LP11 type parallel interface
- Two 250 sheet input trays
- Cut sheet plain paper- standard, legal, and A4 European page sizes with matching paper trays.

This manual contains the information needed to develop system software to use the LN01 as a line printer emulator and as an electronic printer with its full range of capabilities. The LN01 when functioning as a line printer requires minimal changes to existing line printer software unless the system designer wishes to enhance line printer operation to include both the landscape and portrait print orientation. All LP11 functions are supported on the LN01 with the exception of bolding and certain overstriking limits (refer to Chapter 4 for more information).

REFERENCE DOCUMENTS

The following documents should be used for reference information.

ANSI X3.4-1977 - American National Standard Code for Information Interchange

ANSI X3.41-1974 - Code extension techniques for use with the 7-bit coded character set of American National Standard Code for Information Interchange

ANSI X3.64-1979 - Additional controls for use with American National Standard Code for Information interchange

ISO 6937 - Coded Character Sets for Text Communication (Draft International Standard)

CHAPTER 2
FUNCTIONAL DESCRIPTION

GENERAL

This chapter descibes the functional elements of the LNØ1 electronic printer.

OVERVIEW

The LNØ1 stores printable images (characters) and translates them to an intermediate level. When a full page is received, then the final image is constructed (by pieces) to be printed. This technique reduces the memory requirements, but puts limitations on the number of commands that can be stored, and the number of commands that can be processed at printing time (refer to Chapter 4 for limitations).

Figure 2-1 shows a functional block diagram of the LNØ1. The following paragraphs describes each of the major functional elements.

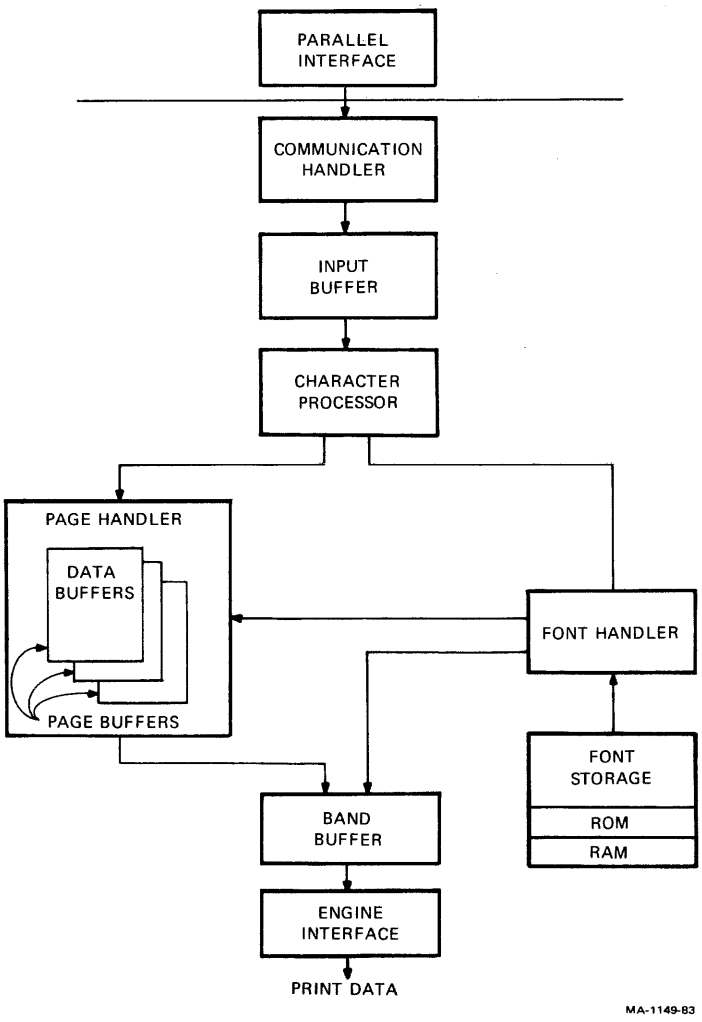


Figure 2-1 LN01 Functional Block Diagram

PRINT MECHANISM
The print mechanism removes individual sheets of paper from one of the paper trays, transports the sheets through the paper path to the printing engine. The image is created using laser electrophotography. The image is then fused on the paper using a heat process, and finally deposited into the output tray.

CONTROLLER
The controller removes data (7 or 8 bit characters) from the parallel interface. The controller then converts the received data to an internal format. The contoller is divided into the following functional parts.

- Communication Handler and Input Buffer
- Font Handler
- Character Processor
- Page Handler
- Band Buffer and Engine Interface

Communication Handler and Input Buffer
The communication handler removes characters from the interface and stores them in the input buffer for further processing. The input buffer stores characters from the communication line at communication rates until they are removed by the character processor.

Character Processor
The character processor removes and analyzes characters from the input buffer. Font loads are directed to the font handler. Printable characters, positioning commands and font invocation commands are transferred to the page handler to create page buffers. Stacking commands are sent to the output tray mechanism.

Page Handler
The page handler processes printable characters, positioning commands and font invocation commands to build page buffers.

Page Buffers
A page buffer stores all of the encoded input data used to define a page. This buffer must be built before a page is physically imaged. This is because once a page is started through the engine it cannot be stopped to wait for additional information.

To achieve higher throughput, the controller will try to keep all of the various elements of the print mechanism busy; each working on a different page. To allow retries (in the event of a paper jam for instance) the controller will maintain the corresponding page buffer until that page is delivered to the output tray.
This process requires that 3 page buffers be maintained; for the paper being picked from one of the paper trays, for the page that is being imaged, and for the page being delivered to the output tray.

If due to page size, there is not enough room for 3 buffers, the controller will wait for the current page to be delivered to the output tray before the next page is started therefore reducing the throughput.

Data Buffers
The page information is structured into data buffers. A data buffer is a collection of printable characters. It roughly corresponds to a line of data. Although various horizontal and vertical positioning commands may cause a new data buffer to be used (refer to Chapter 4 for more information).

Font Handler
The font handler processes font loads and manages font storage memory. It provides the page handler with necessary font position and size information in order to build page buffers. The font handler also provides the band buffer with dot patterns in order to build band buffers.

Two default fonts are stored in ROM. Up to 22 additional fonts can be stored in RAM. The font storage memory can contain up to 187,000 bytes of font information.

Band Buffer and Engine Interface
Once a page buffer is complete, the printing process begins. The actual printed image is produced by combining the page buffer information and the font dot pattern information. The page buffer and font information is split into 32 scans, each 4096 dots (pixels) These bit images are called band buffers. The band buffer drives the laser to create a bit map image of a position on the page.

There are two band buffers, used alternately; one drives the laser while the other is being filled with image data.

CHAPTER 3
PRINTER COMMANDS

CHARACTER PROCESSING

The printer processes characters according to a subset of the American National Standards Institute (ANSI) standards X3.64-1979, X3.4-1977 and X3.41-1977. With the ANSI system, characters are processed according to their position in a standard character chart (Figure 3-1).

The characters in the chart can be divided into two general categories; graphic or printable characters and control characters. Control characters cause some action to be performed by the printer whereas graphic characters are printed. In the sixteen column character chart, the control characters are contained in columns Ø, 1, 8 and 9. Columns zero and one are called the CØ control set. Columns eight and nine are called the C1 control set.

The rest of the chart (columns 2-7 and columns 1Ø-15) contain the graphic or printable characters. Columns 2-7 except SP (hex 2Ø) and DEL (hex 7F), are called the Graphic Left (GL) set. Columns 1Ø-15 except hex AØ and hex FF, are called the Graphic Right (GR) set. The SP, DEL, hex AØ and hex FF are always the same control characters regardless of the character set selected.

> NOTE
> SP can be considered either an information separator control character or a printable character. It can be considered a printable character because it takes up space both in the printer memory, and on the paper when printed.

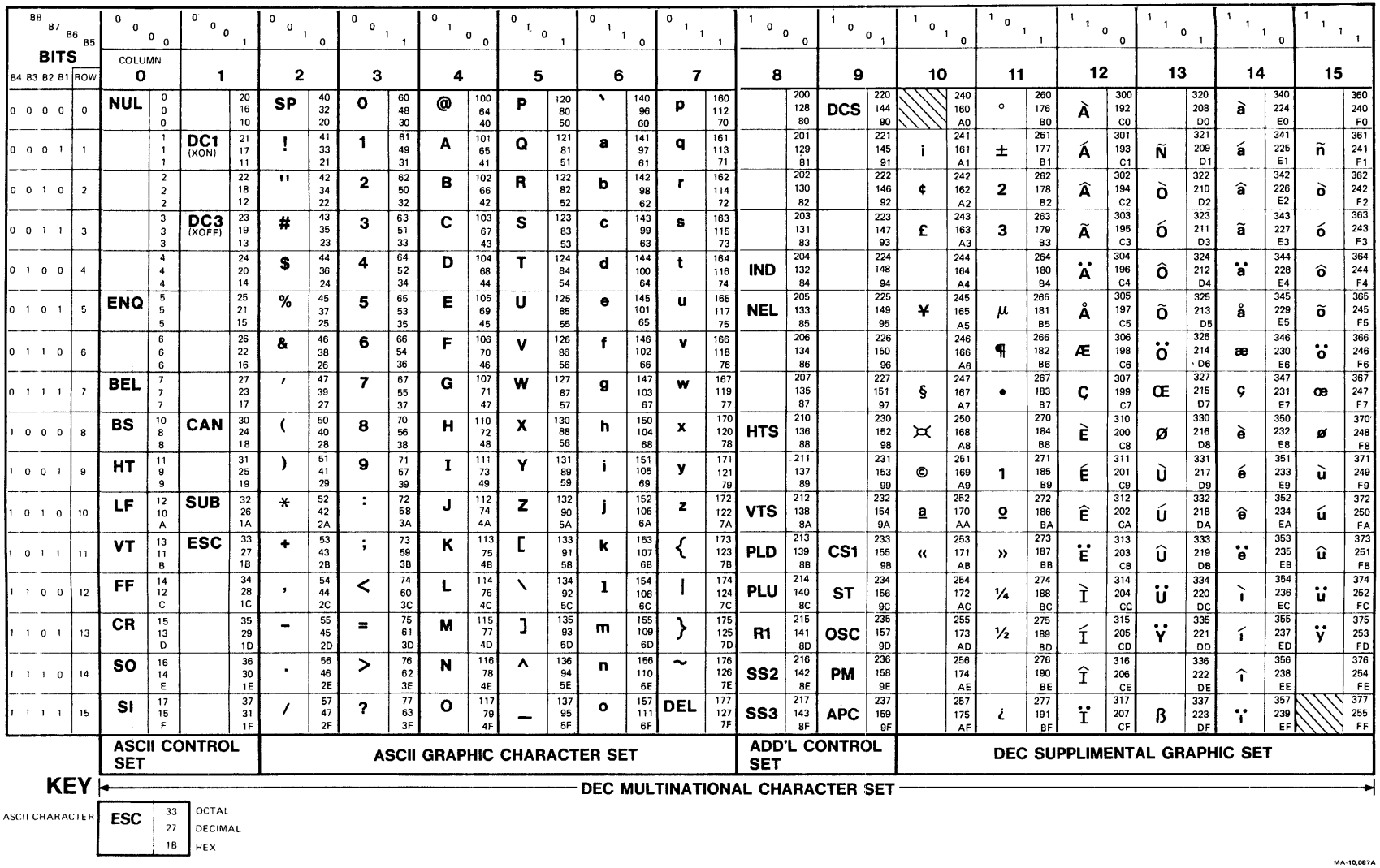


Figure 3-1 Standard Character Chart

PRINTABLE CHARACTERS

Characters in the GL graphic set (hex 21 - 7E inclusive) and characters in the GR graphic set (hex A1 - FE inclusive) cause the LNØ1 to print a character. These characters are received and stored in a page buffer until printing time.

Each received character is printed at the active postion and the active postion then advances unless the active position is equal to the right margin. If the horizontal active position is equal to the right margin, the character is not printed and horizontal active postion is not advanced.

NOTE

The vertical spacing is a function of the font(s) selected for a particular line. Refer to Chapter 4 for more information.

7-BIT AND 8-BIT CHARACTERS

Character processing is described in this document in terms of 7-bit characters. However, the printer recognizes 8-bit characters under some circumstances.

Basically the LNØ1 is a 7-bit device that recognizes 8-bit data. It processes only 7-bit control characters (those in the CØ control set, ØØ hex to 1F hex). All escape sequences, control sequences, and device control strings are processed only in their 7-bit representation. Any 8-bit control characters (those in the C1 control set, 8Ø hex to 9F hex inclusive) are ignored, including the 8-bit representation for the Control Sequence Introducer and Device Control Strings controls.

However, all 8-bit graphic characters (both in the GL and GR sets, 21 hex to 7E hex inclusive and A1 hex to FE hex inclusive) are recognized, without the use of shifting controls.

When the printer is used in a 7-bit environment, only the CØ and GL sets (including space and delete) are available. The 7-bit Shift In (SI) and Shift Out (SO) control characters are ignored.

ERROR CHARACTER

The error character is defined as the reverse question mark. See ANSI X3.4 and X3.32 for detailed information. The LNØ1 prints the reverse question mark for all reserved and empty positions in the default ROM character set. For fonts that are loaded from the host computer, the LNØ1 prints the last character in the character set for any characters greater than the last character. Empty positions in the character sets loaded from the host computer must be filled by the font.

CONTROL CHARACTERS

A control character is a single character control function used to control character processing. Control characters are not printed. The LNØ1 responds to the following subset of the CØ control characters. Each control character is described with its mnemonic, full title, and function. All other control characters received by the printer cause no action.

NOTE

Each control function listed in this manual has a mnemonic. The mnemonic is an abbreviation of the control function name.

Null -- NUL (hex ØØ)

The null character causes no operation in the printer. It is stripped from the data stream upon reception without occupying space in the buffer.

Horizontal Tab -- HT (hex Ø9)

The horizontal tab character advances the horizontal active position to the next horizontal tab stop. If no tab stops are set greater than the horizontal active position, or horizontal tab stops are set past the right margin, the horizontal active position is set to the right margin. Horizontal tab stops are set using the DECSHTS escape sequence.

Refer to the Default Values and States for a description of the default horizontal tab stops.

Line Feed -- LF (hex ØA)

The line feed character increments the vertical active position, unless the vertical active position is at the bottom margin.

NOTE

The vertical spacing is a function of the font(s) selected for a particular line. Refer to Chapter 4 for more information.

If the vertical active position is at the bottom margin, the verticalactive position is set to the top margin of the next page. If linefeed newline mode (LNM) is enabled, the horizontal active position is also set to the left margin.

Vertical Tab -- VT (hex ØB)

The vertical tab character advances the vertical active position to the next vertical tab stop greater than the current vertical active position but no greater than the bottom margin. The horizontal active position remains unchanged. If there are no such tab stops, the vertical tab character increments vertical active position one line spacing. Vertical tab stops are set using the DECSVTS escape sequence.

Refer to the Default Values and States section for a description of the default vertical tab stops.

Form Feed -- FF (hex ØC)
The form feed character advances the vertical active position to the top margin of the next page and sets the horizontal active position to the first character position of the first line. This causes the current page buffer to be printed.

A blank sheet is printed if a form feed is sent without any text on the page.

Carriage Return -- CR (hex ØD)
The carriage return character causes the horizontal active position to be moved to the left margin.

Cancel -- CAN (hex 18)
The cancel character is used to indicate that the data with which it is sent is in error or is to be disregarded. Therefore, the receipt of CAN causes immediate termination, without execution, of any sequence in progress. The CAN character itself receives no further processing. The characters following the CAN are processed as graphic characters, not as part of the escape or control sequence.

Substitute -- SUB (hex 1A)
The substitute character is used to indicate replacement of a character which could not be represented. The receipt of SUB causes immediate termination, without execution, of any sequence in progress. The SUB character itself receives no further processing. The characters following the SUB are processed as graphic characters, not as part of the escape or control sequence.

Delete -- DEL (hex 7F)
The delete character causes no operation in the printer. It is stripped from the data stream upon reception without occupying space in the buffer.

Escape -- ESC (hex 1B)
The escape character introduces an escape sequence. Refer to Appendix A for a description of escape sequence processing.

SEQUENCE AND STRING DEFINITIONS

The following prargraphs describe the LNØ1 escape sequences, control sequences and control strings. A summary of ANSI code extension technique is given in Appendix A.

Loading Fonts

Loading and using fonts from the host computer requires 3 steps:

- Load the font into the LNØ1 font memory
- Assign the font name
- Invoke the font to print using the Select Graphic Rendition (SGR) sequence

Refer to the Load Font Control String, Assign Font Name, and Select Graphic Rendition (SGR) sections for more information.

Load Font Control String

The load font(s) control string writes the new font information over the current contents of the font memory. All previous font information is lost including all font bitmap data and all font assignments. You can not send more than one font load per page because font loads cause page ejects. Therefore, all fonts required for a single page must be loaded before sending any page data.

At power up the default fonts are assigned and invoked. Refer to the Load Font Control String and Assign Font Name in the Default Values and States section for more information.

The LNØ1 allows you to load a font with a name identical to the default fonts. During a font assignment, if a loaded font exists that has an identical name as one of the default fonts, the assignment is made to the default font.

Fonts are accessed with codes from 2Ø to 7E hex inclusive and AØ to FE hex inclusive. Fonts can be loaded such that the characters from 2Ø to 7E hex inclusive and AØ to FE hex inclusive have character cells assigned for them.

```
ESC   P  Ps1  ;  Ps2  y  "Font Record"  ;  "Comment Record" ESC  \
1B    5Ø ***  3B ***  79                 3B                  1B   5C
```

ESC P is the device control string introducer and indicates the beginning of load font dot pattern string. Refer to Control String Format in Appendix A for more information.

Ps1 is the LNØ1 identification number. The numeric value must be 1. The entire Load Font Control String is ignored, if the numeric value is other than 1.

Ps2 indicates whether a summary sheet should be printed or not.

Ps2	Action
Ø 3Ø	Print summary sheet
1 31	Do not print summary sheet

The entire load font control string is ignored, if the numeric value is other than a Ø or 1.

Font Record
Font data is transmitted to the device in the font record. The Device Control String Terminator ESC \ or the delimiter ";" marks the end of the font record. The font record essentially consists of binary data that is converted to sixels. Refer to Appendix B for a description of how binary data is encoded into sixels.

Resources are consumed for each font loaded into the printer. Up to 187,ØØØ bytes of font information can be stored in font memory.

When processing the font record, characters in the hex range of ØØ - 3E inclusive are ignored (except for ESC, and CAN). The font byte count is not affected by the characters in the hex range of ØØ to 3E hex inclusive.

If the LNØ1 receives an ESC or CAN control character in the middle of a Font Record, any fonts already received are loaded and made available for assignment and invokation. Incomplete or partial fonts are ignored.

Comment Record
The LNØ1 prints a summary sheet when you select this option using parameter 2. This summary sheet can contain a comment record. The comment record is a line of user supplied text. The comment record may consist of the printable characters in the hex range of 2Ø - 7E inclusive and AØ - FE inclusive. The maximum number of characters permitted in the comment record is 132 decimal; additional characters are discarded.

The comment record follows the font record. They are separated by the ";" character (3B hex). The comment record is optional. When you select "print summary sheet" and do not supply a comment record, the summary Sheet is printed with blanks where the comment record text would be. When the LNØ1 receives a load font dot pattern control string which selects "no summary sheet printed" and contains a comment record, the summary sheet is not printed and the comment record is discarded.

Assign Font Name

The font assignment sequence assigns or associates a font name to a font number. Once this assignment is made, the select graphic rendition escape sequence is used to invoke the font for printing. A maximum of 20 characters (in the hex range of 20 to 7E and A0 to FE inclusive) is permitted for the name. Each font contains a font name as part of its font record. If a font assignment references a font name that does not exist in any currently loaded fonts, the font assignment sequence is ignored. If a font assignment is received that uses a font number with a font name already assigned to it, the current assignment is discarded and the new font name is assigned to the font number. The font must be reinvoked with the SGR escape sequence.

The LN01 allows you to load a font with a name identical to the default fonts. When the user attempts to assign the font name to a font number with a name identical to one of the default fonts, the default font is used.

Fonts are searched in font load order when the font assignment is made. Therefore, if two fonts are loaded, each containing the same font name, the assignment is made to the font that is loaded first.

A font name can be assigned to multiple font numbers. You can assign the ROM resident fonts to any font number. You can also assign other font names to the default fonts (font numbers 10 and 11).

Up to 10 fonts can be assigned at one time. Font assignments can be made any where in the data stream. You can send an unlimited number of assign character set sequences to the printer.

Assigning a font name to a font number that is invoked for printing does not cause the new font to be invoked for printing. For example, assume font STAR is assigned to font #12 and font #12 is invoked for printing. Characters are printed using the STAR font. If a font named LIGHT is then assigned to font #12, characters are still printed using the font STAR. To print characters using the LIGHT font, you must reinvoke #12 again for printing with the SGR sequence. Refer to the Default Values and States section for the default font assignments.

ESC	P	Ps1	;	Ps2	}	"Font Name"	ESC	\
1B	50	***	3B	***	7D		1B	5C

ESC P is the device control string introducer and indicates the beginning of the assign font name string. Refer to Control String Format in Appendix A for more information.

Ps1 - is the LNØ1 identification number. The numeric value must be 1.

Ps2 - is the font number to be assigned to the font name.

Ps2		Font concerned
1	Ø	primary font
31	3Ø	
1	1	first alternative font
31	31	
1	2	second alternative font
31	32	
1	3	third alternative font
31	33	
1	4	fourth alternative font
31	34	
1	5	fifth alternative font
31	35	
1	6	sixth alternative font
31	36	
1	7	seventh alternative font
31	37	
1	8	eighth alternate font
31	38	
1	9	ninth alternate font
31	39	

Font Name - up to 2Ø character (in the hex range of 2Ø to 7E and AØ to FE inclusive) name used to associate font name with number. The font name may be up to 2Ø characters long. If more than 2Ø characters are received, the extra characters are ignored.

ESC \ is the device control string terminator. It indicates the end of the assign font name string. Refer to the Control String Format in Appendix A for more information.

Select Graphic Rendition
The select graphic rendition sequence allows you to invoke a font for printing and select underlining.

SGR - Font Invocation

Font numbers are assigned to font names using the assign character set control string. Up to 22 fonts can be loaded in addition to the two resident fonts. However, a maximum of 23 fonts can be invoked on a page because one of the resident fonts has the other orientation. Since there are only 10 font numbers that can be assigned or invoked, you must reassign and reinvoke fonts (within the page) in order to print more than 10 fonts per page.

Fonts can be invoked anywhere in the data stream using the SGR sequence. The invoked font remains in effect until another SGR sequence or the RIS sequence is sent. Upon power-up or receipt of the RIS sequence, the fonts are selected according to the setting of dip switch #2. Refer to SGR in the Default Values and States section.

The SGR sequence is ignored if no font name is assigned to the font number. If this happens, it is reported on the summary sheet (if requested).

Each page must be printed in either landscape or portrait orientation. Orientations can not be mixed within one page. If you attempt to invoke a font that is of an opposite orientation to the font currently in use, then printing stops on the current page, and starts on the next page in the new orientation.

NOTE

A specific font can have only one orientation (e.g. all characters must be either landscape or portrait).

SGR - Underline

When underline is selected, all printable characters are underlined, including spaces and tabs. The underline mode stays in effect across line and page boundaries.

NOTE

The selective parameters for invoking fonts and underlining may be included within the same parameter list. Refer to control sequence format in Appendix A for more information.

Select Graphic Rendition -- SGR

```
ESC   [  Ps   m
1B    5B ***  6D
```

Ps	Function
Ø 3Ø	Turn underline off
4 34	Begin underline
1 Ø 31 3Ø	Invoke primary (default) font
1 1 31 31	Invoke first alternative font
1 2 31 32	Invoke second alternative font
1 3 31 33	Invoke third alternative font
1 4 31 34	Invoke fourth alternative font
1 5 31 35	Invoke fifth alternative font
1 6 31 36	Invoke sixth alternative font
1 7 31 37	Invoke seventh alternative font
1 8 31 38	Invoke eighth alternative font
1 9 31 39	Invoke ninth alternative font
2 4 32 34	Turn underline off

Select Size Unit

SSU establishes the unit of measurement in which the numeric parameters used in this document are processed. The unit selected is used until the LNØ1 receives another SSU or RIS.

The SSU sequences affects the numeric parameters in the following escape sequences:

DECSHTS - Set Horizontal Tab Stops
DECSVTS - Set Vertical Tab Stops
DECSLPP - Set Lines Per Physical Page
DECSTBM - Set Top and Bottom Margins
DECSLRM - Set Left and Right Margins
HPA - Horizontal Position Absolute
HPR - Horizontal Position Relative
VPA - Vertical Position Absolute
VPR - Vertical Position Relative
DECVEC - Draw Vector

NOTES

The printer converts decipoints into pixel values by multiplying the decipoint value by 5 and then dividing by 12 with the remainder being discarded. All arithmetic operations are performed using integer instructions. Because of this truncation, decipoint values of 1 or 2 convert to Ø pixels.

There is an accumulative positioning error if decipoints is selected for the size unit and the user is sending relative positioning sequences (HPR and VPR). This error is because the conversion between decipoints and pixels is on an integer basis.

Select Size Unit -- SSU

ESC	[	Ps	SP	I
1B	5B	***	2Ø	49

Ps	Size Unit
Ø 3Ø	No Action
2 32	Decipoint (1/72Ø inch)
7 37	Pixel

Numeric parameter values other than 2 or 7 are ignored.

Active Position

Active position defines a specific point on the paper. It consists of a vertical and horizontal coordinate (horizontal active position and vertical active position).

Horizontal active postion is the horizontal coordinate where the next printable image cell is aligned. Vertical active postion is the vertical coordinate where the next printable image cell is aligned (see Figure 3-2).

NOTE
Coordinates numbers begin with Ø not 1.

Printable characters advance the horizontal active position unless the current active postion is at the right margin. If the current active postion is at the right margin, the character is not printed and the active postion is not advanced.

The following explicit commands also modify horizontal active position:

- Carriage Return
- Horizontal Tab
- Horizontal Position Absolute
- Horizontal Position Relative

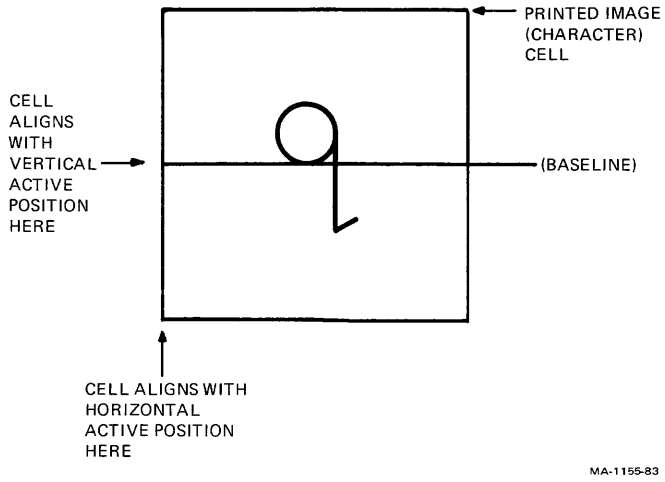


Figure 3-2 Printable Image Cell Orientation

The following commands may implicitly modify horizontal active position:

- Form Feed (Sets horizontal active position to zero)
- Line Feed (when Line Feed New Line mode is selected)
- Set Left and Right Margins
- Set Top and Bottom Margins
- Set Lines Per Physical Page
- Justify
- RIS
- SGR when there is an orientation change (landscape or portrait)

The following explicit commands modify vertical active position:

- Vertical Tab
- Line Feed
- Form Feed
- Vertical Position Absolute
- Vertical Position Relative
- PLU
- PLD

The following commands may implicitly modify vertical active position:

- SGR when there is an orientation change (landscape or portrait)
- Set Lines Per Physical Page
- Set Top and Bottom Margins
- RIS

Adjusting Vertical Active Position

Printing a line of characters at the vertical active position may result in the vertical active position being adjusted in order to prevent large fonts from over printing the previous line. The following sections defines the adjustment to the vertical active position.

Vertical Line Spacing Parameters

H is the offset of the highest ascender above the baseline (300ths of an inch)

B is the offset of the lowest descender below the baseline (300ths of an inch)

LS is the normal line spacing distance for the font including leading (white space above and/or below the character within the printable image cell). The LS value is usually larger than (B + H) by an amount equal to the leading.

Line Spacing (Portrait Orientation)
Line spacing in the portrait orientation is based on two factors; the size of the largest font in the line being processed when the line feed is encountered, and the size of the largest font in the line being processed after the line feed is executed.

If the first line is called line 1 and the second line is called line 2, then the distance moved down a page when a line feed is executed is calculated as follows:

Line Space = Lower Space (1) + Upper Space (2)

where:

Lower Space = B value from font header (of largest font)
Upper Space = (LS - B) value from font header (of largest font)

The largest font on the line is defined as the font with the largest B value. If the largest font on both lines are the same size, then the spacing is:

B + (LS - B) = LS (the line spacing including leading)

Line Spacing (Landscape Orientation)
Line spacing in landscape orientation uses the LS value only (of the largest font on the line). Since imaging in landscape begins at the top of the highest ascender in the font, the H value is used to determine the position of the baseline of a landscape line.

No vertical active position adjustments are made in the following cases:

1. Initial vertical position established using a vertical position absolute command in portrait orientation.

2. Initial vertical position established using a vertical tab in portrait orientation.

Active Position Movement
The HPA, VPA, HPR, VPR sequences allow you to set the active position to a specific location on the paper. The HPA and VPA sequences are bidirectional, that is you can move active position anywhere on the page from anywhere on the page within the margins. The HPR and VPR sequences are unidirectional, that is you can only increase the current active position within the page. If you attempt to print beyond the margins, the active position is held to the margin.

NOTE
The numeric parameter(s) used for the active postition movement sequences can be expressed in units of either decipoints or pixels using the SSU sequence. Refer to the SSU sequence description for more information.

Horizontal Position Absolute - HPA

ESC	[	Pn
1B	5B	*** 60

HPA causes the horizontal active position to be moved to the Pn-th horizontal position of the active vertical position. If an attempt is made to move the horizontal active position past the right margin, the horizontal active position stops at the right margin.

The default value is 1.

Horizontal Position Relative - HPR

ESC	[	Pn	a
1B	5B	***	61

HPR causes the horizontal active position to be moved to Pn + the current horizontal position on the active vertical position. The new horizontal active position is computed by adding Pn to the horizontal active position value. If you attempt to move the horizontal active position past the right margin, the horizontal active position stops at the right margin.

If decipoints is selected as the size unit and you send the relative position sequence with 1 or 2 as its parameter value, due to conversion to pixels the horizontal active position will not change.

The default value is 1.

Vertical Position Absolute - VPA

ESC	[	Pn	d
1B	5B	***	64

VPA causes the vertical active position to be moved to Pn-th vertical position at the active horizontal position. If you attempt to move the vertical active position below the bottom margin, the vertical active position stops at the bottom margin.

The default value is 1.

Vertical Position Relative - VPR

ESC	[	Pn	e
1B	5B	***	65

VPR causes the vertical active position to be moved to Pn + the current vertical active position at the active horizontal position. If you attempt to move the vertical active position below the bottom margin, the vertical active position stops at the bottom margin. The new vertical active position is computed by adding Pn to the current vertical active position.

If decipoints is selected as the size unit and you send the relative position sequence with 1 or 2 as its parameter value, due to conversion to pixels the vertical active position will not move.

The default value is 1.

Horizontal Margins

The left horizontal margin specifies the first printable position on a line; the right horizontal margin specifies the last printable position on a line. Printing is permitted only within the left and right margins inclusive.

Margins are defined as hard margins. That is, neither the horizontal active position nor the printed image may be placed outside the margins. There are two exceptions:

The escape sequence DECVEC - Draw Vector is permitted to draw lines outside the margins.

During justification if the interword spacing is less than 60% of the width of the space character, the text is printed unjustified. The resultant text will exceed the right margin.

NOTE

If the printed image is placed as to exceed the left margin, it is shifted within the left margin. If a printed image is placed as to exceed the right margin, it is ignored.

The set horizontal margins sequence when accompanied by two numeric parameters, sets the left and right margins. If the first parameter is the smaller of the two, the left margin is set to that specified parameter and the right margin is set to the second.

The sequence is ignored if the first parameter is greater than or equal to the second parameter. The sequence is also ignored if one of the specified parameters would set the right margin further right than 13.65 inches (4095 pixels).

If the first parameter in the sequence is omitted, the remaining parameter sets the right margin to the specified value. If an attempt is made to set the right margin to the left of the left margin, the sequence is ignored.

If the second parameter in the sequence is omitted, the first parameter sets the left margin to the specified value. If an attempt is made to set the left margin to the right of the right margin, the sequence is ignored. If the horizontal active position is less than the new left margin, it is set to the new left margin.

If both parameters are zero or omitted, the margins are unchanged. Refer to the Default Values and States section for the default margin settings.

Set Left and Right Margins - DECSLRM

ESC	[	Pn	;	Pn	s
1B	5B	***	3B	***	73

The left margin is set to the value of the first numeric parameter; the right margin is set to the value of the second parameter.

NOTE

The numeric parameter(s) can be expressed in units of either decipoints or pixels using the select size unit (SSU) sequence. Refer to the SSU sequence description for more information.

Horizontal Tabulation Stops

A horizontal tab is a preselected point on a line to which the horizontal active position advances when a horizontal tab control character is received. The printer has 32 possible horizontal tab stops. Each stop may be set independently. Setting a stop already set has no effect; the same is true for clearing a stop already cleared. Tab stops may be set regardless of margins. Refer to the Default Values and States section for a description of the default horizontal tabulation stops.

The set tab control sequence can have a variable number of numeric parameters. Numeric parameters can be sent in any order in the sequence.

NOTE

Four seperate escape sequences are required to load 32 stops, as each sequence can only contain a maximum of eight tab stops.

New tab stops are inserted starting with the tab stop with the lowest value. If more than 32 tab stops exist, the first 32 tab stops are retained, while the tabs with the highest values are discarded.

Horizontal tabulation stops are set to the corresponding absolute position of all lines.

Set Horizontal Tabulation Stops - DECSHTS

ESC	[	Pn	;	...	;	Pn	u
1B	5B	***	3B	...	3B	***	75

NOTE

The numeric parameter(s) can be expressed in units of either decipoints or pixels using the select size unit (SSU) sequence. Refer to the SSU sequence description for more information.

Clearing Tab Stops

Tabulation Clear - TBC

ESC	[	Ps	g
1B	5B	***	67

Clear the tab stops as indicated by Ps.

Ps	Function
Ø 3Ø	No action
3 33	Clear all horizontal tab stops.
4 34	Clear all vertical tab stops.

Set Lines Per Physical Page

This sequence defines the form length and the location of the origin point Ø,Ø. Form length defines the logical length of the form.

If the user attempts to set the form length greater than 14.Ø inches (42ØØ pixels) the form length is set to 14.Ø inches (42ØØ pixels).

The maximum printing length of the LNØ1 is 13.64 inches (4Ø95 pixels). If the form length is set to 14.Ø inches and the user attempts to print images within the first .35 inches the images may be printed at the bottom of the form. Although images may be displaced if printed in the first .35 inch area for 14.Ø inch form length, vertical active position is properly maintained.

For landscape orientation, the origin is fixed at the upper left-hand corner of the paper. That means if the form length is set to 5 inches (for 8.5 inch paper), text is printed on the first 5 inches of the paper. The last 3.5 inches are left blank.

For portrait orientation, the location of the origin is set in relation to the form length. The location of the origin is that point which is n inches from the bottom edge of the paper, where n is the form length. Therefore, for different form lengths, the origin will be at different locations.

For example, if form length is 4 inches (for 11 inch paper), text is printed in the last 4 inches of the paper, with first 7 inches left blank. In this example the Ø,Ø point is 4 inches from the bottom edge of the paper (or 7 inches from the top edge).

Refer to the Default Values and States section for a description of the default values for the DECSLPP sequence.

NOTE

The numeric parameter(s) can be expressed in units of either decipoints or pixels using the SSU sequence. Refer to the SSU sequence description for more information.

Set Lines per Physical Page - DECSLPP

ESC	[	Pn	t
1B	5B	***	74

Set form length to Pn. Set top margin to Ø. Set bottom margin to Pn.

Vertical Margins

The top vertical margin specifies the first printable line; the bottom vertical margin specifies the last printable line. Printing is allowed only on the lines between the top and bottom margins inclusive.

Margins are defined as hard margins. That is, neither the vertical active position nor the printable image can be placed outside the margins. The only exception to this rule is the Draw Vector (DECVEC) sequence. You can use the DECVEC sequence to draw lines outside the margins.

NOTE

If the printed image is placed as to exceed the vertical margins, it is moved within the margins.

When the form length is changed, vertical margins are cleared; the top margin is set to zero and the bottom margin is set to the form length. The following conditions must exist to set new vertical margins.

- Top margin must be greater than or equal to one
- Bottom margin must be greater than or equal to the top margin
- Form length must be greater than or equal to the bottom margin

Attempting to print above the top margin or below the bottom margin automatically advances the vertical active position to the top margin of the next page. For example, a line feed (LF) control character received at the bottom margin causes the printer to perform a form feed.

The set vertical margins sequence, accompanied by two parameters, set the top and bottom margins. If the first parameter is the smaller of the two, the top margin is set to the first parameter and the bottom is set to the second. Then the vertical active position may be repositioned to the new top margin depending on the current active vertical position and printer activity.

The sequence is ignored if the first parameter is greater than or equal to the second parameter. The sequence is also ignored if one of the specified parameters would set the bottom margin past the assigned form length.

If the first parameter in the sequence is omitted, the remaining parameter sets the bottom margin to the specified line. If an attempt is made to set the bottom margin above the top margin, the sequence is ignored.

If the second parameter in the sequence is omitted, the first parameter sets the top margin to the specified line. If an attempt is made to set the top margin below the bottom margin, the sequence is ignored. If the vertical active position is less than the new top margin, it is set to the new top margin.

If both parameters are set to zero or omitted the margins are unchanged. Refer to the Default Values and States section for the default margin settings.

Set Top and Bottom Margins - DECSTBM

ESC	[	Pn1	;	Pn2	r
1B	5B	***	3B	***	72

The top margin is set to Pn1, and the bottom margin is set to Pn2. If the first parameter is greater than the second parameter, the sequence is ignored.

Whenever the vertical active position is less than the new top margin or greater than the new bottom margin, the vertical active position is set to the top margin.

NOTE

The numeric parameter(s) can be expressed in units of either decipoints or pixels using the SSU sequence. Refer to the SSU sequence description for more information.

Vertical Tabulation Stops

A vertical tab is a preselected line to which the vertical active position advances when a vertical tab control character is received. The printer has 32 possible vertical tab positions. Vertical tabs can be set independently. Setting a tab stop already set has no effect; the same is true of clearing a tab stop already cleared. Vertical tab stops can be set regardless of margins. Refer to the Default Values and States section for a description of the default vertical tabulation stops.

Set Vertical Tabulation Stops - DECSVTS

ESC	[	Pn	;	...	;	Pn	v
1B	5B	***	3B	...	3B	***	76

This control sequence can have a variable number of numeric parameters. The numeric parameters can be sent in any order.

NOTE

Four seperate escape sequences are required to load 32 stops, as each sequence can only contain a maximum of eight tab stops.

New tab stops are inserted starting with the tab stop with the lowest value. If more than 32 tab stops exist, the first 32 tab stops are retained, while the tabs with the highest values are discarded.

NOTE

The numeric parameter(s) can be expressed in units of either decipoints or pixels using the select size unit (SSU) sequence. Refer to the SSU sequence description for more information.

Text Justification

When text justification is selected, the LNØ1 printer justifies text lines within the currently defined horizontal margins by varying the spacing between words. The LNØ1 does not determine the end of line nor does it make hyphenation decisions. The spacing between words is adjusted such that the first character of the first word starts on the left margin; the last character of the last word ends on the right margin. The space between words of the justified text line is evenly distributed.

The minimum and maximum distance between words will not be less than 60% nor greater than 200% of the width of the space character in the font from which the words' characters are derived. A line of text will not be justified if the maximum or minimum space size restrictions cannot be honored. In this situation the unjustified text is printed, (even though it may print the line past the right margin).

Text justification is performed on all text which occurs between a start justification and stop justification sequence.

A start justify sequence detected within the line defines the left justification point for that line, and a stop justify sequence detected within a line defines the right justification point for that line. To justify a line of text according to the left and right margins, the start and stop justify sequences must encompass the line beginning and end points.

The following control functions determine the end of the line to be justified when justification is started.

CR - Carriage Return
LF - Line Feed
VT - Vertical Tab
HPA - Horizontal Position Absolute
VPA - Vertical Position Absolute
FF - Form Feed

If justification is turned on then off for only a portion of a text line, those space characters outside the start and stop justify sequence are not modified. Those characters outside of the start and stop justify sequence use the width of the space character in the invoked font.

Justify - JFY

ESC	[	Ps	SP	F
1B	5B	***	2Ø	46

Perform the justification action as indicated by Ps.

Ps	Function
Ø 3Ø	Stop Justification.
2 32	Start Justification.

Paper Tray Selection

The LNØ1 printer has two paper trays; the top and bottom. You can choose either tray, from which paper is used for printing. The LNØ1 printer defaults to alternate usage of the top and bottom trays, starting with the top tray. When the one tray becomes empty, the other tray is automatically selected.

You can explicitly select a tray by sending the DECASFC escape sequence. Explicit selection disables the alternate tray selection method of operation. When an explicitly selected tray becomes empty, the LNØ1 printer waits indefinitely for the operator to fill the tray before proceeding.

The last DECASFC escape sequence received prior to the completion of page composition determines the paper source for the printing of that page.

The default value is Ø.

Automatic Sheet Feeder Control - DECASFC

ESC	[	Ps	!	v
1B	5B	***	21	76

Paper tray selection is enabled as indicated by Ps.

Ps	Function
Ø 3Ø	Enable alternate tray selection
1 31	Select a sheet from the top tray
2 32	Select a sheet from the bottom tray

Line Feed New Line Mode

Line feed new line mode defines the function of the line feed (LF) control character. When this mode is disabled, and the LF control character is received, the printer increments the vertical active position. When this mode is enabled, and the LF control character is received it returns the horizontal active position to the left margin in addition to the usual functions.

New line mode - LNM

ESC	[	Ps	l
1B	5B	***	6C

This sequence disables line feed new line mode as indicated by Ps.

Ps	Function
Ø 3Ø	No action
2 32	LF control character advances only the vertical active position.

ESC	[	Ps	h
1B	5B	***	68

This sequence enables line feed new line mode as indicated by Ps.

Ps	Function
Ø 3Ø	No action
2 Ø 32 3Ø	LF control character causes the active position to be moved to the left margin of the next line.

Document Finishing
The LNØ1 has the ability to deliver documents to the output tray in an "offset" or "no-offset" position. When the no offset position is selected pages are delivered to the output tray in a single stack. When the LNØ1 receives the DECFIN sequence, subsequent pages are delivered to the output tray slightly offset from the previous stack.

The offset remains in effect until it is explicitly changed. The paper offset is reset to the default or "no-offset" position at device power-up.

DECFIN - Document Finishing

ESC	[	Ps	!	}
1B	5B	***	21	7D

Document finishing is enabled as indicated by Ps.

Ps	Finishing operation
Ø 3Ø	No offset
1 31	Toggle the offset of the paper

The default selective parameter value is Ø.

Drawing Vectors

This sequence is used to draw a line of variable thickness. The X direction is parallel to the short edge of the paper, the Y direction is parallel to the long edge of the paper. Before drawing the line, the Draw Vector function saves the current active position. The line is drawn and the active position is restored to its previous location.

NOTE

The numeric parameter(s) used in drawing vector sequence can be expressed in units of either decipoints or pixels using the SSU sequence. Refer to the SSU sequence description for more information.

Draw Vector - DECVEC

ESC	[	Pn1	;	Pn2	;	Pn3	;	Pn4	;	Pn5	!	|
1B	5B	***	3B	***	3B	***	3B	***	3B	***	21	7C

Pn1 - defines the line orientation (axis) along which the length of the line is drawn.

Pn1	Axis
Ø 3Ø	X (parallel to short edge of paper)
1 31	Y (parallel to long edge of paper)

Pn2 - defines the distance from the origin along the X axis where the line starts. The maximum value is 2549 pixels.

NOTE

Origin is defined in the Set Lines Per Physical Page section in this chapter.

Pn3 - defines the distance from the origin along the Y axis where the line starts. The maximum value is 4Ø95 pixels.

Pn4 - defines the length of the line. If the pixel values Ø or 1 are received, Pn4 is set to 2 pixels. The maximum value for a line drawn along the X axis is 255Ø pixels. The maximum value for a line drawn along the Y axis is 4Ø96 pixels.

NOTE

Decipoint values of Ø-5 are rounded up to 5 decipoints (2 pixels).

Pn5 - defines the line thickness/width. The direction of the thickness/width is perpendicular to the length. The thickness of a line always increases active position. If the values Ø or 1 are received, Pn5 is set to 2 pixels. The maximum value for a line drawn along the X axis is 4Ø96 pixels. The maximum value for a line drawn along the Y axis is 255Ø pixels.

NOTES

Decipoint values of Ø-5 are rounded up to 5 decipoints (2 pixels).

Lines may be drawn regardless of margins.

Parameter values exceeding the maximum are truncated to the maximum.

The DECVEC sequence consumes resources within the LNØ1. Refer to Chapter 4 for more information.

Because it is possible for a line to extend beyond the physical limits of a page, the user must ensure that the DECVEC sequence is coordinated with the proper paper size.

Superscripting And Subscripting

The partial line up and partial line down sequences are used to print superscript and subscript characters. The following section describes the partial line up and partial line down sequences.

Partial Line Up - PLU

ESC L
1B 4C

Printing superscript characters can be performed with the PLU escape sequence. The PLU sequence causes the vertical active position to move up in the vertical direction a predefined distance. The distance moved up in the vertical direction is determined by the currently invoked font. The PLD sequence causes the vertical active position to return to the previous baseline. Only one level of superscripting is permitted.

Partial Line Down - PLD

ESC K
1B 4B

Printing subscript characters can be performed with the PLD escape sequence. The PLD sequence causes the vertical active position to move down in the vertical direction a predefined distance. The distance moved down in the vertical direction is determined by the currently invoked font. The PLU sequence causes the vertical active position to return to the previous baseline. Only one level of subscripting is permitted.

NOTE

If the vertical active position is near the top or bottom margins, the PLD and PLU functions may cause the superscripted or subscripted character to be printed outside the margins.

Reset To Initial State
This sequence causes the LNØ1 to reset all features to their default state. Refer to the following section for a list of the features and their default values.

Reset to Initial State - RIS

ESC c
1B 63

Default Values and States
The LNØ1 printer is set to the following values and states upon device power-up or when the RIS escape sequence is received.

Load Font Control String
Upon device power-up the ROM landscape and portrait fonts are available for printing. When the RIS escape sequence is sent to the printer, any currently loaded fonts remain intact.

Assign Font Name
Upon device power-up and receipt of the RIS sequence the landscape ROM font is assigned to font #1Ø and the portrait ROM font is assigned to font #11.

Select Graphic Rendition
Upon device power-up and receipt of the RIS sequence fonts are invoked for printing according to the setting of dip switch #2. When the dip switch is set to the "on" position, the landscape ROM font #10 is selected. When the dip switch is set to the "off" position, portrait ROM font #11 is selected.

NOTES

The landscape ROM font is always used for printing the summary sheet no matter which mode is selected.

Upon device power-up and receipt of the RIS sequence underlining is disabled.

Set Horizontal Tab Stops
Upon device power-up default horizontal tab stops are set one tab stop every 8 characters (see the table below for actual distance). The first default horizontal tab stop is located one tab stop distance from the left margin (8 characters).

Upon receipt of the RIS sequence all tabs currently set are cleared. Tabs are then set to the power-up defaults.

	Distance between tab stops	
Landscape (60 Hz):	0.587 inches	176 pixels
Landscape (50 Hz):	0.640 inches	192 pixels
Portrait (60 Hz):	0.800 inches	240 pixels
Portrait (50 Hz):	0.666 inches	200 pixels

Set Vertical Tab Stops
Upon device power-up default tab stops are set one tab stop every 1 line (see table below for actual distance). The first default vertical tab stop is located one tab stop distance from the top margin (1 line).

Upon receipt of the RIS sequence all tabs currently set are cleared. Tabs are then set to the power-up defaults.

	Distance between tab stops	
Landscape (60 Hz):	0.117 inches	35 pixels
Landscape (50 Hz):	0.117 inches	35 pixels
Portrait (60 Hz):	0.166 inches	50 pixels
Portrait (50 Hz):	0.166 inches	50 pixels

Set Lines per Physical Page
Upon device power-up and receipt of the RIS sequence the default form length is set according to the following table:

	Form Length
Landscape (60 Hz):	8.50 inches 2550 pixels
Landscape (50 Hz):	8.27 inches 2481 pixels
Portrait (60 Hz):	11.00 inches 3300 pixels
Portrait (50 Hz):	11.69 inches 3507 pixels

Set Top and Bottom Margins

NOTE
All measurements are from the nearest edge of the paper.

Landscape (60 Hz):

Top Margin: .4 inches
Bottom Margin: .4 inches

Landscape (50 Hz):

Top Margin: .39 inches
Bottom Margin: .18 inches

Portrait (60 Hz):

Top Margin: .5 inches
Bottom Margin: .5 inches

Portrait (50 Hz):

Top Margin: .845 inches
Bottom Margin: .845 inches

The top and bottom margins are set to produce pages with the following number of lines per page for each of the orientations:

Landscape (60 Hz):

Lines/Page: 66

Landscape (50 Hz):

Lines/Page: 66

Portrait (60 Hz):

Lines/Page: 60

Portrait (50 Hz):

Lines/Page: 60

Set Left and Right Margins

NOTE
All measurements are from the nearest edge of the paper.

Landscape (60 Hz):

Left Margin: .66 inches
Right Margin: .66 inches

Landscape (50 Hz):

Left Margin: .57 inches
Right Margin: .56 inches

Portrait (60 Hz):

Left Margin: .25 inches
Right Margin: .25 inches

Portrait (50 Hz):

Left Margin: .802 inches
Right Margin: .802 inches

The left and right margins are set to produce lines with the following number of maximum characters per line.

Landscape (60 Hz):

Characters/Line: 132

Landscape (50 Hz):

Characters/Line: 132

Potrait (60 Hz):

Characters/Line: 80

Portrait (50 Hz):

Characters/Line: 80

Horizontal and Vertical Position Absolute
Upon power-up or receipt of the RIS sequence, the active position is set to the upper left-hand corner of the paper. This position is where the first line and character would be printed assuming no other position commands.

Justify
Upon power-up or receipt of the RIS sequence, justification is disabled.

Paper Selection
Upon power-up or receipt of the RIS sequence, the top tray is selected, unless empty then the bottom tray is selected. The printer automatically cycles through the paper trays.

Line Feed New Line
Upon power-up or receipt of the RIS sequence, line feed new line mode is enabled.

Document Finishing
Upon power-up documents are delivered to the output tray in the "no-offset" position.

Upon receipt of the RIS sequence the position of the output stacker is not effected.

Select Size Unit
Upon device power-up or receipt of the RIS sequence the printer is set to pixel mode.

Print Summary Sheet
Upon power-up or receipt of the RIS sequence, printing of the summary sheet is disabled.

Summary Sheet
No summary sheet is printed upon receipt of the RIS sequence. Also, the internal error list is cleared when the RIS sequence is received.

LED Display
The operators LED display is set to "Ø1" when the first Form Feed (hex ØC) control character is received after an RIS sequence.

PLD and PLU
Upon receipt of the RIS sequence the vertical active position is reset to the baseline.

CHAPTER 4
PAGE DEMOGRAPHICS

GENERAL

As described in Chapter 2, commands and printable characters are not fully executed when received. They are stored in page buffers and are completed at printing time. Both the storage requirements and the execution time requirements limit the number of characters and commands that a page can contain. This chapter defines the storage and execution time limits of the LNØ1 Printer. These limits are collectively known as page demographics.

LINE PRINTER SUBSET

In general the LNØ1 can be used as a line printer without change to existing software. However, the limitations described in this section must be observed. In addition to line printer emulation, you may provide a set of fonts and switch among them and the resident font within a page. If you use this added functionality, refer to the Full Functionality Demographics section for the corresponding limitations.

Overstriking

Overstriking is the ability to superimpose one line of printed text on top of another. It can be used to form combinations of characters, such as underlining or slashing a Ø.

Overstriking is accomplished by sending a line of text followed by a carriage return with no line feed. The next line of text is then placed over the first.

> NOTE
> For a line printer, bolding is accomplished by overstriking the identical text line several times making the line appear darker. Since the LNØ1 is not an impact printer, overstriking the character will not cause the character to appear bolder.

Observe the following limits when using overstrikes:

- Landscape Orientation (Figures 4-1 and 4-2)
  Up to 142 printable characters per print line with a maximum of five carriage returns (including final (CR) (LF) pair), when printing with the ROM resident DELandscape font.

- Portrait Orientation (Figures 4-3 and 4-4)
  Up to 150 printable characters per print line with a maximum of two carriage returns (including final (CR) (LF) pair), when printing with the ROM resident DETitan font.

For example: On a landscape page, a 71 character print line followed by a CR (carriage return) followed by another 71 character print line followed by (CR) (LF) pair would reach the 142 printable character limit.

Final Page
The final page is not ejected until one of the following page eject commands is received:

- Form Feed
- Positioning command causing a Form Feed
- Page orientation change

The operating sytem software or the operator should be aware of this.

Landscape Print Profile
Up to 132 characters per line and 66 lines per page may be printed in landscape orientation using the ROM Resident Font.

The LN01-AA resident font is DELandscape13.6-@. It has a point size of approximately 6.7 and character cells of 35 by 22 pixels, resulting in a print image area of 7.7 by 9.68 inches. Margins are chosen to optimize this format by centering on 8.5 by 11 inch paper as shown in Figure 4-1.

The LN01-AB resident font is DELandscape12.5-@, it has a point size of approximately 6.7 and character cells of 35 by 24 pixels, resulting in a print image area of 19.5 by 27 cm (7.7 by 10.56 inches). Margins are chosen to optimize this format for 21 by 29.7 cm (8.27 by 11.69 inches) paper as shown in Figure 4-2.

Portrait Print Profile
Up to 80 characters per line and 60 lines per page may be printed in portrait orientation using the ROM resident font.

The LN01-AA resident portrait font is DETitan10-R it has 10 point characters and a cell size of 50 by 30 pixels. Margins are shown in Figure 4-3.

The LN01-AB resident portrait font is DETitan12-T. It has 9 point characters with a cell size of 50 by 25 pixels. Margins are shown in Figure 4-4.

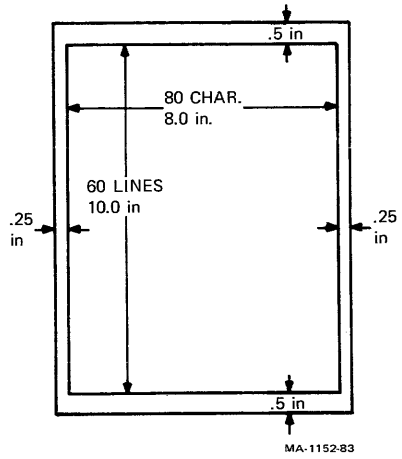


Figure 4-1 LN01-AA ROM Resident Landscape Font

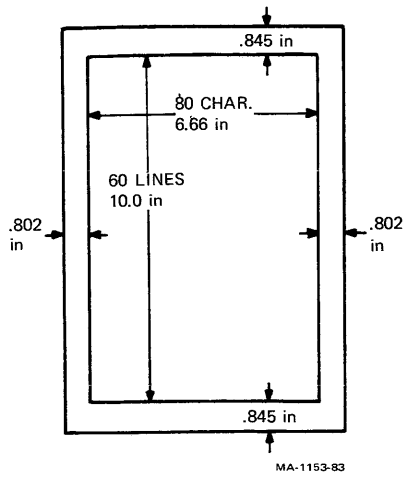


Figure 4-2 LN01-AB ROM Resident Landscape Font

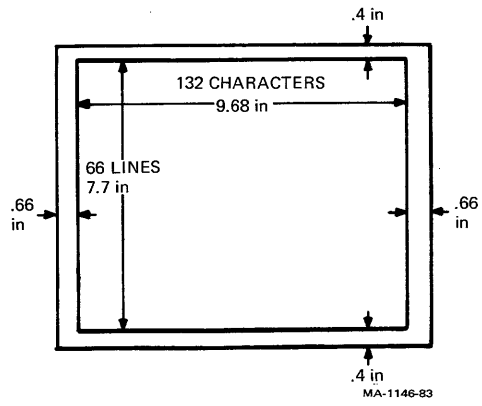


Figure 4-3 LN01-AA ROM Resident Portrait Font

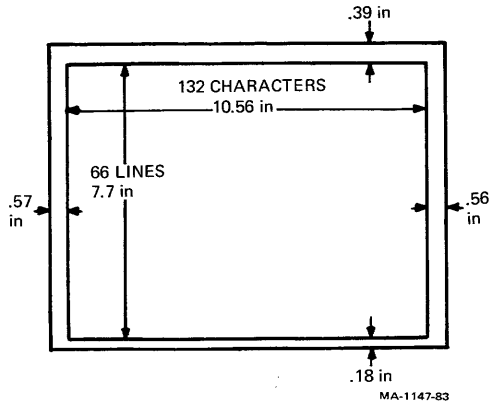


Figure 4-4 LN01-AB ROM Resident Portrait Font

FULL FUNCTIONALITY DEMOGRAPHICS

The LNØ1 supports a wide range of electronic printer features. You can produce landscape or portrait oriented pages, adjust margins and print various font sizes and styles on the page. You can also print Logos and signatures. The following sections list and describe the full functionality page demographic limits.

Limitation Types

There are 4 types of limits for page composition complexity. They are based on specific electro-mechanical implementations, controller software implementations and/or processing time restrictions.

- Page too complex to print
- Page too dense
- Format limits exceeded
- Print format larger than paper

Page Too Complex To Print

If the time required to complete the processing of a line of text is more than the time required to print the preceeding lines (e. g. the scanning laser passes the point of imaging before the processing is complete), the page is too complex to be printed. Excessive use of overstriking, complex juxtapositioning of different sized characters, or printing of many small size characters can cause this to occur.

Page Too Dense

If a page contains more than 3Ø,ØØØ bytes of page data the LNØ1 internal page storage is exceeded. The data in excess of 3Ø,ØØØ bytes is printed on the next page.

Format Limits Exceeded

If a page contains more than 5ØØ data buffers, the LNØ1 internal page storage is exceeded. The data buffers in excess of 5ØØ are printed on the next page.

Print Format Larger Than Paper

The positioning of text is not checked by the LNØ1. Management of paper size to prevent loss of margins or placement of text beyond the paper's boundary is a user responsibility.

Data Buffers

Input data is translated to an internal format stored in data buffers. Basically, a data buffer corresponds to a line of text. However, additional commands such as underline force the LNØ1 to start a new data buffer. Data buffers are collected in order to form pages in the page buffer memory. There are two functionality limits imposed by the data buffers and their format:

- There is a maximum of 5ØØ data buffers within a page. More than 5ØØ data buffers on a page result in the reminder being printed on the next sheet of paper.

- The total number of bytes consumed by data buffers on a page can not exceed that capacity defined in the Page Buffer Capacity section. This byte count includes the header and trailer bytes of each data buffer.

Data buffers are variable in size from 8 to 24Ø bytes. The format is shown in Figure 4-5.

The Image Data area consists of 1 to 233 bytes of data, allocated as follows:

- 1 byte for each printable character.

- 2 bytes every time a font change occurs between two consecutive characters in a data buffer, to indicate the change. (The first font selector is already included in the header.)

- 3 bytes for each horizontal position relative command, horizontal tab command, or space character when the space characters within the buffer are adjusted in size for justification.

- Draw vector command - A draw vector consumes 9 bytes for each 2ØØ pixels or partial 2ØØ pixels along the Y axis. Nine additional bytes are used if the partial 2ØØ pixels is not a multiple of 16,

- Underline - A maximum of 63 underline segments per line.

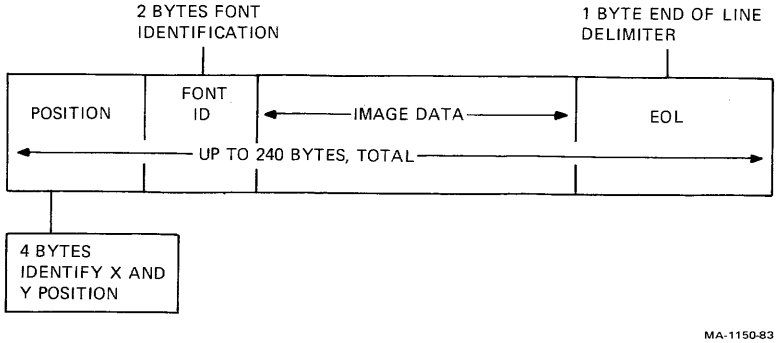


Figure 4-5 Data Buffer Format

Data buffers are created (a new position and font ID header created) for the following conditions:

- When a printable character is invoked and there are no open (header but no eol) data buffers.
- The first underline command on a line results in a data buffer being created for all underlines on the line (a maximum of 63). The data buffer containing the printable characters for the line is filled in parallel to the underline data buffer.
- Draw vector command

Image data is placed into the current open data buffer as each input function is processed.

Data buffers are terminated (the end of data buffer code is placed in the current buffer and the next printable function causes a new buffer and header to be created) upon the following commands or conditions:

- More than 233 bytes of image data for the current buffer
- A vertical positioning command - LF, FF, VPR, VPA, VT, PLU, PLD.
- An absolute horizontal positioning command - CR, HPA.
- A draw vector command *
- A character is printed from a font of different height than the other characters in the data buffer - landscape mode only.

NOTE
Consecutive positioning commands do not consume data buffers.

*Draw vector causes any current data buffer to be terminated, a new buffer to be created, filled with vector image data, and terminated.

Page Buffer

Page buffer memory is used to store whole formated pages of data buffers prior to printing. Thirty thousand bytes of page buffer are available, which may contain a maximum of three pages. The byte count for a page is the sum of all the bytes used for the corresponding data buffer plus 1 byte for page orientation. A single page may consume up to 30,000 bytes. Pages requiring more than 30,000 bytes are split across multiple sheets of paper. Pages of such a size that 3 complete consecutive pages can not be stored simultaneously may result in decreased throughput.

Band Buffer

General

Two ping-pong band buffers are used. Each band buffer is a 4096 by 32 bit buffer within which the data for one "band" (32 scan lines) can be stored. Each bit within a band buffer represents a marked or unmarked pixel on the paper at 300 dots to the inch. While one band buffer's data is being used to produce dots in the printing process, the other band buffer is being filled with pixel data for the next "band".

In the scan direction (long edge of paper), images are started on an even numbered pixel. Although the imagining has a 300 dots per inch resolution, images can only be located with a 150 dots per inch resolution. The actual location of the image is obtained by truncating the logical start address (low order bit forced to a zero).

In landscape mode, the first scan of the first band is vertical position 0, with successive bands being adjacent every 32 scans such that the last scan of the first band is vertical position 31, the first scan of the second band is vertical position 32, and so on for the remainder of the page.

In portrait mode, the first scan of the first band is horizontal position 0, with successive bands being adjacent every 32 scans such that the last scan of the first band is horizontal position 31, the first scan of the second band is horizontal position 32, and so on for the remainder of the line.

Band Buffer Capacity

As the physical length of each scan line in the band buffer is 4096 bits (pixels), the maximum data imaging along the long edge of the paper is 13.65333 inches (4096 pixels/300 pixels per inch imaging resolution). Thus, 14 inch paper cannot be imaged end-to-end. When using 14 inch paper, a minimum right margin of 0.3467 inch is forced for landscape and the origin is at least 0.3467 inch from the top edge of the paper in portrait.

Band Buffer Complexity

There are timing requirements on the number of printable characters and commands that can be processed to create a band buffer. A 32 scan lines band buffer will sometime cover several text lines and a larger number of data buffers.

The number of printable characters included in a band buffer depend on the following variables:

- Heigth of the fonts
- Width of the fonts
- Length of the line
- Length of the page
- Number of fonts in a line

The following formulas provide the maximum line lengths (landscape orientation) or page length (portrait orientation) based on the other variables.

Font Size (LANDSCAPE):

Single Font Sizes
The following example shows the worst case landscape situation involving a single font size. One full character cell and two partial characters are in the band buffer. This condition can exist for all character cells from 16 to 30 pixels high.

Processing Time Based On Font Size (Landscape)
Band buffer image processing time is a complex function of character cell dimensions. Excessive processing times result in data being lost during the printing process. In order to insure that data is not lost, line lengths should be limited based on the relationships listed in Table 4-1.

Multiple Font Sizes
When arbitrarily mixing character cell sizes on a page, to be most conservative, and guarantee that image processing time is sufficient, it should be assumed that all character cells are contained within a hypothetical cell as shown in Figure 4-6.

If multiple font sizes are not arbitrarily mixed but separated vertically (headings in one size, text in another, for example), then the individual expressions given above apply.

NOTE

Use the multiple cell size rule if a size change occurs within a space of 32 pixels (0.107 inches) as in this event, multiple cell sizes can occur in the band buffer.

Table 4-1 Characters Per Line (Landscape)

Character Cell Dimensions				
	Width			
Height	Dots	Bytes	Characters Per Line	Formulas
16-30	< 8	1	123	
	9-16	2	114	
	17-24	3	106	1600
	25-32	4	100	------
	33-40	5	94	12 + w
	41-48	6	89	
31	< 8	1	140	
	9-16	2	129	
	17-24	3	119	1600
	25-32	4	111	--------
	33-40	5	104	10.4 + w
	41-48	6	98	
32 or greater	< 8	1	182	
	9-16	2	163	
	17-24	3	148	1600
	25-32	4	135	-------
	33-40	5	125	7.8 + w
	41-48	6	116	

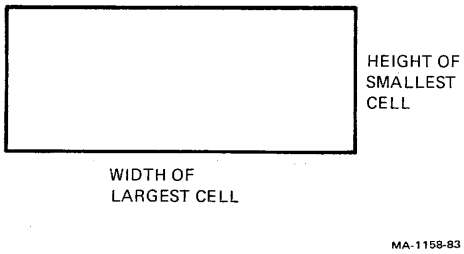


Figure 4-6 Hypothetical Character Cell (Landscape)

Font Size (Portrait)

Single Font Sizes
The following example shows the worst case portrait situation for fixed-pitch characters, involving a single font size. There can be as many as 4 complete and partial characters in the band buffer. With proportional-spaced characters, as many as 6 characters and partial characters can be in a band.

In general, for character cells of various widths:

Cell Range	Maximum number of cells in a band
>30 pixels	2
16-30 pixels	3
11-15 pixels	4
8-10 pixels	5
7 pixels*	6

Image Processing Restrictions (Portrait)
As in the landscape case, band buffer image processing time is a complex function of character cell dimensions. Excessive processing times result in data being lost during the printing process. In order to insure that data is not lost, the number of lines on a page should be limited based on the relationships listed in Table 4-2.

Multiple Font Sizes
When arbitrarily mixing character cell sizes on a page, it should be assumed that all characters are contained within hypothetical cells as shown in Figure 4-7. This is a conservative approach to calculating maximum lines per page which guarantees that image processing time is sufficient.

If multiple font sizes are not arbitrarily mixed, but are isolated (separated from each other horizontally by at least 32 pixels), then the individual expressions given above can be used.

*The smallest 6-point proportional character is 7 pixels wide.

Table 4-2 Lines Per Page (Portrait)

Character Cell Dimensions				
	Height			
Width	Dots	Bytes	Full Lines Per Page	Formulas
7	25-32	4	53	
	33-40	5	52	1600
	41-48	6	50	------
	49-56	7	48	26 + h
	57-64	8	47	
8-10	25-32	4	63	
	33-40	5	60	1600
	41-48	6	58	-------
	49-56	7	56	21.5 + h
	57-64	8	54	
11-15	25-32	4	75	
	33-40	5	71	1600
	41-48	6	68	--------
	49-56	7	65	17.5 + h
	57-64	8	63	
16-30	25-32	4	94	
	33-40	5	89	1600
	41-48	6	84	------
	49-56	7	80	13 + h
	57-64	8	76	
31	25-32	4	103	
	33-40	5	97	1600
	41-48	6	91	--------
	49-56	7	86	11.5 + h
	57-64	8	82	
32 or greater	25-32	4	123	
	33-40	5	114	1600
	41-48	6	107	-----
	49-56	7	100	9 + h
	57-64	8	94	

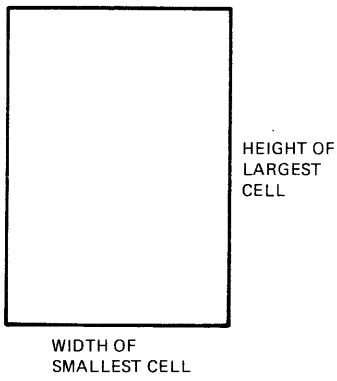


Figure 4-7 Hypothetical Character Cell (Portrait)

Character Size Range
Largest Character Size -- The largest character cell (a character cell includes leading between lines and spacing between characters) is shown in Figure 4-8 for landscape and portrait orientations.

Such "character" cells would not normally be used for type-faces due to the abnormal size and aspect ratios. These cells may be used for the storage of digitized logos and signatures, or portions thereof, however.

Smallest Character Size -- The smallest character cell (a character cell includes leading between lines and spacing between characters) is a nominal 6 points i.e., 25 pixels high by 15 pixels wide as shown in the following example.

Example: 6 Point Helvetica Font

Height = 25 pixels
= 6 points
= .Ø83 inches

Width = 15 pixels
= .Ø5 inches

Aspect Ratio = .6

Lines Per Inch = 12

Char. Per Inch = 2Ø (fixed pitch character)

This is the smallest fixed-pitch character. The smallest proportional-spaced character is 7 pixels (Ø.Ø21 inches) wide.

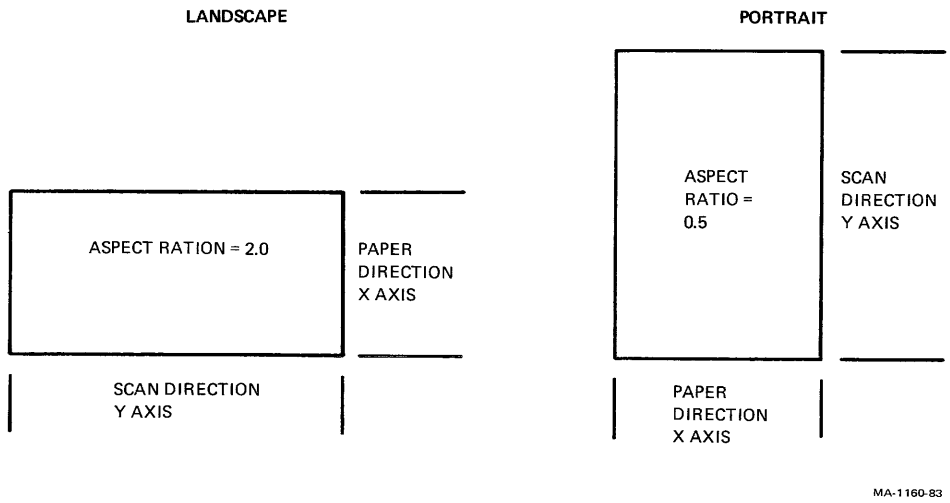


Figure 4-8 Character Cell Example

Band Aligned Printing

The following limits and conditions apply when characters are aligned with the band buffers (using fonts designed for this method). This method may be used to completely image a large area using 16 X 16 pixel cells.

- Character cells must be 16 by 16 pixels in size
- First scan of the cells (top of cell in landscape, left of cell in portrait) must start on either the first or 17th scan of a band buffer.
- A maximum of 155 characters per line in landscape, no maximum number of characters per line in portrait.
- A maximum of 138 lines per page in portrait, no limit for landscape.
- A maximum of 8 font switches per line.

CHAPTER 5
COMMUNICATION

GENERAL

This chapter describes the LNØ1 communication including the interface connector, the interface signals and the internal controls.

INTERFACE CONNECTOR

The LNØ1 parallel interface is terminated with a standard 5Ø pin subminiature "D" type connector.

INTERFACE SIGNALS

This section describes the LNØ1 interface signals. The pin assignments for the parallel interface are listed in Table 5-1. The following paragraphs describe each signal.

On Line

Signal Source - LNØ1

The On Line signal is sent to the user system to indicate the printer is on line. When On Line is turned on (logical 1), the printer is ready to print data. The following conditions exist:

- The Ready signal is turned on
- The printer is on line
- No error conditions are present, e.g. paper out

NOTE

Standard TTL logic levels are used with +5 volts for logical 1 (on) and Ø volts for logical Ø (off).

Demand

Signal Source - LNØ1

The demand line is used to synchronize data between the printer and the user system. When Demand is turned on, the LNØ1 is capable of accepting a character. It remains on until a Data Strobe is received, then is turned off while the character is being stored in the data memory. The Demand signal can only be turned on if the Data Strobe is off. The printer will accept any data sent to it in response to Demand, independent of the On Line state.

Table 5-1 Parallel Interface Pin Assignments

Signal Title	Assigned Pin	Signal Direction To LNØ1	Signal Direction From LNØ1	Bus Nomenclature
DATA BIT 1 Sig	19	X		
DATA BIT 1 Rtn	3	X		
DATA BIT 2 Sig	2Ø	X		
DATA BIT 2 Rtn	4	X		
DATA BIT 3 Sig	1	X		
DATA BIT 3 Rtn	2	X		
DATA BIT 4 Sig	41	X		
DATA BIT 4 Rtn	4Ø	X		
DATA BIT 5 Sig	34	X		
DATA BIT 5 Rtn	18	X		
DATA BIT 6 Sig	43	X		
DATA BIT 6 Rtn	42	X		
DATA BIT 7 Sig	36	X		
DATA BIT 7 Rtn	35	X		
*DATA BIT 8 Sig	28	X		
*DATA BIT 8 Rtn	44	X		
*DATA BIT 8 Sig	3Ø	X		
*DATA BIT 8 Rtn	14	X		
DATA STROBE Sig	38	X		
DATA STROBE Rtn	37	X		
ON LINE Sig	21		X	
ON LINE Rtn	5		X	
DATA DEMAND Sig	23		X	READY
DATA DEMAND Rtn	7		X	
READY Sig	22		X	
READY Rtn	6		X	
INTERFACE CONNECT	45			
VERIFY	46			

* Switch 8 allows either Paper Instruction (Pin 3Ø) or Data Bit 8 (Pin 28) as the functional high order data bit. Refer to Internal Controls for more information.

Data Strobe
Signal Source - User Equipment

The Data Strobe signal line is controlled by the user to define when the information on the Data Lines is valid. Each time a Data Strobe occurs, the printer samples the Data Lines and turns the Demand Line off, while the character is stored.

Ready
Signal Source - LNØ1

When Ready is turned on, power to the printer is on and the warm up cycle is complete.

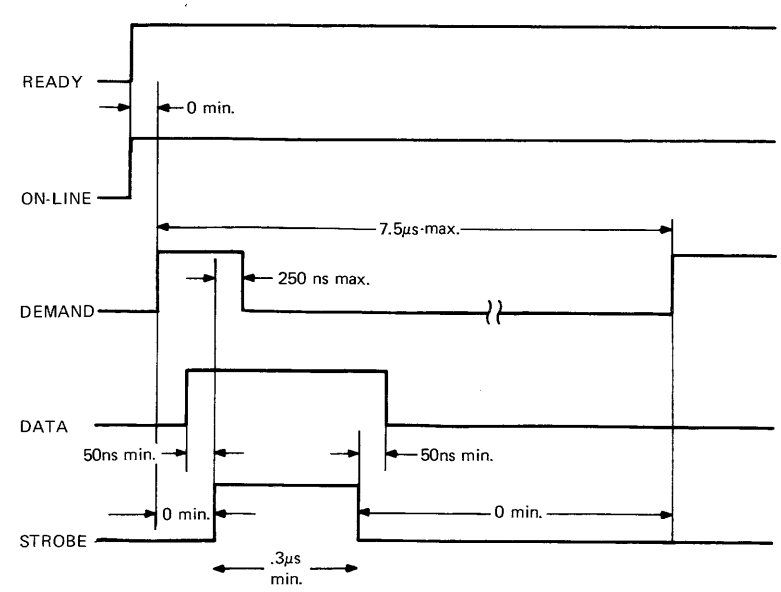
Data
Signal Source - User Equipment

7 or 8 Data Lines - parallel data format

Interface Connect Verify

Ensures that the interface cable is connected to the host computer.

INTERNAL CONTROLS
The LNØ1 has 8 2-position internal switches that are used to specify LNØ1 application options. These controls should be changed by your service representative only. The functions of the internal switches are listed in Table 5-2.



NOTE:
TO ACHIEVE THE 7 1/2 MICRO SECOND TRANSFER RATE, THE STROBE MUST BE RAISED WITHIN 1 MICRO SEC. FROM DEMAND RISING, AND DROP WITHIN 5 MICRO SEC.

MA-1148-83

Figure 5-1 Interface Timing Diagram

Table 5-2 Internal Controls

Switch #	Position	Function
1	ON	Not Used
	OFF	Not Used
2	ON	Landscape Orientation = default
	OFF	Portrait Orientation = default
3	ON	Not Used
	OFF	Not Used
4	ON	7 Bit ASCII
	OFF	8 bit ASCII
5	ON	Enable Printer
	OFF	Enable Data Monitor
6	ON	
	OFF	Enable Data Products Protocol
7	ON	
	OFF	ASCII Encoding
8	ON	Data bit 8 = pin 30
	OFF	Data bit 8 = pin 28

APPENDIX A
CODE EXTENSION TECHNIQUE
(ESCAPE SEQUENCES, CONTROL SEQUENCES, AND CONTROL STRINGS)

ESCAPE AND CONTROL SEQUENCES

Escape and control sequences are used to provide additional controls not provided by the control characters in the character set. These sequences are multiple character control functions used to control character printing and processing. They are not printed. Escape and control sequences are defined in ANSI standards X3.41 -- 1977 and X3.64 -- 1979.

The characters used in escape and control sequences throughout this manual are shown (but not defined) using the ASCII graphic character set and hexadecimal coding. The case of the characters used in a sequence is significant and must be sent to the printer exactly as shown. These characters are spaced apart for clarity only.

NOTE

There is a cross reference to binary, decimal and octal in the character chart.

Escape Sequence Format

The following paragraphs define the format of an escape sequence.

```
ESC            I......I       F
1B             20 -- 2F       30 -- 7E
|                 |              |
Escape         Intermediate     Final
Sequence       Characters       Character
Introducer     (any number      (one character)
               of characters
               -- zero or more)
```

The escape sequence introducer is the ESC control character (hex 1B). When the ESC character is received, the next characters received are not printed, but stored, to be used as part of the sequence.

If the characters received after the ESC character are in the 20 -- 2F hex range they are intermediate characters. These characters are stored as part of the sequence.

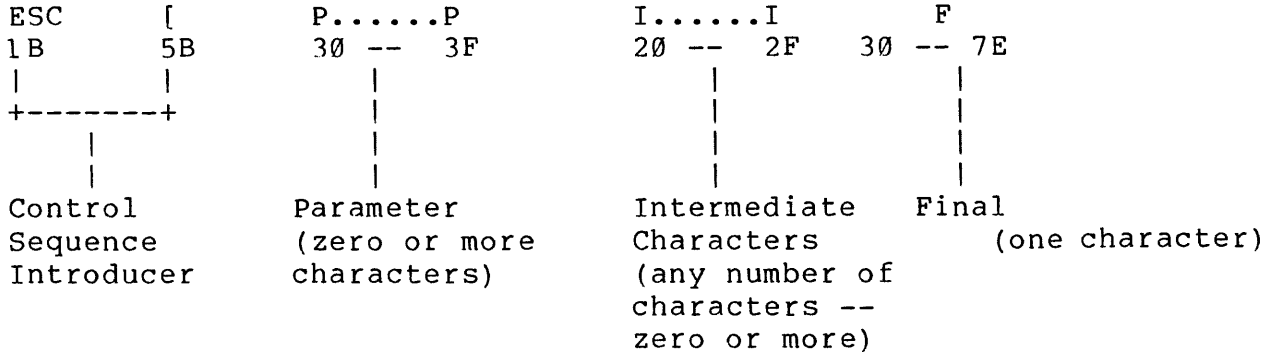
If the character received after the ESC character is in the 30 -- 7E hex range, it is a final character. Final characters in the 40 -- 7E hex range are reserved for standard ANSI use. Final characters in the 30 -- 3F hex range are reserved for private use.

The final character indicates the end of the escape sequence. The intermediate and final characters together define the function of the sequence. The printer performs the action specified by the sequence, then continues to print data.

Escape Sequence Example
The following sequence is used to reset the printer to the default state.

Escape Sequence Introducer	Final character
ESC	c
1B	63

Control Sequence Format
The following paragraphs define the format of a control sequence.



The control sequence introducer (CSI) consists of the ESC (hex 1B) and [(hex 5B) characters. It is used to gain the extended functionality of the 8-bit environment while using 7-bit characters. After the CSI character(s) are received, characters received are not printed, but stored for use as part of the sequence.

If the characters received after the CSI character(s) are in the 30 -- 3F hex range, the characters are parameter characters. A parameter character modifies the action or interpretation of the sequence. Characters in the 30 -- 39 hex range are used to define numbers. The delimiter character ";" (hex 3B) is used to separate a string of parameters. The ? character (hex 3F) is used at the beginning of a string of parameters to indicate a DIGITAL private sequence.

The limit for parameters is 65,536 decimal. You can not use more than 8 parameters in a string. If no decimal value is specified for a parameter character, a zero value is assumed for the parameter.

The printer processes 2 types of parameters; numeric and selective. A numeric parameter is used to indicate a numeric value such as a tab or margin location. Numeric parameters are processed as unsigned decimal integers with the most significant digit transmitted first. Leading zeros are allowed but not necessary.

In this manual, numeric parameters are shown as actual values or are designated as Pn, Pnl, Pn2, etc. In the hex representation of the sequences, parameter characters are shown as three asterisks (***).

Numeric Parameter Example

```
Control Sequence Introducer
   |                              Numeric Parameters
   |                                                    Final Character
+-----+          +-------------------------------+      |
|     |          |                               |      |
ESC   [          3   6   Ø   Ø   ;   5   7   6   Ø      s
1B    5B         33  36  30  30  3B  35  37  36  30     73
```

In this example the left margin is set to 5 inches (decipoint is assumed) and the right margin is set to 8 inches. The numeric parameters are 3600 (5 inches) and 5760 (8 inches). The ";" delimiter character separates the 2 parameters.

A selective parameter is used to represent a function. In this manual, selective parameters are designated as Ps, Psl, Ps2, etc. In the hex representation of the sequences, parameter characters are shown as three asterisks (***).

Selective Parameter Example

```
Control Sequence Introducer
|       Selective Parameter
+-----+ |   Intermediate Character
|     | |   |   Final Character
|     | |   |   |
ESC   [ 2   SP  I
1B    5B 32 20  49
```

In this example, 2 (hex 32) is the selective parameter. It causes the printer to select decipoint as the size unit. If the parameter were 7 (hex 37) the printer selects pixels as the size unit.

In control sequences with more than one selective parameter, the parameters are processed sequentially beginning with the first. A control sequence containing more than one selective parameter has the same effect as a corresponding number of separate control sequences, each with a single parameter.

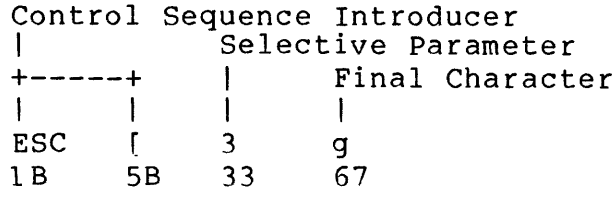
If the characters received after the ESC character are in the 20 -- 2F hex range they are intermediate characters. These characters are stored as part of the sequence.

If the character received after the CSI character(s) is in the 40 -- 7E hex range, it is a final character. Final characters in the 40 -- 6F hex range are reserved for standard ANSI use. Final characters in the 70 -- 7E hex range are reserved for private use.

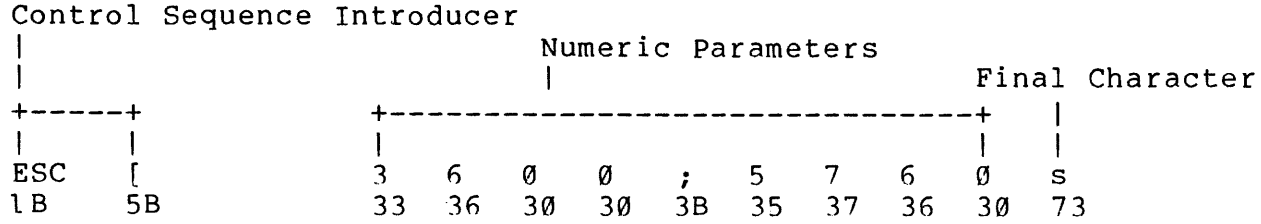
The final character indicates the end of a control sequence. The parameter, intermediate and final characters together define the function of the sequence. The printer performs the action specified by the sequence, then continues to print data.

Control Sequence Examples

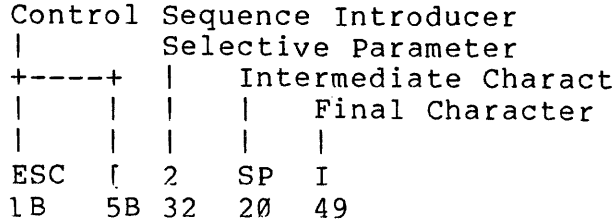
Example 1
The following sequence is used to clear all horizontal tab stops.



Example 2
The following sequence is used to set the left margin to 5 inches and the right margin to 8 inches (decipoint mode is assumed).



Example 3
The following sequence is used to select decipoints as the size unit for numeric parameters.



CONTROL STRINGS

The general format of a control string is:

<String Introducer><Data><String Terminator>

The string introducers are:

DCS (Device Control String)	ESC	P
	1B	5Ø
OSC (Operating System Command)	ESC	]
	1B	5D
PM (Privacy Message)	ESC	^
	1B	5E
APC (Application Program Command)	ESC	
	1B	5F

The string terminator is:

ST (String Terminator)	ESC	\
	1B	5C

The data is defined by private agreement between the sending and receiving device and should not contain any CØ or C1 codes.

The LNØ1 does not implement any APC, OSC or PM strings. When received, these strings are processed as printable characters. The LNØ1 responds to 2 device control strings (refer to the following section on device control strings for more detail).

Device Control String Format

The following paragraphs define the format of a DIGITAL device control string.

<DCS Introducer><Protocol Selector><Command String><String Terminator>

DCS Introducer

The device control string introducer (DCS) is:

ESC	P
1B	5Ø

Protocol Selector
The format of the protocol selector is:

P.....P	I.....I	F
3Ø -- 3F	2Ø -- 2F	4Ø -- 7E

Parameter (zero or more characters)	Intermediate (zero or more characters)	Final (one character)

Refer to the control sequence format for the definition of a parameter, intermediate and final character.

Command String
The format of the command string is determined by the combination of parameters, intermediate characters and final character which precede it.

String Terminator
The Device Control String Terminator is:

ESC	\
1B	5C

ERROR RECOVERY
The LNØ1 usually recovers from control function errors by performing as much of the function as possible. Errors include invalid control functions, control characters embedded in escape or control sequences or parameters out of range. The following paragraphs describe the specific LNØ1 error handling techniques.

Unsupported escape and control sequences (valid escape and control sequences not listed in this document) are ignored.

If the ESC character (hex 1B) is received during an escape or control sequence, the sequence is cancelled and a new sequence is processed beginning with the ESC.

Cancel (hex 18) is used to indicate that the data with which it is sent is in error or is to be disregarded. Therefore, the receipt of CAN causes immediate termination, without execution, of any sequence in progress. The CAN character itself receives no further processing. The characters following the CAN are not processed as part of the escape or control sequence, but rather are processed as printable characters.

Substitute (hex 1A) is used to indicate replacement of a character which could not be represented. The receipt of SUB causes immediate termination, without execution, of any sequence in progress. The SUB character itself, and all subsequent characters, are not interpreted as part of the escape or control sequence, but rather are interpreted normally.

C0 control codes (hex 00 to 1F inclusive) received between the ESC (hex 1B) code and the final character are ignored.

C1 control codes (hex 80 to 9F inclusive) received between the ESC (hex 1B) code and the final character are ignored.

Eight bit graphic codes (hex A0 to FE inclusive) received between the ESC (hex 1B) code and the final character cause the sequence to be ignored.

While processing control sequences, codes from hex 30 to 3F inclusive received after the intermediate character(s) cause the sequence to be ignored.

While processing control sequences, codes from hex A0 to FE received after the intermediate character(s) cause the sequence to be ignored.

Additional numeric parameters (after 8) are discarded.

If characters other than "0" (hex 030) -- "9" (hex 39) and ";" (hex 3B) are processed by the printer while storing the numeric parameter string, the sequence is ignored. An error code 2 is then printed on the next summary sheet.

If more than one intermediate character is processed, the sequence is ignored.

The numeric parameter string is stored in a 144 character buffer. If the buffer overflows, characters are discarded until an intermediate character is processed.

Upon receipt of an intermediate character, printer stops storing the numeric parameter string and converts parameters to binary values.

LN01 PROGRAMMER REFERENCE MANUAL

Your comments and suggestions will help us in our continuous effort to improve the quality and usefulness of our publications.

What is your general reaction to this manual?

	Excellent	Very Good	Good	Fair	Poor
Accuracy	☐	☐	☐	☐	☐
Completeness	☐	☐	☐	☐	☐
Organization	☐	☐	☐	☐	☐
Format	☐	☐	☐	☐	☐

What features are most useful?
(Notes, Tables, Illustrations, etc.) _____

_____

_____

_____

_____

Does the publication satisfy your needs? ☐ Yes ☐ No

What errors have you found?
(Ref. page no., table no., figure no.) _____

_____

_____

_____

_____

Additional Comments _____

_____

_____

Name _____ Street _____

Title _____ City _____

Company _____ State/Country _____

Department _____ Zip _____

The Documentation Products Directory and the DECdirect Guide contain information on the remainder of DIGITAL's technical documentation.

Copies of the above Directory and Guide, as well as additional copies of this document, are available by writing or calling:

Digital Equipment Corporation
Accessories and Supplies Group
P.O. Box CS2008
Nashua, New Hampshire 03061

Attention: Documentation Products
Telephone: 1-800-258-1710

Order No. EK-LN01S-RM-002

MY

Fold Here

DO NOT TEAR – FOLD HERE AND TAPE

digital

No Postage
Necessary
if Mailed in the
United States

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 33 MAYNARD, MA.

POSTAGE WILL BE PAID BY ADDRESSEE

Digital Equipment Corporation
Educational Services/Quality Assurance
12 Crosby Drive, BUO/E08
Bedford, MA 01730

Digital Equipment Corporation • Maynard, MA 01754