7265

Programmer's Manual

Matrix printer

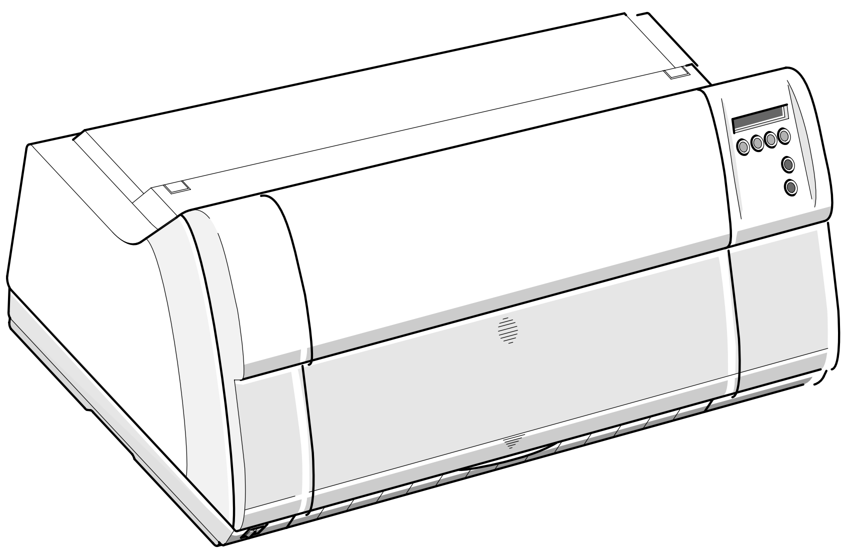


Table of Contents

Table of Contents

Table of Figures

Chapter 1. Introduction

The **TallyGenicom** 7265 provides rugged, versatile impact printers useful in a variety of applications. Flexibility of use is guaranteed by the wide range of software emulations and programming options. The current printer characteristics are the result of years of application experience. It is a hallmark of **TallyGenicom** products that they may be used with a broad range of systems, hardware, emulations, and protocols.

The information in this manual is believed to be accurate and correct. However, **TallyGenicom** makes no claim as to its absolute accuracy and reserves the right to make improvements to this document from time to time.

r 2. ANSI

The TallyGenicom im ple m entation of ANSI X3.64 is the native control protocol for the 7265 printer, as it is for a variety of other TallyGenicom impact matrix printer families. The control functions presented here are a superset of ANSI control sets familiar to our customers. For the most part, ANSI drivers written for other TallyGenicom printers should only need to be modified as features introduced on the 7265 Series printers are exercised. Examples of such new features include user-defined formats, alternate paper paths, user-defined character substitutions, and remote selection of emulations.

Enabling the application to select among emulations provides access to some of the 7265 features not supported in emulated products. Toward this end, the printer can be commanded to maintain the current status of fonts, pitches, tabs, margins and so forth across emulation changes. Although some minute differences due to translations from, say, columns to decipoints may be discernible when switching from one emulation to another, the printer is quite predictable when switching from another emulation to ANSI and then back again.

The 7265 is a decipoint machine. Locations and dimensions such as tabs, margins, absolute and relative positions, and form lengths are described in ANSI control sequences in units of 1/720 inch. If you have the choice of describing your page in one or another of the supported emulations, consider that ANSI gives you more flexible control of locations of objects on the page.

In this chapter we include spaces between characters in escape sequences for clarity. ESC H, for instance, is easier to read than is ESCH. Where the space character forms a component of a control sequence, then it is designated by **<SP>** .

Listed below are the control codes interpreted by this printer, along with page numbers where you can find detailed descriptions.

Control Code Summary

Code	Hex	Description
ETX	03	ETX/ACK Communication Protocol
ACK	06	ETX/ACK Communication Protocol
BEL	07	Sound Beeper
BS	08	Backspace
HT	09	Horizontal Tab
LF	0A	Line Feed
VT	0B	Vertical Tab
FF	0C	Form Feed
CR	0D	Carriage Return
SO	0E	Barcode / Oversized On
SI	0F	Barcode / Oversized Off
DC1(XON)	11	Printer Ready
DC3(XOFF)	13	Printer Busy
ESC	1B	Escape
DEL	7F	Delete
If 8-bit control code interpretation is enabled ...		
IND	84	Index
NEL	85	Next Line
HTS	88	Set Horizontal Tab
VTS	8A	Set Vertical Tab
PLD	8B	Partial Line Down
PLU	8C	Partial Line Up
RI	8D	Reverse Index
DCS	90	Device Control String
PU1	91	Private Use 1
CSI	9B	Control String Introducer
ST	9C	String Terminator
OSC	9D	Straps and Options Introducer

Some of the control codes are interpreted when encountered within a graphics data sequence. Interpretations relative to graphics are shown on page 31.

ompar ison, 7 Bi t/ 8 Bi t

8 B i t	Hex	Description	7 Bit	Hex
IND	84	Index	ESC D	1B 44
NEL	85	Next Line	ESC E	1B 45
HTS	88	Set Horizontal Tab	ESC H	1B 48
VTS	8A	Set Vertical Tab	ESC J	1B 4A
PLD	8B	Partial Line Down	ESC K	1B 4B
PLU	8C	Partial Line Up	ESC L	1B 4C
RI	8D	Reverse Index	ESC M	1B 4D
DCS	90	Device Control String	ESC P	1B 50
PU1	91	Private Use 1	ESC Q	1B 51
CSI	9B	Control String Introducer	ESC [	1B 5B
ST	9C	String Terminator	ESC \	1B 5C
OSC	9D	Straps and Options Introducer	ESC]	1B 5D

8-bit Control Codes

ANSI assigns control functions to characters 80 hex through 9F hex and calls these characters C1 codes. You can set the printer to either interpret 80 hex - 9F hex as control codes or not. The ISU default status of this option is disabled.

You can toggle the status of the 8-bit control code enable from the Emulations Options Menu or with escape sequences. The pertinent escape sequences are the ANSI private use sequences described on page 12. In 8-bit mode, all of the lower contro l codes and ESC sequences remain active. For example, you can use either PLD or ESC K for subscript printing. In 7-bit mode, only ESC K is available.

If 8-bit control code interpretation is disabled, then the printer processes characters 80 hex through 9F hex according to IBM PC character sets 1 or 2, depending on which is currently selected. The default is character set 1. You can toggle the character set selection from the Emulation Options Menu.

If 8-bit control code interpretation is disabled and character set 1 is active, then the printer strips the most significant bit of characters 80 hex through 9F hex. For example, if you send 8A hex when 8-bit control code interpretation is disabled and character set 1 is active, then the printer treats 8A hex as 0A hex and performs a line feed.

xade c imal, Orde red Column|Row

0	1	2	3	4	5	6		7
0				0	@	P	`	p
1		**DC1**	!	1	A	Q	a	q
2			“	2	B	R	b	r
3	**ETX**	**DC3**	#	3	C	S	c	s
4			$	4	D	T	d	t
5	**ENQ**		%	5	E	U	e	u
6	**ACK**		&	6	F	V	f	v
7	**BEL**		‘	7	G	W	g	w
8	**BS**		(	8	H	X	h	x
9	**HT**		)	9	I	Y	I	y
A	**LF**		*	:	J	Z	j	z
B	**VT**	**ESC**	+	;	K	[	k	{
C	**FF**		,	<	L	\	l	**|**
D	**CR**		-	=	M	]	m	}
E	**SO**		.	>	N	^	n	~
F	**SI**		/	?	O	_	o	**DEL**

Figure 2-1 *Default Character Set for ANSI Emulation, 00 to 7F*

Code is Hexadecimal, Ordered Column|Row

	8	9	A	B	C	D	E	F
0		**DCS**	á	!	"	#		$
1		**PU1**	í	%	&	'		±
2		ó		(	)	*		+
3		ú		,	-	#		.
4	**IND**		ñ	/	0	1		2
5	**NEL**		Ñ	3	4	5		6
6		a	_	7	8	9		÷
7		o	_	:	;	<		=
8	**HTS**		¿	>	?	@		°
9			A	B	C	D		•
A	**VTS**		¬	E	F	G		H
B	**PLD**	**CSI**	½	I	J	K		L
C	**PLU**	**ST**	¼	M	N	O	æ	P
D	**RI**	**OSC**	¡	Q	R	S	Ø	²
E			«	T	U	V	W	!
F			»	X	Y	Z	[	

Figure 2-2 *Default Character Set for ANSI Emulation, 80 to FF*

Escape Sequence Summary

In this chart, the 7-bit representation is given. Use the Comparison Chart on page 4 to convert 7 bit to 8 bit sequences.

Code	Description
ESC H	Set Horizontal Tab
ESC J	Set Vertical Tab
ESC K	Subscript Printing
ESC L	Superscript Printing
ESC P	Enter Dot Graphics
ESC c	Restore to Initial State
ESC \	Exit Dot Graphics Modes
ESC [(p1);(p2)<SP>B	Graphic Size Modification
ESC [(p1);(p2)<SP>G	Line/Character Spacing

Code	Descripton
ESC [(p1);(p2)<SP>~	Select Emulation
ESC [(p)`	Horizontal Position Absolute
ESC [(p)a	Horizontal Position Relative
ESC [(p)d	Vertical Position Absolute
ESC [(p)e	Vertical Position Relative
ESC [(p1);(p2)f	Vertical and Horizontal Position Absolute
ESC [(Ps)g	Clear Tabs
ESC [(p1);...(pn)h	Set Auto CR on LF
ESC [>(p1);...(pn)h	Set Mode (GENICOM)
ESC [(p)j	Horizontal Position Backwards
ESC [(p)k	Vertical Position Backwards
ESC [(p1);...(pn)l	Reset Auto CR on LF
ESC [>(p1);...(pn)l	Reset Mode (GENICOM)
ESC [(P1);...(Pn)m	Fonts and Print Modes (SGR)
ESC [(p1)p	Select Paper Path
ESC [(p1)q	Select Graphics Density
ESC [(p1);(p2);(p3)r	Forms Setup
ESC [(p1);(p2)s	Sets Left and Right Margins
ESC [(p1)t	ENA/DIS Oversized/Expanded/Bar Code
ESC [(p1);(p2);...(p22)u	Set Horizontal Tabs at Certain Positions
ESC [(p1);(p2);...(p12)v	Set Vertical Tabs at Certain Positions
ESC [(p1);(p2);…}	Set Bar Code Parameters
ESC [(p)x	National Character Sets

Pr   inter Handshaking

Commands	Description
DC1	PRINTER READY In serial interface, if the printer is strapped for XON/XOFF handshaking, then the printer sends DC1 to the host to signal that the printer is ready to accept data. In parallel interface, if the printer is strapped for select/ deselect by received DC1/DC3 codes, then the host sends DC1 to the printer to set the printer online. **Dec** 17 **Hex** 11
DC3	PRINTER NOT READY In serial interface, if the printer is strapped for XON/XOFF handshaking, then the printer sends DC3 to the host to signal that the printer cannot, for the moment, accept any more data. In parallel interface, if the printer is strapped for select/ deselect by received DC1/DC3 codes, then the host sends DC3 to the printer to set the printer to Standby. In Standby, the printer will accept and acknowledge data from the interface, but will discard all data received. **Dec** 19 **Hex** 13
ETX	ETX/ACK COMMUNICATION PROTOCOL In serial interface, if the printer is strapped for ETX/ACK handshaking, then the host sends the ETX control code at the end of a block of data. **Dec** 3 **Hex** 03
ACK	ETX/ACK COMMUNICATION PROTOCOL In serial interface, if the printer is strapped for ETX/ACK handshaking, then the printer sends an ACK in response to and ETX from the host to indicate that it is ready to receive more data. **Dec** 6 **Hex** 06

Printer Control

Commands	Description
SO	SHIFT OUT You can strap the printer to start either barcode (page 37) or oversized (page 64) mode, whichever is enabled, on receipt of SO. The ISU status of SO/SI control is disabled. You can enable SO/SI control from Software Options on the printed menu, or from the host via the Straps and Options control sequence. See page 14. **Dec** 14 **Hex** 0E
SI	SHIFT IN If the printer is strapped to turn on barcode or oversized on receipt of SO, then SI turns barcode (page 37) or oversized (page 64) mode off. **Dec** 15 **Hex** 0F
ESC	ESCAPE Introduces an escape sequence. The printer evaluates characters following the ESC character to determine if the sequence is valid. If the sequence is valid, then the printer responds to the command. If the sequence is not valid, then the printer aborts the process, discarding at least the first character following ESC. **Dec** 27 **Hex** 1B
ESC P (DCS)	DEVICE CONTROL STRING The control function introducer for the device control string structure, which is used to frame graphics. **Dec** 144 **Hex** 90
ESC [(CSI)	The CSI character is the control function introducer for the ANSI control sequence. Control sequences are multi-character control functions that accept parameters. **Dec** 155 **Hex** 9B
ESC \ (ST)	STRING TERMINATOR Terminates the DCS and the OSC sequence. **Dec** 156 **Hex** 9C
ESC] (OSC)	The OSC character is the introducer for straps and options. **Dec** 157 **Hex** 9D
BEL	BEEPER **Dec** 7 **Hex** 07

Commands	Descripton
DEL	DELETE DEL is a printable character, or is a valid graphics data byte if received in dot graphics mode. **Dec** 127 **Hex** 7F
ESC [(p1);(p2)<SP>~	SELECT EMULATION If p2 is 0, which is the default, then the current settings in the printer remain in effect through the emulation change to the extent that selected features are supported in the target emulation. This affects such things as, font selection, character pitch, margins, tabs, and so forth. If p2 is 1, then the status of such parameters reverts to defaults dependent on the selected emulation. **p1** **selects** 0 GENICOM ANSI 1 Tally ANSI 2-21 Reserved 22 Epson FX286e **p2** **selects** 0-1 full reset Example: Select Epson emulation with full reset: **Dec** 27 91 50 50 59 49 32 126 **Hex** 1B 5B 32 32 3B 31 20 7E
ESC [(p1)p	SELECT PAPER PATH **p1** **selects** 08 unload current path 10 unload current path and load tractor 2 12 unload current path and load tractor 1 Forms in the current path should be torn off before changing paper paths. If you send a command to change paper paths and there is more paper downstream of the print head than 1/2 times the current form length or six inches, whichever is greater, then the printer returns paper to the position that was current before the command was received, and a fault condition is set. If you send a command to change paper paths and the target path is the current path, then the printer executes a form feed. If you load tractor 2 when tractor 2 is not installed, the command will be ignored. Example: Unload tractor 1 and load tractor 2: **Dec** 27 91 49 48 112 **Hex** 1B 5B 31 30 70

Commands	Descripton
ESC c	RESET TO INITIAL STATE Recalls the user format presently assigned to the current paper path. **Dec** 27 99 **Hex** 1B 63
ESC [(p1)t	ENABLE/DISABLE OVERSIZED/EXPANDED/BAR CODE MODES **p1** **Effect** **Page** 0 reset special modes 1 select oversized printing 4-2 2 select expanded printing 4-2 3 select bar code 3-6 Oversized printing and bar codes are explained in separate chapters. When expanded printing is selected, the current font is expanded horizontally and vertically in 100% increments according to expansion factors specified by ESC [(p1);(p2)<SP>B (see page 11). The range of expansion is 1x through 8x. Example: select expanded printing **Dec** 27 91 50 116 **Hex** 1B 5B 32 74
ESC [(p1); (p2)<SP>B	GRAPHIC SIZE MODIFICATION Sets the size of expanded and oversized characters. The effect of this command on oversized is explained in a separate chapter. When expanded print is selected, the argument units for parameters p1 and p2 are percentages of the vertical and horizontal dimensions of the current font. Parameter p1 controls the vertical expansion and p2 controls the horizontal expansion. The argument limits for expanded print are 800;800 (8x the parent font size). Oversized enables much larger expansions. Parameters are rounded down to the nearest 100. **Note:** This printer goes into expanded mode immediately on receiving this command. Any text between this sequence and a countermanding ESC [(p1)t sequence (see page 11) is printed according to the specified expansion. If height and width are different values, the height will be set the same as the width.
ESC [(p1); ...(pn)h	SET MODE (ANSI) **p** **Effect** 20 auto CR on LF **Dec** 27 91 50 48 104 **Hex** 1B 5B 32 30 68

Commands	Descripton
ESC [(p1); ...(pn)l	RESET MODE (ANSI) Resets mode(s) set by ESC [(p1);..(pn)h. **p** **Effect** 20 reset auto CR on LF Example: reset auto CR on LF **Dec** 27 91 50 48 108 **Hex** 1B 5B 32 30 6C
ESC [> (p1);..(pn)h	SET MODE (GENICOM) This is an ANSI private use sequence. **p** **Effect** 1 proportional print 2 reserved 3 80 hex - 9B hex interpreted as C1 control codes 4 bold mode accomplished by overstrike 5 select character set 2 Example: set proportional print **Dec** 27 91 62 49 104 **Hex** 1B 5B 3E 31 68
ESC [> (p1);..(pn)l	RESET MODE (GENICOM) Resets modes set by ESC [>(pn)h. See the previous command. The parameter definitions are the same. Example: reset proportional print **Dec** 27 91 62 49 108 **Hex** 1B 5B 3E 31 6C

Commands	Description
ESC [(p1); (p2);<SP>{	PAPER SHEAR If the paper shear option is installed, then **note** that the appropriate strap must be set in the hardware options menu in order for the paper shear to work. Parameter p1 = 1 selects the paper shear. Parameter p2 = 1 executes a cutting sequence: If the vertical position is not at top-of-form, then a form feed is executed. Paper is advanced to the shear position and cut. Next, it is advanced an inch to eject the sheet that has been cut off; then it is retracted to the top margin on the following form. The horizontal position following a shear is the left margin. The shear position should be set from the control panel so that paper is cut precisely at the perforation. If a non-zero top print reference is needed, then set the top print reference first and then set the shear position. **Dec** 27 91 49 59 49 32 123 **Hex** 1B 5B 31 3B 31 20 7B

Commands	Descripton
ESC]6;4; (p3);(p4) ESC \	STRAPS AND OPTIONS This command enables downline control of options found at various places in the printer menus. This includes the Emulation and Software Options menus, as well as the User-defined Options menu. For details and descriptions, see the User's Manual. p3 = 0 reset 1 set p4 = 4 auto cr on lf 5 auto lf on cr 6 auto cr on vt 14 auto wrap

Graphics Rendition (Fonts and Modes)

Command	Description	
ESC [(p1); ...(pn)m	SELECT GRAPHICS RENDITION	
	Selects print modes and fonts.	
	p1	Printing Mode
	0	normal printing - resets all modes (but does not affect font selection)
	1	set bold print
	4	set underline
	5	set doublewide
	6	set proportional
		*
	10 and 11	Gothic DP
	12	Gothic LQ
	13	Courier DP
	14	Courier LQ
	15	Gothic Italic DP
	16	Gothic Italic LQ
	17	Courier Italic DP
	18	Courier Italic LQ
	19	Wide Gothic DP
		*
	22	cancel bold
	24	cancel underline
	25	cancel doublewide / expanded
	26	cancel proportional
	30	select black ribbon color
	31	select red ribbon color
	32	select green ribbon color
	33	select yellow ribbon color
	34	select blue ribbon color
	35	select magenta ribbon color
	36	select cyan ribbon color

NATIONAL CHARACTER SET

Command	Description
ESC [(p)x	SELECT NATIONAL CHARACTER SET **p** **Character Set** 0 USA 1 Germany 2 French A 6 Italian 7 United Kingdom 8 Spanish 9 Danish/Norwegian A 16 Swedish/Finnish D If a parameter value (p) is not recognized, then the default character set (0 - USA) is selected. To use substitutions that have characters in the 80 hex to 95 hex range, first disable 8-bit control code processing, either by escape sequence or from the Emulation Options Menu. Next, enable character set 2, either by the Straps and Options control sequence on page 14 or from the Emulation Options menu. This causes ASCII codes 80 hex - 9F hex to be mapped to printable characters instead of control codes. Example: To select the United Kingdom character set: **Dec** 27 91 55 120 **Hex** 1B 5B 37 78

D/N = Danish/Norwegian S/F = Swedish/Finnish

	Code D Locatbn H	35 23	36 24	38 26	42 2A	64 40	91 5B	92 5C	93 5D	94 5E	96 60	123 7B	124 7C	125 7D	126 7E	155 9B	157 9C
p	Character Set																
0	USA	#	$	&	*	@	[	\	]	^	`	{	!	}	~	¢	¥
1	Germany	#	$	&	*	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß	¢	¥
2	French A	£	$	&	*	à	°	ç	§	^	`	é	ù	è	¨	¢	¥
6	Italian	£	$	&	*	§	°	ç	é	^	ù	à	ò	è	ì	¢	¥
7	UK	£	$	&	*	@	[	\	]	^	`	{	!	}	~	¢	¥
8	Spanish	^	$	&	*	@	í	Ñ	¿	^	`	¨	ñ	}	~	¢	¥
9	D/N A	#	$	&	*	@	Æ	Ø	Å	^	`	æ	ø	å	~	¢	¥
16	Swedish	¤	&	*	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü	¢	¥	

Figure 2-3 *National Character Substitutions 0 - 18*

Line/Character Spacing

Command	**Description**
ESC [(p1); (p2) <SP> G	LINE/CHARACTER SPACING Sets the vertical and horizontal pitch in decipoints. Parameter (p1) is the spacing between lines and (p2) is the spacing between characters. Unspecified parameters remain at their current values. This command does not affect oversized printing selected by ESC [1t.

Commands	Description
ESC [(p1); (p2) <SP> G	LINE/CHARACTER SPACING p1 = line spacing parameter p2 = character spacing parameter If you select doublewide printing (ESC [5m), then the printer doubles character spacing set by this command. If the designed pitch of the current font is 10 cpi and doublewide is set, then a character spacing argument of 72 results in a pitch of 5 cpi, at which horizontal segments of line-draw characters are contiguous. Example: ESC [90;60 <sp> G sets the vertical spacing to 8 lines per inch and the horizontal spacing to 12 characters per inch. (720 divided by 8 = 90, and 720 divided by 12 = 60). **Dec** 27 91 52 48 59 54 48 32 71 **Hex** 1B 5B 39 30 3B 36 30 20 47

p2	CPI	
120	6	The quick brown fox jumped
90	8	The quick brown fox jumped over
72	10	The quick brown fox jumped over the lazy
60	12	The quick brown fox jumped over the lazy dog’s back.
51	14	The quick brown fox jumped over the lazy dog’s back.
45	16	The quick brown fox jumped over the lazy dog’s back.
40	18	The quick brown fox jumped over the lazy dog’s back.
36	20	The quick brown fox jumped over the lazy dog’s back.

Figure 2-4 *Effects of Character Spacing Parameters*

Forms Setup

Command	Description
ESC [(p1);(p2);(p3);(p4);(p5)r	FORMS SETUP Sets top of form and sets form length, as well as top and bottom margins, in decipoints. Parameter p1 is form length 720ths , p2 is the top margin, p3 sets the length of white space between the baseline of the last allowed line and the end of form. p4 is the top print reference, and p5 is the left print reference Any parameter not specified is assigned its default value. Maximum form length is 15840 decipoints (22 inches). **Default Values** p1 7920 decipoints 11-inch form p2 0 decipoints 0-inch top margin p3 0 decipoints 0-inch bottom margin p4 0 decipoints 0-inch top print reference p5 0 decipoints 0-inch left print reference Example: Set 8-inch form length, one-inch top margin, one-inch bottom margin **Dec** 27 91 53 55 54 48 59 55 50 48 59 55 50 48 114 **Hex** 1B 5B 35 37 36 30 3B 37 32 30 3B 37 32 30 72 **Note:** Before you send this command, verify that the top print reference is zero in the target paper path. One way to do this is to park and then load paper in the target path. The top edge of the paper should line up with the top of the print head. If not, then set the top print reference to zero from the menu and save the change to the format associated with the target paper path. You must once again park and load paper in the target path in order for the new print reference to take effect. Next, send a command to set the vertical position-absolute to zero. If you do not do this, then the printer establishes top-of-form at the current top margin, and form registration is lost.

Superscript/Subscript

Command	Description
ESC K (PLD)	SUBSCRIPT PRINTING Moves paper 1/2 line below the current line for subscript printing. ESC K is used to return to the original line following ESC L (superscript). If 8-bit control code processing is enabled then **Dec** 139 **Hex** 8B otherwise **Dec** 27 75 **Hex** 1B 4B
ESC L (PLU)	SUPERSCRIPT PRINTING Moves paper 1/2 line above the current line for superscript printing. ESC L is used to return to the original line following ESC K (subscript). If 8-bit control code processing is enabled then **Dec** 140 **Hex** 8C otherwise **Dec** 27 76 **Hex** 1B 4C

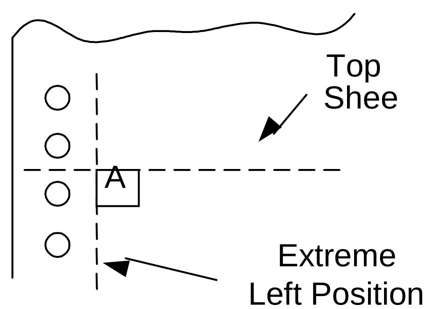
NOTE: The partial line up does not respect top of form; that is, following a form feed, ESC L causes a partial line movement upward. Also, partial line down does not respect bottom of form; that is, when printing the last line on the form, ESC L causes a partial line movement downward.

When printing the last line on a form, do not send a line feed (LF) code to move directly from superscript to subscript. The LF code will be acted upon as the bottom of form terminator.

Vertical Movement

Absolute positions are calculated from the top and left print references without respect to margin settings. The top of a character cell located at vertical position-absolute 0 is at top print reference. The left edge of a character cell located at horizontal position-absolute 0 is at the left print reference. If the default left print reference is current, and if the left tractor position(s) are as shipped, then this is about 0.6 inches from the left edge of the paper.

Top and left print references are adjustable from the printer menu.



Character Printed at Zero Position

CAUTION: Do not attempt reverse paper motion across a perforation, since this may cause the paper to snag on the print head. Vertical position accuracy is not specified for reverse paper moves greater than 1.5 inches.

Vertical Movement

Commands	Description
LF	LINE FEED Advances the paper one line according to the spacing currently in effect. **Dec** 10 **Hex** 0A
VT	VERTICAL TAB Advances paper to the next vertical tab setting. If the current print position is at or beyond the last tab setting, the paper advances to the top of form. If no tabs are set, then VT is processed according to the status of emulation option strap 14. **Dec** 11 **Hex** 0B
FF	FORM FEED Advances the paper to the top margin on the next form. **Dec** 12 **Hex** 0C
ESC D (IND)	INDEX Advances the paper to the next line. The current column is not changed, regardless of the status of auto CR on LF. **Dec** 132 **Hex** 84
ESC E (NEL)	NEXT LINE The next line character advances the paper to the next line. The current column is reset to the left margin. Auto CR on LF has no effect. **Dec** 133 **Hex** 85
ESC M (RI)	REVERSE INDEX Moves the paper back one line. **Dec** 141 **Hex** 8D

Commands	Description
ESC [(p)e	VERTICAL POSITION RELATIVE Advances paper p decipoints. This command enables printing below the bottom margin of the current form and above the top margin of the following form. The example below advances the paper 4 1/4 inches. Example: ESC [3060e **Dec** 27 91 51 48 54 48 101 **Hex** 1B 5B 33 30 36 30 65
ESC [(p)k	VERTICAL POSITION BACKWARDS Moves the vertical position backwards to (p) decipoints above the current position. The horizontal position does not change. If the target position is above the top margin, then the vertical position is the top margin. If no top margin is set, and the target position is above top-of-form, then the vertical position is top-of-form. **NOTE:** The printer economizes vertical motion with vertical logic seeking, so this command may cause reverse paper motion in some instances and not in others. Reverse paper motion can cause problems with some forms. Example: ESC [1080k moves the vertical position up by 1.5 inches. **Dec** 27 91 49 48 56 48 107 **Hex** 1B 5B 31 30 38 30 6B
ESC [(p)d	VERTICAL POSITION ABSOLUTE Moves the current print position to p decipoints from the top of the form. The following example advances the paper to 2 inches below top of form. Example: ESC [1440d **Dec** 27 91 49 52 52 48 100 **Hex** 1B 5B 31 34 34 30 64
ESC[(p1); (p2)f	VERTICAL AND HORIZONTAL POSITION ABSOLUTE Moves the print position to any coordinate on the page. Coordinates are measured in decipoints. Parameter p1 is the vertical coordinate, which is measured from the top print reference. Parameter p2 is the horizontal coordinate, which is measured from the left print reference. The computation of absolute positions is not influenced by margin settings. Top and left print references are adjustable from the control panel. You can print characters beyond the left, top, and bottom margin setting, but no printing is allowed beyond the right margin. EXAMPLE: ESC [1440;2160f Prints the next character 2 inches from the top print reference and 3 inches from the left print reference. **Dec** 27 91 49 52 52 48 59 50 49 54 48 102 **Hex** 1B 5B 31 34 34 30 3B 32 31 36 30 66

Margins

Use the Forms Setup command, page 2-39, to set vertical margins.

Commands	Description
ESC [(p1);6 (p2)s	SET LEFT AND RIGHT MARGIN Sets the left and right margin in decipoints; p1 is the left margin and p2 is the right. Distances are measured from the left print reference. This command takes effect following the next line terminator (you cannot set margins for the current line). EXAMPLE: Set the left margin at 0.4 inch and right margin at 6.9 inches, making a 6.5-inch print line. **Dec** 27 91 50 56 56 59 52 57 54 56 115 **Hex** 1B 5B 32 38 38 3B 34 39 36 38 73
ESC [s	ASSIGN MARGIN DEFAULTS Assigns the default parameters. Sets the left margin to zero, right margin to maximum print line width of the printer. This command takes effect following the next line terminator. **Dec** 27 91 115 **Hex** 1B 5B 73

Horizontal Movement

Commands	Description
HT	HORIZONTAL TAB Causes the current print position to move to the next tab stop. If no tabs are set, then the current position moves one space. If tab(s) are set but no tab(s) are set between the active print position and the right margin, then following characters on the line are either discarded or printed on the next line, depending on the status of auto wrap. **Dec** 9 **Hex** 09
CR	CARRIAGE RETURN Initiates printing and returns the current print position to the left margin. **Dec** 13 **Hex** 0D
ESC [(p)a	HORIZONTAL POSITION RELATIVE Advances the current print position by the distance specified. Parameter p is specified in decipoints. Example: ESC [1080a advances the print position 1.5 inches. **Dec** 27 91 49 48 56 48 97 **Hex** 1B 5B 31 30 38 30 61
ESC [(p)j	HORIZONTAL POSITION BACKWARDS Moves the horizontal position (p) decipoints left of the current position. Actual distance between symbols separated by this command is the argument (p) minus the current horizontal pitch (text or graphics). For example, if you print an uppercase E at 10 cpi, move backwards by (p), and print another uppercase E, then the distance between leading edges of the two characters is (p) - 72 decipoints. If you print graphics at 72 dpi, move backwards by p, and print another graphics column, then the distance between the two graphics columns is (p) – 10 decipoints. This command enables printing left of the left margin. Any data located left of the left print reference is discarded. Example: ESC [1080j moves the horizontal position back by 1.5 inches. **Dec** 27 91 49 48 56 48 106 **Hex** 1B 5B 31 30 38 30 6A

Commands	Description
ESC [(p)`	HORIZONTAL POSITION ABSOLUTE Causes the print position to move (in decipoints) a specified distance from the left print reference. Example: ESC [360` Move print head ½" from left print reference. **Dec** 27 91 51 54 48 96 **Hex** 1B 5B 33 36 30 60
BS	BACKSPACE Prints the data in the print buffer, then moves the print position one character position to the left. **Dec** 8 **Hex** 08

Tabs

This printer stores tab stops in NVRAM while power is off. Therefore, all tab stops should be cleared before new stops are set.

Commands	Description
ESC H (HTS)	SET HORIZONTAL TAB AT CURRENT POSITION If 8-bit control code processing is enabled then both **Dec** 136 **Hex** 88 otherwise only **Dec** 27 72 **Hex** 1B 48
ESC [(p1) (p2);..(p22)u	SET HORIZONTAL TABS AT SPECIFIED POSITIONS Sets up to 22 stops at one time. The value of p1, p2, etc. are in decipoints (1 inch = 720 decipoints). **Dec** 27 91 55 50 48 59 50 56 56 48 117 **Hex** 1B 5B 37 32 30 3B 32 38 38 30 75 The above example sets tabs at 1 inch and 4 inches. Existing tab stops are not cleared. Margin settings have no effect on the positions of tab stops.
ESC J (VTS)	SET VERTICAL TAB AT CURRENT POSITION If 8-bit control code processing is enabled then both **Dec** 138 **Hex** 8A otherwise only **Dec** 27 74 **Hex** 1B 4A
ESC [(p1); (p2);.. (p12)v	SET VERTICAL TABS AT SPECIFIED POSITIONS Sets vertical tabs at positions p1, p2, p3, etc. Up to 12 stops can be set at one time. Tab stops are measured in decipoints from the top print reference. Example: Set tab stops at 4 and 7 inches. **Dec** 27 91 50 56 56 48 59 53 48 52 48 118 **Hex** 1B 5B 32 38 38 30 3B 35 30 34 30 76
ESC [(Ps)g	CLEAR TABS **Ps** **Effect** 0 clear horizontal tab at current print position 1 clear vertical tab at current position 3 clear all horizontal tabs 4 clear all vertical tabs Example: Clear all horizontal tabs **Dec** 27 91 51 103 **Hex** 1B 5B 33 67

Dot Graphics

Dot graphics allows you to print individual dots at any position on the page and thus to print pictures as well as text. When you enter dot graphics, the printer defaults to a dot density of 72 dpi horizontally and vertically, and a line spacing of 12 lpi. Exiting from dot graphics returns the printer to its prior lpi setting. You can also choose a density of 144 dpih x 72 dpiv. The printer powers up in low density. The density remains as last set until it is changed, or until power is turned off.

Each byte of graphics data defines one column of dots that is six dots high. You can think of the least significant six bits in the byte as controlling the topmost six wires of the print head, with the first bit controlling the top wire and the sixth bit controlling the sixth wire down. If you wanted to fire all six wires, then you would send a byte with a decimal value of 63. Byte values of 0 through 31 decimal are interpreted as control codes, so you have to add 64 to any graphics data byte less than 32 decimal.

You would expect to be able to simply add the value 64 decimal to every graphics data byte; in other words, just set the seventh bit. This is risky because some variations within the 3000 product family interpret decimal 127, the Delete character, within a graphics sequence. The effect is to delete everything following the last line terminator. The safest course is:

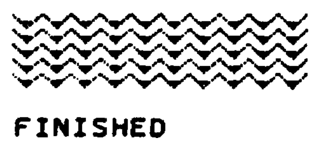
byte_val < 32 then byte_val = byte_val + 64

WEIGHT	WIRE											
1	1	●										●
2	2		●								●	
4	3			●						●		
8	4				●	●	●	●	●			
16	5					●		●				
32	6						●					
		65	66	68	72	88	40	88	72	68	66	65

Figure 2-5 *Dot Column Coding*

Sample Basic Program for Dot Coding

```
10 WIDTH "LPT1:",255
20 LPRINT CHR$(27);"P";         'start graphics
30 FOR N = 1 TO 5
40   FOR K = 1 TO 10
50      LPRINT CHR$(65);CHR$(66);CHR$(68);CHR$(72);
60      LPRINT CHR$(88);CHR$(40);CHR$(88);CHR$(72);
70      LPRINT CHR$(68);CHR$(66);CHR$(65);
80   NEXT K
90   LPRINT                     'cr - lf
100 NEXT N
110 LPRINT CHR$(27);CHR$(92)    'end graphics
120 LPRINT "FINISHED"
```



AE30

Figure 2-6 *Dot Graphics Example*

Graphics Control Functions

The set of control codes that are interpreted while in graphics mode are shown in the following charts. Bytes less than 32 decimal that are not interpreted as control codes are ignored. The printer processes valid escape sequences within a graphics data sequence. This is potentially useful, for instance, in the cases of the absolute and relative position commands. A valid escape sequence that is not applicable to graphics mode, such as bold printing mode, will be implemented for text on exiting graphics. If the printer receives a nonvalid escape sequence while in graphics mode, then it drops out of graphics mode.

Commands	Description
BEL	BELL Causes the beeper to sound for about 1/2 second. **Dec** 7 **Hex** 07
HT	HORIZONTAL TAB Causes the print head to move to the next tab stop. **Dec** 9 **Hex** 09
LF (NEL)	LINE FEED Graphics next line. Causes printing to move to the left margin on the following line. If 8-bit control code processing is enabled, then **NE**xt **L**ine does the same thing. **Dec** 10 or 133 **Hex** 0A or 85
VT	VERTICAL TAB Causes printing to move to the left margin at the next vertical tab stop. **Dec** 11 **Hex** 0B
FF	FORM FEED Causes printing to move to the top left margin on the following form. **Dec** 12 **Hex** 0C
CR	CARRIAGE RETURN Causes printing to move to the left margin on the current line. This permits overprinting the current line. **Dec** 13 **Hex** 0D
IND	INDEX If 8-bit control code processing is enabled, then a line feed is performed at the graphics vertical pitch. The horizontal position stays the same. **Dec** 132 **Hex** 84
RI	REVERSE INDEX If 8-bit control code processing is enabled, then a reverse line feed is performed at the graphics vertical pitch. The horizontal position stays the same. **Dec** 141 **Hex** 8D
ESC P	ENTER DOT GRAPHICS MODE Also sets line spacing to 1/12 inch so that LF produces contiguous graphics lines. **Dec** 27 80 **Hex** 1B 50
ESC \	EXIT DOT GRAPHICS MODE Also restores line spacing to its previously set value. **Dec** 27 92 **Hex** 1B 5C

Commands	Description
ESC [(p1)q	GRAPHICS DENSITY Sets the graphics density. The printer powers up with a graphics density of 72 x 72 dpi; changes are not stored in NVRAM. **p1** **horizontal density** **vertical density** 0 72 dpi 72 dpi 1 144 dpi 72 dpi **Dec** 27 91 48 113 **Hex** 1B 5B 30 71

Chapter 3. ANSI Bar Codes

This chapter describes the characteristics and lists ANSI control sequences for 23 resident bar code styles. You have to be in ANSI emulation to print the resident bar codes. You can adjust various bar code attributes including rotation, height, print density, status of the human-readable line, and element widths. Subject to some constraints based on style and rotation, the human-readable font is also selectable. For some styles, the printer will calculate checksum characters for you.

To print a bar code symbol, you need to:

1) Set the printer to ANSI emulation.
2) Send the escape sequence to set the user-adjustable attributes. Always do this at least once.
3) Send the control sequence to turn on bar code.
4) Send the ASCII representation of symbol data you want to encode.
5) Send the control sequence to turn off bar code.
6) Send a line terminator at some point thereafter.

In bar code parlance, a *module* is the narrowest nominal width of measure, while an *element* is a single bar or space. A *character* is a group of elements that represents a number, letter, or punctuation mark. A *symbol* is a group of characters that can stand alone in terms of being interpreted by the reader. A symbol always comprises one or more data characters framed by white spaces known as *quiet zones*. The symbol frame almost always includes start and stop characters. Depending on the style, some bar code symbols can also include check characters, center characters, and guard bars.

Among printers that generate bar codes internally and that also support any user-adjustment of bar code widths, some enable the specification of module width, while others, such as this one, enable the adjustment of element widths. From the programmer's point of view, adjustable module widths tend to guarantee that the ratio among element widths remains correct, while adjustable element widths afford more flexibility.

When we ship the printer, the ratios among element widths in a given bar code style are within generic specifications for that style. You can restore these default element widths by performing an initial setup (ISU) as described in the User's manual. Since this printer lets you specify element widths independently of each other, it is possible to print bar codes that are outside of generic specifications, or outside of the limits of your particular reading system, or both. You need to know the requirements of your system before you program your printer.

Bar codes printed with this printer are suitable for readers designed for low- or medium-density bar code symbols. Avoid readers with apertures smaller than 7 mils. If bar codes are to be read with an infrared reader, then you must use a special carbon ribbon.

The examples given in this section use the Code 3-of-9 format.

Before You Begin - Set the Form Length

Programmers tend not to trust form feeds and sometimes use line feeds instead to move from one form to another. If there is any advantage to that approach, it is that the vertical move will not depend on the current form length. When printing bar codes on labels, you can greatly simplify your task by setting a form length that exactly matches the length of your physical form and then using form feeds to make vertical moves between labels or rows of labels.

If you print nothing but text, then the vertical position following each line feed lies on a grid that is evenly spaced according to the line feed increment. If you print a bar code symbol, then positions established by subsequent line feeds may not be on that grid. The vertical position following a bar code symbol becomes the origin for a subsequent line feed; this new origin must be computed based on bar code height, and the status of the human-readable line.

If you do not initialize your vertical position with a form feed for each row of labels, then printed objects may creep cumulatively up or down with respect to the top of your physical form.

Setting Bar Code Parameters

The ANSI control sequence that sets bar code parameters has the format:

Command	Description
ESC [p1;p2;p3;p4;p5;p6;p7;p8;p9;p10;p11;p12;p13}	Set Up Bar Code Parameters Determines bar code parameters to be used when bar code mode is enabled. See following sections. **Dec** 27 91 (p1) 59 (p2) 59…(p13)125 **Hex** 1B 5B (p1) 3B (p2) 3B…(p13) 7D

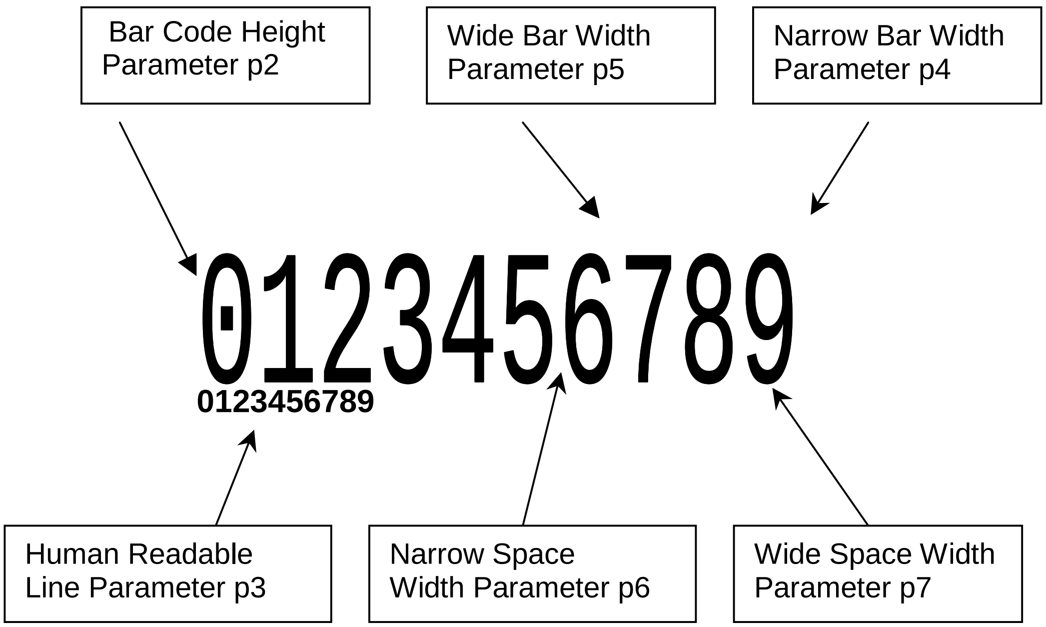


Figure 3-1 *Parts of a Bar Code*

Values of all the adjustable bar code parameters are stored in NVRAM while power is off. You can restore the bar code attributes to the factory settings by performing an ISU on the printer.

pn	Attribute
p1	style
p2	height
p3	human readable line enable
p4	narrow bar width
p5	wide bar width
p6	narrow space width
p7	wide space width
p8	N / A
p9	rotation
p10	horizontal print density
p11	check character
p12	human readable font
p13	height (in 1/24” increments)

General Rules for Assigning Parameters

You can send the sequence to modify bar code parameters any time except when bar code mode is active.

If you assign a parameter value outside the permissible range, then that parameter value reverts to the default value.

Values for narrow bars must be less than values for wide bars. Values for narrow spaces must be less than values for wide spaces. The ratio between wide and narrow elements should be about 3:1.

As with other ANSI escape sequences that take multiple parameters, you can use the semicolon as a place holder when you want to change a higher-numbered parameter while leaving lower-numbered parameters alone. For example, to set the horizontal print density to 60 dpi without changing any other bar code attributes, you could send:

ESC [;;;;;;;;;0;;}

Semicolons that trail the last specified parameter are optional. For example, to turn off the human-readable line without changing any other parameters, you could send:

ESC [;;0}

Or, you could send:

ESC [;;0;;;;;;;;;}

Note: The term “default” is misleading when applied to bar code parameters. This printer stores bar code parameters in nonvolatile RAM while power is off; except for an ISU or an out-of-range argument, you might never see the effects of the factory defaults. For this reason, your application should probably initialize all 12 parameters one time before any symbols are sent. This is particularly true when other bar code applications are sharing the printer.

Bar Code Command Sequences

Command	Description
ESC [3t	ENABLE BAR CODES Enables Bar Codes with the characteristics set by ESC [(p1);(p2)…}. ESC [3t selects Bar Code mode and activates Bar Code printing. **Dec** 27 91 51 116 **Hex** 1B 5B 33 74
ESC [0t	CANCEL OVERSIZED/EXPANDED/BAR CODE MODES **Dec** 27 91 48 116 **Hex** 1B 5B 30 74
SO	SHIFT OUT Turns off bar code printing, then enters bar code mode again. **Dec** 14 **Hex** 0E
SI	SHIFT IN SI turns bar code printing off. **Dec** 15 **Hex** 0F

p1 - Bar Code Style

This printer supports the following styles:

p1	Style
0	Interleaved 2 of 5
1	Bidirectional 2 of 5
2	Matrix 2 of 5
3	Industrial 2 of 5
4	Code 3 of 9 (default)
5	EAN-8
6	EAN-13
7	Code 11
9	Codabar a/t (see note)
10	Codabar b/n (see note)
11	Codabar c/* (see note)
12	Codabar d/e (see note)
13	UPC-A
14	UPC-E
15	Code 93
16	Code 128 (subsets A, B, and C)
17	Reserved
18	Reserved
19	MSI
50	POSTNET

Note: Each of the four Codabar styles will accept any combination of valid start-stop characters that you send. If you do not send any start-stop characters, then the printer generates start-stop characters according to the style in force.

p2 – Height

Height of the bar code in 1/12-inch increments. The default is ¾ inch, which corresponds to an argument of 9.

Caution: In some versions of the printer, the last non-zero value of p13 supplants all subsequent values of p2 until you ISU that printer. (See page 41.)

p3 - Human Readable Enable

An argument of 1 turns on the human-readable line, while 0 turns it off. The default is on. When the human readable line is enabled, it is printed in the font specified by parameter p12. There is 0.1" between the bottom edge of the bar code and the top of the characters in the human-readable line.

Element Widths

The default element widths are the narrowest that we recommend for consistent readability.

The argument units for horizontal bar code element widths are 1/120 inch. The print wire diameter is 1/72 inch; a printed dot is about 1/60 inch in diameter. This is the finest line that the printer can make. The printer accepts an element width of 1 as a valid argument, but the actual width of the resulting element is 1/60 inch, the same as if you had specified a width of 2.

If the element width argument is greater than 2, then the printer rounds down odd numbers. A width argument of 5 produces the same result as a width argument of 4.

p4 - Narrow Bar Width

Argument units are 1/120 inch if the bar code is horizontal (0^o or 180^o rotation) or 1/144 inch if the bar code is vertical (90^o or 270^o rotation). The default is 2.

p5 - Wide Bar Width

Argument units are 1/120-inch increments if the bar code is horizontal or 1/144 inch if the bar code is vertical. The default is 6.

p6 - Narrow Space Width

Argument units are 1/120-inch increments if the bar code is horizontal or 1/144 inch if the bar code is vertical. The default is 2.

p7 - Wide Space Width

Argument units are 1/120-inch increments if the bar code is horizontal or 1/144 inch if the bar code is vertical. The default is 6.

Wide space and wide bar are both set to the smaller of the two values. This sets the proportion of the large to small the same for spaces as it is for bars.

Other Parameters

p9 – Rotation

Sets the absolute rotation in 90 degree increments. If parameter p12 = 0, then the human-readable font is shown in the following table:

p9	Rotation	HR Font if p12 = 0
0	no rotation (default)	current font
1	no rotation	special bar code HR font
2	270^{o}rotation	special bar code HR font
3	no rotation	special bar code HR font
4	270^{o} rotation	special bar code HR font

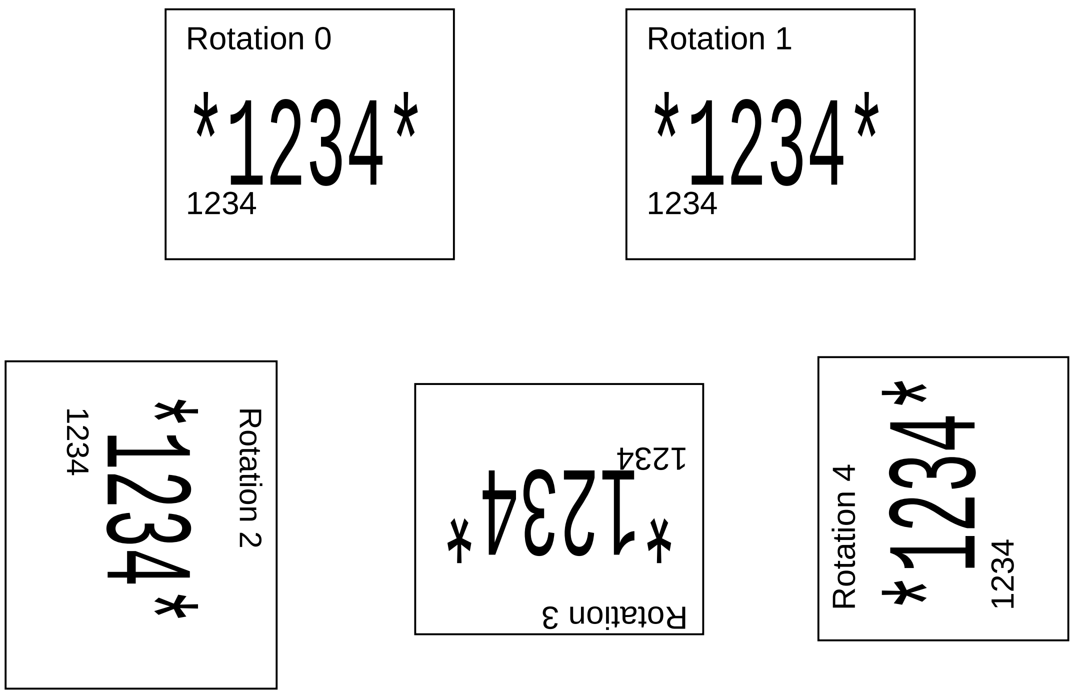


Figure 3-2 *Effects of Rotation Parameters*

p10 - Horizontal Resolution

A value of 0 is 60 dots per inch, while a value of 1 is 1/120 dots per inch. The default depends on the style in force. EAN and UPC styles default to 1. All other styles default to 0.

Note: This parameter affects bar codes printed at 0^{o} or 180^{o} rotation only. If you select the higher resolution, then bar codes are darker and are printed at a somewhat slower speed.

p11 - Check Character Enable

N / A

p12 - Human Readable Font

N / A

p13 - Height in 1/24-inch Increments

Normally, bar code height is specified in 1/12-inch increments by p2. Only specify p13 if your job requires the associated higher resolution of bar code height. A non-zero value of p13 supplants the value of p2.

Caution: In some versions of the printer, a non-zero value of p13 supplants all subsequent values of p2 until you ISU that printer. To be on the safe side, if you do not plan to use p13, then set p13 to 0. NOTE: Bar code height is limited to 10 inches.

Delimiters

A bar code delimiter is a character or control function that tells the printer where one bar code symbol ends and an adjacent symbol begins. Different bar code styles accept different delimiters, which might be spaces, commas, asterisks, horizontal tabs, or horizontal position-relative commands. Valid delimiters for the various styles are listed in the Bar Code Style Characteristics section.

There is always a minimum one-fourth inch of white space called the quiet zone on either side of a bar code symbol, so there is always an irreducible half-inch space between adjacent symbols, no matter which delimiter is used. The space character occupies an additional tenth of an inch. Neither the comma nor the asterisk imposes any additional space between symbols. Asterisks are valid delimiters in Code 3 of 9 (style 4) only, and are normally used in pairs.

```
110 LPRINT CHR$(27);"[0;9;1}";      'height 3/4", hr = on, interleaved 2 of 5
120 LPRINT CHR$(27);"[3t";          'turn on bar code
140 LPRINT "001234,001234";         'comma delimiter
150 LPRINT CHR$(27);"[0t";          'turn off bar code
160 LPRINT "comma delimiter"
170 LPRINT
180 LPRINT CHR$(27);"[3t";          'turn on bar code
190 LPRINT "001234 001234";         'space delimiter
200 LPRINT CHR$(27);"[0t";          'turn off bar code
210 LPRINT "space delimiter"
220 LPRINT
230 LPRINT CHR$(27);"[4}";          'style 4 = 3 of 9
240 LPRINT CHR$(27);"[3t";          'turn on bar code
250 LPRINT "*1234**1234*";          'asterisk delimiter
260 LPRINT CHR$(27);"[0t";          'turn off bar code
270 LPRINT
280 LPRINT "asterisk delimiters are used in pairs"
```

#/E #/E

Interleaved 2 of 5, comma delimiter

#/E #/E

Interleaved 2 of 5, space delimiter

1234 *1234*

1234 *1234*

Asterisk delimiters are used in pairs, 3 of 9

Figure 3-3 *Comma, Space, and Asterisk Delimiters*

Horizontal Tab Delimiter

The HT control code (09) is a valid delimiter in all bar code styles. The leading quiet zone of a subsequent symbol begins at the first tab stop right of the trailing quiet zone of the current symbol. This is evident in the sample, where the quiet zone of the second symbol begins at the third tab stop.

Since this printer stores tabs in nonvolatile RAM while power is off, and since the control sequence that sets horizontal tabs does not clear existing tabs, you should clear all tabs before setting tabs. Note also that new tab stops do not take effect until you send a line terminator; in other words, you cannot set tabs for the current line.

The superfluous tab stops in the following example are included to show you how the function works. If you use horizontal tabs for delimiters, then set no more tab stops than you plan to use.

```
100 LPRINT CHR$(27);"[4;9;1}";          ‘3 of 9
110 LPRINT CHR$(27);"[3g";              'clear all h_tabs
120 'set horizontal tabs at 1/2-inch intervals
130 LPRINT CHR$(27);"[360;720;1080;1440;1800;2160u"
150 FOR K = 1 TO 6                      'show where the tabs are
160 LPRINT CHR$(9);"T";
170 NEXT K
180 LPRINT
190 LPRINT CHR$(27);"[3t";              'turn on bar code
200 LPRINT "1234";CHR$(9);"1234";
210 LPRINT CHR$(27);"[0t";              'turn off bar code
```

```
        T     T     T     T     T     T
*1234*   *1234*
```

Figure 3-4 *Horizontal Tab Delimiter*

Horizontal Position - Relative Delimiter

The horizontal position-relative command is a valid delimiter in all bar code styles except POSTNET.

ESC [(Pn)a

The argument units for this command are decipoints (1/720 inch), which the printer rounds off to the nearest 1/120 inch. This command is a valid delimiter even when issued with an argument of zero.

```
110 LPRINT CHR$(27);"[4;9;1}";        'height 3/4", hr = on
120 LPRINT CHR$(27);"[3t";            'turn on bar code
140 LPRINT "1234";                    'send a symbol
150 LPRINT CHR$(27);"[720a";          'hp_relative 1 inch
160 LPRINT "1234";                    'send a symbol
170 LPRINT CHR$(27);"[0t";            'turn off bar code
180 LPRINT "hp_rel delimiter"
```

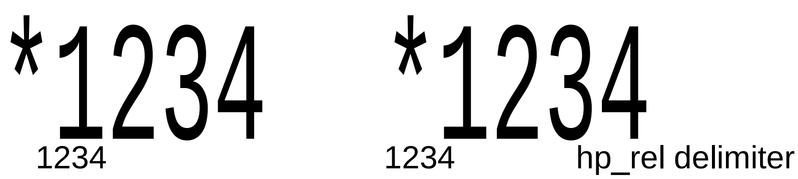


Figure 3-5 *Horizontal Position-Relative Delimiter*

Delimiters and Vertical Bar Codes

If you rotate a bar code 90^o or 270^o, then space imposed by delimiters is not rotated, but is applied horizontally, as the sample shows. The symbol dimensions that used to be horizontal are shortened to 120/144 of what they used to be; this includes the quiet zones.

```
100 LPRINT CHR$(27);"[4;9;1}";          '3 of 9

110 LPRINT CHR$(27);"[3g";              'clear all h_tabs

120 'set horizontal tabs at 1/2-inch intervals

130 LPRINT CHR$(27);"[360;720;1080;1440;1800;2160u"

140 LPRINT CHR$(27);"[0;9;1;;;;;;2}";

150 FOR K = 1 TO 6                      'show where the tabs are

160 LPRINT CHR$(9);"T";

170 NEXT K

180 LPRINT

190 LPRINT CHR$(27);"[3t";      'turn on bar code

200 LPRINT "1234";CHR$(9);"5678";

210 LPRINT CHR$(27);"[0t"       'turn off bar code

220 LPRINT CHR$(27);"[3t";      'turn on bar code

230 LPRINT "2345";CHR$(9);"6789";

240 LPRINT CHR$(27);"[0t"       'turn off bar code
```

T T T T T T

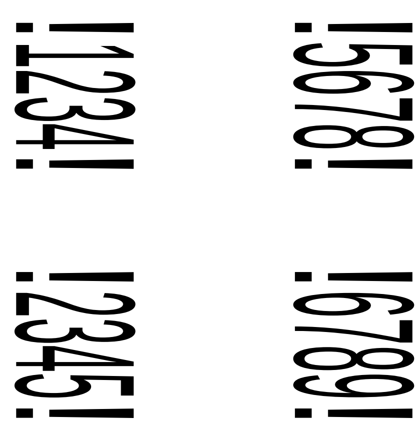


Figure 3-6 *Vertical Bar Code Symbols*

Calculating Characters per Inch

We show characters per inch at the default element widths and $0^{o}/180^{o}$ rotation for each style in the following chapter. All of our numbers ignore the quiet zones; you need to allow 1/2 inch per symbol for the horizontal rotations.

Defining characters per inch for fixed-length codes is straightforward in that the start-stop and center characters always take up the same portion of length of the symbol. The styles that have center characters are all fixed-length.

For variable-length codes, net characters per inch depends on the number of data characters in the symbol. The overhead imposed by start-stop characters is the same whether a symbol represents 1 data character or 10, so the net cpi is greater for longer symbols. We show the overhead imposed by start-stop characters separately for the variable length codes.

The default element widths are the narrowest that we recommend. If you use wider element widths and maintain a 3:1 ratio of wide elements to narrow elements, then you can extrapolate characters-per-inch for wider symbols. If you depart from the 3:1 ratio, then it's simpler for you to print test symbols and derive formulae based on what you measure than it is for us to tell you how to do the computation for each style.

One way to determine the combined length of start-stop characters for a variable-length symbol is to print two symbols, one with twice as many data characters as the other. Measure the shorter symbol and call this L1; measure the longer symbol and call this L2. The combined length of the start-stop characters is:

d = (2 x L1) - L2

To summarize, when using our cpi figures to predict the length of a symbol, remember:

- Allow for two quiet zones per symbol.
- For variable-length symbols, allow the combined start-stop character length for each symbol.
- If you use a check character, then allow for the check character.

Bar Code Style Characteristics

The following pages show specific information about the various styles. A NULL character is a combination of bars and spaces unique to a particular style that is printed when a nonvalid character is encountered in received bar code symbol data. A null character in the symbol shows up as a diamond in the human-readable line if the HRL is enabled.

Interleaved 2 of 5 (Style 0)

1	Bars per unit (2 characters): 5 dark bars and 5 light bars
2	Character set: 0 through 9
3	Start character: 2 dark bars and 2 light bars Stop character: 2 dark bars and 1 light bar
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: Variable
8	Check digit: None.
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters: Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only)
11	CPI at default element widths: 6.857
12	Combined start-stop character length: 0.167"

Bidirectional/Industrial 2 of 5 (Styles 1 and 3)

1	Bars per character: 5 dark bars and 4 light bars
2	Character set: 0 through 9
3	Start/Stop characters: 2 dark bars and 1 light bar
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: Variable
8	Check digit: None
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters: Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only)
11	CPI at default element widths: 4.3
12	Combined start-stop character length: Style 1 = 0.319", Style 3 = 0.139"

Matrix 2 of 5 (Style 2)

1	Bars per character: 3 dark bars and 2 light bars
2	Character set: 0 through 9
3	Start/Stop characters: 3 dark bars and 2 light bars
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: Variable
8	Check digit: None
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters: Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only)
11	CPI at default element widths: 6.0
12	Combined start-stop character length: 0.278"

Code 3 of 9 (Style 4)

1	Bars per character: 5 dark bars and 4 light bars
2	Character set: 0 through 9, upper case letters A through Z characters - . $ / + % and the space character
3	Start/Stop characters: Yes
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: Variable
8	Check digit: None
9	Delimiters: Comma, asterisk (use in pairs--one before the bar code symbol and one following it), horizontal tab, hp_relative
10	HR characters: Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only)
11	CPI at default element widths: 3.78
12	Combined start-stop character length: 0.514"

EAN-8 (Style 5)

1	Bars per character: 2 dark bars and 2 light bars
2	Character set: 0 through 9
3	Start/Stop characters: 2 dark bars and 1 light bar
4	Center character code: Yes
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: 8
8	Check digit: The check digit can be supplied by the data source.
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters (without guard bars): Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only), HR characters (with guard bars): OCR-B font
11	CPI at default element widths: 7.385

EAN-13 (Style 6)

1	Bars per character: 2 dark bars and 2 light bars
2	Character set: 0 through 9
3	Start/Stop characters: 2 dark bars and 1 light bar
4	Center character code: Yes
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: 13
8	Check digit: The check digit can be supplied by the data source.
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters (without guard bars): Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only), HR characters (with guard bars): OCR-B font
11	CPI at default element widths: 8.283

Code 11 (Style 7)

1	Bars per character: 3 dark bars and 2 light bars
2	Character set: 0 through 9 and the dash character
3	Start/Stop characters: Yes
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: Variable
8	Check digit: If needed, it must be generated by the data source
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters: Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only)
11	CPI at default element widths: 6.00
12	Combined start-stop character length: 0.306"

Codabar A/t, B/n, C/, D/e (9, 10, 11, 12)*

1	Bars per character: 4 dark bars and 3 light bars
2	Character set: 0 through 9 and the four additional characters :/ . +
3	Start/Stop characters: Yes (A,B,C,D/A,B,C,D,T,N,*,E)
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: Variable
8	Check digit: If needed, it must be generated by the data source
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters: Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only)
11	CPI at default element widths: 5.05
12	Combined start-stop character length: 0.458"

UPC-A (Style 13)

1	Bars per character: 2 dark bars and 2 light bars
2	Character set: 0 through 9
3	Start/Stop characters: 2 dark bars and 1 light bar
4	Center character code: Yes
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: 12
8	Check digit: The check digit can be supplied by the data source
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters: Currently selected font, special HR
11	CPI at default element widths: 7.64

UPC-E (Style 14)

Ten digits must be entered as input. The first five represent a vendor number and the last five represent a product number. Only 6 of the 10 input digits are encoded in the bar code symbol.

The six characters to be encoded in the symbol are determined as follows:

If the vendor number (first five digits) ends in 000, 100, or 200, then the product number (second five digits) must be between 00000 and 00999. The six digits encoded are the first two characters of the vendor number followed by the last three characters of the product number, followed by the third character of the vendor number. (1st, 2nd, 8th, 9th, 10th, 3rd)

1. If the vendor number ends in 300, 400, 500, 600, 700, 800, or 900, then the product number must be between 00000 and 00099. The six digits encoded are the first three characters of the vendor number followed by the last two characters of the product number, followed by a “3”. (1st, 2nd, 3rd, 9th, 10th, “3”)
2. If the vendor number ends in 10, 20, 30, 40, 50, 60, 70, 80, or 90, then the product number must be between 00000 and 00009. The six digits encoded are the first four characters of the vendor number followed by the last character of the product number, followed by a “4”. (1st, 2nd, 3rd, 4th, 10th, “4”)
3. If the vendor number does not end in zero, then the product number must be between 00005 and 00009. The six digits encoded are all five digits of the vendor number followed by the last character of the product number. (1st, 2nd, 3rd, 4th, 5th, 10th)

1	Bars per character: 2 dark bars and 2 light bars
2	Character set: 0 through 9
3	Start/Stop characters: 2 dark bars and 1 light bar
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: 10
8	Check digit: Automatically generated by the printer and used to check parity. It is not encoded in the bar code symbol.
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters (without guard bars): Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only) HR characters (with guard bars): OCR-B font
11	CPI at default element widths: 7.2

Code 93 (Style 15)

1	Bars per character: 3 dark bars and 3 light bars
2	Character set: All 128 ASCII characters
3	Start/Stop characters: Yes. (Stop character has 4 dark bars and 3 light bars)
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: No
7	Characters per symbol: Variable
8	Check digit: Always required. Must be generated by data source. See page 58
9	Delimiters: Horizontal tab. (When no tabs are defined, a space is inserted.), hp_relative
10	HR characters: Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only)
11	CPI at default element widths: 6.85
12	Combined start-stop character length: 0.333"

Code 128 (Style 16)

1	Bars per character: 3 dark bars and 3 light bars
2	Character set: Any of 3 subsets. The printer automatically shifts among subsets based on the data that you send. Subset A --All standard alphanumeric keyboard characters, control characters and special characters Subset B --All standard alphanumeric keyboard characters, lower case alpha characters and special characters Subset C --Set of 100 digit pairs from 00 to 99
3	Start/Stop characters: Yes
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: Variable
8	Check digit: Automatically generated by the printer. (Cannot be disabled.)
9	Delimiters: Horizontal tab. (When no tabs are defined, a space is inserted.), hp_relative
10	Automatic data compression and subset selection
11	CPI at default element widths: 11.07
12	Combined start-stop character length: 0.583"

MSI (Style19)

1	Bars per character: 4 dark bars and 4 light bars
2	Character set: 0 through 9
3	Start characters: 1 dark bar and 1 light bar Stop characters: 2 dark bars and 1 light bar
4	Center character code: None
5	NULL character: Yes
6	Intercharacter gap: None
7	Characters per symbol: Variable
8	Check digit: The check digit can be supplied by the data source
9	Delimiters: Space, comma, horizontal tab, hp_relative
10	HR characters: Currently selected font, special HR font, special OCR-A/OCR-B font (120 DPI only)
11	CPI at default element widths: 3.8
12	Combined start-stop character length: 0.153"

POSTNET (Style 50)

You cannot adjust POSTNET bar code attributes with ESC[(p1);..(p12)}. All attribute parameters except the style parameter (p1) are ignored by POSTNET, although they are processed, saved, and would apply to a subsequent bar code of another style if not overwritten. To select POSTNET, you need to send only ESC [50}. You cannot print human-readable characters with this bar code style.

There are three variations of POSTNET. They are ZIP, ZIP+4, and ABC. The printer determines which variation is being called for by the number of characters in the symbol string. Legal characters are the numerals 0 - 9 (30 hex - 39 hex) and the dash symbol (2D hex), which is ignored. Any other printable characters in the symbol string cause an error character (diamond) to be printed.

ZIP	5 digits plus 1 check digit
ZIP+4	9 digits plus 1 check digit
ABC	11 digits plus 1 check digit

The check digit is the digit that makes the sum of the digits in the symbol string, including the check digit, evenly divisible by 10. For example, if you send the ZIP symbol string "12345", then the sum of those five digits is 15, so "123455" would be the complete ZIP symbol.

If you do not send a check digit, then the printer computes it and prints it. If you do send a check digit, the printer compares your check digit with its internal computation; and if the two numbers do not agree, then an error character is printed.

1	Bars per character: 2 tall bars and 3 short bars
2	Character set: 0 through 9
3	Start characters: 1 tall bar Stop characters: 1 tall bar
4	Center character code: None
5	NULL character: diamond symbol
6	Intercharacter gap: Yes
7	Characters per symbol: ZIP - 5 + 1 check character ZIP+4 - 9 + 1 check character ABC - 11 + 1 check character
8	Check digit: yes. If a check digit is not sent, then the printer adds it. If a check digit is sent, then the printer verifies it. If a user-supplied check digit is wrong, then an error character is printed.
9	Delimiters: Space, comma, horizontal tab
10	HR characters: none

Calculating the Checksum for Code 3 of 9

A 3 of 9 symbol is a variable-length string which can include the digits 0 - 9, uppercase A - Z, and six additional punctuation characters. The values used to compute the checksum for a 3 of 9 symbol are the locations of individual characters in the unique 3 of 9 character table. Your program will need to refer to a 3 of 9 look-up table to acquire values for the characters you want to encode, and again to convert the computed checksum back to a valid 3 of 9 character.

The check digit is the modulus 43 sum of all the character values in a given symbol.

1. Compute the check character for the character string:

   12345ABCDE/

2. Sum the values of the characters based on their locations in the 3 of 9 character table:

   1 + 2 + 3 + 4 + 5 + 10 + 11 + 12 + 13 + 14 + 40 = 115

3. Compute the modulus 43 of the sum:

   115/43 = 2 Remainder 29

4. The check character is the character corresponding to location 29 in the 3 of 9 character table, which is T. Send to the printer:

   12345ABCDE/T

Code 3 of 9 Character Table

0	0		12	C		24	O		36	- (minus sign)
1	1		13	D		25	P		37	. (period)
2	2		14	E		26	Q		38	(space)
3	3		15	F		27	R		39	$
4	4		16	G		28	S		40	/
5	5		17	H		29	T		41	+
6	6		18	I		30	U		42	%
7	7		19	J		31	V			
8	8		20	K		32	W			
9	9		21	L		33	X			
10	A		22	M		34	Y			
11	B		23	N		35	Z			

Calculating the Checksum for CODE 93

A Code 93 symbol consists of a start code, a variable number of data characters, two check digits (referred to as "C" and "K"), and a stop code.

The values used to compute the checksum for a Code 93 symbol are the locations of individual characters in the Code 93 character table. Check digit "C" is the modulo 47 sum of the character values and a weighting sequence, where the weights from right to left are in the sequence 1, 2, 3...19, 20, 1, 2, 3...19, 20....

Check digit "K" is the modulo 47 sum of the character values and a weighting sequence, where the weights from right to left, starting with the check character "C", are in the sequence

1, 2, 3...14, 15, 1, 2, 3...14,15....

The following example illustrates the calculation of the check digits for the symbol CODE 93.

Data		C	O	D	E	sp	9	3	"C"	"K"
Data Values		12	24	13	14	38	9	3		
C Weights		7	6	5	4	3	2	1		
K Weights		8	7	6	5	4	3	2	1	

1) For "C", sum the products of the (C WEIGHTS * DATA VALUES).

   (1 * 3) + (2 * 9) + (3 * 38) + (4 * 14) + (5 * 13) + (6 * 24) + (7 * 12) = 484

2) Divide 484 by 47.

   484 / 47 = 10 remainder 14.

3) Therefore, the value of "C" is 14, which corresponds to character E.
4) For "K", sum the products of the (K WEIGHTS * DATA VALUES), starting with the newly calculated "C" digit.

   (1*14) + (2*3) + (3*9) + (4*38) + (5*14) + (6*13) + (7*24) + (8*12) = 611

5) Divide 611 by 47.

   611 / 47 = 13 remainder 0

6) The value of "K" therefore is 0, corresponding to character 0. Send to the printer:

   CODE 93E0

The Code 93 character table is the same as the Code 3 of 9 character table, except that there are four additional “control characters” at the end.

Code 93 Character Table

0	0		12	C		24	O		36	- (minus sign)
1	1		13	D		25	P		37	. (period)
2	2		14	E		26	Q		38	(space)
3	3		15	F		27	R		39	$
4	4		16	G		28	S		40	/
5	5		17	H		29	T		41	+
6	6		18	I		30	U		42	%
7	7		19	J		31	V		43	<CTRL> $
8	8		20	K		32	W		44	<CTRL> %
9	9		21	L		33	X		45	<CTRL> /
10	A		22	M		34	Y		46	<CTRL> +
11	B		23	N		35	Z			

Code 93 Checksums for Full ASCII

You can send any of the 128 ASCII characters verbatim in a Code 93 symbol. You would not need to worry about Code 93 control codes if you did not need to compute checksums.

The four control characters in the Code 93 character table do not correspond to any byte values sent to the printer, but instead refer to "precedence codes" that the printer generates. If you send the printer a lowercase "a" in a Code 93 symbol, for instance, then the printer prints bars and spaces that correspond to the <CTRL> + code in front of bars and spaces that correspond to the uppercase "A". The reader then interprets this character combination as a lowercase "a".

So far, this process is transparent both to routines sending symbol strings to the printer and to routines receiving decoded data from the reader. When ASCII characters are represented as character combinations, however, the reader expects to see a checksum based on the values of these combinations; this is the reason that the control codes are assigned values in the Code 93-character table. The following example illustrates the calculation of the check digits for the symbol **Cat** :

Checksum calculation for the symbol Cat

Data		C	<CTRL> +	A	<CTRL> +	T	"C"	"K"
Data Values		12	46	10	46	29		
C Weights		5	4	3	2	1		
K Weights		6	5	4	3	2	1	

1) For "C", sum the products of the (C WEIGHTS * DATA VALUES).

   (1 * 29) + (2 * 46) + (3 * 10) + (4 * 46) + (5 * 12) = 395

2) Divide 395 by 47.

   395 / 47 = 8 remainder 19.

Therefore, the value of "C" is 19, which corresponds to character J.

3) For "K", sum the products of the (K WEIGHTS * DATA VALUES), starting with the newly calculated "C" digit.

   (1*19) + (2*29) + (3*46) + (4*10) + (5*46) + (6*12) = 557

4) Divide 557 by 47.

   611 / 47 = 11 remainder 40

5) The value of “K” therefore is 40, corresponding to character /. Send to the printer:

Code 93 Full ASCII Table

ASCII	Code 93	ASCII	Code 93	ASCII	Code 93	ASCII	Code 93
NUL	%U	SP	space	@	%V	`	%W
SOH	$A	!	/A	A	A	a	+A
STX	$B	"	/B	B	B	b	+B
ETX	$C	#	/C	C	C	c	+C
EOT	$D	$	$	D	D	d	+D
ENQ	$E	%	%	E	E	e	+E
ACK	$F	&	/F	F	F	f	+F
BEL	$G	'	/G	G	G	g	+G
BS	$H	(	/H	H	H	h	+H
HT	$I	)	/I	I	I	i	+I
LF	$J	*	/J	J	J	j	+J
VT	$K	+	+	K	K	k	+K
FF	$L	,	/L	L	L	l	+L
CR	$M	-	-	M	M	m	+M
SO	$N	.	.	N	N	n	+N
SI	$O	/	/	O	O	o	+O
DLE	$P	0	0	P	P	p	+P
DC1	$Q	1	1	Q	Q	q	+Q
DC2	$R	2	2	R	R	r	+R
DC3	$S	3	3	S	S	s	+S
DC4	$T	4	4	T	T	t	+T
NAK	$U	5	5	U	U	u	+U
SYN	$V	6	6	V	V	v	+V
ETB	$W	7	7	W	W	w	+W
CAN	$X	8	8	X	X	x	+X
EM	$Y	9	9	Y	Y	y	+Y
SUB	$Z	:	/Z	Z	Z	z	+Z
ESC	%A	;	%F	[	%K	{	%P
FS	%B	<	%G	\	%L	|	%Q
GS	%C	=	%H	]	%M	}	%R
RS	%D	>	%I	^	%N	~	%S
US	%E	?	%J	_	%O	DEL	%T

Chapter 4. ANSI Oversized Font

This chapter describes the characteristics and lists the ANSI control sequences for the rotatable oversized font, a unique sans-serif font with character shapes designed to accommodate very large expansions. This is an industrial-strength feature. Horizontal resolution is held to 120 dpi to enable oversized printing at carriage speeds of up to 50 inches per second. Combined with vertical logic seeking, this enables the 7265 to print complex mixes of character sizes and rotations at creditable throughputs.

The oversized font is monospaced only. ANSI print modes: bold, underscored, doublewide, and proportional have no effect. You can print characters 20 hex through 7F hex; characters above this range are ignored. User-defined character substitutions do not work in oversized, so you cannot use the IBM line-draw characters. The resident international character substitutions work the same as in normal text.

In contrast to the oversized function on earlier products, the 7265 interprets escape sequences and control codes within an oversized string. As of this writing, however, we recommend that you exit oversized before sending any other control functions.

To maintain compatibility with existing printers, the 7265 printer does not back paper up to print oversized characters on the current baseline. If you enter an oversized mode and immediately print one character, then the top of the character is at the vertical position that was current when you entered oversized. This is no problem if all characters on a line are the same size. To mix oversized with either normal type or with different expansions on the same baseline, however, you will need to write a routine to find the baseline. We cover this later in this chapter.

Some of the oversized control sequences are redundant. The nonrotatable controls that let you optionally toggle oversized with SHIFT OUT and SHIFT IN are retained for compatibility with older printers. You could reasonably choose to use only the escape sequences for rotatable oversized (ESC [(Ps)|) for new applications.

Control Function Summary

Control Function		Description
SO		Activate Oversized, Expanded, Barcode
SI		Exit Oversized, Expanded, Barcode
ESC [0t		Cancel Oversized, Expanded, Barcode
ESC [1t		Non-rotatable Oversized
ESC [2t		Expanded Printing
ESC [Pv;Ph<sp>B		Set Character Size
ESC[0|□ □ □		Exit Oversized
ESC[p1|□ □ □		Set Oversized

Oversized Control Functions

Commands	Description
SO	ACTIVATE OVERSIZED / EXPANDED / BAR CODE **Dec** 14 **Hex** 0E
SI	EXIT OVERSIZED / EXPANDED / BAR CODE **Dec** 15 **Hex** 0F
ESC [0t	CANCEL OVERSIZED / EXPANDED / BAR CODE Cancels oversized, whether set with ESC [1t or ESC [(Ps)|, and also expanded and bar code modes. This command does not clear the oversized dimensions set by ESC [(Pv);(Ph)<sp>B. **Dec** 27 91 48 116 **Hex** 1B 5B 30 74
ESC [1t	NON-ROTATABLE OVERSIZED Included for compatibility with older products. Use if you want to toggle oversized with SHIFT OUT and SHIFT IN. **Dec** 27 91 49 116 **Hex** 1B 5B 31 74
ESC [2t	EXPANDED PRINTING Selects expanded (as opposed to the oversized) printing at the current expansion factor. **Dec** 27 91 50 116 **Hex** 1B 5B 32 74

Commands	Description
ESC [Pv;Ph<sp>B	SET CHARACTER SIZE Sets the expansion of oversized characters in percentage points. Pv is the vertical expansion and Ph is the horizontal expansion. The parent font is a 10-point font. Dimensions established by this sequence are stored in nonvolatile RAM while power is off. **Caution:** Any text between receipt of this command and a countermanding mode selection command, either ESC [(Ps)| or ESC[(Ps)t, will be printed in expanded mode. If you do not want this to happen, then you must follow this command immediately with a mode selection command. The range of valid parameters (in the sense that a valid parameter does not abort the escape sequence) is 0 <= Pn < 32767. The printer rounds off dimension parameters as follows: If Pn < 100, then Pn = 100 If 100 < Pn < 18700, then Pn is rounded down to the nearest 100. If Pn >= 18700, then Pn = 18700 Example - to set 20-point type at 1:1 aspect ratio: **Dec** 27 91 50 48 48 59 50 48 48 32 66 **Hex** 1B 5B 32 30 30 3B 32 30 30 20 42
ESC [0|	EXIT OVERSIZED **Dec** 27 91 48 125 **Hex** 1B 5B 31 7C
ESC [p1|	SET OVERSIZED p1 = 1-4 **Dec** 27 91 49-52 124 **Hex** 1B 5B 31-34 7C

Oversized Versus Expanded

There are two ANSI functions that print characters scaled according to the Graphics Size Modification parameters. Both expanded and oversized are standard features with this printer. You can get into expanded mode accidentally when you send the Graphic Size Modification sequence.

```
100 LPRINT CHR$(27);"[720e";        'vp_relative
110 LPRINT "Expanded ";
120 LPRINT CHR$(27);"[400;400 B";   'graphic size mod
130 LPRINT "ABCj"                   'drops into expanded
140 LPRINT CHR$(27);"[0t";          'cancel mode
150 LPRINT "Oversized ";
160 LPRINT CHR$(27);"[1|";          'set oversized
170 LPRINT "ABCj"
180 LPRINT CHR$(27);"[0t"           'cancel mode
```

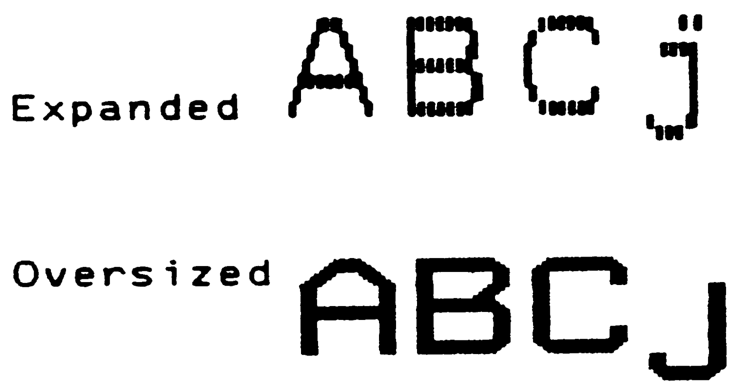


Figure 4-1 Expanded and Oversized

Expanded mode is based on the current font, is limited to 8x the parent font size, and propagates up from the baseline. Reverse paper motion occurs for vertical expansions. The horizontal resolution reflects the current font selection. If you send a line feed character while in expanded mode, then the paper advances by the current vertical expansion factor times the current line-feed increment.

Oversized is based on a unique font, is rotatable, expandable to 1870x, and propagates down from the top of the character cell. The resolution is always 120 dpih x 144 dpiv. If you send a line feed character while in oversized, then paper advances five dot rows for 0^o rotation, or nine dot rows for 270^o rotation. The scaling algorithms for the oversized function are obviously better.

Setting the Expansion

The size of the parent oversized font is 10 points (12 cpi). You can set the horizontal and vertical dimensions of oversized characters separately in increments of 100% of the parent font size. Limits are 100% to 18,700%. The larger number corresponds to an uppercase M about 18.7 inches high. The control sequence that sets the expansion is the ANSI Graphic Size Modification:

ESC [(pv);(ph)<SP>B

where (pv) is the vertical expansion, (ph) is the horizontal expansion, and <SP> is the space character (20 hex).

For 0^o rotation, you can think of parameter (pv) as 10 times the point size of the oversized character. If you set a 600 x 600 expansion and print an uppercase “E” at zero rotation, then the resulting character will nearly match the 60-point Helvetica “E” on a typesetter's scale.

A sideways character shares neither the size nor the aspect ratio of a vertical character printed at the same expansion factors. Relationships between expansion and character dimensions for 270^o rotation reflect differences in the horizontal and vertical resolutions of the printer in oversized mode.

Remember to always follow the graphic size modification command immediately with either ESC [0t or the desired oversized mode selection if you do not want to drop into expanded mode.

Device Timeout with Very Large Characters

Normally, one byte of data in the printer's input buffer represents one text character, which takes milliseconds to print. If our host creates a huge character using dot graphics, then one byte of data in the buffer corresponds to one dot column. If we set oversized, however, then one byte of data can command the printer to render several hundred thousand dot columns. A big oversized character can take a while to print.

If we fill the printer's input data buffer while a big oversized character is printing, then the printer goes BUSY. The big character, as well as a buffer full of following objects, has to be printed before the printer goes READY again.

If the host now tries for some period of time to send data to the printer without seeing a READY at the interface, then the host might decide that the printer is not working, show you a device timeout message, and abort the data transfer.

This same possibility exists with large bar codes, and we discuss remedies in the ANSI Bar Code chapter.

What is a Character Cell?

A character cell is the rectangular array of locations at which dots can be printed to form one character.

All character cells for a given monospaced font at a given pitch are the same size. If you print a line of characters with no countermanding motion instructions, then no character cell will impinge horizontally on the character cell of an adjacent character. You can determine character cell width in a monospaced font by measuring the distance from the leading edge of, say, an uppercase “E” to the leading edge of an adjacent uppercase “E”.

In an impact-matrix printer font, the printed portion of a character is often centered horizontally in the character cell. There might be a fixed number of dot columns on either end of the character cell that are never printed. This is analogous to the side bearings in a typeset character.

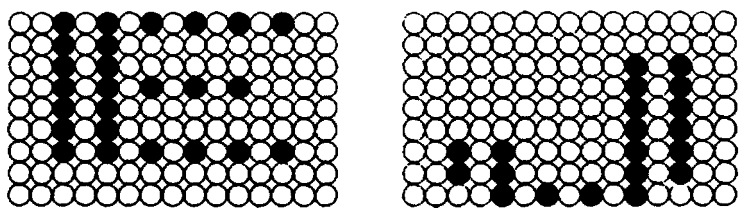


Figure 4-2 *Character Cells*

In the oversized font, on the other hand, characters are left-justified in the cells. This lets you print a larger character when you are printing, say, one huge character on a sheet. The 7265 printer prints as much of an oversized character as will fit between the margins. If your character is going to be clipped at the right margin, then you might rather clip the side bearing than clip a printed portion of that character.

In the oversized font, the topmost dot in the uppercase “E”, for instance, is centered on the upper boundary of the character cell. The lowest dot on the descender of the lowercase “j” is centered on the lower boundary of the character cell.

Oversized Character Dimensions – 0°

The width of an oversized character cell is 0.72 decipoints times the horizontal expansion factor. In thousandths of an inch, it is the same as the horizontal expansion factor. If you set a horizontal expansion of 1000, then characters will be 1 inch, or 720 decipoints, apart.

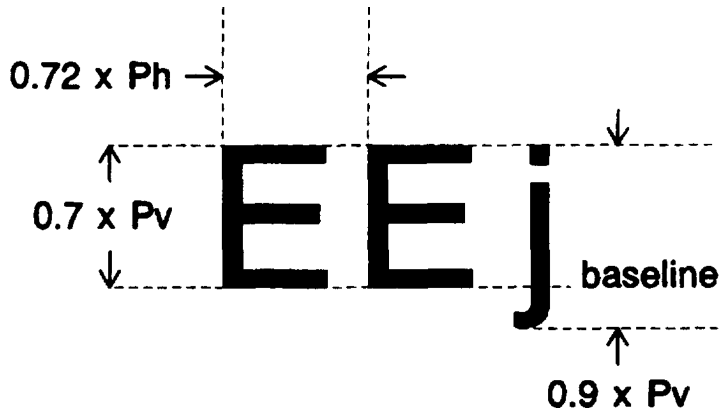


Figure 4-3 *Vertical Character Dimensions*

The height of an oversized character cell is 0.9 decipoints times the vertical expansion factor. In thousandths of an inch, it is 1.25 times the vertical expansion factor. If you set a vertical expansion of 1000, then the top of the uppercase "E" is 1.25 inches, or 900 decpoints, from the bottom of the lowercase "j".

The distance from the top of an oversized character cell to the baseline is 0.7 decipoints times the vertical expansion factor. Top of the character cell means the upper boundary of the cell before it was rotated.

Oversized Character Dimensions - 270°

The width of a character cell is 0.6 decipoints times the horizontal expansion factor. Width means the dimension that is left-to-right as you read the character.

The height of a character cell is 1.08 decipoints times the vertical expansion factor. Height means the dimension that is vertical as you read the character.

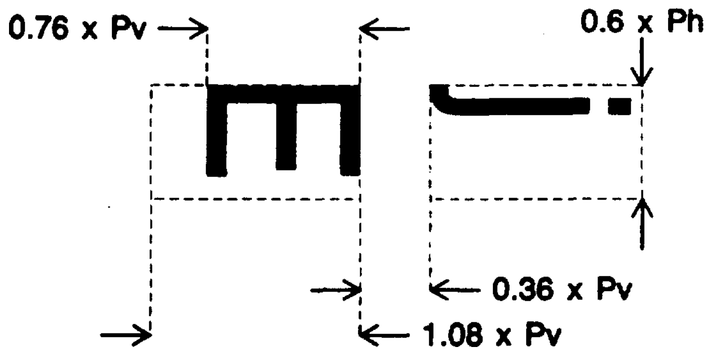


Figure 4-4 *Sideways Character Dimensions*

The distance from the top of the character cell to the baseline is 0.76 decipoints times the vertical expansion factor. Top-of-cell means the upper boundary of the character cell as you read the character.

When printing strings of sideways characters, the printer adds horizontal space between character cells. This space is equal to 0.36 decipoints times the vertical expansion factor.

Vertical Position-Relative in Oversized

Normally, the origin of a vertical position-relative move is the current position. If the last object printed on the current line was an oversized character, however, then the origin of a vertical position-relative move is the top of that character cell, where top-of-cell means the character cell boundary that is uppermost on the paper. This is true whether or not you are still in oversized when the vertical position-relative command is sent.

Implementing the Line Feed Function - 0^o

A good technique is to exit oversized with ESC [0t and send a vertical and horizontal position absolute command:

ESC [(Pv);(Ph)f

where (Pv) is decipoints from the top print reference and (Ph) is decipoints from the left print reference. The logic seeking of the printer minimizes redundant paper motion associated with position-absolute commands. You could also use vertical position-relative to implement line feeds.

Try a line feed increment of 1.2 decipoints times the vertical expansion factor.

If you send a line feed character while in oversized, then paper advances five dot rows for the 0^o rotation.

Line Spacing and Intercharacter Spacing - 270^{o}

If you print sideways text, then the vertical move that otherwise corresponded to a line feed now sets the distance from the leading edge of one character cell to the leading edge of the next character cell (as you read the characters). A good starting value for this move is the character cell width, which is 0.6 decipoints times the horizontal expansion factor.

Strings of sideways characters propagate left-to-right across the portrait page. The printer makes “ line feeds” for you by inserting space between the trailing cell boundary of the previous character and the leading cell boundary of the next character. This space is 0.36 decipoints times the vertical expansion factor, which is generous line spacing.

If this line spacing is not satisfactory, then send sideways text one character at a time, preceded by a locating command for each character. If you use the resident line spacing for sideways lines, then you will probably use a character array to rotate text. In that case, remember to fill your array with space characters (20 hex) before you copy strings into it.

Appendix A. Commands and Exceptions

Supported Control Codes and Escape Sequences

ACK	Acknowledge
BEL	Bell
BS	Back Space
CR	Carriage Return
CSI	Control String Introducer, ESC [
DC1	Select printer (Data Control 1)
DC3	Deselect printer (Data Control 3)
DEL	Delete
ENQ	Enquiry
ESC	Escape
ESC \ or ST	String Terminator
ESC c	Reset to Initial State (RIS)
ESC D or IND	Index
ESC E or NEL	Next line
ESC H or HTS	Horizontal Tab Setting
ESC J or VTS	Vertical Tab Setting
ESC K or PLD	Partial Line Down
ESC L or PLU	Partial Line Up
ESC M or RI	Reverse Index
ESC P data or DCS	Enter Dot Graphics Mode (Device Control String)
ESC [p1; pn }	Sets bar code parameters (GENBC)
ESC [p1;....pn m	Select Graphics Rendition (SGR), Character Pitch, Print Modes and Color Ribbon color.
ESC [p1; p2 SP B	Graphic Size Modification (GSM)
ESC [p1; p2 s	Left/Right Margin Set (GENSLR)
ESC[p1; pn u	Sets horizontal tab stops at specified positions Multiple Horizontal Tab Set (GENHTS)
ESC [p1; pn g	Tab Clear (TBC)
ESC [p1; pn v	Sets vertical tab stops at specified positions (Multiple Vertical Tab Set - GENVTS)
ESC [p1; p2; p3 r	Form Definition (GENFD)
ESC [p1; p2 <SP> G	Sets the line/character spacing
ESC [p1;...pn h	Set Mode (SM)
ESC [p1; pn l	Reset mode (RM)
ESC [p1 ’	Horizontal Position Absolute (HPA)
ESC [p1 a	Horizontal Position Relative (HPR)
ESC [p1 d	Vertical Position Absolute (VPA)
ESC [p1 j	Horizontal Position Backward (HPB)
ESC [p1 k	Vertical Position Backward (VPB)
ESC [p1 q	Select Graphics Mode/Density(GENGRM)
ESC [p1 t	Special Print Mode (Oversize/Expanded/Bar code Mode - GENSPM)
ESC [p1; p2 f	Horizontal and Vertical Position Absolute (HVP)
ESC [p1 e	Vertical Position Relative (VPR)
ETX	End of transmission
FF	Form Feed
HT	Horizontal tab
LF	Line Feed
NUL	Ignored
OSC	Operating System Command, ESC]
SI	Shift In
SO	Shift Out
SP	Space
VT	Vertical tab

Commands not implemented in the 7265 ANSI Emulation

ESC] p1;pn<ST>	User Defined Character Substitution (GENUDS)
ESC] p1;p2;p3;data<ST>	Operating system command (Load Mode OSC)
ESC [p1;p2 SP ~	Selects emulation (GENEMU)
ESC [p1;pn {	Unidirectional Printing (GENUPD)
ESC [p1;p2;p3 SP |	Customer Setup Save/Unsave (GENCSX)
ESC [p1;pn p	Assign Source for Forms (GENASF)
ESC [p1 x	Selects National character set (Select National Characters - GENSNC)
ESC k	Print Test Character (GENPTC)
ESC Q or PU1	Executes Self Test

Exceptions

1.) Barcode ratios are controlled by one parameter that affects both width and height.
2.) Block characters print at a fixed size.
3.) Block characters printed with a fixed rotation.
4.) The Genicom printer saves ESC sequence changes to powerup; we load from the powerup menu.
5.) <DEL> character prints a space; Genicom prints a “house” character.
6.) Across 13.6 inches we print one less character per line in the following CPI's:
    6, 6.25, 6.67, 8.33, 8.57, 9, 12, 12.5, 13.3, 17.14 and 18
7.) If we set Auto CR to ON then form feeds also get a CR, where Genicom does not add CR to form feeds.
    Genicom does add CR to FF if "Auto CR on Vertical Tab" is enabled.
8.) Different combinations of adding and removing vertical tabs will produce different results compared to the Genicom printer.
9.) Differences when mixing some form lengths with top and bottom margins.
10.) Some illegal parameters are treated different than the Genicom printer.
11.) Default horizontal and vertical tabs are not defined at power up.

Programmer's Reference

EPSON Emulation

Table of Contents

Description of Escape Sequences

Appendix A Summary of Possible Codes

Appendix B Character Sets

Index

Description of Sequences

Introduction

Paper and Text Formatting

Character Spacing

Line Spacing

Character Styling

Print Positioning

Graphics

Download Character Generator

Miscellaneous

Control Codes

Introduction

This User's Manual describes the sum of all EPSON Escape sequences, regardless of the specific EPSON emulation used by your printer. The only differentiation is made between 9 and 24 needle printers as well as the maximum possible paper width of 8 inches (80 column printer) or 13.6 inches (136 column printer). Please note that the print quality LQ (Letter Quality) is only available in the 24 needle printer.

Please be sure to observe the notes and steps described in the Operator's Manual as well as the specific EPSON emulations implemented in your printer and which of the described sequences are thus not available for that specific emulation.

Select the EPSON mode as described in the Operator's Manual. Selecting this emulation mode will automatically select the Epson character set.

Read the Operator's Manual to see whether it is also possible to use ANSI (MTPL) sequences in addition to the special Epson sequences.

The following explanations will help you understand the sequences better:

Every sequence description begins with a header, in which the function and the short form of the sequence are listed without parameters, e.g.:

Setting the form length in lines	ESC C
Setting the form length in inches	ESC C NUL

The **ESC** control code (hex. 1B, dec. 27) introduces every Escape sequence. The characters following the ESC control code (here **C**) determine the sequence's function.

This is followed by the **Data Structure** in ASCII, hexadecimal and decimal syntax with the necessary parameters, e.g.:

ASCII	ESC	"C"	<n>	Setting the form length in lines
hex.	1B	43	<n>	
dec.	27	67	<n>	

For the parameter (here n) it is necessary to differentiate between two types of syntax.

- if the parameter is in pointed parentheses, the decimal value is transferred
- if the parameter is **not** in pointed parentheses, the ASCII value is transferred

Example:

Parameter syntax:	**<n>**, with n=0
to be transmitted:	ASCII "NUL" (hex.00, dec.0)
Parameter syntax:	**n**, with n=0
to be transmitted:	ASCII "0" (hex.30, dec.48)

Character explanation and symbol description

ℓ	Lower case "l"
i	Information
9	Sequence only applies for 9 needle printer
24	Sequence only applies for 24 needle printer
ESC/P2	Sequence only applies for ESC/P2 printers

On the next page you will find examples for Escape sequences complete with BASIC programming examples.

Setting the form length to 72 lines

Escape Sequence		ESC	C	<n> with n=72
Transfer	ASCII	ESC	"C"	"H"
	hex.	1B	43	48
	dec.	27	67	72

```
100 REM Sample for the ESC C <n> sequence in ASCII,
110 REM using form with 72 lines.
120 REM Please note, in ASCII-Syntax you can use ASCII values
130 REM equal or bigger codetable no. 32 only.
140 REM LPRINT CHR$(27);"C";"H": REM mixed syntax;
150 REM set form length up to 72 lines
160 REM The same sequence written in hexadecimal syntax
170 LPRINT CHR$(&H1B);CHR$(&H43);CHR$(&H48);
180 REM set form length up to 12 lines
190 REM The same sequence written in decimal syntax
200 LPRINT CHR$(27);CHR$(67);CHR$(72);
210 REM set form length up to 72 lines
```

Setting the form length to 12 inches

Escape Sequence		ESC	C	NUL	<n> with n=12
Transfer	ASCII	ESC	"C"	NUL	FF
	hex.	1B	43	00	0C
	dec.	27	67	0	12

```
100 REM Sample for the ESC C NUL <n> sequence in ASCII,
110 REM using form with 12 inch length.
120 REM Please note, in ASCII-Syntax you can use ASCII values
130 REM equal or bigger codetable no. 32 only.
140 REM LPRINT CHR$(27);"C";CHR$(0);CHR$(12): REM mixed syntax;
150 REM set form length up to 12 inch
160 REM The same sequence written in hexadecimal syntax
170 LPRINT CHR$(&H1B);CHR$(&H43);CHR$(&H0);CHR$(&HC);
180 REM set form length up to 12 inch
190 REM The same sequence written in decimal syntax
200 LPRINT CHR$(27);CHR$(67);CHR$(0);CHR$(12);
210 REM set form length up to 12 inch
```

Form Length in Lines — ESC C
Form Length in Inches — ESC C NUL

Data Structure

ASCII	ESC	"C"	<n>		set form length in lines	❶❸
hex.	1B	43	<n>			
dec.	27	67	<n>			
ASCII	ESC	"C"	<0>	<n>	set form length in inches	❷❸
hex.	1B	43	0	<n>		
dec.	27	67	00	<n>		

Description

❶ This code sequence sets the form length to n times the current line feed pitch (in inches). Also the current print position is simultaneously defined as top of form.

The value specified for n must be an integer in the range of 1 to 127. This value is multiplied by the current line feed pitch to obtain the form length. For example, if 60 is specified for n and the current line feed pitch is $\frac{1}{6}$ inch, the form length is set to 10 inches. Once set by this sequence, the form length is not affected by changing the line spacing. This code sequence is ignored if the value specified for n is not in the range of 1 to 127.

❷ This code sequence sets the form length to the number of inches specified by n. It also defines the current print position as top of form.

The value specified for n must be an integer in the range of 1 to 22; otherwise the sequence will be ignored.

❸ This code sequence resets the perforation skip function (ESC N).

Example

```
10  REM form length
20  FF$=CHR$(12)
30  K=1
40  LPRINT CHR$(27);"C";CHR$(4);
50  FOR F=1 TO 3
60  FOR L=1 TO K
70  LPRINT "length 4: form";F;"line";L
80  NEXT L
90  K=K+1
100 LPRINT FF$;
110 NEXT F

120 K=1
130 LPRINT CHR$(27);"C";CHR$(3);
140 FOR F=1 TO 3
150 FOR L=1 TO K
160 LPRINT "length 3: form";F;"line";L
170 NEXT L
180 K=K+1
190 LPRINT FF$;
200 NEXT F
210 END
```

```
length 4: form 1 line 1



length 4: form 2 line 1
length 4: form 2 line 2


length 4: form 3 line 1
length 4: form 3 line 2
length 4: form 3 line 3

length 3: form 1 line 1


length 3: form 2 line 1
length 3: form 2 line 2

length 3: form 3 line 1
length 3: form 3 line 2
length 3: form 3 line 3
```

Set Page Length in Units — ESC (C

Data Structure

ASCII	ESC	"("	"C"	<n_L> <n_H> <m_L><m_H>	ESC/P2	set page length in units
hex.	1B	28	43	<n_L> <n_H> <m_L><m_H>		
dec.	27	40	67	<n_L> <n_H> <m_L><m_H>		

Description

Valid values:

$n_L = 2,\ n_H = 0$

$0 < ((m_H \times 256) + m_L) \times (\text{defined unit}) \leq 22$

Defines page length in untis previously defined with the sequence ESC (U (see page 62).

$(\text{page length}) = ((m_H \times 256) + m_L) \times (\text{defined unit})$

$$m_H = \mathrm{INT}\left(\frac{(\text{page length}) \times \frac{1}{(\text{defined unit})}}{256}\right)$$

$$m_L = \mathrm{MOD}\left(\frac{(\text{page length}) \times \frac{1}{(\text{defined unit})}}{256}\right)$$

This sequence sets top and bottom margins to default.

Set page length first, then load paper. Using this sequence within the form sets top of form position at the current position.

Set Page Format ESC (c

Data Structure

ASCII	ESC	"("	"c"	$<n_L>$	$<n_H>$	$<t_L>$	$<t_H>$	$<b_L>$	$<b_H>$	ESC/P2	set page format
hex.	1B	28	63	$<n_L>$	$<n_H>$	$<t_L>$	$<t_H>$	$<b_L>$	$<b_H>$		
dec.	27	40	99	$<n_L>$	$<n_H>$	$<t_L>$	$<t_H>$	$<b_L>$	$<b_H>$		

Description

Valid values:

$n_L = 4,\ n_H = 0$

$((t_H \times 256) + t_L) < ((b_H \times 256) + b_L)$ top margin < bottom margin

$((b_H \times 256) + b_L) \times (\text{defined unit}) \leq 22$ bottom margin < 22 inches

Defines top and bottom margins previously defined with the sequence ESC (U (see page 62).

(top margin) = $((t_H \times 256) + t_L) \times (\text{defined unit})$

$$t_H = \text{INT}\left(\frac{(\text{top margin}) \times \dfrac{1}{(\text{defined unit})}}{256}\right)$$

$$t_L = \text{MOD}\left(\frac{(\text{top margin}) \times \dfrac{1}{(\text{defined unit})}}{256}\right)$$

(top margin) = $((t_H \times 256) + t_L) \times (\text{defined unit})$

$$b_H = \text{MOD}\left(\frac{(\text{bottom margin}) \times \dfrac{1}{(\text{defined unit})}}{256}\right)$$

$$b_L = \text{MOD}\left(\frac{(\text{bottom margin}) \times \dfrac{1}{(\text{defined unit})}}{256}\right)$$

This sequence sets new top and bottom. it does not affect the current page length setting.

Set top and bottom margin first, then load paper. Using this sequence within the form sets top margin at the current position.

Perforation Skip — ESC N
Reset Perforation Skip — ESC O

Data Structure

ASCII	ESC	"N"	<n>	set space before perforation (perforation skip)	❶
hex.	1B	4E	<n>		
dec.	27	78	<n>		
ASCII	ESC	"O"		reset perforation skip	❷
hex.	1B	4F			
dec.	27	79			

Description

❶ This code sequence enables the perforation skip function and sets the bottom margin (distance between the last print line of one page and the first print line of the following page) to n lines. The value specified for n must be in the range of 1 to 127 and must be smaller than the form length minus the top margin.
The printer automatically advances the paper to the first printable line of the next page whenever the current print line falls within the margin area defined as n lines from the bottom of the current page (n being the value specified for the bottom margin area). The margin area before and after the perforation is n times the current line spacing. Once this margin area has been defined, it will not be affected by changing the line spacing and remains effective until the printer receives code sequence ESC O (reset perforation skip), or until form length is changed by ESC C or ESC C 0.

❷ This code sequence deselects the perforation skip function set by ESC N <n>.The bottom margin is set to 0 lines, thus disabling perforation skip. This results in continuous printout, unless the software used in the computer defines the form length by counting the printed lines.

This function only applies for fanfold paper, not for cut sheet processing or with mounted sheet feeder.

Example

```
10 REM    skip
20 LPRINT CHR$(27);"C";CHR$(6);:REM form length
30 LPRINT CHR$(27);"N";CHR$(1);:REM set skip
40 FOR F=1 TO 2
50 FOR L=1 TO 5
60 LPRINT "form";F;" line";L
70 NEXT L:NEXT F
80 END
```

```
form 1  line 1
form 1  line 2
form 1  line 3
form 1  line 4
form 1  line 5

form 2  line 1
form 2  line 2
form 2  line 3
form 2  line 4
form 2  line 5
```

Right Margin	**ESC Q**
Left Margin	**ESC ℓ**

Data Structure

ASCII	ESC	"Q"	<n>	set right margin	❶
hex.	1B	51	<n>		
dec.	27	81	<n>		
ASCII	ESC	"ℓ"	<n>	set left margin	❷
hex.	1B	6C	<n>		
dec.	27	108	<n>		

Description

	Values for n	
Character density	80 columns printer	136 columns printer
10 cpi	1... 80	1...136
12 cpi	1... 96	1...163
15 cpi	1...120	1...204
17.1 cpi	1...136	1...232
20 cpi	1...160	1...255

Values for n are valid for a line length of 8 inch (80 columns printer) or 13.6 inch (136 columns printer) set in the menu.

The absolute margin position depends on whether enlarged print mode, compressed print mode, Pica or Elite are selected. When using proportional spacing, the setting of the margins corresponds to the setting when in Pica font type (10 cpi).

The command for setting the margins must be given at the beginning of a line, all data of the same line are lost in the print buffer.

❶ The right margin is set to n columns, depending on the character density selected. As soon as the right margin is reached after this command has been given, a carriage return and a line feed are added to the characters to be printed.

❷ The left margin is set to n columns, depending on the character density selected.

Example

```
10  REM    left and right margin
20  LPRINT "         1         2         3         4         5"
30  LPRINT "12345678901234567890123456789012345678901234567890123"
40  LPRINT CHR$(27);"l";CHR$(10);CHR$(27);"Q";CHR$(53);
50  LPRINT "Now the left margin is set to column 10 and";
60  LPRINT "the right margin is set to column 53."
70  LPRINT CHR$(27);"l";CHR$(5);CHR$(27);"Q";CHR$(41);
80  LPRINT "Now the left margin has been reduced";
90  LPRINT "to column 5 and the right margin has";
100 LPRINT "been set to column 41."
110 END
```

```
         1         2         3         4         5
12345678901234567890123456789012345678901234567890123

          Now the left margin is set to column 10 and
          the right margin is set to column 53.
     Now the left margin has been reduced
     to column 5 and the right margin has
     been set to column 41.
```

Justification and Centering — ESC a

Data Structure

ASCII	ESC	"a"	<n>
hex.	1B	61	<n> or n
dec.	27	97	<n> or n

Description

In some cases it is necessary to center headlines or to have the text printed in a way that left or right margin or both margins are justified. By means of this code sequence the printer automatically justifies the text.

Justification mode depends on the value selected for n:

0 (hex.00 or hex.30)	left justification (default setting)
1 (hex.01 or hex.31)	centered
2 (hex.02 or hex.32)	right justification
3 (hex.03 or hex.33)	left and right justification ❶

> The justification mode must always be set at beginning of the line.
> Justification mode can be used in all print qualities.

❶ If left and right justification is activated, the validity of codes ESC $, ESC\ and HT at the beginning of the line is checked.
Left and right justification is only achieved if the length of the data line (measured from the start of data input up to a CR or LF code) fills 75% to 125% of the given printed area.
At less than 75%, left and right justification justification is not achieved. If the length of the printed line extends more than 25% outside of the printed area, left and right justification justification is carried out for the character spacing which comes closest to the value of 100% within the printed area. The remaining data are printed in the next line.

Example

```
10  REM    justification and centering
20  WIDTH  "LPT1:",255
30  LPRINT CHR$(27);"Q";CHR$(48);:REM set right margin
40  LPRINT CHR$(27);"a";CHR$(1);"CENTERED"
50  LPRINT "HEADLINE"
55  LPRINT CHR$(27);"a";CHR$(3)
60  LPRINT "Now the right and the left justification ";
70  LPRINT "is on. The lines are filled with blanks ";
80  LPRINT "until they are long enough to reach the ";
90  LPRINT "right margin."
100 LPRINT
110 LPRINT CHR$(27);"a";CHR$(2);"These lines are only"
120 LPRINT "justificated on the"
130 LPRINT "right margin."
140 LPRINT
150 LPRINT CHR$(27);"a";CHR$(0);"Now the default setting"
160 LPRINT "is selected again. The lines are"
170 LPRINT "justified on the left margin."
180 END
```

```
                CENTERED
                HEADLINE

Now   the   right   and   the   left
justification  is  on.  The lines are
filled  with  blanks  until  they  are
long enough to reach the right margin.

                  These lines are only
                   justificated on the
                         right margin.

Now the default setting
is selected again. The lines are
justified on the left margin.
```

Character Spacing

Set HMI (Horizontal Motion Index)	**ESC c**

Data Structure	ASCII	ESC	"c"	<n_L> <n_H>	ESC/P2	set HMI (Horizontal Motion Index)
	hex.	1B	63	<n_L> <n_H>		
	dec.	27	99	<n_L> <n_H>		

Description

Valid values:

$0 \leq n_H \leq 4$

$0 \leq n_L \leq 255$

$0 < ((n_H \times 256) + n_L)) \leq 1080$; HMI $\leq$ 3.00 inches

The normal print density is enlarged according this formula:

$$HMI = \frac{(n_H \ \ x\ 256) + n_L}{360} \ \text{inch}$$

$$n_H = INT \frac{HMI \ x\ 360}{256}$$

$$n_L = MOD \frac{HMI \ x\ 360}{256}$$

This sequence terminates previously set additional character space defined with ESC SP.

Following sequences will cancel the HMI mode:

ESC !	print mode selection
ESC @	printer initialization
DC2	reset condensed print
DC4	reset enlarged print mode (set by <SO> or ESC <SO>)
ESC g	character density 15 cpi
ESC M	character density 12 cpi (Elite)
ESC P	character density 10 cpi (Pica)
ESC p	proportional spacing ON/OFF
ESC SP	character spacing
ESC W	continuous enlarged print mode ON/OFF
SI	condensed print ON
SO	enlarged print mode (automatically reset after one line)

Set enlarged Print Mode	**SO/ESC SO**
Reset enlarged Print Mode	**DC4**
Continuous enlarged Print Mode	**ESC W**

Syntax

ASCII	SO	or	ESC	SO	set enlarged print mode (for one line)	❶
hex.	0E	or	1B	0E		
dec.	14	or	27	14		
ASCII	DC4				reset enlarged print mode (set by SO or ESC SO)	❷
hex.	14					
dec.	20					
ASCII	ESC	"W"	<n> or n		set/reset continuous enlarged print mode	❸
hex.	1B	57	<n> or n			
dec.	27	88	<n> or n			

Description

❶ Either one of these codes switches the printer to the enlarged print mode. In enlarged print mode, the width of printed characters are twice that in the normal mode. Enlarged printing can be used with all print qualities.
Enlarged print mode set by SO or ESC SO is reset by LF, CR, FF, VT, ESC !, ESC W <0> and DC4.

❷ This code resets the printer from enlarged print mode to normal (standard size) mode.The code is ignored except when the printer has been set to enlarged mode by SO or ESC SO. It does not affect the enlarged mode set by ESC W or by ESC ! sequences.

❸ This code sequence switches the printer to continuous enlarged print mode when 1 is specified for n, and resets printing to normal mode when 0 is specified.
Enlarged print activated by the code sequence SO or ESC SO is also reset to normal mode when n=0. Valid values for n are hex.00, hex.01, hex.30 and hex.31.

Example

```
10  REM enlarged character densities
20  LPRINT CHR$(27);"P";: REM 10 cpi
30  LPRINT "normal 10 cpi character density"
40  LPRINT CHR$(27);"W";CHR$(1);: REM continuous enlarged density
50  LPRINT "10 cpi continuous enlarged"
60  LPRINT CHR$(27);"M";: REM 12 cpi
70  LPRINT "12 cpi continuous enlarged"
80  LPRINT CHR$(27);"W0";: REM normal character density
90  LPRINT CHR$(14);: REM enlarged character density for one line
100 LPRINT "this is enlarged only"
110 LPRINT "for one line"
120 END
```

```
normal 10 cpi character density
1 0   c p i   c o n t i n u o u s   e n l a r g e d
1 2  c p i  c o n t i n u o u s  e n l a r g e d
t h i s   i s   e n l a r g e d   o n l y
for one line
```

Condensed Character Density — SI/ESC SI
Reset Condensed Character Density — DC2

Data Structure

ASCII	SI	or	ESC	SI	condensed print ON	❶
hex.	0F	or	1B	0F		
dec.	15	or	27	15		
ASCII	DC2				reset condensed print	❷
hex.	12					
dec.	18					

Description

❶ When the printer receives this code, the data will be printed approx. 40% smaller than in normal mode. Condensed print mode can be used in all print qualities. It can be used during printing in the double strike or bold modes. Once set, this code remains effective until it is reset by code DC2.

❷ When the printer receives this code, the condensed print mode is reset.

This code is ignored except when the printer has been set to condensed mode by SI or ESC SI.

9 Proportionally spaced characters cannot be condensed.

Example

```
10 REM   condensed character densities
20 LPRINT CHR$ (15);"this is condensed density";
30 LPRINT CHR$(18);" and this is normal density"
40 END
```

this is condensed density and this is normal density

Character Density 10 cpi	**ESC P**
Character Density 12 cpi	**ESC M**
Character Density 15 cpi	**ESC g**

Data Structure

ASCII	ESC	"P"	set character density to 10 cpi (Pica)		❶
hex.	1B	50			
dec.	27	80			
ASCII	ESC	"M"	set character density to 12 cpi (Elite)		❷
hex.	1B	4D			
dec.	27	77			
ASCII	ESC	"g"	set character density to 15 cpi	24	❸
hex.	1B	67			
dec.	27	103			

Description

❶ This code sequence sets the character density to 10 cpi (Pica). This is also the default character density.

❷ This code sequence sets the character density to 12 cpi (Elite).

❸ This code sequence sets the character density to 15 cpi. With this code sequence not only the character spacing is altered, but also the size of the characters (approx. 2.3 mm high and 1 mm wide). That is why the font set with this sequence is referred to as microfont.

All three control codes neutralise each other.
If the character spacing is changed using one of the sequences – with proportional spacing selected this change only becomes effective when the printer leaves the proportional mode.

Example

```
10 REM    single character densities
20 LPRINT CHR$(27);"P";"this is pica sized"
30 LPRINT CHR$(27);"M";"and this is elite sized"
40 LPRINT CHR$(27);"g";"and this is 15 cpi"
50 END
```

```
this is pica sized
and this is elite sized
and this is 15 cpi
```

Proportional Spacing — ESC p

Data Structure

ASCII	ESC	"p"	<n> or n	proportional spacing ON/OFF
hex.	1B	70	<n> or n	
dec.	27	112	<n> or n	

Description

If 1 is specified for n, proportional spacing is ON, if 0 is specified for n, proportional spacing is OFF.

When in default font type, all characters are assigned a fixed, equally wide space. When in proportional spacing, the spaces between each character are set to the actual width of the characters; on account of these “proportional” characters reading of the text becomes more easy.

Proportional spacing selects automatically LQ as print quality. Condensed is reset. The set character densities are only effective again, when proportional spacing is switched off.

Example

```
10  REM    proportional spacing
20  LPRINT CHR$(27);"p";CHR$(1);
30  LPRINT "This is proportional spacing"
40  LPRINT CHR$(27);"p";CHR$(0);
50  LPRINT "This is normal spacing"
60  END
```

This is proportional spacing
This is normal spacing

Character Spacing — ESC SP

Data Structure

ASCII	ESC	SP	<n>
hex.	1B	20	<n>
dec.	27	32	<n>

Description

The size of the spaces between the characters is increased by n $*$ $1/180$ inch in NLQ and LQ and in Draft by n $*$ $1/120$ inch. This space is added to the spaces used for character definition.

The value specified for n must be in the range of 0 to 127.

Example

```
10 REM    character spacing
20 LPRINT CHR$(27);"@"
25  LPRINT CHR$(27);"x1";:REM High Print quality
30 LPRINT "this is normal width"
40 LPRINT CHR$(27);" ";CHR$(3);
50 LPRINT "now 3/180 inch is added to normal width"
60 LPRINT CHR$(27);" ";CHR$(6);
70 LPRINT "now 6/180 inch is added"
80 END
```

```
this is normal width
n o w  3 / 1 8 0  i n c h  i s  a d d e d  t o  n o r m a l  w i d t h
n o w   6 / 1 8 0   i n c h   i s   a d d e d
```

Line Spacing

Fixed Line Spacing 1/8 Inch	**ESC 0**
Fixed Line Spacing 7/72 Inch	**ESC 1**
Fixed Line Spacing 1/6 Inch	**ESC 2**

Data Structure

ASCII	ESC	"0"	set line spacing to 1/8 inch	❶
hex.	1B	30		
dec.	27	48		
ASCII	ESC	"1"	set line spacing to 7/72 inch 9	❷
hex.	1B	31		
dec.	27	50		
ASCII	ESC	"2"	set line spacing to 1/6 inch	❸
hex.	1B	32		
dec.	27	50		

Description

❶ This code sequence sets line spacing to $\frac{1}{8}$ inch.
Once the line spacing has been changed by this sequence, $\frac{1}{8}$ inch line feeds are made until line spacing is changed by code sequences ESC 1, ESC 2, ESC 3 n, ESC A n or ESC + n.

❷ This code sequence sets line spacing to $\frac{7}{72}$ inch.
Once the line spacing has been changed by this sequence, $\frac{7}{72}$ inch line feeds are made until line spacing is changed by code sequences ESC 0, ESC 2, ESC 3 n, ESC A n or ESC + n.

❸ This code sequence sets line spacing to $\frac{1}{6}$ inch.
Once the line spacing has been changed by this sequence, $\frac{1}{6}$ inch line feeds are made until line spacing is changed by code sequences ESC 0, ESC 1, ESC 3 n, ESC A n or ESC + n.

Note that changing the line spacing also changes the number of lines per page. Changing the line spacing does not affect the current tab stops or form length.

Example

```
10 REM    fix line spacing
20 LPRINT CHR$(27);"0"
30 LPRINT "these two lines are printed with"
40 LPRINT "a spacing of 1/8 inch"
50 LPRINT CHR$(27);"2"
60 LPRINT "and these two lines are printed with"
70 LPRINT "a spacing of 1/6 inch"
80 END
```

```
these two lines are printed with
a spacing of 1/8 inch

and these two lines are printed with
a spacing of 1/6 inch
```

Variable Line Spacing n/180 Inch or n/216 Inch	ESC 3
Variable Line Spacing n/60 Inch or n/72 Inch	ESC A
Variable Line Spacing n/360 Inch	ESC +

Data Structure

ASCII	ESC	"3"	<n>	set n/180 inch line spacing	24	❶
hex.	1B	33	<n>	set n/216 inch line spacing	9	
dec.	27	51	<n>			
ASCII	ESC	"A"	<n>	set n/60 inch line spacing	24	❷
hex.	1B	41	<n>	set n/72 inch line spacing	9	
dec.	27	65	<n>			
ASCII	ESC	"+"	<n>	set n/360 inch line spacing	24	❸
hex.	1B	2B	<n>			
dec.	27	43	<n>			

Description

❶ When the printer receives this code sequence it sets the line feed pitch to $^n/_{180}$ inches (24 needle version) or to $^n/_{216}$ inches (9 needle version). The value specified for n must be in the range of 0 to 255.

❷ This code sequence sets the line feed pitch to $^n/_{60}$ inch (24 needle version) or to $^n/_{72}$ inches (9 needle version). The value specified for n must be in the range of 1 to 127.

❸ This code sequence sets the line feed pitch to $^n/_{360}$ inch (24 needle version). The 9 needle printer ignores this sequence.
The value specified for n must be in the range of 1 to 127.

Once line spacing has been set with one of these sequences, an appropriate line feed is made whenever code LF is received until line spacing is changed with one of the code sequences ESC 0, ESC 1, ESC 2, ESC 3 n, ESC + n or ESC A n.

Example **24**

```
10 REM    variable line spacing
20 FOR N= 20 TO 40 STEP 10
30 LPRINT CHR$(27);"3";CHR$(N)
40 LPRINT "these two lines are printed with"
50 LPRINT "a spacing of";N;"/180 inches"
60 NEXT
70 END
```

```
these two lines are printed with
a spacing of 20 /180 inches

these two lines are printed with
a spacing of 30 /180 inches

these two lines are printed with
a spacing of 40 /180 inches
```

9

```
10 REM    variable line spacing
20 FOR N= 20 TO 40 STEP 10
30 LPRINT CHR$(27);"3";CHR$(N)
40 LPRINT "these two lines are printed with"
50 LPRINT "a spacing of";N;"/216 inches"
60 NEXT
70 END
```

```
these two lines are printed with
a spacing of 20 /216 inches

these two lines are printed with
a spacing of 30 /216 inches

these two lines are printed with
a spacing of 40 /216 inches
```

Character Styling

Super-/Subscript Mode ON	ESC S
Super-/Subscript Mode OFF	ESC T

Data Structure

ASCII	ESC	"S"	<n> or n	super/subscript mode ON	❶
hex.	1B	53	<n> or n		
dec.	27	83	<n> or n		
ASCII	ESC	"T"		super/subscript mode OFF	❷
hex.	1B	54			
dec.	27	84			

Description

❶ This code sequence switches the printer to superscript or subscript mode. The printer is switched to superscript mode when 0 (hex.00 or hex.30) is set for n, and to subscript mode when 1 (hex.01 or hex.31) is set. The subsequent characters are printed with about $2/3$ of their normal height, either in the upper or lower half of the corresponding character location.
If superscript or subscript characters are underlined, the underline character is printed at its normal position.
The characters hex.B0 to hex.DF (dec. 176...dec.223) and hex.F0 to hex.FE (dec.240...dec.254) of the extended EPSON graphics character set cannot be set to superscript or subscript mode.

❷ This code sequence resets the superscript or subscript mode to normal mode.

Example

```
10 REM    superscript and subscript
20  LPRINT "E=M*C";
30  LPRINT CHR$(27);"S";CHR$(0);
40  LPRINT "2";
50  LPRINT CHR$(27);"T";
60  LPRINT " is Einsteins most famous formula."
70  LPRINT "H";
80  LPRINT CHR$(27);"S";CHR$(1);"2";CHR$(27);"T";
90  LPRINT "O is simply water."
100 END
```

$E=M*C^2$ is Einsteins most famous formula.

H_2O is simply water.

Double Strike ON — ESC G
Double Strike OFF — ESC H

Data Structure

ASCII	ESC	"G"	double strike ON	❶
hex.	1B	47		
dec.	27	71		
ASCII	ESC	"H"	double strike OFF	❷
hex.	1B	48		
dec.	27	72		

Description

❶ This code sequence switches the printer to double strike mode. In double strike mode characters are printed in two printing passes at the same horizontal position; thus print appears thicker and fuller. Print speed is reduced since the characters are printed twice.
Double strike mode can be used in combination with emphasized mode (ESC E).
Double strike mode is reset when the printer receives the code sequence ESC H.

❷ This code sequence resets the double strike mode set by ESC G or ESC !.
This code sequence is ignored when the printer is not in double strike mode.

Example

```
10 REM    double strike
20 LPRINT CHR$(27);"H";
30 LPRINT "this is normal printing ";
40 LPRINT CHR$(27);"G";
50 LPRINT "and this is double strike printing"
60 END
```

this is normal printing **and this is double strike printing**

Select Font and Pitch by Point — ESC X

Data Structure

ASCII	ESC	"X"	<m> <n_L> <n_H>	ESC/P2 — select font and pitch by point
hex.	1B	58	<m> <n_L> <n_H>	
dec.	27	88	<m> <n_L> <n_H>	

Description

Valid values:

$5 \le m \le 127$ $m = 0.1$

$0 \le n_L \le 255$

$0 \le n_H \le 127$

Selects the pitch and point attributes for scalable fonts.

Pitch:

m = 0	No change in pitch
m = 1	Selects proportional spacing
$m \ge 5$	Selects fixed pitch equal to 360/m cpi
m = 36	default (10 cpi)

Points:

$$\text{(point size)} = \frac{(n_H \times 256) + n_L}{2} \text{ inch}$$

1 point $\triangleq$ 1⁄72 inch

$$n_H = \text{INT}\,\frac{\text{(point size)} \times 2}{256}$$

$$n_L = \text{MOD}\,\frac{\text{(point size)} \times 2}{256}$$

$n_H = n_L = 0$ No change in point size

$\left.\begin{array}{l} n_H = 0 \\ n_L = 21 \end{array}\right\}$ default (10.5 points)

STOP

Not all printer fonts are scaleable. Please refer to the printer's reference manual which fonts are supported. Scaleable fonts can only be printed in one quality (LQ).

Points:
Following point sizes (1/72 inch) are available:

8, 10 (10.5), 12, 14, 16, 18, 20 (21), 22, 24, 26, 28, 30, 32

Pitch:
Previously set pitch settings will be cancelled.

Following sequences are ignored in scaleable font mode:

ESC SI	condensed print
ESC SO	enlarged print mode (automatically reset after one line)
ESC SP	character spacing
ESC W	continuous enlarged print mode ON/OFF
ESC w	double height
SI	condensed print ON
SO	enlarged print mode (automatically reset after one line)

Following sequences will cancel the scaleable font mode:

ESC !	print mode selection
ESC @	printer initialization
ESC g	character density 15 cpi
ESC M	character density 12 cpi (Elite)
ESC P	character density 10 cpi (Pica)
ESC p	proportional spacing ON/OFF

Print Quality — ESC x

Data Structure

ASCII	ESC	"x"	<n> or n	set print quality (PQ)
hex.	1B	78	<n> or n	
dec.	27	120	<n> or n	

Description

n	Selected via control panel		
	DPQ	NLQ	LQ
0	DPQ	DPQ	DPQ
1	LQ	NLQ	LQ

Valid values for n are either ASCII Code 0 and 1 (hex00 and hex.01) or ASCII characters "0" and "1" (hex.30 and hex.31).

Example

```
10 REM   print quality
20 LPRINT CHR$(27);"x1";"This is high print quality "
30 LPRINT CHR$(27);"x0";"and this is draft print quality."
40 END
```

This is high print quality
and this is draft print quality.

Font Type Selection ESC k

Data Structure

ASCII	ESC	"k"	<n>
hex.	1B	6B	<n>
dec.	27	107	<n>

Description

Corresponding to the value specified for n, one of the printer’s internal fonts or a font of an optional font card is selected. Please refer to the printer’s reference manual which fonts are supported by your printer type.

This sequence is **not** valid in Draft Mode.

Example (depending on the printer)

0	Roman	11	Quadrato	
1	S_Serif	12	Kyrillic	
2	Courier	13	Arabic_I	
3	Prestige	14	Arabic_II	
4	Script	15	Farsi_I	
5	OCR-B	16	Farsi_II	
6	OCR-A			
7	Modern	20	Roman_T	ESC/P2
8	Kaufmann	21	S_Serif_H	ESC/P2
9	Gothic			
10	Helvetica	66	Courier_I	

These fonts can be selected also via the menu of the printer.
Roman_T and S_Serif_H are scaleable fonts.

Example

```
10  REM font selection
20  LPRINT CHR$(27);"x1"; : REM high print quality
30  FOR i=0 TO 9
40  LPRINT CHR$(27);"k";CHR$(i);
50  LPRINT "This is the font selected with ";i"
60  NEXT i
70  END
```

```
This is the font selected with  0
This is the font selected with  1
This is the font selected with  2
This is the font selected with  3
This is the font selected with  4
This is the font selected with  5
This is the font selected with  6
THIS IS THE FONT SELECTED WITH  7
This is the font selected with  8
This is the font selected with  9
```

Typestyle Selection — ESC q

Data Structure

ASCII	ESC	"q"	<n> or n	24
hex.	1B	71	<n> or n	
dec.	27	113	<n> or n	

Description

This sequence selects normal typestyle, outline typestyle, shaded typestyle or shaded outline typestyle. All the characters of the extended EPSON character set with the exception of the characters hex.B0 to hex.DF (dec.176...dec.233), hex.F4 (dec.244) and hex.F5 (dec.245) can be printed in the selected typestyle.

The following parameter assignment applies:

n	Typestyle
0 (hex.00 or hex.30)	Normal typestyle
1 (hex.01 or hex.31)	Outline typestyle
2 (hex.02 or hex.32)	Shaded typestyle
3 (hex.03 or hex.33)	Shaded Outline typestyle

Beispiel

```
10  REM character style
20  LF$=CHR$(10)
30  LPRINT CHR$(27);"q";CHR$(0);
40  LPRINT "<0> normal characters";LF$
50  LPRINT CHR$(27);"q";CHR$(1);
60  LPRINT "<1> outline characters";LF$
70  LPRINT CHR$(27);"q";CHR$(2);
80  LPRINT "<2> shadow characters";LF$
90  LPRINT CHR$(27);"q";CHR$(3);
100 LPRINT "<3> outline shadow characters";LF$
110 END
```

```
<0>  normal characters

<1>  outline characters

<2>  shadow characters

<3>  outline shadow characters
```

Emphasized Mode ON	ESC E
Emphasized Mode OFF	ESC F

Data Structure

ASCII	ESC	"E"	emphasized mode ON	❶
hex.	1B	45		
dec.	27	69		
ASCII	ESC	"F"	emphasized mode OFF	❷
hex.	1B	46		
dec.	27	70		

Description

❶ This code sequence switches the printer to emphasized mode. During printing in emphasized mode, each character is printed twice, slightly shifted in horizontal direction. Since the needles in the print head are fired twice (instead of once as in standard mode), print speed is reduced.
This code sequence is ignored if received while the printer is already in emphasized mode.

Emphasized mode plus condensed or Elite is not possible.

❷ This code sequence resets the emphasized mode initiated by ESC E or ESC ! (Master Select).
This code sequence is ignored if the printer is not in emphasized mode.

Example

```
10 REM    emphasized mode
20 LPRINT CHR$(27);"F";"this is printed in the normal mode
30 LPRINT CHR$(27);"E";"and this is in the emphasized on"
40 END
```

```
this is printed in the normal mode
and this is in the emphasized on
```

Underline Mode — ESC -

Data Structure

ASCII	ESC	"-"	<n> or n	underline mode ON/OFF
hex.	1B	2D	<n> or n	
dec.	27	45	<n> or n	

Description

This code sequence switches the automatic underline function ON or OFF. When the automatic underline function is ON, all printed characters including spaces are automatically underlined.

The automatic underline function is ON when 1 is specified for n, and is OFF when 0 is specified. Once the printer receives ESC -<1>, all subsequent characters including spaces are automatically underlined until ESC -<0> is received.

Spaces between tab codes (HT) will not be underlined.

The characters hex.B0 to hex.DF (hex.176...dec.223) and hex.F0 to hex.FE (dec.240...dec.254) of the extended EPSON graphics character set cannot be underlined using this mode.

Example

```
10 REM    underline mode
20 LPRINT "the most ";
30 LPRINT CHR$(27);"-";CHR$(1);
40 LPRINT "important";
50 LPRINT CHR$(27);"-";CHR$(0);
60 LPRINT " word must be underlined."
70 END
```

the most <u>important</u> word must be underlined.

Score Selection ESC (-

Data Structure

ASCII	ESC	"("	"-"	<3>	NUL	<1>	<n_1>	<n_2>
hex.	1B	28	2D	03	00	01	<n_1>	<n_2>
dec.	27	40	45	3	0	1	<n_1>	<n_2>

Description

This Escape sequence switches on and off the underscore, overscore and strike-through modes. The lines can be single or double, broken or continuous.

n_1 = loc
n_2 = type

loc (n_1)

The parameter n_1 indicates the location of the score to be printed.
The following values are avaiable:

1 (hex.01) = underscore mode
2 (hex.02) = strike-through mode
3 (hex.03) = overscore mode

type (n_2)

The parameter n_2 indicates the type of score to be printed.
The following values are avaiable:

0 (hex.00) = cancel scoring selected by n_1 (loc)
1 (hex.01) = single continuous line
2 (hex.02) = double continuous line
5 (hex.05) = single broken line
6 (hex.06) = double broken line

This command can be used more than once to activate the underscore, strike-through and overscore modes at the same time with different line types.

This mode cannot be used to underscore, strike-through or overscore block graphic characters.

Example

```
10  REM Score Selection
20  LPRINT CHR$(27);"(-";CHR$(3);CHR$(0);CHR$(1);CHR$(1);CHR$(2);
30  LPRINT "1. Underline - Double continous line"
40  LPRINT CHR$(27);"(-";CHR$(3);CHR$(0); CHR$(1);CHR$(2);CHR$(2);
50  LPRINT "2. Strike through and Underline - Double continous line"
60  REM cancel strike through - double continous line
70  LPRINT CHR$(27);"(-";CHR$(3);CHR$(0);CHR$(1);CHR$(2);CHR$(0);
80  LPRINT CHR$(27);"(-";CHR$(3);CHR$(0);CHR$(1);CHR$(2);CHR$(1);
90  LPRINT "3. Strike through - Single continous line and"
100 LPRINT "   Underline - Double continous line"
110 REM cancel underline - double continous line
120 LPRINT CHR$(27);"(-";CHR$(3);CHR$(0);CHR$(1);CHR$(1);CHR$(0);
130 LPRINT CHR$(27);"(-";CHR$(3);CHR$(0);CHR$(1);CHR$(3);CHR$(1);
140 LPRINT "4. Overscore and strike through - Single continous line"
150 END
```

1. Underline - Double continous line

~~**2. Strike through and Underline - Double continous line**~~

~~**3. Strike through - Single continous line and**~~
~~**Underline - Double continous line**~~

~~**4. Overscore and strike through - Single continous line**~~

Italic Mode ON — ESC 4
Italic Mode OFF — ESC 5

Data Structure

ASCII	ESC	"4"	italic mode ON	❶
hex.	1B	34		
dec.	27	52		
ASCII	ESC	"5"	italic mode OFF	❷
hex.	1B	35		
dec.	27	53		

Description

❶ All characters following this code sequence are printed in italics. Italic characters are also printed, if codes with the eight bit set by ESC > are sent to the printer.

> The characters hex.B0 to hex.DF (hex.176...dec.223) and hex.F0 to hex.FE (dec.240...dec.254) of the extended EPSON graphics character set cannot be printed in italic mode.

❷ Italic print mode set by ESC 4 or ESC ! (Master Select) is reset. Italic mode can be used in all print qualities (DPQ, NLQ, LQ).

Example

```
10  REM    italic mode
20  LPRINT CHR$(27);"4";
30  GOSUB 70
40  LPRINT CHR$(27);"5";
50  GOSUB 70
60  END
70  LPRINT "Matrix Printer"
80  RETURN
```

Matrix Printer
Matrix Printer

Print Mode Selection — ESC !

Data Structure

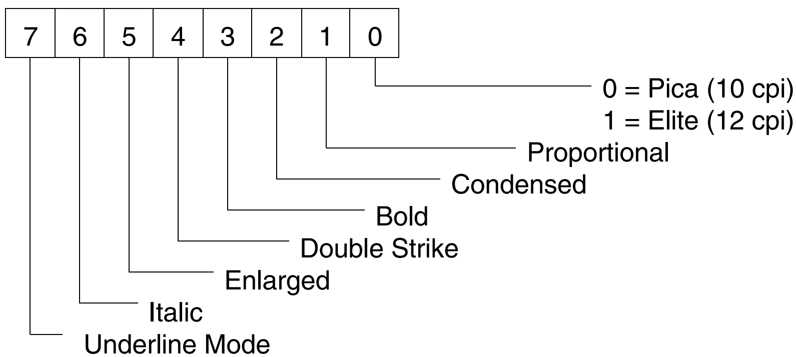
ASCII	ESC	"!"	<n>
hex.	1B	21	<n>
dec.	27	33	<n>

Description

This code sequence allows simultaneous control of print features like the character size and attributes characters. By this code sequence several print features can be simultaneously set via the value of n. This command can also be used to reset all character styling features or to reset the printer to standard Pica font type by specifying n=0.

The value specified for n must be in the range of 0 to 255. See the following table for possible values of n.

Print mode selection byte:



		Standard	Condensed	Enlarged	Cond. enlarg.	Proportional	Prop.-/ enlarg.	LQ	LQ enlarged
Pica	- normal	0	4	32	36	2	34	0	32
	- emphas.	8	–	40	–	18	50	8	40
	- double	16	20	48	52	–	–	–	–
	- em./double	24	–	56	–	–	–	–	–
Pica italic	- normal	64	68	96	100	66	98	–	–
	- emphas.	72	–	104	–	82	114	–	–
	- double	80	84	112	116	–	–	–	–
	- em./double	88	–	120	–	–	–	–	–
Pica underlined	- normal	128	132	160	164	130	162	128	160
	- emphas.	136	–	168	–	146	178	136	168
	- double	144	148	176	180	–	–	–	–
	- em./double	152	–	184	–	–	–	–	–
Pica italic underlined	- normal	192	196	224	228	194	226	–	–
	- emphas.	200	–	232	–	210	242	–	–
	- double	208	212	240	244	–	–	–	–
	- em./double	216	–	248	–	–	–	–	–
Elite	- normal	1	5	33	37	–	–	–	–
	- double	17	21	49	53	–	–	–	–
Elite italic	- normal	65	69	97	101	–	–	–	–
	- double	81	85	113	117	–	–	–	–
Elite underlined	- normal	129	133	161	165	–	–	–	–
	- double	145	149	177	181	–	–	–	–
Elite italic underlined	- normal	193	197	225	229	–	–	–	–
	- double	209	213	241	245	–	–	–	–

Example

```
10 REM    print mode selection
20 LPRINT CHR$(27);"D";CHR$(40);CHR$(0);
30 FOR N = 0 TO 255 STEP 7
40 LPRINT CHR$(27);"!";CHR$(N);
50 LPRINT "Matrix Printer";CHR$(9);CHR$(27);"!";CHR$(0);N
60 NEXT N
70 END
```

```
Matrix Printer        0
Matrix Printer        7
Matrix Printer        14
Matrix Printer        21
Matrix Printer        28
Matrix Printer        35
Matrix Printer        42
Matrix Printer        49
Matrix Printer        56
Matrix Printer        63
Matrix Printer        70
Matrix Printer        77
Matrix Printer        84
Matrix Printer        91
Matrix Printer        98
Matrix Printer        105
Matrix Printer        112
Matrix Printer        119
Matrix Printer        126
Matrix Printer        133
Matrix Printer        140
Matrix Printer        147
Matrix Printer        154
Matrix Printer        161
Matrix Printer        168
Matrix Printer        175
Matrix Printer        182
Matrix Printer        189
Matrix Printer        196
Matrix Printer        203
Matrix Printer        210
Matrix Printer        217
Matrix Printer        224
Matrix Printer        231
Matrix Printer        238
Matrix Printer        245
Matrix Printer        252
```

Double Height	ESC w

Data Structure

ASCII	ESC	"w"	<n>	double height ON/OFF
hex.	1B	77	<n> or n	
dec.	27	119	<n> or n	

Description

By means of this code sequence the characters of the subsequent text are specified with double height.

ESC w followed by n=1 (hex.01 or hex.31) sets double height printing. To reset the double height to the normal height, specify n=0 (hex.00 oder hex.30).
The line spacing is not altered. This command cannot be combined with superscript/subscript or condensed print commands.

ESC w with parameter n=0 (hex.00 or hex.30) resets character representation back to normal character height.

Example

```
10  REM    double height
20  LPRINT "for ";
20  LPRINT CHR$(27);"w";CHR$(1);
30  LPRINT "Headlines ";
40  LPRINT CHR$(27);"w";CHR$(0);
50  LPRINT "double height printing is recommended"
```

for Headlines double height printing is recommended

Print Positioning

Horizontal Tab Stops — ESC D

Data Structure

ASCII	ESC	"D"	$<n_1> ...<n_{32}>$	NUL	set horizontal tab stops	❶
hex.	1B	44	$<n_1> ...<n_{32}>$	00		
dec.	27	68	$<n_1> ...<n_{32}>$	0		
ASCII	ESC	"D"	NUL		clear all horizontal tab stops	❷
hex.	1B	44	00			
dec.	27	68	0			

Description

❶ This code sequence is used for setting or clearing horizontal tab stops. It replaces all previous tab stops and can be used to set a maximum of 32 tab stops.

For example,

CHR$(27);"D";CHR$(1);CHR$(11);CHR$(21);CHR$(0);

sets horizontal tab stops at positions 1, 11, and 21, position 1 being the position of the first character at the left margin.

Double width printing does not influence the physical positions of tab stops.

Tab stops are set in those positions which are specified by character codes following ESC D. These character codes must be arranged in ascending order.

The length of this code sequence varies according to the number of tab stops being set. Code hex.00 is used as a terminator to signal the end of the data string.

If a character code with a smaller numeric value than the preceding is among those codes, the code with the smaller value is ignored. Horizontal tab stops can be set at positions 1 to 255 (136 columns printer) resp. 1 to 160 (80 columns printer). Horizontal tab stops set by this sequence remain effective until the printer again receives code sequence ESC D or until it is initialized by code sequence ESC @.

For proportional spacing, tab stops are set in pica spacing (10 cpi).

A horizontal tabulation is executed by HT control code (hex.09, dec.9).

❷ This code sequence cancels all previous tab stops.

When the printer is switched on or ESC @ is sent, horizontal tab stops are set at every eight position (9, 17, 25, etc.).

Example

```
10  REM    horizontal tabs
20  LPRINT CHR$(27);"D";CHR$(0) :REM clear old tabs
30  LPRINT CHR$(27);"D";CHR$(10);CHR$(20);CHR$(30);CHR$(0);
40  LPRINT CHR$(18);:REM 10 cpi
50  LPRINT "123456789012345678901234567890123456789"
60  GOSUB 120
70  LPRINT CHR$(14);:REM double width
80  GOSUB 120
90  LPRINT CHR$(15);:REM condensed
100 GOSUB 120
110 END
120 LPRINT CHR$(9);"tab1";
130 LPRINT CHR$(9);"tab2";
140 LPRINT CHR$(9);"tab3"
150 RETURN
```

```
123456789012345678901234567890123456789
          tab1      tab2      tab3
          t a b 1   t a b 2   t a b 3
          tab1      tab2      tab3
```

Set Horizontal and Vertical Step Width	ESC e

Data Structure

ASCII	ESC	"e"	NUL	<n>	Set horizontal step width	❶
hex.	1B	65	00	<n>		
dec.	27	101	0	<n>		
ASCII	ESC	"e"	SOH	<n>	Set vertical step width	❷
hex.	1B	65	01	<n>		
dec.	27	101	1	<n>		

Description

❶ This sequence sets the horizontal step width n according to the current character density. When an HT command is received, a horizontal tab stop is carried out.

The following maximum values apply for n:

21 (Pica)
25 (Elite)
36 (narrow Pica)
42 (narrow Elite)

The horizontal tab stop positions are set relative to the left margin.

❷ This sequence defines the vertical step width n as the number of line feeds corresponding to the line spacing set for LF. On receipt of a VT command, a vertical tab stop is carried out.

The following maximum values apply for n:

9 $n \leq 127$
24 $n \leq 255$

Horizontal and Vertical Step Forward — ESC f

Data Structure

ASCII	ESC	"f"	NUL	<n>	Set horizontal step forward	❶
hex.	1B	66	00	<n>		
dec.	27	102	0	<n>		
ASCII	ESC	"f"	SOH	<n>	Set vertical step forward	❷
hex.	1B	66	01	<n>		
dec.	27	102	1	<n>		

Description

❶ At the next print position n spaces without carriage return (CR) are inserted. As soon as this command is received, all data in the buffer are printed. This command is ignored when justification is activated. If the next print position is on or beyond the right margin, a line feed is carried out and printing starts at the left margin.

❷ On receipt of this sequence the printer carries out n line feeds corresponding to the current line spacing. A carriage return (CR) is not carried out. As soon as this command is received, all data in the buffer are printed.

Absolute Horizontal Step **ESC $**

Data Structure	ASCII	ESC	"$"	<n_1> <n_2>	set absolute horizontal step
	hex.	1B	24	<n_1> <n_2>	
	dec.	27	36	<n_1> <n_2>	

Description

This code sequence moves the print head to an absolute print position independent of the current character density. The positioning is in increments of $1/60$ inches. The print position from the left hand margin is then calculated as: ($n_1 + n_2 * 256$).

The value specified for n_1 must be in the range of 0 to 255, the value specified for n_2 must be in the range of 0 to 3.

This code sequence can be used in all print qualities (DPQ, NLQ, LQ).

Example for calculating n_1 and n_2:

The print position is 4.5 inches, which corresponds to a number of dots of 270 (4.5 inch x 60 dots/inch = 270 dots).

n_2 = number of dots divided by 256
= $270/256$
= dec.1
= hex.01

n_1 = remainder of division of n_2
= dec.14
= hex.0E

Note that you obtain the same result by using the formula given for calculating n_1 and n_2 for relative horizontal step (see sequence ESC \). Only the way to get it is different. The reason for these different descriptions is that customers' programming experience is not the same. Therefore this is meant to be a little help for you to choose the easiest way.

Example

```
10  REM  absolute horizontal position
20  LPRINT CHR$(27);"x1";:REM select LQ
30  LPRINT CHR$(27);"$";CHR$(10);CHR$(0);"10";
40  LPRINT CHR$(27);"$";CHR$(100);CHR$(0);"100";
50  LPRINT CHR$(27);"$";CHR$(170);CHR$(0);"170";
60  LPRINT
70  LPRINT CHR$(27);"W1";
80  LPRINT CHR$(27);"$";CHR$(10);CHR$(0);"10";
90  LPRINT CHR$(27);"$";CHR$(100);CHR$(0);"100";
100 LPRINT CHR$(27);"$";CHR$(170);CHR$(0);"170";
110 LPRINT
120 END
```

```
10                  100            170
1 0                 1 0 0          1 7 0
```

Relative Horizontal Step ESC \

Data Structure

ASCII	ESC	"\"	$<n_1>$ $<n_2>$	set relative horizontal step
hex.	1B	5C	$<n_1>$ $<n_2>$	
dec.	27	92	$<n_1>$ $<n_2>$	

Description

This code sequence moves the print head to a relative print position without regard to the current character density. This print position is calculated according to the formula ($n_1 + 256 * n_2$).Each dot, i. e. each print position has a width of $^1/_{120}$ inch in DPQ and $^1/_{180}$ inch in NLQ and LQ.

Calculation of the values of n_1 and n_2 first requires the prefix (n dots) to be calculated. If the movement is directed towards the left side, the result is subtracted from 65536. Subsequently the values for n_1 and n_2 can be calculated according to the formula:

n_1 = n MOD 256
n_2 = INT($^n/_{256}$)

If the print position exceeds the currently set margins after execution of the horizontal step, this code sequence is ignored.

This code sequence can be used in all print qualities (DPQ, NLQ, LQ).

Example for calculating n_1 and n_2:

Movement of 4 inches towards the right side:

n = $4 * 120$
= 480

n_2 = INT ($^{480}/_{256}$)
= dec.1
= hex.01

$$n_1 = 480 \text{ MOD } 256 = \text{dec.}224 = \text{hex.E0}$$

Movement of $5/3$ inches (1.67 inches) towards the left side:

$$n = 65536 - 1.67 * 120 = 65336$$

$$n_2 = \text{INT}\left(65336/256\right) = \text{dec.}255 = \text{hex.FF}$$

$$n_1 = 65336 \text{ MOD } 256 = \text{dec.}56 = \text{hex.}38$$

Note that you obtain the same result by using the formula given for calculating n_1 and n_2 for absolute horizontal step (see sequence ESC $). Only the method of calculations is different. The two different methods give the same result and therefore the programmer should use the method best suited for his application.

Example

```
10  REM    relative horizontal position
20  LPRINT CHR$(27);"\";CHR$(224);CHR$(1);"+480";
30  LPRINT CHR$(27);"\";CHR$(56);CHR$(255);"-200";
40  LPRINT CHR$(27);"\";CHR$(50);CHR$(0);"+50";
50  LPRINT
60  END
```

```
                          -200     +50  +480
```

Vertical Tab Stops — ESC B

Data Structure

ASCII	ESC	"B"	$<n_1>...<n_{16}>$	NUL	set vertical tab stops	❶
hex.	1B	42	$<n_1>...<n_{16}>$	00		
dec.	27	66	$<n_1>...<n_{16}>$	0		
ASCII	ESC	"B"	NUL		reset all vertical tab stops	❷
hex.	1B	42	00			
dec.	27	66	0			

Description

❶ By means of this code sequence a maximum of 16 vertical tab stops can be set.

For example,

CHR$(27);"B";CHR$(1);CHR$(11);CHR$(21);CHR$(0);

sets vertical tab stops at lines 1, 11 and 21, line 1 being the first line of the page.

Tab stops are set in those positions which are specified by character codes following ESC B. These character codes must be arranged in ascending order.

The length of this code sequence varies according to the number of tab stops being set. Code <NUL> (hex. 00) is used as a terminator to signal the end of the data string.

If any character code with a smaller numeric value than the preceding is among those codes, the code with the smaller value is ignored. Vertical tab stops set by this sequence remain effective until code sequence ESC B is received again or form length is changed by ESC C.

A vertical tabulation is executed by the VT control code (hex.0B, dec.11).

❷ This code sequence cancels all previous tab stops.

For example,

CHR$(27);"B";CHR$(0)

clears all vertical tab stops.

When the printer is switched on no vertical tab stops are set.

Example

```
10  REM    vertical tabs
20  LPRINT CHR$(27);"C";CHR$(10);:REM set form length
30  LPRINT CHR$(27);"B";CHR$(0);:REM clear all vtabs
40  LPRINT CHR$(27);"B";CHR$(3);CHR$(5);CHR$(9);CHR$(0);
50  LPRINT CHR$(11);"this is the first  vertical tab, line 3"
60  LPRINT CHR$(11);"this is the second vertical tab, line 5"
70  LPRINT CHR$(11);"this is the third  vertical tab, line 9"
80  END
```

```
this is the first  vertical tab, line 3

this is the second vertical tab, line 5



this is the third  vertical tab, line 9
```

Set/Reset Vertical Tabs in Channels Select Vertical Tab Channel	ESC b ESC /

Data Structure

ASCII	ESC	"b"	<m>	<n_1>...<n_{16}> NUL	set vertical tabs in channels	❶
hex.	1B	62	<m>	<n1>...<n16> 00		
dec.	27	98	<m>	<n1>...<n16> 0		
ASCII	ESC	"b"	<m>	NUL	clear all tab settings	
hex.	1B	62	<m>	00		
dec.	27	98	<m>	0		
ASCII	ESC	"/"	<m>		select vertical tab channel	❷
hex.	1B	2F	<m>			
dec.	27	47	<m>			

Description

❶ Up to 16 vertical tabs for one channel <m> can be set. the number of tab channels m ranges from 0 to 7. The value of n, given in lines, ranges from 1 to 255.
Channel 0 can be set via ESC B. The default value is m=0.

❷ This command is used to select one of the eight vertical tab channels. The value of m ranges from 0 to 7.

Example

```
10  REM vertical tabs in tab channels
20  LPRINT CHR$(27);"C";CHR$(24);   'form length 24 lines
30  LPRINT CHR$(27);"b";CHR$(0);CHR$(2);CHR$(4);CHR$(6);CHR$(0)
40  REM Channels 0 tabs set at lines 2, 4, and 6
50  LPRINT CHR$(27);"b";CHR$(1);CHR$(9);CHR$(11);CHR$(14);CHR$(0);
60  REM Channel 1 tabs set at lines 9, 11, and 14
70  LPRINT CHR$(27);"/";CHR$(0);
80  REM Selects channel 0
90  LPRINT "Channel 0"
100 GOSUB 180
110 LPRINT CHR$(27);"/";CHR$(1);
120 REM Selects channel 1
130 LPRINT "Channel 1"
140 GOSUB 180
150 END
160 FOR I=1 TO 3
170 LPRINT CHR$(11);
180 LPRINT "VT Position #";I
190 NEXT
200 RETURN
```

```
Channel 0
VT Position # 1

VT Position # 2

VT Position # 3
Channel 1

VT Position # 1

VT Position # 2


VT Position # 3
```

Set Absolute Vertical Print Position **ESC (V**

Data Structure

ASCII	ESC	"("	"V"	$<n_L>$ $<n_H>$ $<m_L>$ $<m_H>$	ESC/P2 set absolute vertical print pos.
hex.	1B	28	56	$<n_L>$ $<n_H>$ $<m_L>$ $<m_H>$	
dec.	27	40	86	$<n_L>$ $<n_H>$ $<m_L>$ $<m_H>$	

Description

Valid values:

$n_L = 2$, $n_H = 0$

$0 \leq m_L \leq 255$; $0 \leq m_H \leq 127$

Defines the vertical print position in units previously defined with the sequence ESC (U (see page 62).

(vertical position) = $((m_H \times 256) + m_L)$ x (defined unit) + (top-margin position)

$$m_H = \mathrm{INT}\left(\frac{((\text{vertical position}) - (\text{top–margin position})) \times \frac{1}{(\text{defined unit})}}{256}\right)$$

$$m_L = \mathrm{MOD}\left(\frac{((\text{vertical position}) - (\text{top–margin position})) \times \frac{1}{(\text{defined unit})}}{256}\right)$$

The default unit for this sequence is $1/360$".
Only positions within a valid form can be achieved.
Positions below bottom margin will
a) eject paper (single sheet) and
b) set new position to top of form at next page.

Set Relative Vertical Print Position — ESC (v

Data Structure

ASCII	ESC	"("	"v"	<n_L> <n_H> <m_L><m_H>	ESC/P2 set relative vertical print pos.
hex.	1B	28	76	<n_L> <n_H> <m_L><m_H>	
dec.	27	40	118	<n_L> <n_H> <m_L><m_H>	

Description

Valid values:

$n_L = 2,\ n_H = 0$

$0 \le m_L \le 255;\ 0 \le m_H \le 127$

Defines the vertical print position relatively up and down in units previously defined with the sequence ESC (U (see page 62).

(horizontal position) = ((m_H x 256) + m_L) x (defined unit) + (current position)

Downwards:

$$m_H = \mathrm{INT}\left(\frac{((\text{vertical position}) - (\text{current position})) \times \dfrac{1}{(\text{defined unit})}}{256}\right)$$

$$m_L = \mathrm{MOD}\left(\frac{((\text{vertical position}) - (\text{current position})) \times \dfrac{1}{(\text{defined unit})}}{256}\right)$$

Upwards:

$$m_H = 32768 - \mathrm{INT}\left(\frac{((\text{current position}) - (\text{vertical position})) \times \dfrac{1}{(\text{defined unit})}}{256}\right)$$

$$m_L = 32768 - \mathrm{MOD}\left(\frac{((\text{current position}) - (\text{vertical position})) \times \dfrac{1}{(\text{defined unit})}}{256}\right)$$

The default unit for this sequence is $^1\!/_{360}$".
Only positions within a valid form can be achieved.
The current position is base for the newly defined relative positions.
Positions exceeding the defined top margin will be ignored.
Positions below bottom margin will
a) eject paper (single sheet) and
b) set new position to top of form at next page.

Set Unit ESC (U

Data Structure

ASCII	ESC	"("	"U"	<n_L> <n_H> <m>	ESC/P2	set unit
hex.	1B	28	55	<n_L> <n_H> <m>		
dec.	27	40	85	<n_L> <n_H> <m>		

Description

Valid values:

$n_L = 1$, $n_H = 0$

m = 10, 20, 30, 40, 50, 60

Sets the units to $m/3600$ inch.

Following commands are using this unit:
ESC (V
ESC (v
ESC (C
ESC (c

The default values of these sequences is defined at $1/360$ inch.

Graphics

Graphics 60 dpi	**ESC K**
Graphics 120 dpi	**ESC L**
High Speed Graphics 120 dpi	**ESC Y**
Graphics 240 dpi	**ESC Z**
Graphic Modes	**ESC ***

Data Structure	ASCII	ESC	"K"	<n_1>	<n_2>		graphics 60 dpi
	hex.	1B	4B	<n_1>	<n_2>		
	dec.	27	75	<n_1>	<n_2>		
	ASCII	ESC	"L"	<n_1>	<n_2>		graphics 120 dpi
	hex.	1B	4C	<n_1>	<n_2>		
	dec.	27	76	<n_1>	<n_2>		
	ASCII	ESC	"Y"	<n_1>	<n_2>		high speed graphics 120 dpi
	hex.	1B	59	<n_1>	<n_2>		
	dec.	27	89	<n_1>	<n_2>		
	ASCII	ESC	"Z"	<n_1>	<n_2>		graphics 240 dpi
	hex.	1B	5A	<n_1>	<n_2>		
	dec.	27	90	<n_1>	<n_2>		
	ASCII	ESC	"*"	<0>	<n_1>	<n_2>	graphic modes
	hex.	1B	2A	00	<n_1>	<n_2>	
	dec.	27	42	0	<n_1>	<n_2>	

Description

Data following the graphics ESC-sequence is printed out as a dot pattern. The number of bytes of the graphics string is defined by n_1 and n_2. You will find a list of all sequences and the available parameter values m for the ESC * code sequences on the following page.

Special 8 Point Sequences	ESC * <m> ... -Sequences	Print Mode	Density in dpi (Dots per Inch)		Needles in use	Adja-cent Dots
			horizont.	vertic.		
ESC K...	ESC * <0>...	single density	60	72 9 60 24	8	possible
ESC L...	ESC * <1>...	double density	120		8	possible
ESC Y...	ESC * <2>...	double density/ high speed	120		8	not possible
ESC Z...	ESC * <3>...	quadruple density	240		8	not possible
–	ESC * <4>...	screen graphics	80		8	possible
–	ESC * <5>... 9	plot (1:1)	72		8	possible
–	ESC * <6>...	screen graphics II	90		8	possible
–	ESC * <7>... 9	plot (double density)	144		8	possible
–	ESC * <32>... 24	single density	60	180	24	possible
–	ESC * <33>... 24	double density	120		24	possible
–	ESC * <38>... 24	screen graphics II	90		24	possible
–	ESC * <39>... 24	triple density	180		24	possible
–	ESC * <40>... 24	sixfold density	360		24	not possible

Example for Calculation of n_1 and n_2:

The number of graphics data columns is 80.

n_2 = number of data divided by 256
= $^{80}/_{256}$
= dec.0
= hex.00

n_1 = rest of division of n_2
= dec.80
= hex.50

Relationship between bits of graphics data and needles (9 needle version)

Each needle in a column is assigned a bit of the data byte.

Needle		Data Byte 1
1	●	Bit 7
2	●	Bit 6
3	●	Bit 5
4	●	Bit 4
5	●	Bit 3
6	●	Bit 2
7	●	Bit 1
8	●	Bit 0

Relationship between bits of graphics data and needles (24 needle version)

When using 24-pin graphics modes, three bytes of data are required for each dot column.

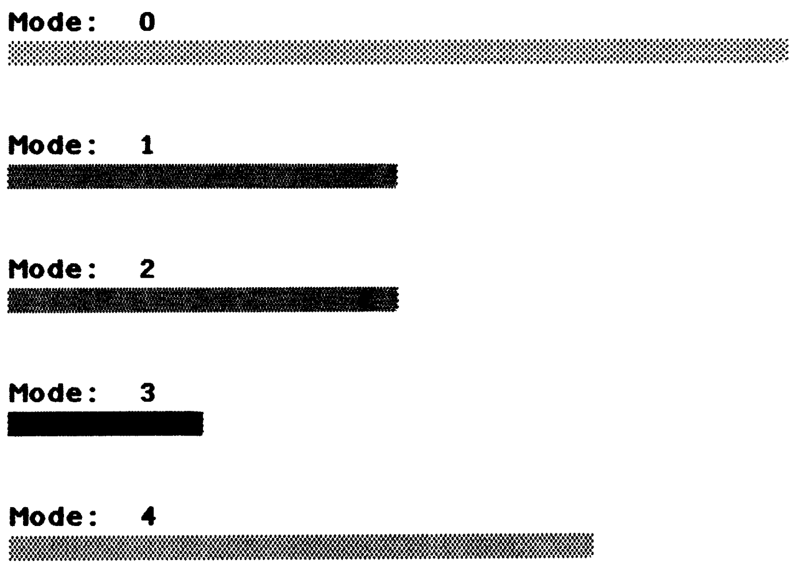
Needle		Data Byte 1	Needle		Data Byte 2	Needle		Data Byte 3
1	●	Bit 7	9	●	Bit 7	17	●	Bit 7
2	●	Bit 6	10	●	Bit 6	18	●	Bit 6
3	●	Bit 5	11	●	Bit 5	19	●	Bit 5
4	●	Bit 4	12	●	Bit 4	20	●	Bit 4
5	●	Bit 3	13	●	Bit 3	21	●	Bit 3
6	●	Bit 2	14	●	Bit 2	22	●	Bit 2
7	●	Bit 1	15	●	Bit 1	23	●	Bit 1
8	●	Bit 0	16	●	Bit 0	24	●	Bit 0

Graphics and text can be mixed within one line.

If image data, lying outside the printable area, are input these data are ignored.

Example

```
10  REM bit image print
20  WIDTH "LPT1:",255
30  FOR i=0 TO 4:
40    LPRINT "Mode: ";STR$(i)
50    LPRINT CHR$(27);"*";CHR$(i);CHR$(254);CHR$(0);
60    FOR j=1 TO 127
70    LPRINT STRING$(1,64+16+4+1);STRING$(1,128+32+8+2);
80    NEXT j
90    LPRINT CHR$(10);CHR$(13)
100 NEXT i
110 END
```



Select Graphics Mode	ESC (G

Data Structure

ASCII	ESC	"("	"G"	<n_L> <n_H> <m>	ESC/P2	select graphics mode
hex.	1B	28	47	<n_L> <n_H> <m>		
dec.	27	40	71	<n_L> <n_H> <m>		

Description

Valid values:

$n_L = 1$

$n_H = 0$

m = 1, 49

Enters graphics mode. Prepares printer for the sequence ESC **.** (see page 76).

Following sequences are valid after entering graphics mode:

ESC (c	Set page format
ESC (C	Set page length in units
ESC (V	Set absolute vertical print position
ESC (v	Set relative vertical print position
ESC \	Relative horizontal step
ESC $	Absolute horizontal step
ESC r	Select printing color
ESC U	Set/reset unidirectional printing
ESC +	Variable line spacing n/360 inch
ESC (U	Set unit
ESC EM	Sheet feeder
ESC @	Printer initialization
ESC .	Print raster graphics
LF	Line feed
FF	Form feed
CR	Carriage return

The graphics mode can be exited with sequence ESC @ (see page 114).

In this mode text and graphics cannot be mixed on the same page.
This sequence does not allow text and user defined character printing.
Vertical and horizontal tab settings are cleared.

Graphics Mode Reassignment **ESC ?**

Data Structure

ASCII	ESC	"?"	<s>	<n>	reassign graphics mode
hex.	1B	3F	<s>	<n>	
dec.	27	63	<s>	<n>	

Description

This code sequence redefines one of the four alternate bit image codes ESC K, ESC L, ESC Y or ESC Z as one of the bit density numbers used with the ESC * <m>... code sequence.

Available values are:

s = 75, 76, 89, 90 (ASCII = K, L, Y, Z)

n = 0	single density	
n = 1	double density	
n = 2	double density, high speed	
n = 3	quadruple density	
n = 4	screen graphics	
n = 5	plot graphics (1:1)	9
n = 6	screen graphics II	
n = 7	plot graphics, double density	9
n = 32	single density	24
n = 33	double density	24
n = 38	screen graphics II	24
n = 39	triple density	24
n = 40	sixfold density	24

For example, if you send the following code before you run a graphics program, it will change every instance of mode K (single density) to mode 3 (quadruple density):

```
LPRINT CHR$(27);"?K";CHR$(3);
```

Example

```
10  REM Reassigns graphics sequence
20  LPRINT CHR$(27);"K";CHR$(60);CHR$(0);
30  REM Standard ESC K graphics sequence
40  FOR X=1 TO 60:REM    60 Columns
50  LPRINT CHR$(255);:REM    one byte per columns
60  NEXT X:LPRINT
70  LPRINT CHR$(27)"?K";CHR$(3)
80  REM Reassigns ESC K to quadruple density
90  LPRINT CHR$(27)"K";CHR$(60);CHR$(0);
100 FOR Z=1 TO 60:REM    60 Columns
110 LPRINT CHR$(255);CHR$(255);CHR$(255);
130 NEXT Z
140 END
```

Graphics Printing with 9 Print Dots ESC ^

Data Structure	ASCII	ESC	"^"	$<m>$	$<n_1>$	$<n_2>$
	hex.	1B	5E	$<m>$	$<n_1>$	$<n_2>$
	dec.	27	94	$<m>$	$<n_1>$	$<n_2>$

Description

With this Escape sequence, graphics printing with 9 print dots per dot row is selected. For each row of dots two graphics bytes are required. The first byte controls the upper eight print dots and the second controls only the lowest print dot.

m = 0 selects normal graphics resolution (60 dpi).
m = 1 selects double graphics resolution (120 dpi).

The data following the graphics ESC sequence are printed as a dot pattern. The parameters n1 and n2 define the number of graphics bytes.

9

Print Head			Data Byte	Value	
Needle	1	●	Bit 7	128	Byte 1
	2	●	Bit 6	64	
	3	●	Bit 5	32	
	4	●	Bit 4	16	
	5	●	Bit 3	8	
	6	●	Bit 2	4	
	7	●	Bit 1	2	
	8	●	Bit 0	1	
	9	●	Bit 7	128	Byte 2

Example for the calculation of n1 and n2:

The number of graphics bytes is 80.

n_2	=	number of bytes divided by 256
	=	$^{80}/_{256}$
	=	dec.0
	=	hex.00
n_1	=	remainder from division of n_2
	=	dec.80
	=	hex.50

Graphics and text data can be used in combination in a line.

For 9-print-dots-graphics-printing, two graphics bytes must be transferred per column.

Download Character Generator

General

Your printer offers you the possibility to design your own characters in normal print mode (DPQ), in NLQ, NLQ proportional, LQ and LQ proportional. These characters can either represent modified characters of the printer's internal character sets or can replace other characters.

These special characters, symbols, etc., can be defined in the Download Character Generator and then called "download characters" (DLL). You can define and store in the printers memory a maximum of 96 DLL characters.

	00	10	20	30	40	50	60	70
00								
01								
02		DC2						
03		DC3						
04		DC4						
05								
06								
07	BEL							
08	BS	CAN						
09	HT							
0A	LF							
0B	VT	ESC						
0C	FF							
0D	CR							
0E	SO							
0F	SI							DEL

printable characters

If only single characters of an internal character set should be changed, it is possible to copy the internal character set to the Download Character Generator and redefine these single characters.

After power switch-off or initialization of the printer, the contents of the Download Character Generator is lost. Download characters are possible in draft print quality (DPQ), near letter quality (NLQ) and in letter quality (LQ).

To define download characters it is helpful to do this in a defined way:

- Draft print quality or letter quality or letter quality proportional characters?
- Superscript or subscript character, 12 cpi or 15 cpi character (DPQ + LQ)?

The desired print mode (DPQ or LQ or LQ proportional) must be selected in the printer.

- Load characters from ROM to RAM using ESC : if necessary.
- Define the new character in the RAM using ESC &.
- Select the RAM character set using ESC %.

Copy ROM into RAM **ESC :**

Data Structure

ASCII	ESC	":"	NUL	<n>	NUL	copy ROM CG
hex.	1B	3A	00	<n>	00	
dec.	27	58	0	<n>	0	

Description

This Escape sequence copies the specified character set (see ESC k <n>) from ROM or from the font module to RAM. There the individual characters can be altered by the user.

Example

```
10 REM    Copy ROM CG into Download CG
20 LPRINT CHR$(27);":";CHR$(0);CHR$(0);CHR$(0);
30 REM Select Download CG
40 LPRINT CHR$(27);"%";CHR$(1);CHR$(0);
50 LPRINT "[ \ ] ` { | } ~"
60 END
```

```
[ \ ] ` { ¦ } ~
```

Print Raster Graphics **ESC .**

Data Structure

											ESC/P2
ASCII	ESC	"**.**"	<c>	<v>	<h>	<m>	<n_L>	<n_H>	<d_1>...<d_x>		
hex.	1B	2E	<c>	<v>	<h>	<m>	<n_L>	<n_H>	<d_1>...<d_x>		
dec.	27	46	<c>	<v>	<h>	<m>	<n_L>	<n_H>	<d_1>...<d_x>		

Description

Valid values:

c = 0
c = 1
v = 10, 20
h = 10, 20
m = 1, 8, 24

$0 \le n_L \le 255$
$0 \le n_H \le 127$
$0 \le d \le 255$

This sequence supports following print densities:

v	h	v (dpi)	h (dpi)	m
20	20	180	180	1, 8 or 24
20	20	180	360	1, 8 or 24
10	10	360	360	1, 8 or 24

"ESC **.**" allows printing dot graphics in raster format. A compression algorithm is provided for multiple printing of a particular byte of data.

Description of parameters:

c = 0	Normal mode (not compressed)
c = 1	Data Length Encoding (DLE, compressed)
v	Vertical resolution in dpi: 360, 180 dpi
h	Horizontal resolution in dpi: 360, 180 dpi
m	Vertical dot count (rows of dot graphics)
n_L, n_H	Horizontal dot count (columns of dot graphics)

Calculating algorithm:

$$n_H = \mathrm{INT}\,\frac{(\text{horizontal dot count})}{256}$$

$$n_L = \mathrm{MOD}\,\frac{(\text{horizontal dot count})}{256}$$

x Total number of data bytes

$$x = m \times \mathrm{INT}\,\frac{(n_H \times 256) + n_L + 7}{8}$$

d During normal mode (c=0)

Graphics data
d_1 ... d_x

d During DLE (c=1)

The first data byte is treated as a counter. Graphics data bytes then alternate with a data counter byte:

$0 \leq$ (counter byte) ≤ 127

“a” Counter specifies the number of data bytes following:

(counter byte) + 1 = (number of data bytes to follow) or
(counter byte) = (number of data bytes to follow) – 1
128 ≤ (counter byte) ≤ 255

“b” Counter specifies the number of times to repeat the next byte of data:

256 – (counter byte) + 1 = (number of times to repeat next byte)
(counter byte) = 257 – (number of times to repeat next byte)

STOP

Data which will exceed the right margin are discarded.
The vertical movement cannot be defined smaller than the current print density: density must not be changed within Graphics Mode.
Avoid moving the print position upwards while in Graphics Mode.

This sequence is valid only in Graphics Mode (**ESC (G**, see page 67).
After printing raster graphics the actual print position is the most right dot + 1 dot of the image.
Counter (a) and counter (b) can be mixed within the same sequence.
The dot count should be a multiple of 8. Otherwise the remaining data < 8 will be discarded.

Activate User Defined Character Set — ESC %

Data Structure

ASCII	ESC	"%"	SOH	activate user defined character set in RAM
hex.	1B	25	01	
dec.	27	37	1	

ASCII	ESC	"%"	NUL	activate normal character set in ROM or Font Card
hex.	1B	25	00	
dec.	27	37	0	

Example

```
Example   30 LPRINT "[ \ ] ` { | } ~"
40 REM Copy ROM CG into Download CG
50 LPRINT CHR$(27);":";CHR$(0);CHR$(0);CHR$(0);
60 REM Select Download CG
70 LPRINT CHR$(27);"%";CHR$(1);
80 LPRINT "[ \ ] ` { | } ~"
90 END
```

```
[ \ ] ` { ¦ } ~
[ \ ] ` { ¦ } ~
```

Download Characters — ESC &

The sequence for the definition of download characters receives different parameters for 9 needle and 24 needle printers. Use the data structure available for your printer.

9

Definition of Draft Characters (9 Needle Printer)

Data Structure

ASCII	ESC	&	NUL	<x> <y> <A> <n_1> <n_2> <n_3>...<n_{11}>	define
hex.	1B	26	00	...	download
dec.	27	38	0	...	characters

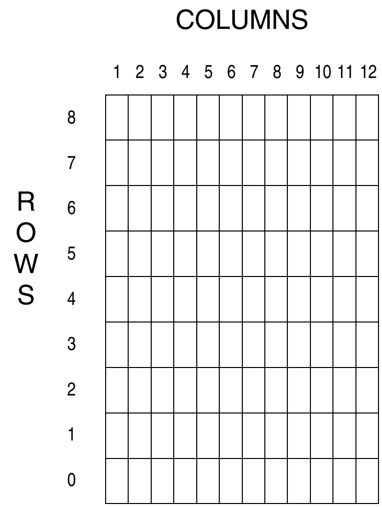
Description

x	first character to be redefined
y	last character to be redefined
A	an attribute which consists of descending data and proportional data (see “Computing the Attribute”)
n_1	needle information for column 1
n_2	needle information for column 2
n_3	needle information for column 3
.	
.	
.	
n_{11}	needle information for column 11

This code sequence assigns the character pattern defined by n_1, n_2, n_3 to n_{11} as download character to ASCII codes x to y. If the download character is only being assigned to one character code, specify x = y. "A" determines the descender and proportional print attributes.

Defining Character (9 Needle Printer)

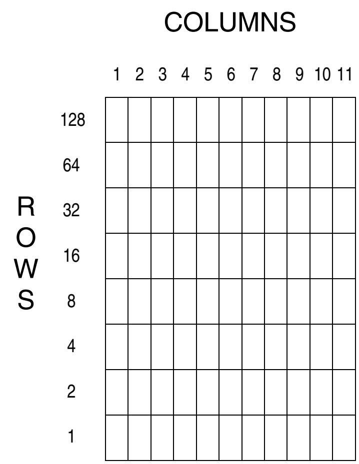
Each downloaded character is defined in a matrix field of 9 rows high and 12 columns wide.



It is only possible to use column 1 up to column 11. Column 12 is the space between one character and the next. Therefore it is always left blank by automatically setting dots in column 12 to zero.

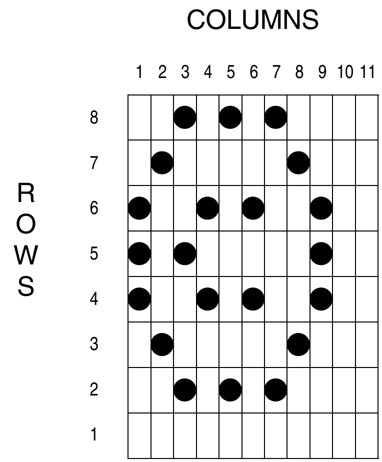
Also, only 8 of 9 dots in any one column can be printed (see also “Descenders”).

Each column is defined by a single byte where each row in that column corresponds to the LSB (dec.1) and the top row to the MSB (dec.128).

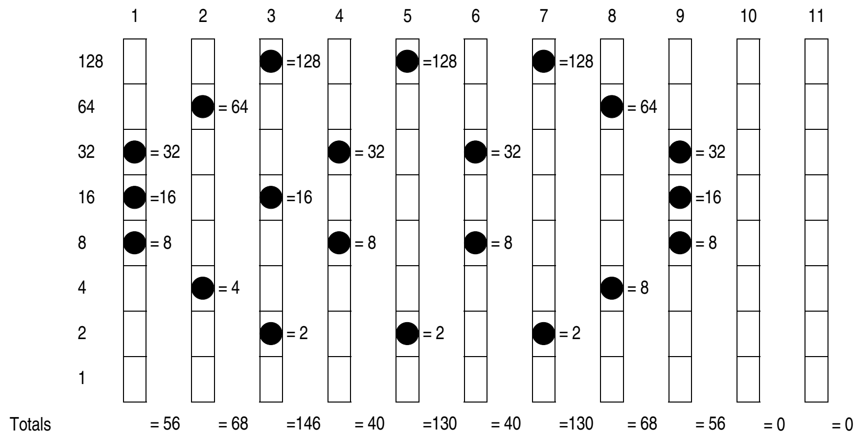


Example (9 needle printer):

The following example illustrates the definition of a character, the copyright symbol (a small “c” in a circle), instead of the character “@”.

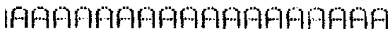


The evaluation of the bytes in each column is as follows:



Example

```
10 REM    Download Character Definition
20 LPRINT CHR$(27);"x";CHR$(0);:REM select draft print quality
30 LPRINT CHR$(27);"&";CHR$(0);"@@";CHR$(139);
40 LPRINT CHR$(56);CHR$(68);CHR$(146);CHR$(40);
50 LPRINT CHR$(130);CHR$(40);CHR$(130);CHR$(68);
60 LPRINT CHR$(56);CHR$(0);CHR$(0);
70 LPRINT CHR$(27);"%";CHR$(1);CHR$(0);
80 LPRINT "@ @ @ @ @ @ @ @ @ @"
90 END
```



Attribute “A” (9 needle printer)

Attribute “A” consists of descender data and proportional data. The descender data determines whether or not the ninth needle will be used, and the proportional data determines the starting and ending positions of the character in the print area.

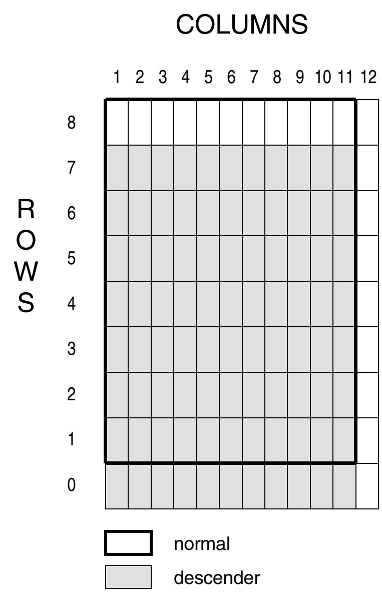
Descenders (9 needle printer)

Some characters, like “g” or “y”, extend below the line. To define a character that extends below the line, the ninth needle, which corresponds to row 0 of the matrix field, must be used. Characters which use row 0 cannot use the top needle or row 8 of the matrix field.

Bit 8 of the attribute determines whether the matrix-field is shifted down one row or not.

Normal matrix field (row 1 up to row 8) is used by setting bit 8 to 1.

Descender matrix field (row 0 up to row 7) is used by setting bit 8 to 0 (See “Computing the Attribute”).



Proportional Data (9 needle printer)

Not all characters are the same size. “W” is considerably wider than “i”. In the proportional print mode, it is necessary to specify how much space a character takes by specifying the first and last printable columns in that character’s matrix.

Seven bits of the attribute are used for specifying the print position of a character within its matrix. The three high-order bits of the seven bits of proportional data represent the starting position of the character in the print area and the remaining four bits indicate the ending position.

All characters in proportional mode are printed as emphasized characters. Therefore, when defining a download character for printing in proportional mode, specify the ending print position one or more blank columns greater than the actual width of the character. This is also true for enlarged mode.

The maximum print area width of a download character is 12 columns, but column 12 must be blank, or if dots are specified in this column, they will automatically set to 0. The minimum width of a download character is 5 columns. All of the download character data n_1 to n_{11} must be sent to the printer. Note also that, if horizontally adjacent dots are specified, they will be automatically ignored.

Computing the Attribute (9 needle printer)

The 8 bits that make up attribute A are broken down as follows:

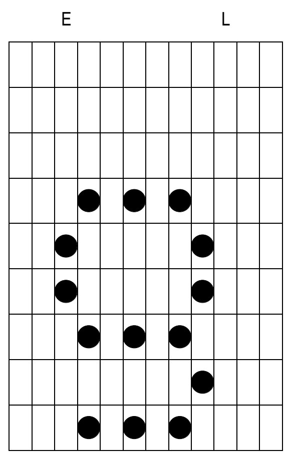
		Bit	dec.
Part 1	When using normal matrix field, this bit is 1. For descending matrix field it is 0.	8	128
Part 2	These 3 bits specify the first column of the proportional character.	7 6 5	64 32 16
Part 3	These 4 bits specify the last column of the proportional character.	4 3 2 1	8 4 2 1

Just add up the values of the three parts of the byte to get the value of A.

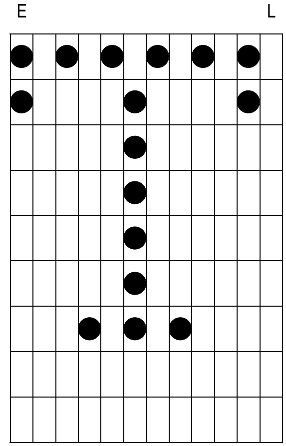
1.	Normal	=	128
	Descender	=	0
2.	First column is 1	=	0
	First column is 2	=	16
	First column is 3	=	32
	First column is 4	=	48
	First column is 5	=	64
	First column is 6	=	80
	First column is 7	=	96
	First column is 8	=	112
3.	Last column is 5	=	4
	Last column is 6	=	5
	Last column is 7	=	6
	Last column is 8	=	7
	Last column is 9	=	8
	Last column is 10	=	9
	Last column is 11	=	10
	Last column is 12	=	11

Example:

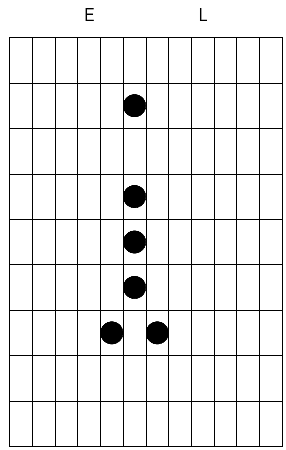
Descender matrix field	=	0
First column is 3	=	32
Last column is 10	=	9
Total value of <A>	=	41



Normal matrix field	=	128
First column is 1	=	0
Last column is 12	=	11
Total value of <A>	=	139



Normal matrix field	=	128
First column is 4	=	48
Last column is 9	=	8
Total value of <A>	=	184



Example

```
10  REM    Copy US ASCII caracter set
20  LPRINT CHR$(27);"R";CHR$(0);
30  LPRINT CHR$(27);":";CHR$(0);CHR$(0);CHR$(0);
40  REM Define character into # code and 4 code
50  LPRINT CHR$(27);"&";CHR$(0);"#$";
60  REM First character without descenter
70  LPRINT CHR$(137);
80  LPRINT CHR$(56);CHR$(68);CHR$(146);CHR$(40);
90  LPRINT CHR$(130);CHR$(40);CHR$(130);CHR$(68);
100 LPRINT CHR$(56);CHR$(0);CHR$(0);
110 REM Second character with descender
120 LPRINT CHR$(9);
130 LPRINT CHR$(56);CHR$(68);CHR$(146);CHR$(40);
140 LPRINT CHR$(130);CHR$(40);CHR$(130);CHR$(68);
150 LPRINT CHR$(56);CHR$(0);CHR$(0);
160 REM Select Download Character Generator
170 LPRINT CHR$(27);"%";CHR$(1);CHR$(0);
180 LPRINT "*** Draft print quality ***"
190 PRINT
200 LPRINT "First character without descender ---> ##########"
210 LPRINT "Second character with descender -----> $$$$$$$$$$"
220 LPRINT "Both characters mixed ---------------> #$#$#$#$#$"
230 LPRINT :LPRINT :LPRINT
240 LPRINT CHR$(27);"p1";
250 LPRINT "*** Proportional mode ***"
260 LPRINT
270 LPRINT "First character without descender ---> ##########"
280 LPRINT "Second character with descender -----> $$$$$$$$$$"
290 LPRINT "Both characters mixed ---------------> #$#$#$#$#$"
300 END
```

```
*** Draft print quality ***

First character without descender ---> ΘΘΘΘΘΘΘΘΘΘ
Second character with descender -----> ΘΘΘΘΘΘΘΘΘΘ
Both characters mixed ---------------> ΘΘΘΘΘΘΘΘΘΘ


*** Proportional mode ***

First character without descender ---> ΘΘΘΘΘΘΘΘΘΘ
Second character with descender -----> ΘΘΘΘΘΘΘΘΘΘ
Both characters mixed ---------------> ΘΘΘΘΘΘΘΘΘΘ
```

9 Definition of NLQ Characters (9 needle printer)

The definition of NLQ characters is more comprehensive than the definition of draft characters, since a larger number of dots is required.

> User-defined NLQ characters can only be generated in the NLQ mode of the printer.

Data structure

ASCII	ESC	"&"	NUL	<x>	<y>	<d_0> <d_1> <d_2> <n_1>...<n_k> (data 1. character)
						<d_0> <d_1> <d_2> <n_1>...<n_k> (data 2. character)
hex.	1B	26	00			
dez.	27	38	0			

Description

x	first character to be redefined
y	last character to be redefined
d_0	leading space of the character
d_1	number of columns of the defineded character
d_2	following space of the character
k	number of data bytes to transmit
$n_1 ... n_k$	data bytes (3 data-bytes for one column)

Values of parameters:

$0 \le d_0 \le 20$
$0 \le d_1 \le 20$
$0 \le d_2 \le 20$
$k = 3 * d_1$

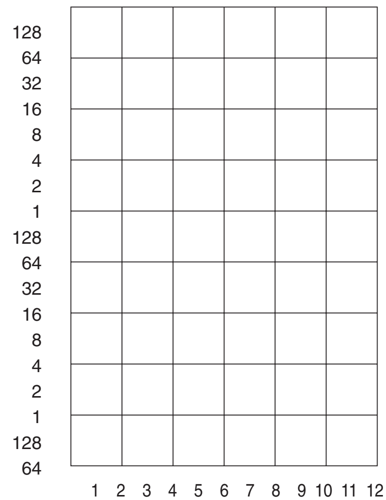
If only a single download character is to be defined, x = y is to be set.

> When this command is transferred to the printer, first of all, all data in the print buffer are output.

Definition of the Character Grid

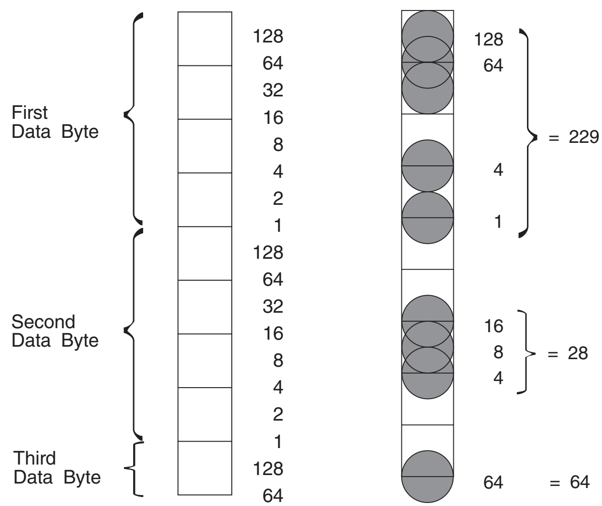
NLQ characters consist of 18 vertical dots and 12 horizontal dots.

Within this grid every numbered line or gap can be occupied, including the lowest and the furthest right line. You should, however, leave one or two columns empty, so that neighboring characters do not touch.

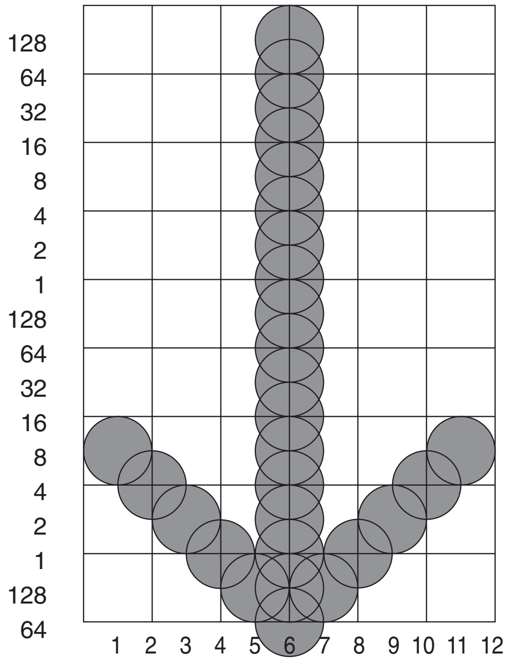


For each column of the NLQ character it is necessary to define 3 Bytes. Each bit of the data bytes corresponds to a certain value. The example on the right shows the assignment of print dots to data bytes and the calculation of the values.

In order to determine the numeric value of a column, it is necessary to first of all add the values of the 3 bytes separately. A byte with no print dot receives the numeric value 9.



The following example shows a user-defined arrow with the list of the individual column values (each 3 byte) which must be transferred to the printer.



Data byte 1	0	0	0	0	0	255	0	0	0	0	0	0
Data byte 2	8	4	2	1	0	255	0	1	2	4	8	0
Data byte 3	0	0	0	0	128	192	128	0	0	0	0	0

See next page for the corresponding programming example.

Example

```
100 LPRINT CHR$(27);"x";CHR$(1);
110 LPRINT CHR$(27);"&";CHR$(0);";;";
120 LPRINT CHR$(0);CHR$(12);CHR$(0);
130 FOR I=1 TO 36
140 READ D
150 LPRINT CHR$(0);
160 NEXT I
170 LPRINT "; ; ; ; ; ;"
180 LPRINT CHR$(27);"%";CHR$(1);
190 LPRINT "; ; ; ; ; ;"
200 LPRINT CHR$(27);"%";CHR$(0);
210 LPRINT "; ; ; ; ; ;"
220 END
230 DATA 0,8,0,0,4,0,0,2,0,0,1,0
240 DATA 0,0,128,255,255,192,0,0,128,0,1,0
250 DATA 0,2,0,0,4,0,0,8,0,0,0,0
```

```
; ; ; ; ; ;
↓ ↓ ↓ ↓ ↓ ↓
; ; ; ; ; ;
```

24 needle printer

Data Structure

ASCII	ESC	"&"	NUL	<x>	<y>	<d_0> <d_1> <d_2> $n_1...n_3$ (data first character)
						<d_0> <d_1> <d_2> $n_1...n_3$ (data second character) ...
hex.	1B	26	...			
dec.	27	38	...			

Description

x	first character to be redefined
y	last character to be redefined
d_0	leading space of the character
d_1	number of columns of the defineded character
d_2	following space of the character
$n_1...n_3$	three data bytes for one column in DPQ and LQ, two data bytes for superscript or subscript characters or 15 cpi character (DPQ + LQ)

If the download character is only being assigned one character code, specify x = y.

Limitations for d_0, d_1 and d_2

print quality	d_0	d_1	d_2	$d_0+d_1+d_2$
DPQ 10 cpi	≥ 0	≤ 12	≥ 0	≤ 12
DPQ 12 cpi	≥ 0	≤ 10	≥ 0	≤ 10
DPQ 15 cpi	≥ 0	≤ 8	≥ 0	≤ 8
LQ 10 cpi	≥ 0	≤ 36	≥ 0	≤ 36
LQ 12 cpi	≥ 0	≤ 30	≥ 0	≤ 30
LQ 15 cpi	≥ 0	≤ 24	≥ 0	≤ 24
LQ proportional	≥ 0	≤ 42	≥ 0	≤ 42
LQ super/subscript	≥ 0	≤ 37	≥ 0	≤ 37

If defined character density (10/12/15 cpi) is selected, d_0 and d_2 will be ignored and the defined character is centered.

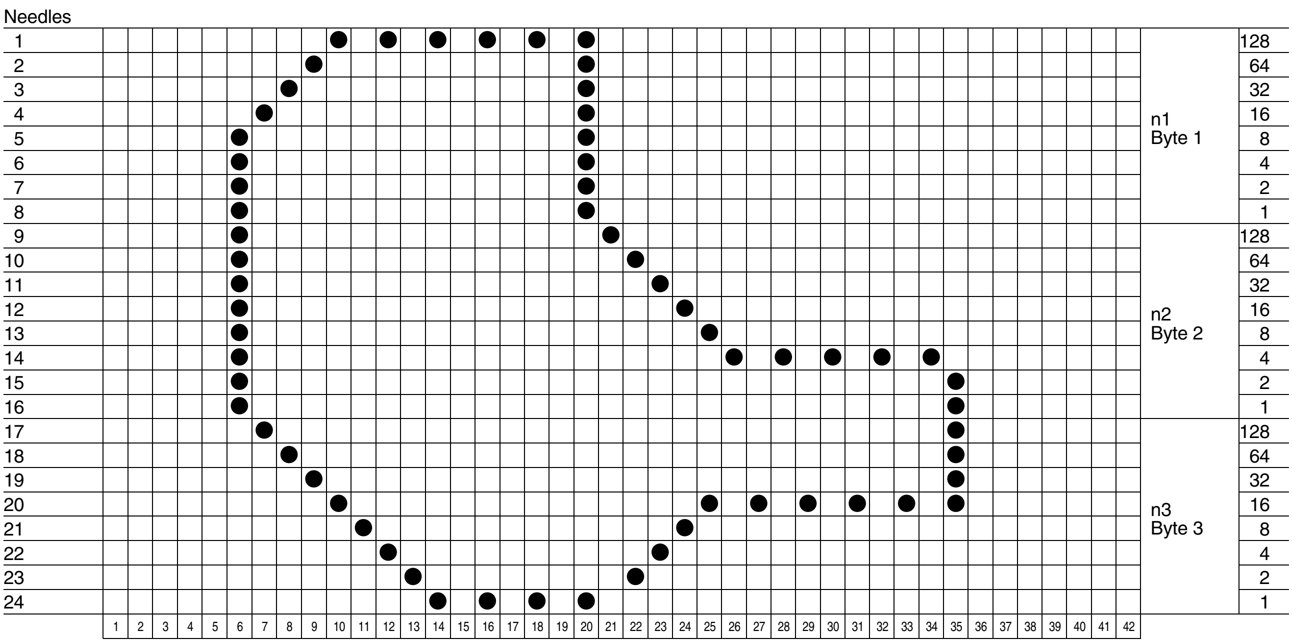
If more columns than d_1 will be defined for one character, it is printed more to the right and can overlap the next character. The absolute limit for d_1 is 15 columns in DPQ and 42 columns in LQ.

For LQ proportional only, it is necessary to define all 41 columns; column 42 should be free. All other print modes use the number of columns as defined by d_1.

Superscript and subscript characters and 15 cpi characters can be assigned needles 1 to 16. The super and sub shifting will be done by corresponding escape sequence.

If the same needle is set to print adjacent columns, the printer will ignore one of them, i.e. it will print in alternating order.

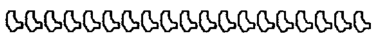
Example:



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42
Code 1						15	16	32	64	128	0	128	0	128	0	128	0	128	0	255	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							
Code 2						255	0	0	0	0	0	0	0	0	0	0	0	0	0	0	128	64	32	16	8	4	0	4	0	4	0	4	0	4	3							
Code 3						0	128	64	32	16	8	4	2	1	0	1	0	1	0	1	0	2	4	8	16	0	16	0	16	0	16	0	16	0	240							

Example

```
10 REM    Download Character Definition
20  LPRINT CHR$(27);"x1"
30  LPRINT CHR$(27);"&";CHR$(0);"AA";
40  LPRINT CHR$(5);CHR$(30);CHR$(0);
50  LPRINT CHR$(15);CHR$(255);CHR$(0);
60  LPRINT CHR$(16);CHR$(0);CHR$(128);
70  LPRINT CHR$(32);CHR$(0);CHR$(64);
80  LPRINT CHR$(64);CHR$(0);CHR$(32);
90  LPRINT CHR$(128);CHR$(0);CHR$(16);
100 LPRINT CHR$(0);CHR$(0);CHR$(8);
110 LPRINT CHR$(128);CHR$(0);CHR$(4);
120 LPRINT CHR$(0);CHR$(0);CHR$(2);
130 LPRINT CHR$(128);CHR$(0);CHR$(1);
140 LPRINT CHR$(0);CHR$(0);CHR$(0);
150 LPRINT CHR$(128);CHR$(0);CHR$(1);
160 LPRINT CHR$(0);CHR$(0);CHR$(0);
170 LPRINT CHR$(128);CHR$(0);CHR$(1);
180 LPRINT CHR$(0);CHR$(0);CHR$(0);
190 LPRINT CHR$(255);CHR$(0);CHR$(1);
200 LPRINT CHR$(0);CHR$(128);CHR$(0);
210 LPRINT CHR$(0);CHR$(64);CHR$(2);
220 LPRINT CHR$(0);CHR$(32);CHR$(4);
230 LPRINT CHR$(0);CHR$(16);CHR$(8);
240 LPRINT CHR$(0);CHR$(8);CHR$(16);
250 LPRINT CHR$(0);CHR$(4);CHR$(0);
260 LPRINT CHR$(0);CHR$(0);CHR$(16);
270 LPRINT CHR$(0);CHR$(4);CHR$(0);
280 LPRINT CHR$(0);CHR$(0);CHR$(16);
290 LPRINT CHR$(0);CHR$(4);CHR$(0);
300 LPRINT CHR$(0);CHR$(0);CHR$(16);
310 LPRINT CHR$(0);CHR$(4);CHR$(0);
320 LPRINT CHR$(0);CHR$(0);CHR$(16);
330 LPRINT CHR$(0);CHR$(4);CHR$(0);
340 LPRINT CHR$(0);CHR$(3);CHR$(240);
350 LPRINT CHR$(27);"%";CHR$(1);CHR$(0);
360 LPRINT:LPRINT
370 LPRINT "AAAAAAAAAAAAAAAAAAA"
380 LPRINT CHR$(27);"x0"
390 LPRINT CHR$(27);"%";CHR$(0);CHR$(0);
400 LPRINT
410 LPRINT "AAAAAAAAAAAAAAAAAAA"
420 END
```



AAAAAAAAAAAAAAAAAAA

Miscellaneous

Print Data as Characters — ESC (^

Data Structure

ASCII	ESC	"("	"^"	<n_L> <n_H> <d_1>...<d_n>	ESC/P2	print data as characters
hex.	1B	28	54	<n_L> <n_H> <d_1>...<d_n>		
dec.	27	40	94	<n_L> <n_H> <d_1>...<d_n>		

Description

Valid values:

$0 \leq n_L \leq 255$

$0 \leq n_H \leq 127$

Data bytes d_1 to d_n are printed as characters rather than control codes.

$k = ((n_H \times 256) + n_L)$

$$n_H = \mathrm{INT}\,\frac{k}{256}$$

$$n_L = \mathrm{MOD}\,\frac{k}{256}$$

Data is ignored if no character is assigned to that character in the active character table.

Printing Direction unidirectional for one line — ESC <
Set/Reset Unidirectional Printing — ESC U

Data Structure

ASCII	ESC	"<"		unidirectional printing for one line	❶
hex.	1B	3C			
dec.	27	60			
ASCII	ESC	"U"	<n> or n	set/reset unidirectional printing	❷
hex.	1B	55	<n> or n		
dec.	27	85	<n> or n		

Description

❶ This code sequence sets unidirectional printing mode only for one line. The printing is carried out from left to right.

❷ This code sequence sets or resets unidirectional printing mode. Unidirectional printing is enabled when 1 is specified for n and is disabled when 0 is specified.

In text mode, printing is faster when unidirectional printing is disabled. However, characters on one line may be slightly shifted with respect to those in the preceding line when bidirectional printing is used.

In its default setting the printer is set to bidirectional printing.

Half-Speed Printing — ESC s

Data Structure

ASCII	ESC	"s"	<n>
hex.	1B	73	<n>
dec.	27	115	<n>

Description

With n=1 the print speed is reduced by half. The result is a lower printer noise level.

With n=0 the normal print speed is set.

Example

```
10  REM    selects half-speed printing
20  LPRINT "This is full speed print in LQ mode"
30  LPRINT CHR$(27);"s";CHR$(1);:REM    set half speed
40  LPRINT "This is half speed print in LQ mode"
50  LPRINT "              (Quiet Mode)
60  LPRINT CHR$(27);"s";CHR$(0);:REM    Set full speed
70  LPRINT CHR$(27);"x";CHR$(0):REM    Set draft mode"
80  LPRINT "This is full speed print in draft mode"
90  LPRINT CHR$(27);"s";CHR$(1);:REM    set half speed
100 LPRINT "This is half speed print in draft mode"
110 LPRINT "              (Quiet Mode)
120 LPRINT CHR$(28);"@"    'Resets printer
130 END
```

```
This is full speed print in LQ mode
This is half speed print in LQ mode
              (Quiet Mode)

This is full speed print in draft mode
This is half speed print in draft mode
              (Quiet Mode)
```

Single n/180 Inch or n/216 Inch Line Feed — ESC J
Single n/180 Inch or n/216 Inch Back Feed — ESC j

Data Structure

ASCII	ESC	"J"	<n>	n/180 inch line feed	24	❶
hex.	1B	4A	<n>	n/216 inch line feed	9	
dec.	27	74	<n>			
ASCII	ESC	"j"	<n>	n/180 inch back feed	24	❷
hex.	1B	6A	<n>	n/216 inch back feed	9	
dec.	27	106	<n>			

Description

❶ When the printer receives this code sequence it advances the paper by $n/180$ inches (24 needle printer) or by $n/216$ inches (9 neadle printer). This sequence does not effect a CR (reset line pointer).

❷ When the printer receives this code sequence it moves the paper backwards by $n/180$ inches (24 needle printer) or by $n/216$ inches (9 needle printer).

Unlike an ordinary line feed code, code ESC J does not advance the paper to the top of the next page when the print position enters the perforation area (Perforation-skip operation is not performed). The value specified for n must be in the range of 0 to 255. This code sequence does not affect the amount by which the paper is advanced by subsequent LF or VT codes.

Colour Selection — ESC r

Data Structure

ASCII	ESC	"r"	<n>
hex.	1B	72	<n>
dec.	27	114	<n>

Description

According to the value specified for n one of the subsequent colours is selected:

9

0 black
1 magenta
2 cyan
3 blue
4 yellow
5 red
6 green

24

0 black
1 red
2 blue
3 violet
4 yellow
5 orange
6 green

The available colours depend on the ribbon used. The ribbon used must be matched in the user menu (PRINTER OPTIONS, RIBBON OPTIONS). If a ribbon with several black tracks is used, then it is automatically switched every 60 lines between the black tracks.

The relationship between the track and the ribbon colour for various ribbon cassettes is shown in the following table:

track	magenta, cyan, yellow, black ribbon	red, green, blue, black ribbon	3 * black, red ribbon
1	black	black	black
2	cyan	blue	black
3	magenta	red	black
4	yellow	green	red

The relationship between colour selection sequences and the resulting colours for several ribbon cassettes is shown in the following table:

selected colour	magenta, cyan, yellow, black ribbon	red, green, blue, black ribbon	3 * black, red ribbon
black	black	black	black
red	red *)	red	red
green	green **)	green	****)
yellow	yellow	green	****)
blue	blue ***)	blue	****)
magenta	magenta	red	red
cyan	cyan	blue	****)

* the red is made from yellow + magenta
** the green is made from yellow + cyan
*** the blue is made from magenta + cyan
} two-pass print is automatically executed
**** Sequence is ignored

Example

```
10 REM    colour selection
20 LPRINT CHR$(27);"r";CHR$(5);
30 LPRINT "this is written with red colour"
40 LPRINT CHR$(27);"r";CHR$(3);
50 LPRINT "this is written with blue colour"
60 END
```

this is written with red colour
this is written with blue colour

International Character Sets — ESC R

Data Structure

ASCII	ESC	"R"	<n>	select international character set
hex.	1B	52	<n>	
dec.	27	82	<n>	

Description

According to the value specified for n the corresponding international character set can be selected:

0	USA
1	France
2	Germany
3	United Kingdom (UK)
4	Denmark I
5	Sweden
6	Italy
7	Spain I
8	Japan
9	Norway
10	Denmark II
11	Spain II
12	Latin America
13	Corea
64	Legal

For character sets see Appendix B.

Example

```
10  REM    international character sets
20  LPRINT CHR$(27);"R";CHR$(0) ;"ASCII        :     ";:GOSUB 160
30  LPRINT CHR$(27);"R";CHR$(1) ;"FRANCE       :     ";:GOSUB 160
40  LPRINT CHR$(27);"R";CHR$(2) ;"GERMANY      :     ";:GOSUB 160
50  LPRINT CHR$(27);"R";CHR$(3) ;"UK           :     ";:GOSUB 160
60  LPRINT CHR$(27);"R";CHR$(4) ;"DENMARK I    :     ";:GOSUB 160
70  LPRINT CHR$(27);"R";CHR$(5) ;"SWEDEN       :     ";:GOSUB 160
80  LPRINT CHR$(27);"R";CHR$(6) ;"ITALY        :     ";:GOSUB 160
90  LPRINT CHR$(27);"R";CHR$(7) ;"SPAIN I      :     ";:GOSUB 160
100 LPRINT CHR$(27);"R";CHR$(8) ;"JAPAN        :     ";:GOSUB 160
110 LPRINT CHR$(27);"R";CHR$(9) ;"NORWAY       :     ";:GOSUB 160
120 LPRINT CHR$(27);"R";CHR$(10);"DENMARK II   :     ";:GOSUB 160
130 LPRINT CHR$(27);"R";CHR$(11);"SPAIN II     :     ";:GOSUB 160
140 LPRINT CHR$(27);"R";CHR$(12);"LATIN AMERICA:     ";:GOSUB 160
150 END
160 LPRINT CHR$(35);CHR$(36);CHR$(64);CHR$(91);
170 LPRINT CHR$(92);CHR$(93);CHR$(94);CHR$(96);
180 LPRINT CHR$(123);CHR$(124);CHR$(125);CHR$(126);
190 RETURN
```

```
ASCII        :     #$@[\]^`{¦}~
FRANCE       :     #$à°ç§^`éùè¨
GERMANY      :     #$§ÄÖÜ^`äöüß
UK           :     £$@[\]^`{¦}~
DENMARK I    :     #$@ÆØÅ^`æøå~
SWEDEN       :     #¤ÉÄÖÅÜéäöåü
ITALY        :     #$@°\é^ùàòèì
SPAIN I      :     ₧$@¡Ñ¿^`¨ñ}~
JAPAN        :     #$@[¥]^`{¦}~
NORWAY       :     #¤ÉÆØÅÜéæøåü
DENMARK II   :     #$ÉÆØÅÜéæøåü
SPAIN II     :     #$á¡Ñ¿é`íñóú
LATIN AMERICA:     #$á¡Ñ¿éüíñóú
```

Character Sets — ESC t

Data Structure

ASCII	ESC	"t"	<d_1>	select character table
hex.	1B	74	<d_1>	
dec.	27	116	<d_1>	

Description

Valid values:

$0 \leq d_1 \leq 3$, $48 \leq d_1 \leq 51$

By means of this code sequence the upper half of character code table (dec. 128 to dec. 255) can be selected.

		default	Description
d_1 = 0 or 48	Character table 0	italic	The Epson Standard Character Set is selected. The upper half contains control codes and italic characters.
d_1 = 1 or 49	Character table 1	Table 437	The Epson Graphic Character Set is selected. The upper half contains international and graphic characters (see IBM Character Set 2, from hex.A0 to hex.FF)
d_1 = 2 or 50	Character table 2	DLL *)	The upper half contains user-defined characters, if created by ESC &. When creating characters, assign code value in the range of 0 to 127. For using add 128 to the codes. Using of n = 2 without user-defined characters causes same contents of upper and lower half of the character code table.
d_1 = 3 or 5 ESC/P2	Character table 3	Table 437	The IBM Code Page Table 437 is selected

* user defined character set

You can either register 0 - 3. Register 3 is only available in ESC/P2 mode.
The command ESC (t can be used to assign character sets to character table (d_1).
If d_1 = 2 you cannot use user-defined characters defined previously with ESC (t (see page 105).

Example

```
10  REM  upper half of code table
20  LPRINT CHR$(27);"t0";
30  GOSUB 90
40  LPRINT CHR$(27);"t1";
50  GOSUB 90
60  LPRINT CHR$(27);"t2";
70  GOSUB 90
80  END
90  RESTORE
100 FOR I = 1 TO 15
110 READ D
120 LPRINT CHR$(D);
130 NEXT I
140 LPRINT
150 RETURN
160 DATA 65,66,67,68,69,165,166,167,168,169,170,171,172,173,174
```

```
ABCDE%&´()*+,-.
ABCDEÑªº¿⌐¬½¼¡«
ABCDE%&´()*+,-.
```

Assign Character Table ESC (t

Data Structure

ASCII	ESC	"("	"t"	$<n_L>$ $<n_H>$ $<d_1>$ $<d_2>$ $<d_3>$	ESC/P2 assign character table
hex.	1B	28	74	$<n_L>$ $<n_H>$ $<d_1>$ $<d_2>$ $<d_3>$	
dec.	27	40	116	$<n_L>$ $<n_H>$ $<d_1>$ $<d_2>$ $<d_3>$	

Description

Valid values:

$n_L = 3, n_H = 0$

$0 \le d_1 \le 3, 48 \le d_1 \le 51$

$0 \le d_2 \le 255$

$0 \le d_3 \le 255$

Assignes the choosen character table to the register d_1 (0 - 3). The ESC t command (see page 103) can be used to activate the defined character set.

d_2	d_3	Table name
0	0	Italic
1	0	Table 437 (US)
3	0	Table 850 (Multilingual)
7	0	Table 860 (Portugal)
8	0	Table 863 (Canada-French)
9	0	Table 865 (Norway)
10	0	Table 852 (East Europe)
11	0	Table 857 (Turkish)
14	0	Table 866 (Russian)
24	0	Table 861 (Iceland)
25	0	BRASCII
(continued on next page)		

d_2	d_3	Table name
26	0	Abicomp
29	16	ISO 8859-1 (Latin 1)
35	0	Roman 8
127	2	ISO 8859-2 (Latin 2)

Please refer to the printer's reference manual which character sets are supported.

Input Data Control Bit 8 = 1 — ESC >
Input Data Control Bit 8 = 0 — ESC =
Input Data Control Bit 8 unchanged — ESC

Data Structure

ASCII	ESC	">"	set bit 8 = 1
hex.	1B	3E	
dec.	27	62	
ASCII	ESC	"="	set bit 8 = 0
hex.	1B	3D	
dec.	27	61	
ASCII	ESC	"#"	Removing the settings set for Bit 8 (MSB, most significant bit), which were set with the use of either ESC = or ESC >
hex.	1B	23	
dec.	27	35	

Example

```
10  REM    input data control
20  LPRINT CHR$(27);"#";
30  GOSUB 90
40  LPRINT CHR$(27);">";
50  GOSUB 90
60  LPRINT CHR$(27);"=";
70  GOSUB 90
80  END
90  FOR I=97 TO 103:LPRINT CHR$(I);:NEXT I
100 FOR I=161 TO 167:LPRINT CHR$(I);:NEXT I
110 LPRINT LPRINT
120 RETURN
```

abcdefg*!"#$%&'*

abcdefg!"#$%&'

abcdefg!"#$%&'

Sheet Feeder — ESC EM

Data Structure

ASCII	ESC	EM	"0"	deactivate sheet feeder operation
hex.	1B	19	30	
dec.	27	25	48	
ASCII	ESC	EM	"1"	select magazine 1 (front)
hex.	1B	19	31	
dec.	27	25	49	
ASCII	ESC	EM	"2"	select magazine 2 (rear)
hex.	1B	19	32	
dec.	27	25	50	
ASCII	ESC	EM	"4"	activate sheet feeder operation
hex.	1B	19	34	
dec.	27	25	51	
ASCII	ESC	EM	"R"	eject paper
hex.	1B	19	52	
dec.	27	25	82	

Description

This code sequence should only be used when the sheet feeder is mounted and selected. Parameters for sheet feeder operation set via the control panel are deactivated.

When sheet feeder operation is activated, the paper is fed automatically by a received code FF. If the last print position is outside the defined print area, the paper will additionally be fed by codes LF, VT, or ESC J.

The form length must be defined in accordance with the paper format used.

Example

```
10 REM   sheet feeder
20 LPRINT CHR$(27);CHR$(25);"1";
30 LPRINT "This text is on a form"
40 LPRINT "from magazine one."
50 LPRINT CHR$(12);:REM form feed
60 LPRINT CHR$(27);CHR$(25);"2";
70 LPRINT "This text is on a form"
80 LPRINT "from magazine two."
90 END
```

```
This text is on a form
from magazine one.
```

```
This text is on a form
from magazine two.
```

Print Code Area Expansion ON — ESC 6
Print Code Area Expansion OFF — ESC 7

Data Structure

ASCII	ESC	"6"	print ASCII codes 128 to 159 and 255	❶
hex.	1B	36		
dec.	27	54		
ASCII	ESC	"7"	suppress ASCII codes 128 to 159 and 255	❷
hex.	1B	37		
dec.	27	55		

Description

❶ This code sequence causes the printout of the higher control codes (ASCII 128 to 159 and 255) in the form of characters and symbols instead of the repetition of the functions of ASCII codes 0 to 31 and 127.

For this function the Graphic character set must be selected by ESC t <1> or by menu (see IBM Character Set 2, from hex.80 to hex.9F).

❷ When the printer receives this code sequence, the ASCII codes 128 to 159 and 255 are used as duplicates of the functions of the lower control codes 0 to 31 and 127.

For character sets see Appendix B.

Example

```
10  REM    code aera expansion
15  LPRINT CHR$(27);"t1";
20  LPRINT CHR$(27);"6";
30  GOSUB 70
40  LPRINT CHR$(27);"7";
50  GOSUB 70
60  END
70  RESTORE
80  FOR I = 1 TO 12
90  READ D
100 LPRINT CHR$(D);
110 NEXT I
120 LPRINT
130 RETURN
140 DATA 65,66,67,68,69,128,129,130,131,132,133,134
```

ABCDEÇüéâäàå
ABCDE

Activate/Deactivate Graphics Characters — ESC m

Data Structure

ASCII	ESC	"m"	<n>
hex.	1B	6D	<n>
dec.	27	109	<n>

Description

On receipt of this command, the ASCII codes hex.90 to hex.9F can be output as special graphics characters.

Valid values for n are:

n = 0 (hex.00)	activates the control codes
n = 4 (hex.04)	activates the graphics characters

Activate/Deactivate Direct Print Mode — ESC i

Data Structure

ASCII	ESC	"i"	<n> or n
hex.	1B	69	<n> or n
dec.	27	105	<n> or n

Description

In this mode no complete lines from the print buffer are printed, but each character is printed immediately on receipt of the character, just like in a typewriter. In addition the printer pushes the paper up, so that the user can immediately see the printed character and then pulls the page down again to continue printing.

Valid values for n are:

n = 1 (dec.1 or dec.49, hex.01 or hex.31)	activates the mode
n = 0 (dec.0 or dec.48, hex.00 or hex.30)	deactivates the mode

International Code Table ESC I

Data Structure

ASCII	ESC	"I"	<n> or n	9
hex.	1B	49	<n> or n	
dec.	27	73	<n> or n	

Description

The ASCII codes hex.00 to hex.1F (dec.0 to dec.31) and hex.80 to hex.9F (dec.128 to dec.159) are as a rule assigned with non-printable characters. These codes can, however, be converted to printable characters using ESC I.

Valid values for n are either ASCII code hex.00 and hex.01 or ASCII characters "0" and "1" (hex.30 and hex.31).

n = 1 (hex.01 or hex.31) ➜ Access to user-defined or international characters.

n = 0 (hex.00 or hex.30) ➜ Hex.00 to hex.1F and hex.80 to hex.9F assigned with non-printable codes.

For character sets see appendix B "Character Sets".

Printer Initialization — ESC @

Data Structure

ASCII	ESC	"@"	printer initialization
hex.	1B	40	
dec.	27	64	

Description

Printer initialization means that the printer is in the same state as it is after switch-on. All parameters set by means of ESC sequences are cleared.

> All print data transferred after the last print control code (CR, LF ...) are lost. For this reason first send the respective print control code (CR, LF) and then the ESC @ sequence.

Example

```
10  REM   initialization
15  LPRINT CHR$(27);"l";CHR$(10):REM set left margin
20  LPRINT CHR$(27);CHR$(15):REM set condensed
30  LPRINT CHR$(27);"G":REM set NLQ
35  LPRINT CHR$(27);"S1";:REM set subscript
40  GOSUB 90
70  LPRINT CHR$(27);"@";
80  GOSUB 90
90  END
100 LPRINT "Matrix Printer"
110 LPRINT:LPRINT
120 END
```

Matrix Printer

Matrix Printer

Activate Paper End Sensor — ESC 8
Deactivate Paper End Sensor — ESC 9

Data Structure

ASCII	ESC	"8"	Deactivate paper end sensor	❶
hex.	1B	38		
dec.	27	56		
ASCII	ESC	"9"	Activate paper end sensor	❷
hex.	1B	39		
dec.	27	57		

Description

❶ This sequence deactivates the paper end sensor, so that a cut sheet can be printed all the way to the bottom. With the sheet feeder activated this command is ignored.

❷ Activates the paper end sensor. As soon as a certain area near the lower paper edge is reached, an acoustic signal sounds and the printing process is interrupted.

Control Codes

BEL (hex.07) (dec.7)

Sounds acoustic signal.

BS (hex.08) (dec.8)

Print contents of line buffer, then move the print head one position back. When auto right justification is selected by code sequence ESC a<3>, <BS> will be ignored.

CAN (hex.18) (dec.24)

When the printer receives this code, the print buffer is deleted. This code only clears the contents of the print buffer; it does not affect any control codes which have previously been sent to the printer.

CR (hex.0D) (dec.13)

When the printer receives this code, it prints any data in the print buffer and resets the line pointer to the left margin.

Code CR can also give a line feed if automatic line feed is selected. Refer to the Operator's Manual.

DC1 (hex.11) (dec.17)

The control code DC1 switches the printer into Online Mode. If the printer has been set Offline with the control code DC3, then DC1 is the only control code which can be carried out. DC1 has no effect if the printer has been set Offline via the control panel.

DC1 and DC3 have no effect if the Centronics interface signal SELECT is switched off (LOW).

DC2 (hex.12) (dec.18)

Reset condensed print mode.

DC3 (hex.13) (dec.19)

The control code DC3 switches the printer into Offline Mode. If the printer has been set Offline with the control code DC3, then DC1 (switching printer Online) is the only control code which can be carried out. If the printer has been switched Offline via DC3, it can only be switched into the Online mode again using DC1 and not via the control panel.

DC1 and DC3 have no effect if the Centronics interface signal SELECT is switched off (LOW).

DC4 (hex.14) (dec.20)

Reset enlarged print mode set by <SO>.

DEL	(hex.7F) (dec.127)	Clear previous characters in the print buffer.
ESC	(hex.1B) (dec.27)	Start character for a subsequent program sequence.
FF	(hex.0C) (dec.12)	When the printer receives this code, it prints the data in the print buffer, then advances the paper from the current print position to the top-of-form position on the next page. The next line is beginning at the left margin.
HT	(hex.09) (dec.9)	This code causes the following characters to be printed starting from the first horizontal tab stop after the current print position. When the printer is switched on, tab stops are set at every eighth print position. The amount by which the print position is shifted by tabulation depends on the selected character spacing. When horizontal tabulation is carried out e.g. at a position being ten spaces to the right of the current print position, code HT will shift printing in normal mode one inch to the right. Except for enlarged mode, the print position at which the character spacing is changed is defined as the first column provided that character spacing is changed within a line. If there are no tab stops following the current print position, code HT is ignored. If the next tab stop exceeds the right margin, the printer carries out a carriage return and line feed on receipt of this code.
LF	(hex.0A) (dec.10)	When the printer receives this code, it prints all the data in the print buffer and then advances the paper by one line. The amount by which the paper is advanced depends on the current line spacing.
NUL	(hex.00) (dec.0)	Null
SI	(hex.0F) (dec.15)	Selection of condensed print mode.
SO	(hex.0E) (dec.14)	Code for double width print within one line (enlarged printing).

SOH	(hex.01) (dec.01)	**S**tart **O**f **H**eading.
SP	(hex.20) (dec.32)	Space
VT	(hex.0B) (dec.11)	When the printer receives this code, it prints the data in the print buffer and then advances the paper to the next vertical tab stop position. The next character is printed in the column following the last print position, if no CR is transmitted. The amount by which the paper is advanced varies depending on line spacing. When the printer is switched on, no vertical tab stops are set. If there are no vertical tab stops between the current print position and the end of the page, or if there are no vertical tabs set, VT assumes the same function as LF.

Appendix A

Summary of Possible Codes

Sorted by sequences

Sequence	FX80+/FX100+	FX85/FX105	FX850/FX1050	LQ850/LQ1050	LQ850+/LQ1050+	LQ2500	LQ2550	LQ2170	Function	Page
BEL	✔	✔	✔	✔	✔	✔	✔	✔	Bell	116
BS	✔	✔	✔	✔	✔	✔	✔	✔	Control Code BS (Backspace)	116
CAN	✔	✔	✔	✔	✔	✔	✔	✔	Control Code CAN (Cancel)	116
CR	✔	✔	✔	✔	✔	✔	✔	✔	Control Code CR (Carriage Return)	116
DEL	✔	✔	✔	✔	✔	✔	✔	✔	Control Code DEL (Delete)	116
DC1	✔	✔	✔	✔	✔	✔	✔	✔	set printer online	116
DC2	✔	✔	✔	✔	✔	✔	✔	✔	reset condensed print	20
DC3	✔	✔	✔	✔	✔	✔	✔	✔	set printer offline	116
DC4		✔	✔	✔	✔	✔	✔	✔	reset enlarged print mode (set by <SO> or ESC <SO>)	116
ESC	✔	✔	✔	✔	✔	✔	✔	✔	Control Code ESC	117
ESC !<n>	✔	✔	✔	✔	✔	✔	✔	✔	print mode selection	43
ESC #	✔	✔	✔	✔	✔	✔	✔	✔	input data control bit 8 unchanged	107
ESC $<n1><n2>		✔	✔	✔	✔	✔	✔	✔	absolute horizontal step	51
ESC % NUL	✔	✔	✔	✔	✔	✔	✔	✔	select download character generator	79
ESC % <1>	✔	✔	✔	✔	✔	✔	✔	✔	select ROM character generator	79
ESC & NUL <x> <y> <A> ...	✔	✔	✔						define draft download characters (9 needle version)	80
ESC & NUL <x> <y> <d_0> ...	✔	✔	✔						define NLQ download characters (9 needle version)	88
ESC & NUL <x> <y> <d_0> ...				✔	✔	✔	✔	✔	define download characters (24 needle version)	92
ESC (- <3> NUL <n1> <n2>				✔	✔				score selection	40
ESC (^								✔	print data as characters	95
ESC (C								✔	set page length in units	7
ESC (c								✔	set page format	7
ESC (G								✔	select graphics mode	67
ESC (t								✔	assign character table	105
ESC (U								✔	set unit	62
ESC (V								✔	set absolute vertical print position	59
ESC (v								✔	set relative vertical print position	59
ESC * <0><n1><n2>	✔	✔	✔	✔	✔	✔	✔	✔	graphics 60 dpi	63
ESC * <1><n1><n2>	✔	✔	✔	✔	✔	✔	✔	✔	graphics 120 dpi	63
ESC * <2> <n1> <n2>	✔	✔	✔	✔	✔	✔	✔	✔	high speed graphics 120 dpi	63
ESC * <3><n1><n2>	✔	✔	✔	✔	✔	✔	✔	✔	graphics 240 dpi	63
ESC * <4><n1><n2>	✔	✔	✔	✔	✔	✔	✔	✔	graphics 80 dpi	63

Sequence	FX80+/FX100+	FX85/FX105	FX850/FX1050	LQ850/LQ1050	LQ850+/LQ1050+	LQ2500	LQ2550	LQ2170	Function	Page
ESC * <5><n1><n2>	✔	✔	✔						graphics 72 dpi	63
ESC * <6><n1><n2>	✔	✔	✔	✔	✔	✔	✔	✔	graphics 90 dpi	63
ESC * <7><n1><n2>	✔	✔	✔						graphics 144 dpi	63
ESC * <32> <n1> <n2>				✔	✔	✔	✔	✔	graphics 60 dpi	63
ESC * <33> <n1> <n2>				✔	✔	✔	✔	✔	graphics 120 dpi	63
ESC * <38> <n1> <n2>				✔	✔	✔	✔	✔	graphics 90 dpi	63
ESC * <39> <n1> <n2>				✔	✔	✔	✔	✔	graphics 180 dpi	63
ESC * <40> <n1> <n2>				✔	✔	✔	✔	✔	graphics 360 dpi	63
ESC .								✔	print raster graphics	76
ESC + <n>				✔	✔		✔	✔	variable line spacing n/360 inch	27
ESC - <n>	✔	✔	✔	✔	✔	✔	✔	✔	underline mode ON/OFF	39
ESC / <m>	✔	✔	✔	✔	✔	✔	✔		vertical tab channel selection	57
ESC : NUL <n> NUL	✔	✔	✔	✔	✔	✔	✔	✔	copy ROM character generator	75
ESC <	✔	✔	✔	✔	✔	✔	✔	✔	unidirectional printing for one line	96
ESC =	✔	✔	✔	✔	✔	✔	✔	✔	input data control bit 8 = 0	107
ESC >	✔	✔	✔	✔	✔	✔	✔	✔	input data control bit 8 = 1	107
ESC ? <s> <0>	✔	✔	✔	✔	✔	✔	✔	✔	reassign graphics mode	107
ESC ? <s> <1>		✔	✔	✔	✔	✔	✔	✔	reassign graphics mode	107
ESC ? <s> <2>	✔	✔	✔	✔	✔	✔	✔	✔	reassign graphics mode	107
ESC ? <s> <3>	✔	✔	✔	✔	✔	✔	✔	✔	reassign graphics mode	107
ESC ? <s> <4>	✔	✔	✔	✔	✔	✔	✔	✔	reassign graphics mode	107
ESC ? <s> <5>	✔	✔	✔						reassign graphics mode	107
ESC ? <s> <6>	✔	✔	✔	✔	✔	✔	✔	✔	reassign graphics mode	107
ESC ? <s> <7>	✔								reassign graphics mode	107
ESC ? <s> <32>				✔		✔			reassign graphics mode	107
ESC ? <s> <33>				✔		✔			reassign graphics mode	107
ESC ? <s> <38>				✔		✔			reassign graphics mode	107
ESC ? <s> <39>				✔		✔			reassign graphics mode	107
ESC ? <s> <40>				✔		✔			reassign graphics mode	107
ESC @	✔	✔	✔	✔	✔	✔	✔	✔	printer initialization	114
ESC \ <n1> <n2>		✔	✔	✔	✔	✔	✔	✔	relative horizontal step	53
ESC ^<m> <n1> <n2>	✔	✔	✔						select 9 dots graphics mode	71

Sequence	FX80+/FX100+	FX85/FX105	FX850/FX1050	LQ850/LQ1050	LQ850+/LQ1050+	LQ2500	LQ2550	LQ2170	Function	Page
ESC 0	✔	✔	✔	✔	✔	✔	✔	✔	set line feed to 1/8 inch	25
ESC 1	✔	✔	✔						set line feed to 7/72 inch	25
ESC 2	✔	✔	✔	✔	✔	✔	✔	✔	set line feed to 1/6 inch	25
ESC 3 <n>	✔	✔	✔	✔	✔	✔	✔	✔	set line feed to n/216 inch	27
ESC 4	✔	✔	✔	✔	✔	✔	✔	✔	italic mode ON	42
ESC 5	✔	✔	✔	✔	✔	✔	✔	✔	italic mode OFF	42
ESC 6	✔	✔	✔	✔	✔	✔	✔	✔	print ASCII codes 128 to 159	110
ESC 7	✔	✔	✔	✔	✔	✔	✔	✔	suppress ASCII codes 128 to 159	110
ESC 8	✔	✔	✔						paper end sensor, deactivation	115
ESC 9	✔	✔	✔						paper end sensor, activation	115
ESC A <n>	✔	✔	✔	✔	✔	✔	✔	✔	line feed n/60 inch or n/72 inch	27
ESC a <n>		✔	✔	✔	✔	✔	✔		justification and centering	14
ESC B NUL	✔	✔	✔	✔	✔	✔	✔	✔	reset all vertical tab stops	55
ESC B <n1>...<n16> NUL	✔	✔	✔	✔	✔	✔	✔	✔	vertical tab stops	55
ESC b <m> NUL	✔	✔	✔	✔	✔	✔	✔		clear all tab settings	57
ESC b <m> <n> NUL	✔	✔	✔	✔	✔	✔	✔		vertical tab stops in channels	57
ESC c								✔	set HMI (horizontal motion index)	16
ESC C NUL <n>	✔	✔	✔	✔	✔	✔	✔	✔	form length in inches	5
ESC C <n>	✔	✔	✔	✔	✔	✔	✔	✔	form length in lines	5
ESC D NUL	✔	✔	✔	✔	✔	✔	✔	✔	clear all horizontal tab stops	47
ESC D <n1>...<n32> NUL	✔	✔	✔	✔	✔	✔	✔	✔	horizontal tab stops	47
ESC E	✔	✔	✔	✔	✔	✔	✔	✔	emphasized mode ON	38
ESC e									horizontal and vertical step width	49
ESC EM 0	✔	✔	✔	✔	✔	✔	✔	✔	deselect sheet feeder	108
ESC EM 1			✔		✔	✔	✔	✔	sheet feeder (magazine 1)	108
ESC EM 2			✔		✔	✔	✔	✔	sheet feeder (magazine 2)	108
ESC EM 4	✔	✔	✔	✔	✔	✔			select sheet feeder	108
ESC EM R			✔		✔	✔	✔	✔	sheet feeder (eject paper)	108
ESC F	✔	✔	✔	✔	✔	✔	✔	✔	emphasized mode OFF	38
ESC f NUL <n>									horizontal step forward	50
ESC f SOH <n>									vertical step forward	50
ESC G	✔	✔	✔	✔	✔	✔	✔	✔	double strike print mode ON	31

Sequence	FX80+/FX100+	FX85/FX105	FX850/FX1050	LQ850/LQ1050	LQ850+/LQ1050+	LQ2500	LQ2550	LQ2170	Function	Page
ESC g				✔	✔	✔	✔	✔	character density 15 cpi	21
ESC H	✔	✔	✔	✔	✔	✔	✔	✔	double strike print mode OFF	31
ESC I <n>	✔	✔	✔						printout of control codes	113
ESC i		✔							direct print mode, activate/deactivate	112
ESC J <n>	✔	✔	✔	✔	✔	✔	✔	✔	line feed n/216 inch	98
ESC j <n>		✔							back feed n/216 inch	98
ESC K <n1> <n2>	✔	✔	✔	✔	✔	✔	✔	✔	graphics 60 dpi	63
ESC k <0>			✔		✔	✔	✔	✔	font type selection (Roman)	35
ESC k <1>			✔	✔	✔	✔	✔	✔	font type selection (Gothic)	35
ESC k <2>				✔	✔	✔	✔	✔	font type selection (Courier)	35
ESC k <3>				✔	✔	✔	✔	✔	font type selection (Prestige)	35
ESC k <4>				✔	✔	✔	✔	✔	font type selection (Script)	35
ESC k <5>				✔	✔	✔	✔	✔	font type selection (OCR-B)	35
ESC k <6>				✔	✔		✔	✔	font type selection (OCR-A)	35
ESC k <7>				✔	✔		✔	✔	font type selection (Orator)	35
ESC k <8>				✔	✔		✔	✔	font type selection (Orator-S)	35
ESC L <n1> <n2>	✔	✔	✔	✔	✔	✔	✔	✔	graphics 120 dpi	63
ESC l <n>	✔	✔	✔	✔	✔	✔	✔	✔	left margin	12
ESC M	✔	✔	✔	✔	✔	✔	✔	✔	character density 12 cpi (Elite)	21
ESC m									graphics characters activate/deactivate	112
ESC N <n>	✔	✔	✔	✔	✔	✔	✔	✔	perforation skip	10
ESC O	✔	✔	✔	✔	✔	✔	✔	✔	reset perforation skip	10
ESC P	✔	✔	✔	✔	✔	✔	✔	✔	character density 10 cpi (Pica)	21
ESC p <n>	✔	✔	✔	✔	✔	✔	✔	✔	proportional spacing ON/OFF	23
ESC Q <n>	✔	✔	✔	✔	✔	✔	✔	✔	right margin	12
ESC q				✔	✔	✔	✔	✔	typestyle selection	37
ESC R <n>	✔	✔	✔	✔	✔	✔	✔	✔	national character set selection	101
ESC r <n>						✔	✔	✔	color selection	99
ESC S <n>	✔	✔	✔	✔	✔	✔	✔	✔	super/subscript mode ON	29
ESC s	✔	✔	✔	✔	✔	✔			half-speed printing	97
ESC SI		✔	✔	✔	✔	✔	✔	✔	condensed print	20
ESC SO		✔	✔	✔	✔	✔	✔	✔	enlarged print mode (automatically reset after one line)	18

Sequence	FX80+/FX100+	FX85/FX105	FX850/FX1050	LQ850/LQ1050	LQ850+/LQ1050+	LQ2500	LQ2550	LQ2170	Function	Page
ESC SP <n>		✔	✔	✔	✔	✔	✔	✔	character spacing	24
ESC T	✔	✔	✔	✔	✔	✔	✔	✔	super/subscript mode OFF	29
ESC t <0>			✔	✔	✔	✔	✔	✔	character set selection (italic)	103
ESC t <1>			✔	✔	✔	✔	✔	✔	character set selection (graphics)	103
ESC t <2>				✔	✔	✔	✔	✔	character set selection (download)	103
ESC U <n>	✔	✔	✔	✔	✔	✔	✔	✔	unidirectional printing ON/OFF	96
ESC W <n>	✔	✔	✔	✔	✔	✔	✔	✔	continuous enlarged print mode ON/OFF	18
ESC w <n>			✔	✔	✔		✔	✔	double height	46
ESC x <n>		✔	✔	✔	✔	✔	✔	✔	print quality	34
ESC Y <n1> <n2>	✔	✔	✔	✔	✔	✔	✔	✔	high speed graphics 120 dpi	63
ESC Z <n1> <n2>	✔	✔	✔	✔	✔	✔	✔	✔	graphics 240 dpi	63
ESC X								✔	select font by pitch and point	32
FF	✔	✔	✔	✔	✔	✔	✔	✔	form feed	117
HT	✔	✔	✔	✔	✔	✔	✔	✔	horizontal tab	117
LF	✔	✔	✔	✔	✔	✔	✔	✔	line feed	117
NUL	✔	✔	✔	✔	✔	✔	✔	✔	ASCII character 0	117
SI	✔	✔	✔	✔	✔	✔	✔	✔	condensed print ON	20
SO	✔	✔	✔	✔	✔	✔	✔	✔	enlarged print mode (automatically reset after one line)	18
SOH	✔	✔	✔	✔	✔	✔	✔	✔	ASCII character 1 (Start Of Heading)	118
SP	✔	✔	✔	✔	✔	✔	✔	✔	Space	118
VT	✔	✔	✔	✔	✔	✔	✔	✔	vertical tab	118

Sorted by Functions

Function	Sequence	Page
absolute horizontal step	ESC $<n1><n2>	51
ASCII character 0	NUL	117
assign character table	ESC (t	105
back feed n/216 inch	ESC j <n>	98
bell	BEL	116
character density 10 cpi (Pica)	ESC P	21
character density 12 cpi (Elite)	ESC M	21
character density 15 cpi	ESC g	21
character set selection	ESC t <n>	103
character spacing	ESC SP <n>	24
clear all horizontal tab stops	ESC D NUL	47
clear all tab settings	ESC b <m> <0>	57
color selection	ESC r <n>	99
condensed print	ESC SI	20
condensed print ON	SI	20
continuous enlarged print mode ON/OFF	ESC W <n>	18
control code BS (Backspace)	BS	116
control code CAN (Cancel)	CAN	116
control code CR (Carriage Return)	CR	116
control code DEL (Delete)	DEL	116
control code ESC (Escape)	ESC	117
copy ROM character generator	ESC : <0> <n> <m>	75
define download characters (24 needle version)	ESC & <s> <x> <y> <A> ...	92
define draft download characters (9 needle version)	ESC & NUL <x> <y> <A> ...	80
define NLQ download characters (9 needle version)	ESC & NUL <x> <y> <d$_0$> ...	88
direct print mode, activate/deactivate	ESC i	112
double height	ESC w <n>	46
double strike print mode OFF	ESC H	31
double strike print mode ON	ESC G	31
emphasized mode OFF	ESC F	38
emphasized mode ON	ESC E	38
enlarged print mode (automatically reset after one line)	ESC SO	18
enlarged print mode (automatically reset after one line)	SO	18
font type selection	ESC k <n>	35

Function	Sequence	Page
form feed	FF	117
form length in inches	ESC C NUL <n>	5
form length in lines	ESC C <n>	5
graphics 120 dpi	ESC * <1><n1><n2>	63
graphics 120 dpi	ESC * <33> <n1> <n2>	63
graphics 120 dpi	ESC L <n1> <n2>	63
graphics 120 dpi	ESC Y <n1> <n2>	63
graphics 144 dpi	ESC * <7><n1><n2>	63
graphics 180 dpi	ESC * <39> <n1> <n2>	63
graphics 240 dpi	ESC * <3><n1><n2>	63
graphics 240 dpi	ESC Z <n1> <n2>	63
graphics 360 dpi	ESC * <40> <n1> <n2>	63
graphics 60 dpi	ESC * <0><n1><n2>	63
graphics 60 dpi	ESC * <32> <n1> <n2>	63
graphics 60 dpi	ESC K <n1> <n2>	63
graphics 72 dpi	ESC * <5><n1><n2>	63
graphics 80 dpi	ESC * <4><n1><n2>	63
graphics 90 dpi	ESC * <6><n1><n2>	63
graphics 90 dpin	ESC * <38> <n1> <n2>	63
graphics characters activate/deactivate	ESC m	112
half-speed printing	ESC s	97
high speed graphics 120 dpi	ESC * <2> <n1> <n2>	63
horizontal and vertical step forward	ESC f	50
horizontal and vertical step width	ESC e	49
horizontal tab	HT	117
horizontal tab stops	ESC D <n1>...<n32> <0>	47
input data control bit 8 unchanged	ESC #	107
input data control bit 8 = 0	ESC =	107
input data control bit 8 = 1	ESC >	107
italic mode OFF	ESC 5	42
italic mode ON	ESC 4	42
justification and centering	ESC a <n>	14
left margin	ESC l <n>	12
line feed	LF	117
line feed n/216 inch	ESC J <n>	98
line feed n72 inch	ESC A <n>	27

Function	Sequence	Page
national character set selection	ESC R <n>	101
paper end sensor, activation	ESC 9	115
paper end sensor, deactivation	ESC 8	115
perforation skip	ESC N <n>	10
print ASCII codes 128 to 159	ESC 6	110
print data as characters	ESC (^	95
print mode selection	ESC !<n>	43
print raster graphics	ESC .	76
print quality	ESC x <n>	34
printer initialization	ESC @	114
printout of control codes	ESC I <n>	113
proportional spacing ON/OFF	ESC p <n>	23
reassign graphics mode	ESC ? <s> <m>	107
relative horizontal step	ESC \ <n1> <n2>	53
reset all vertical tab stops	ESC B <0>	55
reset condensed print	DC2	20
reset enlarged print mode (set by <SO> or ESC <SO>)	DC4	116
reset perforation skip	ESC O	10
right margin	ESC Q <n>	12
score selection	ESC (-	40
select 9 dots graphics mode	ESC ^<m> <n1> <n2>	71
select download character generator	ESC % NUL NUL	79
select font by pitch and point	ESC X	32
select graphics mode	ESC (G	67
select ROM character generator	ESC % <1> NUL	79
set absolute vertical print position	ESC (V	59
set HMI (horizontal motion index)	ESC c	16
set line feed to 1/6 inch	ESC 2	25
set line feed to 1/8 inch	ESC 0	25
set line feed to 7/72 inch	ESC 1	25
set line feed to n/216 inch	ESC 3 <n>	27
set page format	ESC (c	7
set page length in units	ESC (C	7
set printer offline	DC3	116
set printer online	DC1	116
set relative vertical print position	ESC (v	59

Function	Sequence	Page
set unit	ESC (U	62
sheet feeder	ESC EM <n>	108
Space	SP	118
super/subscript mode OFF	ESC T	29
super/subscript mode ON	ESC S <n>	29
suppress ASCII codes 128 to 159	ESC 7	110
typestyle selection	ESC q	37
underline mode ON/OFF	ESC - <n>	39
unidirectional printing for one line	ESC <	96
unidirectional printing ON/OFF	ESC U <n>	96
variable line spacing n/360 inch	ESC + <n>	27
vertical tab	VT	118
vertical tab channel selection	ESC / <m>	57
vertical tab stops	ESC B <n1>...<n16> <0>	55
vertical tab stops in channels	ESC b <m> <n> <0>	57

Appendix B

Character Sets

The following character sets can be selected via control panel or by Escape sequences, which correspond to the selected emulation mode.

The following example shows you how to find the hexadecimal value for a character from the character set table.

hex / dec	0	1	2	3	4
0	NUL 0	16	SP 32	0 48	@ 64
1	1	17	! 33	1 49	A 65
2	2	DC2 18	" 34	2 50	B 66
3	3	19	# 35	3 51	C 67

ASCII "B" ; dec.66 ; hex.42

Standard Character Set

hex / dec	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL 0	16	32	0 48	@ 64	P 80	` 96	p 112	NUL 128	144	160	0 176	@ 192	P 208	` 224	p 240
1	1	DC1 17	! 33	1 49	A 65	Q 81	a 97	q 113	129	DC1 145	! 161	1 177	A 193	Q 209	a 225	q 241
2	2	DC2 18	" 34	2 50	B 66	R 82	b 98	r 114	130	DC2 146	" 162	2 178	B 194	R 210	b 226	r 242
3	3	DC3 19	# 35	3 51	C 67	S 83	c 99	s 115	131	DC3 147	# 163	3 179	C 195	S 211	c 227	s 243
4	4	DC4 20	$ 36	4 52	D 68	T 84	d 100	t 116	132	DC4 148	$ 164	4 180	D 196	T 212	d 228	t 244
5	5	21	% 37	5 53	E 69	U 85	e 101	u 117	133	149	% 165	5 181	E 197	U 213	e 229	u 245
6	6	22	& 38	6 54	F 70	V 86	f 102	v 118	134	150	& 166	6 182	F 298	V 214	f 230	v 246
7	BEL 7	23	' 39	7 55	G 71	W 87	g 103	w 119	BEL 135	151	' 167	7 183	G 199	W 215	g 231	w 247
8	BS 8	CAN 24	(40	8 56	H 72	X 88	h 104	x 120	BS 136	CAN 152	(168	8 184	H 200	X 216	h 232	x 248
9	HT 9	25	) 41	9 57	I 73	Y 89	i 105	y 121	HT 137	153	) 169	9 185	I 201	Y 217	i 233	y 249
A	LF 10	26	* 42	: 58	J 74	Z 90	j 106	z 122	LF 138	154	* 170	: 186	J 202	Z 218	j 234	z 250
B	VT 11	ESC 27	+ 43	; 59	K 75	[91	k 107	{ 123	VT 139	ESC 155	+ 171	; 187	K 203	[219	k 235	{ 251
C	FF 12	28	, 44	< 60	L 76	\ 92	l 108	| 124	FF 140	156	, 172	< 188	L 204	\ 220	l 236	| 252
D	CR 13	29	- 45	= 61	M 77	] 93	m 109	} 125	CR 141	157	- 173	= 189	M 205	] 221	m 237	} 253
E	SO 14	30	. 46	> 62	N 78	^ 94	n 110	~ 126	SO 142	158	. 174	> 190	N 206	^ 222	n 238	~ 254
F	SI 15	31	/ 47	? 63	O 79	_ 95	o 111	DEL 127	SI 143	159	/ 175	? 191	O 207	_ 223	o 239	DEL 255

International Substitution Table - normal font

<table>
<tr><th rowspan="2"></th><th>dec</th><th>35</th><th>36</th><th>64</th><th>91</th><th>92</th><th>93</th><th>94</th><th>96</th><th>123</th><th>124</th><th>125</th><th>126</th></tr>
<tr><th>hex</th><th>23</th><th>24</th><th>40</th><th>5B</th><th>5C</th><th>5D</th><th>5E</th><th>60</th><th>7B</th><th>7C</th><th>7D</th><th>7E</th></tr>
<tr><td colspan="2">US ASCII</td><td>#</td><td>$</td><td>@</td><td>[</td><td>\</td><td>]</td><td>^</td><td>`</td><td>{</td><td>|</td><td>}</td><td>~</td></tr>
<tr><td colspan="2">French</td><td>#</td><td>$</td><td>à</td><td>°</td><td>ç</td><td>§</td><td>^</td><td>`</td><td>é</td><td>ù</td><td>è</td><td>¨</td></tr>
<tr><td colspan="2">German</td><td>#</td><td>$</td><td>§</td><td>Ä</td><td>Ö</td><td>Ü</td><td>^</td><td>`</td><td>ä</td><td>ö</td><td>ü</td><td>ß</td></tr>
<tr><td colspan="2">UK</td><td>£</td><td>$</td><td>@</td><td>[</td><td>\</td><td>]</td><td>^</td><td>`</td><td>{</td><td>|</td><td>}</td><td>~</td></tr>
<tr><td colspan="2">Danish I</td><td>#</td><td>$</td><td>@</td><td>Æ</td><td>Ø</td><td>Å</td><td>^</td><td>`</td><td>æ</td><td>ø</td><td>å</td><td>~</td></tr>
<tr><td colspan="2">Swedish</td><td>#</td><td>¤</td><td>É</td><td>Ä</td><td>Ö</td><td>Å</td><td>Ü</td><td>é</td><td>ä</td><td>ö</td><td>å</td><td>ü</td></tr>
<tr><td colspan="2">Italian</td><td>#</td><td>$</td><td>@</td><td>°</td><td>\</td><td>é</td><td>^</td><td>ù</td><td>à</td><td>ò</td><td>è</td><td>ì</td></tr>
<tr><td colspan="2">Spanish I</td><td>₧</td><td>$</td><td>@</td><td>¡</td><td>Ñ</td><td>¿</td><td>^</td><td>`</td><td>¨</td><td>ñ</td><td>}</td><td>~</td></tr>
<tr><td colspan="2">Japanese</td><td>#</td><td>$</td><td>@</td><td>[</td><td>¥</td><td>]</td><td>^</td><td>`</td><td>{</td><td>|</td><td>}</td><td>~</td></tr>
<tr><td colspan="2">Norwegian</td><td>#</td><td>¤</td><td>É</td><td>Æ</td><td>Ø</td><td>Å</td><td>Ü</td><td>é</td><td>æ</td><td>ø</td><td>å</td><td>ü</td></tr>
<tr><td colspan="2">Danish II</td><td>#</td><td>$</td><td>É</td><td>Æ</td><td>Ø</td><td>Å</td><td>Ü</td><td>é</td><td>æ</td><td>ø</td><td>å</td><td>ü</td></tr>
<tr><td colspan="2">Spanish II</td><td>#</td><td>$</td><td>á</td><td>¡</td><td>Ñ</td><td>¿</td><td>é</td><td>`</td><td>í</td><td>ñ</td><td>ó</td><td>ú</td></tr>
<tr><td colspan="2">L.American</td><td>#</td><td>$</td><td>á</td><td>¡</td><td>Ñ</td><td>¿</td><td>é</td><td>ü</td><td>í</td><td>ñ</td><td>ó</td><td>ú</td></tr>
</table>

International Substitution Table - italic font

	dec	163	164	192	219	220	221	222	224	251	252	253	254
	hex	A3	A4	C0	DB	DC	DD	DE	E0	FB	FC	FD	FE
US ASCII		#	$	@	[	\	]	^	`	{	|	}	~
French		#	$	à	°	ç	§	^	`	é	ù	è	¨
German		#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
UK		£	$	@	[	\	]	^	`	{	|	}	~
Danish I		#	$	@	Æ	Ø	Å	^	`	æ	ø	å	~
Swedish		#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Italian		#	$	@	°	\	é	^	ù	à	ò	è	ì
Spanish I		₧	$	@	¡	Ñ	¿	^	`	¨	ñ	}	~
Japanese		#	$	@	[	¥	]	^	`	{	|	}	~
Norwegian		#	¤	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Danish II		#	$	É	Æ	Ø	Å	Ü	é	æ	ø	å	ü
Spanish II		#	$	á	¡	Ñ	¿	é	`	í	ñ	ó	ú
L.American		#	$	á	¡	Ñ	¿	é	ü	í	ñ	ó	ú

Graphics Character Set

hex / dec	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL 0	16	32	0 48	@ 64	P 80	` 96	p 112	NUL 128	144	á 160	░ 176	└ 192	╨ 208	α 224	≡ 240
1	1	DC1 17	! 33	1 49	A 65	Q 81	a 97	q 113	129	DC1 145	í 161	▒ 177	┴ 193	╤ 209	β 225	± 241
2	2	DC2 18	" 34	2 50	B 66	R 82	b 98	r 114	130	DC2 146	ó 162	▓ 178	┬ 194	╥ 210	Γ 226	≥ 242
3	3	DC3 19	# 35	3 51	C 67	S 83	c 99	s 115	131	DC3 147	ú 163	│ 179	├ 195	╙ 211	π 227	≤ 243
4	4	DC4 20	$ 36	4 52	D 68	T 84	d 100	t 116	132	DC4 148	ñ 164	┤ 180	─ 196	╘ 212	Σ 228	⌠ 244
5	5	21	% 37	5 53	E 69	U 85	e 101	u 117	133	149	Ñ 165	╡ 181	┼ 197	╒ 213	σ 229	⌡ 245
6	6	22	& 38	6 54	F 70	V 86	f 102	v 118	134	150	ª 166	╢ 182	╞ 298	╓ 214	µ 230	÷ 246
7	BEL 7	23	' 39	7 55	G 71	W 87	g 103	w 119	BEL 135	151	º 167	╖ 183	╟ 199	╫ 215	τ 231	≈ 247
8	BS 8	CAN 24	(40	8 56	H 72	X 88	h 104	x 120	BS 136	CAN 152	¿ 168	╕ 184	╚ 200	╪ 216	Φ 232	° 248
9	HT 9	25	) 41	9 57	I 73	Y 89	i 105	y 121	HT 137	153	⌐ 169	╣ 185	╔ 201	┘ 217	Θ 233	• 249
A	LF 10	26	* 42	: 58	J 74	Z 90	j 106	z 122	LF 138	154	¬ 170	║ 186	╩ 202	┌ 218	Ω 234	· 250
B	VT 11	ESC 27	+ 43	; 59	K 75	[91	k 107	{ 123	VT 139	ESC 155	½ 171	╗ 187	╦ 203	█ 219	δ 235	√ 251
C	FF 12	28	, 44	< 60	L 76	\ 92	l 108	| 124	FF 140	156	¼ 172	╝ 188	╠ 204	▄ 220	∞ 236	η 252
D	CR 13	29	- 45	= 61	M 77	] 93	m 109	} 125	CR 141	157	¡ 173	╜ 189	═ 205	▌ 221	φ 237	^{2} 253
E	SO 14	30	. 46	> 62	N 78	^ 94	n 110	~ 126	SO 142	158	« 174	╛ 190	╬ 206	▐ 222	ε 238	■ 254
F	SI 15	31	/ 47	? 63	O 79	_ 95	o 111	DEL 127	SI 143	159	» 175	┐ 191	╧ 207	▀ 223	∩ 239	255

Standard Character Set, extended

hex / dec	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL 0	16	32	0 48	@ 64	P 80	` 96	p 112	à 128	§ 144	160	0 176	@ 192	P 208	` 224	p 240
1	1	DC1 17	! 33	1 49	A 65	Q 81	a 97	q 113	è 129	β 145	! 161	1 177	A 193	Q 209	a 225	q 241
2	2	DC2 18	" 34	2 50	B 66	R 82	b 98	r 114	ù 130	Æ 146	" 162	2 178	B 194	R 210	b 226	r 242
3	3	DC3 19	# 35	3 51	C 67	S 83	c 99	s 115	ò 131	æ 147	# 163	3 179	C 195	S 211	c 227	s 243
4	4	DC4 20	$ 36	4 52	D 68	T 84	d 100	t 116	ì 132	Ø 148	$ 164	4 180	D 196	T 212	d 228	t 244
5	5	21	% 37	5 53	E 69	U 85	e 101	u 117	° 133	ø 149	% 165	5 181	E 197	U 213	e 229	u 245
6	6	22	& 38	6 54	F 70	V 86	f 102	v 118	£ 134	¨ 150	& 166	6 182	F 298	V 214	f 230	v 246
7	BEL 7	23	' 39	7 55	G 71	W 87	g 103	w 119	¡ 135	Ä 151	' 167	7 183	G 199	W 215	g 231	w 247
8	BS 8	CAN 24	(40	8 56	H 72	X 88	h 104	x 120	¿ 136	Ö 152	(168	8 184	H 200	X 216	h 232	x 248
9	HT 9	25	) 41	9 57	I 73	Y 89	i 105	y 121	Ñ 137	Ü 153	) 169	9 185	I 201	Y 217	i 233	y 249
A	LF 10	26	* 42	: 58	J 74	Z 90	j 106	z 122	ñ 138	ä 154	* 170	: 186	J 202	Z 218	j 234	z 250
B	VT 11	ESC 27	+ 43	; 59	K 75	[91	k 107	{ 123	¤ 139	ö 155	+ 171	; 187	K 203	[219	k 235	{ 251
C	FF 12	28	, 44	< 60	L 76	\ 92	l 108	| 124	₧ 140	ü 156	, 172	< 188	L 204	\ 220	l 236	| 252
D	CR 13	29	- 45	= 61	M 77	] 93	m 109	} 125	Å 141	É 157	- 173	= 189	M 205	] 221	m 237	} 253
E	SO 14	30	. 46	> 62	N 78	^ 94	n 110	~ 126	å 142	é 158	. 174	> 190	N 206	^ 222	n 238	~ 254
F	SI 15	31	/ 47	? 63	O 79	_ 95	o 111	DEL 127	ç 143	¥ 159	/ 175	? 191	O 207	_ 223	o 239	ø 255

Graphics Character Set, extended

hex / dec	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL 0	16	32	0 48	@ 64	P 80	` 96	p 112	Ç 128	É 144	á 160	░ 176	└ 192	╨ 208	α 224	≡ 240
1	1	DC1 17	! 33	1 49	A 65	Q 81	a 97	q 113	ü 129	æ 145	í 161	▒ 177	┴ 193	╤ 209	β 225	± 241
2	2	DC2 18	" 34	2 50	B 66	R 82	b 98	r 114	é 130	Æ 146	ó 162	▓ 178	┬ 194	╥ 210	Γ 226	≥ 242
3	3	DC3 19	# 35	3 51	C 67	S 83	c 99	s 115	â 131	ô 147	ú 163	│ 179	├ 195	╙ 211	π 227	≤ 243
4	4	DC4 20	$ 36	4 52	D 68	T 84	d 100	t 116	ä 132	ö 148	ñ 164	┤ 180	─ 196	╘ 212	Σ 228	⌠ 244
5	5	§ 21	% 37	5 53	E 69	U 85	e 101	u 117	à 133	ò 149	Ñ 165	╡ 181	┼ 197	╒ 213	σ 229	⌡ 245
6	6	22	& 38	6 54	F 70	V 86	f 102	v 118	å 134	û 150	ª 166	╢ 182	╞ 298	╓ 214	μ 230	÷ 246
7	BEL 7	23	' 39	7 55	G 71	W 87	g 103	w 119	ç 135	ù 151	º 167	╖ 183	╟ 199	╫ 215	τ 231	≈ 247
8	BS 8	CAN 24	(40	8 56	H 72	X 88	h 104	x 120	ê 136	ÿ 152	¿ 168	╕ 184	╚ 200	╪ 216	Φ 232	° 248
9	HT 9	25	) 41	9 57	I 73	Y 89	i 105	y 121	ë 137	Ö 153	⌐ 169	╣ 185	╔ 201	┘ 217	Θ 233	∙ 249
A	LF 10	26	* 42	: 58	J 74	Z 90	j 106	z 122	è 138	Ü 154	¬ 170	║ 186	╩ 202	┌ 218	Ω 234	· 250
B	VT 11	ESC 27	+ 43	; 59	K 75	[91	k 107	{ 123	ï 139	¢ 155	½ 171	╗ 187	╦ 203	█ 219	δ 235	√ 251
C	FF 12	28	, 44	< 60	L 76	\ 92	l 108	| 124	î 140	£ 156	¼ 172	╝ 188	╠ 204	▄ 220	∞ 236	ⁿ 252
D	CR 13	29	- 45	= 61	M 77	] 93	m 109	} 125	ì 141	¥ 157	¡ 173	╜ 189	═ 205	▌ 221	φ 237	² 253
E	SO 14	30	. 46	> 62	N 78	^ 94	n 110	~ 126	Ä 142	₧ 158	« 174	╛ 190	╬ 206	▐ 222	ε 238	■ 254
F	SI 15	31	/ 47	? 63	O 79	_ 95	o 111	DEL 127	Å 143	ƒ 159	» 175	┐ 191	╧ 207	▀ 223	∩ 239	255

International Code Table

hex / dec	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	à 0	§ 16	32	0 48	@ 64	P 80	` 96	p 112	à 128	§ 144	160	0 176	@ 192	P 208	` 224	p 240
1	è 1	β 17	! 33	1 49	A 65	Q 81	a 97	q 113	è 129	β 145	! 161	1 177	A 193	Q 209	a 225	q 241
2	ù 2	DC2 18	" 34	2 50	B 66	R 82	b 98	r 114	ù 130	DC2 146	" 162	2 178	B 194	R 210	b 226	r 242
3	ò 3	DC3 19	# 35	3 51	C 67	S 83	c 99	s 115	ò 131	DC3 147	# 163	3 179	C 195	S 211	c 227	s 243
4	ì 4	DC4 20	$ 36	4 52	D 68	T 84	d 100	t 116	ì 132	DC4 148	$ 164	4 180	D 196	T 212	d 228	t 244
5	° 5	ø 21	% 37	5 53	E 69	U 85	e 101	u 117	° 133	ø 149	% 165	5 181	E 197	U 213	e 229	u 245
6	£ 6	¨ 22	& 38	6 54	F 70	V 86	f 102	v 118	£ 134	¨ 150	& 166	6 182	F 298	V 214	f 230	v 246
7	BEL 7	Ä 23	' 39	7 55	G 71	W 87	g 103	w 119	BEL 135	Ä 151	' 167	7 183	G 199	W 215	g 231	w 247
8	BS 8	CAN 24	(40	8 56	H 72	X 88	h 104	x 120	BS 136	CAN 152	(168	8 184	H 200	X 216	h 232	x 248
9	HT 9	Ü 25	) 41	9 57	I 73	Y 89	i 105	y 121	HT 137	Ü 153	) 169	9 185	I 201	Y 217	i 233	y 249
A	LF 10	ä 26	* 42	: 58	J 74	Z 90	j 106	z 122	LF 138	ä 154	* 170	: 186	J 202	Z 218	j 234	z 250
B	VT 11	ESC 27	+ 43	; 59	K 75	[91	k 107	{ 123	VT 139	ESC 155	+ 171	; 187	K 203	[219	k 235	{ 251
C	FF 12	ü 28	, 44	< 60	L 76	\ 92	l 108	| 124	FF 140	ü 156	, 172	< 188	L 204	\ 220	l 236	| 252
D	CR 13	É 29	– 45	= 61	M 77	] 93	m 109	} 125	CR 141	É 157	– 173	= 189	M 205	] 221	m 237	} 253
E	SO 14	é 30	. 46	> 62	N 78	^ 94	n 110	~ 126	SO 142	é 158	. 174	> 190	N 206	^ 222	n 238	~ 254
F	SI 15	¥ 31	/ 47	? 63	O 79	_ 95	o 111	DEL 127	SI 143	¥ 159	/ 175	? 191	O 207	_ 223	o 239	DEL 255

International Code Table, extended

hex / dec	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	à 0	§ 16	SP 32	0 48	@ 64	P 80	` 96	p 112	à 128	§ 144	160	0 176	@ 192	P 208	` 224	p 240
1	è 1	β 17	! 33	1 49	A 65	Q 81	a 97	q 113	è 129	β 145	! 161	1 177	A 193	Q 209	a 225	q 241
2	ù 2	DC2 18	" 34	2 50	B 66	R 82	b 98	r 114	ù 130	Æ 146	" 162	2 178	B 194	R 210	b 226	r 242
3	ò 3	DC3 19	# 35	3 51	C 67	S 83	c 99	s 115	ò 131	æ 147	# 163	3 179	C 195	S 211	c 227	s 243
4	ì 4	DC4 20	$ 36	4 52	D 68	T 84	d 100	t 116	ì 132	Ø 148	$ 164	4 180	D 196	T 212	d 228	t 244
5	° 5	ø 21	% 37	5 53	E 69	U 85	e 101	u 117	° 133	ø 149	% 165	5 181	E 197	U 213	e 229	u 245
6	£ 6	¨ 22	& 38	6 54	F 70	V 86	f 102	v 118	£ 134	¨ 150	& 166	6 182	F 298	V 214	f 230	v 246
7	BEL 7	Ä 23	' 39	7 55	G 71	W 87	g 103	w 119	¡ 135	Ä 151	' 167	7 183	G 199	W 215	g 231	w 247
8	BS 8	CAN 24	(40	8 56	H 72	X 88	h 104	x 120	¿ 136	Ö 152	(168	8 184	H 200	X 216	h 232	x 248
9	HT 9	Ü 25	) 41	9 57	I 73	Y 89	i 105	y 121	Ñ 137	Ü 153	) 169	9 185	I 201	Y 217	i 233	y 249
A	LF 10	ä 26	* 42	: 58	J 74	Z 90	j 106	z 122	ñ 138	ä 154	* 170	: 186	J 202	Z 218	j 234	z 250
B	VT 11	ESC 27	+ 43	; 59	K 75	[91	k 107	{ 123	¤ 139	ö 155	+ 171	; 187	K 203	[219	k 235	{ 251
C	FF 12	ü 28	, 44	< 60	L 76	\ 92	l 108	| 124	₧ 140	ü 156	, 172	< 188	L 204	\ 220	l 236	| 252
D	CR 13	É 29	– 45	= 61	M 77	] 93	m 109	} 125	Å 141	É 157	– 173	= 189	M 205	] 221	m 237	} 253
E	SO 14	é 30	. 46	> 62	N 78	^ 94	n 110	~ 126	å 142	é 158	. 174	> 190	N 206	^ 222	n 238	~ 254
F	SI 15	¥ 31	/ 47	? 63	O 79	_ 95	o 111	DEL 127	ç 143	¥ 159	/ 175	? 191	O 207	_ 223	o 239	ϕ 255

Table of Character Width for Proportional Printing

On the following pages you will find tables of character widths for proportional printing. The characters are represented by the printout, the decimal and the hexadecimal value.

Characters without decimal and hexadecimal codes are international characters or graphics. Please note the International Substitution Characters on pages B-3 and B-4 and the International Code Tables on page B-8 and B-9. The international characters can be selected via the sequences ESC R and ESC t (see Description of Sequences).

Please note the following for the different types of printers:

9 The values of the character widths are units of $1/120$ inch.
For example: The value in the field for character width is 12, that means, the width of the character is $12/120$ inch.
If the field consists of two values separated by a slash (for example 12/5) the first value stands for the italic character, the second for graphics.

24 The values of the character widths are units of $1/360$ inch.
For example: The value in the field for character width is 30, that means, the width of the character is $30/360$ inch.
If the field consists of two values separated by a slash (for example 30/20) the first value stands for the normal character, the second for the character width with super- or subscript activated.

ESC/P2 The values of the character widths are depending on your selected point size if using scaleable fonts. In normal print mode values described in **24** -table are also valid in **ESC/P2** -mode.

ESC/P2

Proportional width during scaleable font mode

The values of the character widths are depending on the selected point size at your printer.

The width of the scaleable fonts are based on the width of a proportional 10.5 point character.

Calculate the width as follows:

$$(\text{character width}) = \frac{\mathrm{INT}\left(\frac{(\text{point size}) \times (\text{base width})}{10.5} + 0.5\right)}{360} \text{ inch}$$

Example for a 24 point "0" *)

$$(\text{character width}) = \frac{\mathrm{INT}\left(\frac{(24) \times (30)}{10.5} + 0.5\right)}{360} \text{ inch}$$

$$(\text{character width}) = \frac{\mathrm{INT}\ (69.07)}{360} \text{ inch}$$

$$(\text{character width}) = \frac{69}{360} \text{ inch}$$

* The base width for the large character "0" is 30. Please refer to the table 24 .

Code		Char.	Width in 1/360"	Code		Char.	Width in 1/360"	Code		Char.	Width in 1/360"
hex.	dec.			hex.	dec.			hex.	dec.		
20	32		30/20	3B	59	;	18/12	56	86	V	36/24
21	33	!	18/12	3C	60	<	30/20	57	87	W	42/28
22	34	"	30/20	3D	61	=	30/20	58	88	X	36/24
23	35	#	30/20	3E	62	>	30/20	59	89	Y	36/24
24	36	$	30/20	3F	63	?	30/20	5A	90	Z	30/20
25	37	%	36/24	40	64	@	36/24	5B	91	[	24/16
26	38	&	36/24	41	65	A	36/24	5C	92	\	30/20
27	39	'	18/12	42	66	B	36/24	5D	93	]	24/16
28	40	(	24/16	43	67	C	36/24	5E	94	^	30/20
29	41	)	24/16	44	68	D	36/24	5F	95	_	30/20
2A	42	*	30/20	45	69	E	36/24	60	96	`	18/12
2B	43	+	30/20	46	70	F	36/24	61	97	a	30/20
2C	44	,	18/12	47	71	G	36/24	62	98	b	36/24
2D	45	-	30/20	48	72	H	36/24	63	99	c	30/20
2E	46	.	18/12	49	73	I	24/16	64	100	d	36/24
2F	47	/	30/20	4A	74	J	30/20	65	101	e	30/20
30	48	0	30/20	4B	75	K	36/24	66	102	f	24/16
31	49	1	30/20	4C	76	L	36/24	67	103	g	36/24
32	50	2	30/20	4D	77	M	42/28	68	104	h	36/24
33	51	3	30/20	4E	78	N	36/24	69	105	i	18/12
34	52	4	30/20	4F	79	O	36/24	6A	106	j	24/16
35	53	5	30/20	50	80	P	36/24	6B	107	k	36/24
36	54	6	30/20	51	81	Q	36/24	6C	108	l	18/12
37	55	7	30/20	52	82	R	36/24	6D	109	m	42/28
38	56	8	30/20	53	83	S	36/24	6E	110	n	36/24
39	57	9	30/20	54	84	T	36/24	6F	111	o	30/20
3A	58	:	18/12	55	85	U	42/28	70	112	p	36/24

Code		Char.	Width in 1⁄360"	Code		Char.	Width in 1⁄360"	Code		Char.	Width in 1⁄360"
hex.	dec.			hex.	dec.			hex.	dec.		
71	113	q	36/24			ì	18/12			¿	30/20
72	114	r	30/20			Ä	36/24			⌐	30/20
73	115	s	30/20			Å	36/24			¬	30/20
74	116	t	24/16			É	36/24			½	30/20
75	117	u	36/24			æ	42/28			¼	30/20
76	118	v	36/24			Æ	42/28			¡	30/20
77	119	w	42/28			ô	30/20			«	30/20
78	120	x	30/20			ö	30/20			»	30/20
79	121	y	36/24			ò	30/20	B0	176	░	30
7A	122	z	30/20			û	36/24	B1	177	▒	30
7B	123	{	24/26			ù	36/24	B2	178	▓	30
7C	124	|	18/12			ÿ	36/24	B3	179	│	30
7D	125	}	24/16			Ö	36/24	B4	180	┤	30
7E	126	~	30/20			Ü	42/28	B5	181	╡	30
		Ç	36/24			¢	30/20	B6	182	╢	30
		ü	36/24			£	30/20	B7	183	╖	30
		é	30/20			¥	36/24	B8	184	╕	30
		â	30/20			₧	42/28	B9	185	╣	30
		ä	30/20			ƒ	30/20	BA	186	║	30
		à	30/20			á	30/20	BB	187	╗	30
		å	30/20			í	18/12	BC	188	╝	30
		ç	30/20			ó	30/20	BD	189	╜	30
		ê	30/20			ú	36/24	BE	190	╛	30
		ë	30/20			ñ	36/24	BF	191	┐	30
		è	30/20			Ñ	36/24	C0	192	└	30
		ï	18/12			ª	30/20	C1	193	┴	30
		î	18/12			º	30/20	C2	194	┬	30

Code		Char.	Width in 1⁄360"
hex.	dec.		
C3	195	├	30
C4	196	─	30
C5	197	┼	30
C6	198	╞	30
C7	199	╟	30
C8	200	╚	30
C9	201	╔	30
CA	202	╩	30
CB	203	╦	30
CC	204	╠	30
CD	205	═	30
CE	206	╬	30
CF	207	╧	30
D0	208	╨	30
D1	209	╤	30
D2	210	╥	30
D3	211	╙	30
D4	212	╘	30
D5	213	╒	30
D6	214	╓	30
D7	215	╫	30
D8	216	╪	30
D9	217	┘	30
DA	218	┌	30
DB	219	█	30
DC	220	▄	30
DD	221	▌	30
DE	222	▐	30
DF	223	▀	30
E0	224	α	30/20
E1	225	ß	30/20
E2	226	Γ	30/20
E3	227	π	30/20
E4	228	Σ	30/20
E5	229	σ	30/20
E6	230	µ	30/20
E7	231	τ	30/20
E8	232	Φ	30/20
E9	233	Θ	30/20
EA	234	Ω	30/20
EB	235	δ	30/20
EC	236	∞	30/20
ED	237	φ	30/20
EE	238	ε	30/20
EF	239	∩	30/20
F0	240	≡	30
F1	241	±	30
F2	242	≥	30
F3	243	≤	30
F4	244	⌠	30
F5	245	⌡	30
F6	246	÷	30
F7	247	≈	30
F8	248	°	30
F9	249	•	30
FA	250	·	30
FB	251	√	30
FC	252	^{n}	30
FD	253	^{2}	30
FE	254	■	30
		°	24/16
		¤	30/20
		ß	36/24
		Ø	36/24
		ø	30/20
		¨	30/20
		§	30/20

Code hex.	Code dec.	Character	Width in 1/120"
20	32		12
21	33	!	5
22	34	"	8
23	35	#	12
24	36	$	12
25	37	%	12
26	38	&	12
27	39	'	5
28	40	(	6
29	41	)	6
2A	42	*	12
2B	43	+	12
2C	44	,	7
2D	45	-	12
2E	46	.	6
2F	47	/	10
30	48	0	12
31	49	1	8
32	50	2	12
33	51	3	12
34	52	4	12
35	53	5	12
36	54	6	12
37	55	7	12
38	56	8	12
39	57	9	12
3A	58	:	6

Code hex.	Code dec.	Character	Width in 1/120"
3B	59	;	6
3C	60	<	10
3D	61	=	12
3E	62	>	10
3F	63	?	12
40	64	@	12
41	65	A	12
42	66	B	12
43	67	C	12
44	68	D	12
45	69	E	12
46	70	F	12
47	71	G	12
48	72	H	12
49	73	I	8
4A	74	J	11
4B	75	K	12
4C	76	L	12
4D	77	M	12
4E	78	N	12
4F	79	O	12
50	80	P	12
51	81	Q	12
52	82	R	12
53	83	S	12
54	84	T	12
55	85	U	12

Code hex.	Code dec.	Character	Width in 1/120"
56	86	V	12
57	87	W	12
58	88	X	10
59	89	Y	12
5A	90	Z	10
5B	91	[	8
5C	92	\	10
5D	93	]	8
5E	94	^	12
5F	95	_	12
60	96	`	5
61	97	a	12
62	98	b	11
63	99	c	11
64	100	d	11
65	101	e	12
66	102	f	10
67	103	g	11
68	104	h	11
69	105	i	8
6A	106	j	9
6B	107	k	10
6C	108	l	8
6D	109	m	12
6E	110	n	11
6F	111	o	12
70	112	p	11

9

Code hex.	Code dec.	Character	Width in 1⁄120"
71	113	q	11
72	114	r	11
73	115	s	12
74	116	t	11
75	117	u	12
76	118	v	12
77	119	w	12
78	120	x	10
79	121	y	12
7A	122	z	10
7B	123	{	9
7C	124	¦	5
7D	125	}	9
7E	126	~	12
7F	127		–

Code hex.	Code dec.	Character italic	Character graph.	Width in 1⁄120"
			Ç	12
			ü	11
			é	12
			â	12
			ä	12
			à	12
			å	12
			ç	11
			ê	12
			ë	12
			è	12
			ï	8
			î	10
			ì	8
			Ä	12
			Å	12
			É	12
			æ	12
			Æ	12
			ô	10
			ö	10
			ò	10
			û	11
			ù	11
			ÿ	12
			Ö	12
			Ü	12

Code hex.	Code dec.	Character italic	Character graph.	Width in 1⁄120"
			¢	11
			£	12
			¥	12
			₧	12
			ƒ	11
			á	12/12
		!	í	10/8
		"	ó	10/10
		#	ú	12/11
		$	ñ	11/11
		%	Ñ	12/12
		&	ª	12/12
		'	º	5/12
		(	¿	8/12
		)	⌐	8/12
		*	¬	12/12
		+	½	12/12
		,	¼	8/12
		-	¡	12/5
		.	«	7/12
		/	»	10/12
B0	176	0	░	12/12
B1	177	1	▒	9/12
B2	178	2	▓	12/12
B3	179	3	│	12/12
B4	180	4	┤	12/12
B5	181	5	╡	12/12

Code		Character		Width in 1/120"
hex.	dec.	italic	graph.	
B6	182	*6*	╢	11/12
B7	183	*7*	╖	12/12
B8	184	*8*	╕	12/12
B9	185	*9*	╣	11/12
BA	186	*:*	║	8/12
BB	187	*;*	╗	9/12
BC	188	*<*	╝	10/12
BD	189	*=*	╜	11/12
BE	190	*>*	╛	9/12
BF	191	*?*	┐	11/12
C0	192	*@*	└	12/12
C1	193	*A*	┴	12/12
C2	194	*B*	┬	12/12
C3	195	*C*	├	12/12
C4	196	*D*	─	12/12
C5	197	*E*	┼	12/12
C6	198	*F*	╞	12/12
C7	199	*G*	╟	12/12
C8	200	*H*	╚	12/12
C9	201	*I*	╔	10/12
CA	202	*J*	╩	12/12
CB	203	*K*	╦	12/12
CC	204	*L*	╠	10/12
CD	205	*M*	═	12/12
CE	206	*N*	╬	12/12
CF	207	*O*	╧	12/12
C0	208	*P*	╨	12/12

Code		Character		Width in 1/120"
hex.	dec.	italic	graph.	
D1	209	*Q*	╤	12/12
D2	210	*R*	╥	12/12
D3	211	*S*	╙	12/12
D4	212	*T*	╘	12/12
D5	213	*U*	╒	12/12
D6	214	*V*	╓	11/12
D7	215	*W*	╫	12/12
D8	216	*X*	╪	12/12
D9	217	*Y*	┘	12/12
DA	218	*Z*	┌	12/12
DB	219	*[*	█	11/12
DC	220	**	▄	7/12
DD	221	*]*	▌	11/12
DE	222	^	▐	10/12
DF	223	_	▀	12/12
E0	224	`	α	5/12
E1	225	*a*	β	11/11
E2	226	*b*	Γ	11/10
E3	227	*c*	π	11/12
E4	228	*d*	Σ	12/10
E5	229	*e*	σ	11/11
E6	230	*f*	µ	12/11
E7	231	*g*	τ	11/12
E8	232	*h*	Φ	11/10
E9	234	*i*	Θ	9/12
EA	235	*j*	Ω	10/12
EB	236	*k*	δ	11/12

Code		Character		Width in 1/120"
hex.	dec.	italic	graph.	
EC	236	*l*	∞	9/12
ED	237	*m*	ø	11/12
EE	238	*n*	∈	10/10
EF	239	*o*	∩	11/10
F0	240	*p*	≡	11/12
F1	241	*q*	±	11/12
F2	242	*r*	≥	10/10
F3	243	*s*	≤	11/10
F4	244	*t*	⌠	10/12
F5	245	*u*	⌡	11/12
F6	246	*v*	÷	10/12
F7	247	*w*	≈	12/12
F8	248	*x*	°	12/8
F9	249	*y*	•	11/6
FA	250	*z*	·	126
FB	251	*{*	√	10/12
FC	252	*|*	ⁿ	9/8
FD	253	*}*	²	10/8
FE	254	~	■	12/8
FF	255			12/12

TRADEMARK ACKNOWLEDGES

“EPSON” and “EPSON ESC/P” are registered trademarks and EPSON ESC/P 2 is a trademark of SEIKO EPSON Corporation.
“IBM” is a trademark of International Business Machines Corporation.

Programmer’s Reference

Tally ANSI (MTPL) Emulation

Table of Contents

Description of MTPL-Sequences

Appendix A Summary of Possible Codes

Appendix B Character Sets

Appendix C Additional Technical Description

Index

Description of MTPL-Sequences

Introduction

Regardless of the specific MTPL sequences of your printer, this User's Manual describes the sum of all MTPL sequences and control codes. The only differentiation is made between 9 and 24 needle printers as well as the maximum possible paper width of 8 inches (80 column printer) or 13.6 inches (132 column printer). Please note that the print quality LQ (Letter Quality) is only available in the 24 needle printer.

Please be sure to observe the notes and steps described in the Operator's Manual as well as the specific MTPL sequences implemented in your printer and which of the described sequences are thus not available.

Select the MTPL sequence as described in the Operator's Manual, chapter 2.
Apart from the command language MTPL your printer also understands other printer emulation sequences and commands. MTPL and an additional emulation can be active at the same time. For this refer to the chapter 2, "Emulations", Operator's Manual.

The following explanations will help you understand the sequences better:

Every sequence description begins with a header, in which the function and the short form of the sequence are listed without parameters, e.g.:

Set Form Lenght in Lines	CSI ... t

With the **CSI** (hex.9B, dec.155) control code the most MTPL sequences are introduced.

*) MTPL: "Tally Printer Language"

CSI means “**C**ontrol **S**equence **I**ntroducer” and is described by the code 9B (hex.) or ESC [:

7-Bit-environment → hex. 1B, hex. 5B (= ESC [ )
8-Bit-environment → hex. 9B

It is sensible to use only ESC [as “Control Sequence Introducer” if MTPL is used associated with any other printer emulation.

The last character (in the following example **t**) specifies the function of the sequence.

The header is followed by the **Data Structure** in ASCII, hexadecimal und decimal syntax with the necessary parameters, e.g.:

ASCII	CSI	n	"t"	[A] set form lenght in lines
hex.	9B	n	74	
dec.	155	n	116	

For the parameter (here **n**) it is necessary to differentiate between two types of syntax:

- if the parameter is in pointed parentheses, the decimal value is transferred
- if the parameter is **not** in pointed parentheses, the ASCII value is transferred

Example:

Parameter syntax: **<n>**, with n=0
to be transmitted: dec.0 (hex.00)

Parameter syntax: **n**, with n=0
to be transmitted: ASCII "0" (hex.30, dec.48)

Each sequence description or a number of the available parameters is additionally specified with a code level from A to C in square brackets (e.g. “[A]”).

This three code levels are defined as follows:

[A] This code level is to be used for creating new printer driver versions.
[B] This is an optional extension of the MTPL standard.
[C] This function is to be used to ensure compatibility tp previous products. When creating new printer driver versions this code level should not be used anymore.

Character explanation and symbol description

- ℓ — Lower case "ℓ"
- i — Informations
- 9 — Sequence only applies for 9 needle printer
- 24 — Sequence only applies for 24 needle printer

In the following you will find an example for a MTPL sequence with a Basic programming example:

Set to a form length of 72 lines:

MTPL-Sequence		CSI	n with n=72	t
Transmission	ASCII	CSI	"72"	"t"
	hex.	9B	37 32	74
	dec.	155	55 50	116

```
100 REM Sample for the CSI n t sequence in ASCII,
110 REM using formulars with 72 lines.
120 REM Please note, in ASCII-Syntax you can use ASCII values
130 REM equal or bigger codetable no. 32 only.
140 LPRINT CHR$(155);"72";"t": REM mixed syntax;
150 REM set form length up to 72 lines
160 REM The same sequence written in hexadecimal syntax
170 LPRINT CHR$(&H9B);CHR$(&H37);CHR$(&H32);CHR$(&H74);
180 REM set form length up to 12 lines
190 REM The same sequence written in decimal syntax
200 LPRINT CHR$(155);CHR$(55);CHR$(50);CHR$(116);
210 REM set form length up to 72 lines
```

Form Length — CSI ... t

Data Sructure	ASCII	CSI	n	"t"	[A] set form lenght in lines
	hex.	9B	n	74	
	dec.	155	n	116	

Description

PUM turned OFF:

n = 1...132 lines at a line spacing of 1⁄6 inch (6 lpi)
= 1...176 lines at a line spacing of 1⁄8 inch (8 lpi)

This MTPL-sequence sets the form length to n times the current line feed pitch (in inches). Also the current print position is simultaneously defined as top of form.

The value specified for n must be in the range 1 to 255. This value is multiplied by the current line feed pitch to obtain the form length. For example, if 60 is specified for n and the current line feed is 1⁄6 inch, the form length is set to 10 inches. Once set by this sequence, the form length is not affected by changing the line spacing.

PUM turned ON:

The unit for n is either the decipoint or defined by the SSU function (Select Size Unit, see page 8).

n = 0: Reset form lenght to default value.
(print menu setting)

Example

```
10  REM    form length
20  LPRINT CHR$(27);"[4t";
30  FOR F=1 TO 2
40  FOR L=1 TO 3
50  LPRINT "length 4:form";F;"  line";L
60  NEXT L:LPRINT CHR$(12);:NEXT F
70  LPRINT
80  LPRINT CHR$(27);"[3t";
90  FOR F=1 TO 3
100 FOR L=1 TO 2
110 LPRINT "length 3:form";F;"  line";L
120 NEXT L:LPRINT CHR$(12);:NEXT F
130 END
```

```
length 4:form 1   line 1
length 4:form 1   line 2
length 4:form 1   line 3

length 4:form 2   line 1
length 4:form 2   line 2
length 4:form 2   line 3


length 3:form 1   line 1
length 3:form 1   line 2

length 3:form 2   line 1
length 3:form 2   line 2

length 3:form 3   line 1
length 3:form 3   line 2
```

Turn ON Positioning Unit Mode (SM) — CSI 11 h
Turn OFF Positioning Unit Mode (RM) — CSI 11 ℓ

Data Structure

ASCII	CSI	"1"	"1"	"h"	[A] turn PUM ON
hex.	9B	31	31	68	
dec.	155	49	49	104	
ASCII	CSI	"1"	"1"	"ℓ"	[A] turn PUM OFF
hex.	9B	31	31	69	
dec.	155	49	49	105	

Description

PUM = **P**ositioning **U**nit **M**ode

This mode decides, wether positioning commands or functions that deal with horizontal or vertical position parameters, have characters and line positions as parameters or some fixed units defined by the SSU control function (**S**elect **S**ize **U**nit, see the following sequence).

If PUM is turned on and no unit is defined, the units shall default to decipoints ($1/720$ of an inch).

If PUM is turned off, the units shall default to line or character distances.

By default the PUM mode is turned off. If PUM ist activated the accuracy cannot exceed the vertical or horizontal resolution of your printer.

Select Size Unit (SSU) — CSI ... SP I

Data Structure

ASCII	CSI	n	SP	"I"	[B] select size unit
hex.	9B	n	SP	49	
dec.	155	n	SP	73	

Description

For n the following parameters are available:

n	hex.	dec.	size unit
0	30	48	Decipoints ($1/720$ inch = 0,0353 mm)
1	31	49	Millimeters
2	32	50	Decipoints (0,0353 mm)
3	33	51	Decididots (0,0376 mm)
4	34	52	Mils ($1/1000$ inch = 0,0254 mm)
5	35	53	Basic Measuring Unit BMU ($1/1200$ inch = 0,02117 mm)
6	36	54	Micrometer (0,001 mm)
7	37	55	Pixels ❶

❶ This unit is machine dependent and may be of different size horizontally and vertically. Page printers support typically 200, 300 or 600 dpi, matrix printers e.g. 180 or 360 dpi.

Example:

CSI	720	SP	I
1B 55	37 32 30	20	49

results in one inch

By default the size unit is the decipoint. Other size units may not be available due used printer type.

PUM must be set to ON (see page 7)

Set Top and Bottom Margin — CSI ... r
Clear Top and Bottom Margin — CSI r

Data Structure

ASCII	CSI	n_1	";"	n_2	"r"	[A] set top and bottom margin
hex.	9B	n_1	3B	n_2	72	
dec.	155	n_1	59	n_2	114	

ASCII	CSI	"r"	[A] clear top and bottom margin
hex.	9B	72	
dec.	155	114	

Description

n_1 specifies the value of the top margin
n_2 specifies the value of the bottom margin

n_1, n_2 = 1...132 lines at a line spacing of 1⁄6 inch (6 lpi)
= 1...176 lines at a line spacing of 1⁄8 inch (8 lpi)

Examples:

CSI n_1 r	set top margin, set bottom margin to default
CSI ; 0 r	clear top and bottom margin
CSI ; n_2 r	clear top margin, set bottom margin

Any change of margin settings is only effective from the next page on.

The setting of the top and bottom margins depends on the line spacing which is defined before setting the margins. If line spacing is set after the setting of the top and bottom margins, the margin positions on this page are not influenced.

If the form length is changed, the top margin is set to the first line, the bottom margin is set to the last line (= form length).

If there are any logical problems (for example, the setting of top margin is larger then the page length), the sequence is ignored. If only one margin is changed the other margin is automatically reset to the default value.

The unit depends on PUM and is either character positions or defined by the SSU function (Select Size Unit, see page 8).

Example

```
10  REM    top and bottom margin
20  LPRINT CHR$(27);"[6t";:REM form length
30  LPRINT CHR$(27);"[2;5r":REM set margins
40  FOR F=1 TO 2
50  FOR L=2 TO 5
60  LPRINT "form";F;"  line ";L
70  NEXT L:NEXT F
80  LPRINT
90  LPRINT CHR$(27);"[0;0r";:REM clear margins
100 FOR F=1 TO 2
110 FOR L=1 TO 6
120 LPRINT "form";F;"  line ";L
130 NEXT L:NEXT F
140 END
```

```
form 1   line  2
form 1   line  3
form 1   line  4
form 1   line  5


form 2   line  2
form 2   line  3
form 2   line  4
form 2   line  5



form 1   line  1
form 1   line  2
form 1   line  3
form 1   line  4
form 1   line  5
form 1   line  6
form 2   line  1
form 2   line  2
form 2   line  3
form 2   line  4
form 2   line  5
form 2   line  6
```

Set Left and Right Margin	**CSI ... s**
Clear Left and Right Margin	**CSI s**

Data Structure

ASCII	CSI	n_1	";"	n_2	"s"	[A] set left and right margin
hex.	9B	n_1	3B	n_2	73	
dec.	155	n_1	59	n_2	115	

ASCII	CSI	"s"	[A] clear left and right margin
hex.	9B	73	
dec.	155	115	

Description

n_1 specifies the value of the left margin
n_2 specifies the value of the right margin

	Print width 8 inches	Print width 13,6 inches
n_1, n_2	= 1... 80 at 10 cpi	= 1...136 at 10 cpi
	= 1... 96 at 12 cpi	= 1...163 at 12 cpi
	= 1...120 at 15 cpi	= 1...204 at 15 cpi
	= 1...136 at 17.1 cpi	= 1...232 at 17.1 cpi
	= 1...160 at 20 cpi	= 1...272 at 20 cpi

Values for n_1, n_2 are valid for a line length of 8 inches (80 column printer) or 13.6 inches (136 column printer) set in the printer menu.

Example:

CSI n_1 s	set left margin, set right margin to default
CSI s	clear left and right margin
CSI ; n_2 s	clear left margin, set right margin

The setting of the left and right margins depends on the current character density.
Any later setting of the character density will not influence the positions of the left and right margins on this page.

The unit depends on PUM and is either character positions or defined by the SSU function (Select Size Unit, see page 8).

If only one margin is changed, the other margin will be reset to default value.

If there are any logical problems (for example, left margin > right margin), the sequence is ignored.

Example

```
10 REM    left and right margin
20 WIDTH "LPT1:",255
30 LPRINT CHR$(27);"[10;53s";
40 LPRINT "Now the left margin is set to column 10 and the ";
50 LPRINT "right margin is set to column 53."
60 LPRINT CHR$(27);"[5;41s";
70 LPRINT "Now the left margin has been reduced to column 5 ";
80 LPRINT "and the right margin has been set to column 41."
90 END
```

```
     Now the left margin is set to column 10 and
     the right margin is set to column 53.
Now the left margin has been reduced
to column 5 and the right margin has
been set to column 41.
```

Set Line Home (SLH) — CSI ... SP U
Set Line Limit (SLL) — CSI ... SP V

Data Structure

ASCII	CSI	n	SP	"U"	[A] set line home	❶
hex.	9B	n	20	55		
dec.	155	n	32	85		
ASCII	CSI	n	SP	"V"	[A] set line limit	❷
hex.	9B	n	20	55		
dec.	155	n	32	85		

Description

n = 1...maximal print area

❶ Default setting: Physical left margin of medium or device.

n specifies the left margin. The first position is 1. The unit depends on PUM (Positioning Unit Mode) and is either characters or defined by SSU (Select Size Unit, see page 8).

Example:

CSI SP U	reset to physical left margin
CSI 10 SP U	set left margin to position 10

❷ Default setting: Physical right margin (=1) of medium or device.

n specifies the right margin.
The unit depends on PUM (Positioning Unit Mode) and is either character positions or defined by SSU (Select Size Unit, see page 8).

Example:

CSI SP V	reset to physical right margin
CSI 80 SP V	set right margin to position 80

Right Justification ON	**CSI 8 y**
Centering ON	**CSI 9 y**
Right Justification and Centering OFF	**CSI 10 y**
Right Justification and Centering OFF	**CSI : y**

Data Structure

ASCII	CSI	n	"y"	justification
hex.	9B	n	79	
dec.	155	n	121	

Description

For n the following characters are available:

n	hex.	dec.	justification	
8	38	56	[A] right justification ON	❶
9	39	57	[A] centering ON	❷
10	31 30	49 48	[A] right justification and centering OFF	
:	3A	58	[C] right justification and centering OFF	

❶ Leading blanks will not be corrected. Blanks at line end will be ignored.

❷ The text is centered between the active margins.

Character Spacing

Spacing Increment (SPI) — CSI ... SP G

Data Structure

ASCII	CSI	n_1	";"	n_2	SP	"G"	[B] [C]	set spacing increment
hex.	9B	n_1	3B	n_2	20	47		
dec.	155I	n_1	59	n_2	32	71		

Description

n_1 specifies the line spacing
n_2 specifies the character spacing

The unit is expressed in decipoints or other units defined by SSU (Select Size Unit, see page 8).

For this note also the sequence “Set Line Spacing (SLS)” on page 26.

Example:

CSI 120 ; 0 G ≙ 6 lpi, basic cpi
CSI 90 ; 60 G ≙ 8 lpi, 12 cpi

The default setting of the spacing increment normally is 10 cpi.

n = 0: The distance is reset to the values set in the printer menu.

Horizontal Spacing — CSI ... w

Data Structure

ASCII	CSI	n	"w"	[A] set cpi
hex.	9B	n	77	
dec.	155	n	119	

Description

For n the following parameters are available:

n	hex.	dec.	horizontal spacing	
0	30	48		5 cpi
1	31	49		6 cpi
2	32	50		7,5 cpi
3	33	51	[B] [C]	8,6 cpi
4	34	52		10 cpi
5	35	53		12 cpi
6	36	54		15 cpi
7	37	55	[B] [C]	17,1 cpi
11	31 31	49 49	[B] [C]	20 cpi
12	31 32	49 50	[C]	10* cpi

10* cpi in this sequence means higher horizontal resolution of the printed characters (emphasized printing), but lower print speed.

Various character densities can also be used within one line.

The default value for the character density is the printer menu setting.

Example

```
10  REM    character densities
20  LPRINT CHR$(27);"[4wthis is 10 cpi, ";
30  LPRINT CHR$(27);"[12wthis is 10 cpi (high resolution), "
40  LPRINT CHR$(27);"[5wthis is 12 cpi, ";
50  LPRINT CHR$(27);"[6wthis is 15 cpi, ";
60  LPRINT CHR$(27);"[7wthis is 17.1 cpi, ";
70  LPRINT CHR$(27);"[11wthis is 20 cpi, "
80  LPRINT CHR$(27);"[0wthis is 5 cpi, ";
90  LPRINT CHR$(27);"[1wthis is 6 cpi, "
100 LPRINT CHR$(27);"[2wthis is 7.5 cpi, ";
110 LPRINT CHR$(27);"[3wand 8.6 cpi"
120 END
```

this is 10 cpi, this is 10 cpi (high resolution),
this is 12 cpi, this is 15 cpi, this is 17.1 cpi, this is 20 cpi,
this is 5 cpi, this is 6 cpi,
this is 7.5 cpi, and 8.6 cpi

Horizontal Character Spacing (SHS) — CSI ... SP K

Data Structure

ASCII	CSI	n	SP	"K"	[A] set cpi
hex.	9B	n	20	4B	
dec.	155	n	32	75	

Description

For n the following parameters are available:

n	hex.	dec.	character spacing
0	30	48	10 cpi
1	31	49	6 cpi
2	32	50	12 cpi
3	33	51	15 cpi

With this sequence the horizontal character spacing is defined.

The change of the character dimensions is implementation dependent.

Character Spacing (SCS)	CSI ... SP g

Data Structure

ASCII	CSI	n	SP	"g"	[A] set character spacing
hex.	9B	n	20	67	
dec.	155	n	32	103	

Description

n specifies the character spacing

The unit is either decipoints or depends on the SSU selection (Select Size Unit, see page 8).

The character size will not be changed.

Additional Character Spacing (SACS) — CSI ... SP \

Data Structure

ASCII	CSI	n	SP	" \"	[A] set additional character spacing
hex.	9B	n	20	5C	
dec.	155	n	32	92	

Description

n specifies the additional character spacing

The units are either decipoints or defined by SSU (Select Size Unit, see page 8).

This function enlarges the inter character spacing. The function will typically be used with proportional spacing to get effects like “spaced out” or right justification.

With fixed spacing, functions like SCS (Set Character Spacing, see previous page 19) should be used.

The character size will not be changed.

Reduced Character Spacing (SRCS) — CSI ... SP f

Data Structure

ASCII	CSI	n	SP	"f"	[B] set reduced character spacing
hex.	9B	n	20	66	
dec.	155	n	32	102	

Description

n specifies the reduced character spacing

The units are either decipoints or defined by SSU (Select Size Unit, see page 8).

This function reduces the inter character spacing. The function will typically be used with proportional spacing to get special effects like e.g. kerning.

With fixed spacing, functions like SCS (Set Character pacing, see page 19) should be used.

The character size will not be changed.

Horizontal Motion Index (HMI) — CSI ... p

Data Sructure

ASCII	CSI	n	"p"	[C] set HMI
hex.	9B	n	70	
dec.	155	n	112	

Description

n = 0...20

The normal character density is enlarged by n ∗1⁄120 inch. If n is set to zero (default setting), HMI is disabled.

Example

```
10 REM    horizontal motion index
20 LPRINT "this is normal width"
30 LPRINT CHR$(27);"[3p";
40 LPRINT "now 3/120 inch is added to normal width"
50 LPRINT CHR$(27);"[6p";
60 LPRINT "now 6/120 inch is added to normal width"
70 END
```

```
this is normal width
n o w  3 / 1 2 0  i n c h  i s  a d d e d  t o  n o r m a l  w i d t h
n o w   6 / 1 2 0   i n c h   i s   a d d e d   t o   n o r m a l   w i d t h
```

Horizontal Step (Microspace) — CSI ... q

Data Structure

ASCII	CSI	n	"q"	[C] set horizontal step
hex.	9B	n	71	
dec.	155	n	113	

Description

n = 1...255

The space between characters is enlarged by n $*\frac{1}{120}$ inch at the current print position.

Examples:

CSI q	enlarging character space by n $*\frac{1}{120}$ inch
CSI 10 q	inserts a $\frac{1}{12}$ inch space

Example

```
10 REM    horizontal step
20 LPRINT "in the word ‘hori";
30 LPRINT CHR$(27);"[20q";
40 LPRINT "zontal’ there is a step of 20/120 inch"
50 END
```

```
in the word `hori  zontal` there is a step of 20/120 inch
```

Line Spacing

Vertical Spacing (SVS) CSI ... SP L

Data Structure

ASCII	CSI	n	SP	"L"	[A] set vertical spacing (lpi)
hex.	9B	n	20	4C	
dec.	155	n	32	76	

Description

For n the following parameters are available:

n	hex.	dec.	vertical spacing
0	30	48	6 lpi
1	31	49	4 lpi
2	32	50	3 lpi
3	33	51	12 lpi
4	34	52	8 lpi
9	39	57	2 lpi

Note that changing the line spacing also changes the number of lines per page. Changing of line spacing does not affect the form length set before hand.

Line Density 6 lpi	**CSI 3 z**
Line Density 8 lpi	**CSI 4 z**

Data Structure

ASCII	CSI	"3"	"z"	[C] line density 6 lpi (= line spacing 1/6 inch)
hex.	9B	33	7A	
dec.	155	51	122	
ASCII	CSI	"4"	"z"	[C] line density 8 lpi (= line spacing 1/8 inch)
hex.	9B	34	7A	
dec.	155	52	122	

Description

These sequences set the line spacing to 1⁄6 inch (6 lpi line density) or 1⁄8 inch (8 lpi line density).

Note that changing the line spacing also changes the number of lines per page. Changing of line spacing does not affect the current vertical tab stops or form length.

Example

```
10 REM    line spacing
20 LPRINT CHR$(27);"[4z";
30 LPRINT "these two lines are printed with"
40 LPRINT "a spacing of 1/8 inch"
50 LPRINT
60 LPRINT CHR$(27);"[3z";
70 LPRINT "these two lines are printed with"
80 LPRINT "a spacing of 1/6 inch"
90 END
```

```
these two lines are printed with
a spacing of 1/8 inch

these two lines are printed with
a spacing of 1/6 inch
```

Set Line Spacing (SLS) — CSI ... SP h

Data Structure

ASCII	CSI	n	SP	"h"	[A] set line spacing
hex.	9B	n	20	68	
dec.	155	n	32	104	

Description

n specifies the line spacing

The unit is either decipoints or depends on the SSU selection (Select Size Unit, see page 8). Note also the sequence SPI (Spacing Increment, see page 15).

Character Styling

Superscript ON	**CSI 0 z**
Subscript ON	**CSI 1 z**
Microscript ON	**CSI 10 z**
Super-/Sub-/Microscript OFF	**CSI 2 z**

Data Structure

ASCII	CSI	n	"z"	[A] character styling
hex.	9B	n	7A	
dec.	155	n	122	

Description

For n the following parameters are availble:

n	hex.	dec.	character styling	
0	30	48	superscript ON	❶
1	31	49	subscript ON	❶
2	32	50	super-/sub-/microscript OFF	❷
10	31 30	49 48	microscript ON	❶

❶ These sequences switch the printer to superscript, subscript or microscript mode. The subsequent characters are printed with about ½ of their normal height. If superscript or subscript characters are underlined, the underline character is printed at its normal position.

❷ This sequence resets the printer to normal mode.
The base line for microprint is the same as for normal print.

The sequence **CSI 2 z** also resets double height.
(For this see also the following page 29, Double Height OFF)

Example

```
10  REM    super/sub/microscript
20  LPRINT "E=M*C";
30  LPRINT CHR$(27);"[0z";
40  LPRINT "2";
50  LPRINT CHR$(27);"[2z";
60  LPRINT " is Einsteins most famous formula."
70  LPRINT "H";
80  LPRINT CHR$(27);"[1z";"2";CHR$(27);"[2z";
90  LPRINT "O is simply water."
100 LPRINT "Micro";CHR$(27);"[10zscript";CHR$(27);"[2z";
110 LPRINT "is printed in the base line"
120 END
```

E=M*C^2 is Einsteins most famous formula.

H$_2$O is simply water.

Microscript is printed in the base line

Double Height ON (Upper Half)	**CSI 12 z**
Double Height ON (Lower Half)	**CSI 13 z**
Double Height OFF	**CSI 2 z**

Data Structure

ASCII	CSI	"1"	"2"	"z"	[C] double height ON (upper half)	❶
hex.	9B	31	32	7A		
dec.	155	49	50	122		
ASCII	CSI	"1"	"3"	"z"	[C] double height ON (lower half)	❶
hex.	9B	31	33	7A		
dec.	155	49	51	122		
ASCII	CSI	"2"	"z"		[A] double height OFF	❷
hex.	9B	32	7A			
dec.	155	50	122			

Description

❶ The same character must be sent to both lines to form a full character. Afterwards double height must be reset. If only a part of the line is printed double height, the positioning of upper/lower half must be done by spaces. The line spacing should be set to 1⁄6 inch.

❷ This sequence also resets superscript, subscript and microscript.

Note: For selection of double height together with the proportional mode or other print attributes the start of printing has to be identic for the top and bottom half.

Example

```
10  REM   double height
20  LPRINT "For ";CHR$(27);"[12zHeadlines ";CHR$(27);"[2z";
30  LPRINT "printing in double"
40  LPRINT CHR$(27);"[13z    Headlines";CHR$(27);"[2z"
50  LPRINT "height is recommended"
60  END
```

```
For Headlines printing in double

height is recommended
```

Print Quality

CSI ... SP X
CSI ... y

Data Structure

ASCII	CSI	n	SP	"X"	[A] select print quality	❶
hex.	9B	n	20	58		
dec.	155	n	32	88		
ASCII	CSI	n	"y"		select print quality	❷
hex.	9B	n	79			
dec.	155	n	121			

Description

❶ This sequence defines the print quality. Print throughput changes with the print quality.

For n the following parameters are available:
(printer menu setting is valid)

n	hex.	dec.	print quality	
0	30	48	[A] [B]	high
1	31	49	[A]	medium
2	32	50	[A]	low

❷ With the sequence **CSI n y** for n the following parameters are available:

n	hex.	dec.	typeface
0	30	48	[A] draft print quality (DPQ)
1	31	49	[A] near letter quality (NLQ)
4	34	52	[C] NLQ at 10 cpi
5	35	43	[C] NLQ at 12 cpi
11	31 31	49 49	[B] fast draft print quality
12	31 32	49 50	[A] letter print quality (LQ) - e.g. 24∗36 matrix
13	31 33	49 51	[B] high resolution print quality - e.g. 24∗48 matrix

Example

```
10 REM    print quality
20 LPRINT CHR$(27);"[0yThis is draft quality ";
30 LPRINT CHR$(27);"[1yand this is NLQ printing."
40 LPRINT CHR$(27);"[5yThis is NLQ with 12 cpi."
50 END
```

```
This is draft quality and this is NLQ printing.
This is NLQ with 12 cpi.
```

Select Typestyle — CSI ... m

Data Structure

ASCII	CSI	n	"m"	[A] select typestyle
hex.	9B	n	6D	
dec.	155	n	109	

Description

This sequence defines the typestyle.

For n the following parameters are available:

n	hex.	dec.	typestyle (number is printer dependent)
10	31 30	49 48	font 0
11	31 31	49 49	font 1
12	31 32	49 50	font 2
13	31 33	49 51	font 3
14	31 34	49 52	font 4
15	31 35	49 53	font 5
16	31 36	49 54	font 6
17	31 37	49 55	font 7
18	31 38	49 56	font 8
19	31 39	49 57	font 9
0	30	48	clear all selected fonts

Depending on the value specified for n, one of the internal printer fonts or a font of an optional font card is selected. Refer to the Operator's Manual for information on printer-resident fonts and font cards which can be installed.

The order of the font 0 to font 9 not corresponds to the fonts displayed on your printer. The sequence and allocation is defined by the sequence **CSI n_1 ; n_2 SP D** (see FNT, Font Selection, page 43).

A maximum of 16 CSI...m -sequences can be joined in one sequence: e.g. as follows: **CSI n_1 ; n_2 ; n_x m.**

Example Valid for most MTPL printers.

Fontregister	Font
0	Draft
1	NLQ Courier
2	LQ Courier
3	NLQ Sans Serif
4	LQ Sans Serif
5	LQ Roman
6	LQ Script
7	LQ Prestige
8	LQ OCR-B
9	LQ OCR-A

Emphasized Mode ON — CSI = z
Emphasized Mode OFF — CSI > z

Data Structure

ASCII	CSI	"="	"z"	[C] emphasized mode ON
hex.	9B	3D	7A	
dec.	155	61	122	

ASCII	CSI	">"	"z"	[C] emphasized mode OFF
hex.	9B	3E	7A	
dec.	155	62	122	

Description

During printing in emphasized mode, each dot is printed twice, slightly shifted in horizontal direction.

Emphasized mode can be used in all print qualities and character densities.

Example

```
10 REM    emphasized mode
20 LPRINT CHR$(27);"[z";"this is printed in the normal mode"
30 LPRINT CHR$(27);"[=z";"and this in the emphasized one"
40 END
```

```
this is printed in the normal mode
and this in the emphasized one
```

Italic Mode ON	**CSI 3 m**
Italic Mode OFF	**CSI 23 m**

Data Structure

ASCII	CSI	"3"	"m"		[A] italic mode ON
hex.	9B	33	6D		
dec.	155	51	109		
ASCII	CSI	"2"	"3"	"m"	[A] italic mode OFF
hex.	9B	32	33	6D	
dec.	155	50	51	109	

The sequence **CSI 0 m** resets all CSI...m-sequences!

A maximum of 16 CSI...m -sequences can be joined in one sequence, e.g. as follows:
CSI n_1;n_2;n_x m

Example

```
10 REM   italic mode
20 LPRINT CHR$(27);"[3m"
30 GOSUB 70
40 LPRINT CHR$(27);"[23m"
50 GOSUB 70
60 END
70 LPRINT "Matrix Printer"
80 RETURN
```

Matrix Printer
Matrix Printer

Proportional Mode ON (SGR)	**CSI 26 m**
	CSI 2 y
Proportional Mode OFF	**CSI 50 m**
	CSI 7 y

Data Structure

ASCII	CSI	"2"	"6"	"m"	[A] proportional mode ON
hex.	9B	32	36	6D	
dec.	155	50	54	109	
ASCII	CSI	"2"	"y"		[C] proportional mode ON
hex.	9B	32	79		
dec.	155	50	121		
ASCII	CSI	"5"	"0"	"m"	[A] proportional mode OFF
hex.	9B	35	30	6D	
dec.	155	53	48	109	
ASCII	CSI	"7"	"y"		[C] proportional mode OFF
hex.	9B	37	79		
dec.	155	55	121		

Description

When in default font type, all characters are assigned a fixed, equally wide space. When in proportional mode, the spaces between each character are set to the actual width of the characters; on account of these “proportional spaces” reading of the text becomes more easy.

Various font types basically are printed in proportional mode, e.g. Script, Kaufmann etc.

The sequence **CSI 0 m** resets all CSI...m-sequences!

A maximum of 16 CSI...m -sequences can be joined in one sequence, e.g. as follows:
CSI n_1;n_2;n_x m

Example

```
10 REM    proportional spacing
20 LPRINT CHR$(27);"[2y";
30 LPRINT "this is an example with proportional spacing"
40 LPRINT CHR$(27);"[7y";
50 LPRINT "this is an example with normal spacing"
60 END
```

```
this is an example with proportional spacing
this is an example with normal spacing
```

Underline Mode ON (SGR)	**CSI 4 m**
Underline Double Mode	**CSI 21 m**
Underline Mode OFF	**CSI 24 m**

Data Structure

ASCII	CSI	"4"	"m"		[A] underline mode ON	❶
hex.	9B	34	6D			
dec.	155	52	109			

ASCII	CSI	"2"	"1"	"m"	[A] underline double mode
hex.	9B	32	31	6D	
dec.	155	50	49	109	

ASCII	CSI	"2"	"4"	"m"	[A] underline mode OFF
hex.	9B	32	34	6D	
dec.	155	50	52	109	

Description

❶ All printed characters including spaces are automatically unerlined. Spaces between tab codes (HT) are underlined.

The sequence **CSI 0 m** resets all CSI...m-sequences!

A maximum of 16 CSI...m -sequences can be joined in one sequence, e.g. as follows:
CSI $n_1;n_2;n_x$ m

Example

```
10 REM    underline mode
20 LPRINT "the most ";
30 LPRINT CHR$(27);"[4m";
40 LPRINT "important";
50 LPRINT CHR$(27);"[24m";
60 LPRINT " word must be underlined."
70 END
```

the most important word must be underlined.

Overline Mode ON (SGR) — CSI 53 m
Overline Mode OFF — CSI 55 m

Data Structure

ASCII	CSI	"5"	"3"	"m"	[A] overline mode ON
hex.	9B	35	33	6D	
dec.	155	53	51	109	

ASCII	CSI	"5"	"5"	"m"	[A] overline mode OFF
hex.	9B	35	35	6D	
dec.	155	53	53	109	

Description

This sequence switches the automatic overline function on or off. When the automatic overline function is on, all printed characters including spaces are automatically underlined (also see SGR, Underline Mode ON, previous page 39).

The sequence **CSI 0 m** resets all CSI...m-sequences!

A maximum of 16 CSI...m -sequences can be joined in one sequence, e.g. as follows: **CSI n_1;n_2;n_x m**

Double Strike ON	CSI 9 w
Double Strike OFF	CSI 8 w

Data Structure

ASCII	CSI	"9"	"w"	[B] [C] double strike print mode ON
hex.	9B	39	77	
dec.	155	57	119	

ASCII	CSI	"8"	"w"	[B] [C] double strike print mode OFF
hex.	9B	38	77	
dec.	155	56	119	

Description

In double strike mode characters are printed twice in two printing passes. Double strike mode can be used in all print qualities and character densities. Double strike mode can also be used in combination with emphasized mode.

STOP

The use of this sequence in combination with LQ and emphasized printing is not recommended.

Example

```
10 REM double strike
20 LPRINT CHR$(27);"[8w";
30 LPRINT "this is normal printing";
40 LPRINT CHR$(27);"[9w";
50 LPRINT "and this is double strike printing"
60 END
```

this is normal printing

and this is double strike printing

Graphic Size Selection (GSS)	**CSI ... SP C**
Graphic Size Modification (GSM)	**CSI ... SP B**

Data Structure

ASCII	CSI	n	SP	"C"			[B] graphic size selection	❶
hex.	9B	n	20	43				
dec.	155	n	32	67				

ASCII	CSI	n_1	;	n_2	SP	"B"	[A] graphic size modification	❷
hex.	9B	n_1	3B	n_2	20	42		
dec.	155	n_1	59	n_2	32	66		

Description

❶ n specifies the hight of the used font

The units are either decipoints or defined by the SSU function (Select Size Unit, see page 8).

The width is implicitly defined by the height.

Example:

CSI 120 SP C set 12 point font (= 120 decipoints)

❷ n_1 specifies the height of the used fonts (default setting: 100)
n_2 specifies its width (default setting: 100)

These parameters are given as a percent value of the size establishment by sequence ❶.

Example:

CSI SP B	standard height and width
CSI 50 ; 50 SP B	half sized characters (e.g. for indices or exponents)
CSI 200 ; 200 SP B	double sized characters (e.g. for headlines)
CSI 100 ; 67 SP B	compresses a 10 pitch font for 15 cpi

If the selected character size is not printable, the next available character size is used.

Font Designation (FNT) — CSI ... SP D

Data Structure

ASCII	CSI	n_1	;	n_2	SP	"D"	[A] designate font
hex.	9B	n_1	3B	n_2	20	44	
dec.	155	n_1	59	n_2	32	68	

Description

n_1 = font number 0...9
n_2 = identification (ID) of the desired font

For n_2 the following parameters are available:

n_2	hex.	dec.	identification
0	30	48	Roman
1	31	49	Sans Serif
2	32	50	Courier
3	33	51	Prestige
4	34	52	Script
5	35	53	OCR B
6	36	54	OCR A
7	37	55	Modern
8	38	56	Kaufmann
9	39	57	Gothic
10	31 30	49 48	Swiss
11	31 31	49 49	Quadrato
66	36 36	54 54	Courier IBM

The first parameter (n_1) selects one of ten font registers to which the second parameter (n_2) designates one font of the whole repertory in the device. With the sequence CSI 1...m the designated font can be selected (see page 33, Select Typestyle).

n_1 is in the range of zero to nine and selects primary font, first alternate font etc. up to the nineth alternate font.

The second parameter (n_2) is defined by your printer type and represents an identification of the desired font.

Font attributes, e.g. “bold” can be specified after designation through corresponding ESC sequences.

Technical Details Font Designation Codes

The use of the font designation code **CSIn$_1$;n$_2$space D** needs further explanation. (Note: CSI is 9B; you can use ESC[, 1B 5BH, instead if you wish.)

The parameter n$_1$ Register to use.

This paramer can take values of 0 to 9. The default allocation of fonts to these registers is given in the printer’s manual.

The parameter n$_2$ What to put in the register.

This parameter is held as a 16 bit number in the printer. It is made up as below:

The lower eight bits (0 to 255)
These encode the typeface and are as given in the manual with the exception that 5 is OCR-B and 6 is OCR-A.

0 = Roman	1 = Sans Serif	2 = Courier	4 = Script
5 = OCR-B	6 = OCR-A	7 = Modern	8 = Kaufmann
9 = Gothic	10 = Helvette	11 = Quadrato	66 = Courier IBM

The upper eight bits
Only bits 9 (512H) and 11 (2048H) are used. All the other bits must be left at 0. Bits 9 and 11 set print quality:

Bit 11	Bit 9	
0	0	Letter Quality
0	1	Draft
1	0	Near Letter Quality
1	1	Reserved

When Draft (01) is selected the lower eight bits have a new meaning:
0 = Multicopy Draft 1 = Fast Draft 2 = Normal Draft

to specify a type face with a print quality add the decimal values and convert to an ASCII string.

Example 1:	Courier LQ	= 00 + 2	Number = 2
Example 2:	Script NLQ	= 10 (2048) + 4	Number = 2052
Example 3:	Draft	= 01 (512) + 1	Number = 513

So **CSI7;513 D** i.e.: 9B 37 3B 35 31 33 20 44H (or **ESC[7;513 D** i.e. 1B 5B 37 3B 35 31 33 20 44 H) will put Fast Draft in font register 7. The sequence **ESC[17m** will select this register as the font to use.

Print Positioning

Horizontal Tab Stop — HT

Data Structure

ASCII	HT	[A]	horizontal tab stop
hex.	09		
dec.	9		

Description

The HT-Code moves the active print position to the next horizontal tabulation stop on the same line. If there is no tab stop reachable - no more tab stops are set or the next tab stop is beyond the defined right margin - spaces are inserted.

The graphic renditions, e.g. underlining, overscoring etc., being active during the tab is executed, apply also to the whitespace produced by two tab stops.

Set Horizontal Tab Stop at Current Position — HTS

Data Structure

ASCII	HTS	[A]	set horizontal tab stop
hex.	88		
dec.	136		

Description

This code sets a tab stop at the current horizontal position.

Horizontal Tab Stops ON	**CSI ... u**
	ESC H
Horizontal Tab Stops OFF (TBC)	**CSI ... g**

Data Structure

ASCII	CSI	n_1	"u"	[A] set horizontal tab stop	❶
hex.	9B	n_1	75		
dec.	155	n_1	117		
ASCII	ESC	"H"		[C] set horizontal tab stop at current position	❷
hex.	1B	48			
dec.	27	72			
ASCII	CSI	n	"g"	[A] clear all horizontal tab stops	❸
hex.	9B	n	67		
dec.	155	n	103		

Description

❶

	Print width 8 inches	Print width 13,6 inches
n_1	= 1... 80 at 10 cpi	= 1...136 at 10 cpi
	= 1... 96 at 12 cpi	= 1...163 at 12 cpi
	= 1...120 at 15 cpi	= 1...204 at 15 cpi
	= 1...136 at 17.1 cpi	= 1...232 at 17.1 cpi
	= 1...160 at 20 cpi	= 1...272 at 20 cpi

Values for n_1 are valid for a line length of 8 inches (80-column printer) or 13.6 inches (136-column printer) set in the menu.

Up to 16 tab stops can be joined in one sequence as follows:

CSI n_1 ; n_2 ; n_3 ; ... ; n_{16} u

The parameters do not have to be sorted. The positions of the horizontal tab stops depend on the current character density.

❷ This sequence is interpreted as a Epson/IBM-sequence (ESC H $\triangleq$ LQ off), if the emulation MTPL+Epson/IBM is set.

❸ With the sequence **CSI n g** for n the following parameters are available:

n	hex.	dec.	function
0	30	48	clear horizontal tab stop at current position
2	32	50	clear all horizontal tab stops in current line
3	33	51	clear all horizontal tab stops in all lines

Examples:

CSI g	clear horizontal tab stop at current position
CSI 3;4 g	clear all horizontal and vertical tab stops (for this also see page 52, Vertical Tab Stops ON/OFF)

Later changes of character density will not influence the physical position of the tab stops.

A horizontal tabulation is executed by the HT control code (hex.09, dec.9); for this see page 45, Horizontal Tab Stop.

Example 1

```
10  REM    horizontal tabs
20  LPRINT CHR$(27);"[4w";:REM 10 cpi
30  LPRINT CHR$(27);"[10;20;30u";
40  LPRINT "123456789012345678901234567890123456789"
50  GOSUB 110
60  LPRINT CHR$(27);"[0w";:REM 5 cpi
70  GOSUB 110
80  LPRINT CHR$(27);"[11w";:REM 20 cpi
90  GOSUB 110
100 END
110 LPRINT CHR$(9);"tab1";
120 LPRINT CHR$(9);"tab2";
130 LPRINT CHR$(9);"tab3"
140 RETURN
```

```
123456789012345678901234567890123456789
         tab1      tab2      tab3
         t a b 1   t a b 2   t a b 3
         tab1      tab2      tab3
```

Example 2

```
10 FOR I=1 TO 3
20 LPRINT "1234567890";CHR$(27);"H";
30 NEXT I
40 LPRINT
50 FOR J=1 TO 3
60 LPRINT CHR$(9);"tab";
70 NEXT J
80 LPRINT
90 END
```

```
123456789012345678901234567890
          tab       tab       tab
```

Line Feed **LF**

Data Structure	ASCII	LF	[A] line feed
	hex.	0A	
	dec.	10	

Description

When the printer receives this code, it prints all the data in the print buffer, then advances the paper one line.

The LF code can also give a carriage return if automatic carriage return is selected. If no CR is given, the next printing starts in the column following the end of printing.

The amount by which the paper is advanced depends on the current line spacing. The default line spacing of the printer is ⅙ inch.

Vertical Tab Stop VT

Data Structure

ASCII	VT	vertical tab stop
hex.	0B	
dec.	11	

Description

When the printer receives this code, it prints all the data in the print buffer, then advances the paper to the next vertical tab stop position.

The VT code can also give a carriage return if automatic carriage return is selected (refer to Operator’s Manual). If no CR is given, the next printing starts in the column following the end of printing.

The amount by which the paper is advanced depends on the current line spacing. After switching on the printer no vertical tab stops are set (default configuration). If there are no vertical tab stops between the current print position and the end of the page, or if there are no vertical tab stops set, VT assumes the same function as LF.

Set Vertical Tab Stop at Current Position **VTS**

Data Structure	ASCII	VTS	[A] set vertical tab stop
	hex.	8A	
	dec.	138	

Description This code sets an horizontal tab stop at the current vertical position.

Vertical Index **IND**

Data Structure	ASCII	IND	[A] vertical index
	hex.	84	
	dec.	132	

Description The code IND moves the active print position to the following line. The difference to Line Feed (LF) is that IND does not permit an CR option.

Vertical Tab Stops ON Vertical Tab Stops OFF (TBC)	**CSI ... v** **CSI 1 g** **CSI 4 g**

Data Structure

ASCII	CSI	n	"v"	[A] set vertical tab stop ❶
hex.	9B	n	76	
dec.	155	n	118	
ASCII	CSI	"1"	"g"	[A] clear all vertical tab stops at current position
hex.	9B	31	67	
dec.	155	49	103	
ASCII	CSI	"4"	"g"	[A] clear all vertical tab stops
hex.	9B	34	67	
dec.	155	52	103	

Description

❶ n = 1...132 lines at a line spacing of 1⁄6 inch (6 lpi)
= 1...176 lines at a line spacing of 1⁄8 inch (8 lpi)

Up to 16 tab stops can be joined in one sequence as follows:

CSI n_1 ; n_2 ; n_3 ; ... ; n_{16} v

The parameters do not have to be sorted. The positions of the horizontal tab stops depend on the current character density.

Examples:

CSI g	clear horizontal tab stop at current position
CSI 3;4 g	clear all horizontal and vertical tab stops (for this also see page 46, Horizontal Tab Stops ON/OFF)

Later changes of character density will not influence the physical position of the tab stops.

A vertical tabulation is executed by the VT control code (hex.0B, dec.11); also see page 50, Vertical Tab Stop.

Example

```
10 REM    vertical tabs
20 LPRINT CHR$(27);"[10t";:REM set form length
30 LPRINT CHR$(27);"[4g";:REM clear all vtabs
40 LPRINT CHR$(27);"[3;5;9v";
50 LPRINT CHR$(11);"this is the first  vertical tab, line 3"
60 LPRINT CHR$(11);"this is the second vertical tab, line 5"
70 LPRINT CHR$(11);"this is the third  vertical tab, line 9"
80 END
```

```
this is the first  vertical tab, line 3

this is the second vertical tab, line 5



this is the third  vertical tab, line 9
```

Horizontal Posititon Absolute (HPA) — CSI ... ’

Data Structure

ASCII	CSI	n	" ‘ "	[A] set horizontal position absolute
hex.	9B	n	60	
dec.	155	n	96	

Description

n = 1 (default setting) ≙ current left margin

This sequence moves the print head to a horizontal print position absolute.

The unit depends on PUM (Positioning Unit Mode) and is either character positions or defined by the SSU-function (Select Size Unit, see page 8).

Example:

CSI 40 ‘	next character goes to position 40
CSI ‘	next character goes to the left margin

> Positions outside the right margin are ignored.
>
> Make sure you do not confuse the character “ **‘** ” with the apostrophe “ **’** ” !

Horizontal Position Relative (HPR) — CSI ... a

Data Structure

ASCII	CSI	n	"a"	[A] set horizontal position relative
hex.	CSI	n	61	
dec.	CSI	n	97	

Description

n = 1 (default setting)

This sequence moves the print head to the right.

The unit depends on PUM (Positioning Unit Mode) and is either character positions or defined by the SSU-function (Select Size Unit, see page 8).

Example:

CSI 8 a	next character goes 8 positions to the right
CSI a	next character goes one position to the right

Positions beyond the right margin are ignored.

Horizontal Position Backward (HPB) — CSI ... j

Data Structure

ASCII	CSI	n	"j"	[A] set horizontal position backward
hex.	9B	n	6A	
dec.	155	n	106	

Description

n = 1 (default setting)

This sequence moves the print head to the left.

The unit depends on PUM (Positioning Unit Mode) and is either character positions or defined by the SSU-function (Select Size Unit, see page 8).

Example:

CSI 8 j	next character goes 8 positions to the left
CSI j	next character goes one position to the left

Positions outside the left margin are ignored.

Vertical Position Absolute (VPA) CSI ... d

Data Structure	ASCII	CSI	n	d	[A] set vertical position absolute
	hex.	9B	n	64	
	dec.	155	n	100	

Description n = 1 (default setting) ≙ current top margin

This sequence moves the print head to a vertical print position absolute.

The unit depends on PUM (Positioning Unit Mode) and is either lines or defined by the SSU-function (Select Size Unit, see page 8).

Example:

CSI 40 d	move the active print position to line (or vertical position) 40
CSI d	move the active print position to top margin

Positions outside the bottom margin are ignored.

Vertical Position Relative (VPR) — CSI ... e

Data Structure

ASCII	CSI	n	"e"	[A] set vertical position relative
hex.	9B	n	65	
dec.	155	n	101	

Description

n = 1 (default setting)

This sequence moves the print head lines or vertical positions down.

The unit depends on PUM (Positioning Unit Mode) and is either lines or defined by the SSU-function (Select Size Unit, see page 8).

Example:

CSI 12 e	move print position 12 lines (or vertical positions) down
CSI e	move print position one line down

Positions beyond the bottom margin are ignored.

Vertical Position Backward (VPB) — CSI ... k

Data Structure

ASCII	CSI	n	"k"	[A] set vertical position backward
hex.	9B	n	6B	
dec.	155	n	107	

Description

n = 1 (default setting)

This sequence moves the print head lines or vertical positions upward.

The unit depends on PUM (Positioning Unit Mode) and is either lines or defined by the SSU-function (Select Size Unit, see page 8).

Positions beyond the top margin are ignored. This sequence is not valid in Single Sheet Mode.

Horizontal and Vertical Position Absolute (HVP) — CSI ... f

Data Structure

ASCII	CSI	n_1	;	n_2	"f"	[B] [C] set horizontal and vertical position absolute
hex.	9B	n_1	3B	n_2	66	
dec.	155	n_1	59	n_2	102	

Description

n_1 = vertical position
n_2 = horizontal position

This sequence moves the print head to a horizontal and vertical print position absolute.

The unit depends on PUM (Positioning Unit Mode) and is either lines or defined by the SSU-function (Select Size Unit, see page 8).

Example:

CSI f	move the print position to the top left position on page
CSI 20 ; 35 f	move the print position to line (vertical position) 20 and horizontal character position 35 (if PUM is turned off)

Note that during manual and automatic cut sheet feeding your printer can only reverse the form a limited distance.

Backspace BS

Data Structure	ASCII	BS	[A] Backspace
	hex.	08	
	dec.	8	

Description

After receiving this code the content of line buffer is printed out.

Then, the print head is moved one position to the left (depending on the current values of cpi).

Carriage Return CR

Data Structure	ASCII	CR	[A] carriage return
	hex.	0D	
	dec.	13	

Description

When the printer receives this code, it prints any data in the print buffer and resets the line pointer. The active print position is set to the left margin (first possible character position).

The code CR can also give a line feed if automatic line feed is selected (refer to Operator's Manual).

Form Feed FF

Data Structure	ASCII	FF	[A] form feed
	hex.	0C	
	dec.	12	

Description

When the printer receives this code, it prints the data in the print buffer, then advances the paper from the current pritn position to the top-of-form position on the next page.

Cut sheet:

Forms are ejected. The first print position on the following paper is the first (top) physically possible position.

Fanfold paper:

The paper is moved by the form length given before. The first print position on the following paper is the first (top) physically possible position.

The FF code can also give a carriage return if automatic carriage return is selected (refer to Operator’s Manual). The next print is started in the column following the print end of the last print-out if no CR is transmitted.

After switching on the printer the default form length is set to 12 inches (72 lines) per page (Europe 12"; USA 11").

Partial Line Down **PLD**

Data Structure	ASCII	PLD	[A] partial line down
	hex.	8B	
	dec.	139	

Description

If the preceding character is in normal position, PLD moves the active position down to a sufficient distance to give the following characters the appearence of being subscripted. If the preceding character is in a superscript position, PLD moves to normal position.

Partial Line Up **PLU**

Data Structure	ASCII	PLU	[A] partial line up
	hex.	8C	
	dec.	140	

Description

If the preceding character is in normal position, PLU moves the active position up a sufficient distance to give the following characters the appearence of being superscripted. If the preceding character is in a subscript position, PLU moves to normal position.

Reverse Index **RI**

Data Structure	ASCII	RI	[B] reverse index
	hex.	8D	
	dec.	141	

Description The code RI moves the active position to the preceding line. An CR option is not permitted.

Next Line **NEL**

Data Structure	ASCII	NEL	[B] next line
	hex.	85	
	dec.	133	

Description The code NEL moves the active position to the left margin of the next line. This gives the same result as the combination of CR (Carrigage Return) and LF (Line Feed).

Graphics

Graphic Modes

ESC % ...
CSI 6 z
CSI 5 z

Data Structure

ASCII	ESC	"%"	n	n_1	;	n_2	;	DATA	[C] set dpi for graphics
hex.	1B	25	n	n_1	3B	n_2	3B	DATA	
dec.	27	37	n	n_1	59	n_2	59	DATA	

ASCII	CSI	"6"	"z"	[C] graphics invers
hex.	9B	36	7A	
dec.	155	54	122	

ASCII	CSI	"5"	"z"	[C] graphics unchanged
hex.	9B	35	7A	
dec.	155	53	122	

Description

For n the following parameters are available:

n			dpi
3	33	51	graphics 60 dpi
4	34	52	graphics 120 dpi
5	35	53	graphics 80 dpi
6	36	54	graphics 240 dpi
7	37	55	graphics 240 dpi
8	38	56	graphics 72 dpi
9	39	57	graphics 90 dpi

Data following the graphic-CSI-sequence are printed out as dot pattern. n_1 and n_2 define the length of the data sequence.

The vertical resolution is 72 dpi.

The graphics image is printed vertically closed at a line feed of 80⁄720 inch. (Also see the sequences Spacing Increment (SPI), page 15, and Set Line Spacing (SLS), page 26. Example: **CSI 80 ; SP G**).

Example for calculation of n_1 and n_2:

The length of graphics is 80 bytes.

n_2 = number of bytes divided by 256
= int (80⁄256)
= dec.0
= hex.00

n_1 = remainder of division of n_2
= 80 mod 256
= dec.80
= hex.50

Graphics data and text data can be mixed within one line.

24 Relationship between graphics data and needles:

Needle	Data byte
1/3	Bit 7
4/5	Bit 6
6/8	Bit 5
9/10	Bit 4
11/13	Bit 3
14/15	Bit 2
16/18	Bit 1
19/20	Bit 0

9 Relationship between graphics data and needles:

Needle	Data byte
1	Bit 7
2	Bit 6
3	Bit 5
4	Bit 4
5	Bit 3
6	Bit 2
7	Bit 1
8	Bit 0

Example 1

```
10  REM   bit image print
20  WIDTH "LPT1:",255
30  LPRINT CHR$(27);"%3";CHR$(80);CHR$(0);
40  GOSUB 100:LPRINT " 60 dpi"
50  LPRINT CHR$(27);"%4";CHR$(80);CHR$(0);
60  GOSUB 100:LPRINT " 120 dpi"
70  LPRINT CHR$(27);"%7";CHR$(80);CHR$(0);
80  GOSUB 100:LPRINT " 240 dpi"
90  END
100 FOR N=1 TO 80
110 LPRINT CHR$(255);
120 NEXT N
130 RETURN
```

Example 2

```
10  REM   bit image print
20  WIDTH "LPT1:",255
30  LPRINT CHR$(27);"%3";CHR$(10);CHR$(0);
40  GOSUB 130:REM 60 dpi
50  LPRINT CHR$(27);"%4";CHR$(10);CHR$(0);
60  GOSUB 130:REM 120 dpi
70  LPRINT CHR$(27);"[6z";
80  LPRINT CHR$(27);"%3";CHR$(10);CHR$(0);
90  GOSUB 130:REM reverse 60 dpi
100 LPRINT CHR$(27);"[5z";
110 LPRINT
120 END
130 RESTORE
140 FOR I=1 TO 10
150 READ D
160 LPRINT CHR$(D);
170 NEXT I
180 LPRINT "   ";
190 RETURN
200 DATA 34,80,138,0,143,0,138,80,34,0
```

Load Menu — CSI ... SP q

Data Structure

ASCII	CSI	n	SP	"q"	[B] load menu
hex.	9B	n	20	71	
dec.	155	n	32	113	

Description

For n the following parameters are available:

n	hex.	dec.	Loading of
0	30	48	Default menu
1	31	49	Menu No. 1
2	32	50	Menu No. 2
3	33	51	Menu No. 3
4	34	52	Menu No. 4 (reserved)

Up to 3 (4) various configurations can be saved with the printer.
Setting the configuration is made by the control panel of the printer (refer to Operator’s Manual, chapter 2, “Menu”).

Example:
CSI 0 SP q or CSI SP q load default menu

By using this sequence all settings changed by sequences are reset!

Automatic Gap Adjustment (AGA) — CSI ... + z
Direct Setting of the Print Head Distance — CSI ... + y

Data Structure

ASCII	CSI	n	"+"	"z"	[B] automatic gap adjustment ❶
hex.	9B	n	2B	7A	
dec.	155	n	43	122	

ASCII	CSI	n	"+"	"y"	[B] direct setting of the print head distance ❷
hex.	9B	n	2B	79	
dec.	155	n	43	121	

Description

❶ For the sequence **CSI n + z** the following parameters are available for n:

n	hex.	dec.	Adjustment (automatic)
0	30	48	automatic adjustment OFF (print head distance must be set at the printer)
1	31	49	automatic adjustment ON

❷ For the sequence **CSI n + y** the following parameters are available for n:

n	hex.	dec.	Setting (direct)
0	30	48	Default value (is set at the printer)
n			print head distance in *n* -steps

Setting the print head distance accords to the respective paper path selected. Unit, adjust range and default setting depend on the device used.
If the Automatic Gap Adjustment is set to ON, the sequence ❷ is ignored.

Example:

Settings of T2060:	Unit =	10 Micrometers
	Adjust range =	1 – 100
	Default value =	18

Paper Handling — CSI Ps1 Ps2 Ps3 Pn4 Pn5 + v

Data Structure

ASCII	CSI	Ps1 Ps2 Ps3 Pn4 Pn5	"+"	"v"
hex.	9B	Ps1 Ps2 Ps3 Pn4 Pn5	2B	76
dec.	155	Ps1 Ps2 Ps3 Pn4 Pn5	43	118

Description

Ps1	missing or	0:	Printhead does not lift off at the fold
		1:	Printhead lifts off at the fold
Ps2	missing or	0:	Printhead position at left margin during paper movement allowed
		1:	Printhead position at left margin during paper movement not allowed
Ps3	missing or	0:	Printhead position at right margin during paper movement allowed
		1:	Printhead position at right margin during paper movement not allowed
Pn4	missing or	<20 or >136:	No limitation of print width
	20 … 136:		Print width in 1/10 inch
Pn5	missing or	0 or >30:	No displacement of the start of the print area (left margin);
	1 … 20:		Displacement of the start of the print area in 1/10 inches

Shared Interface — CSI Pn + w

Data Structure

ASCII	CSI	Pn	"+"	"w"
hex.	9B	Pn	2B	77
dec.	155	Pn	43	119

Description

Sequence to set hold time of shared I/O after selecting alternate interface.

Pn	missing or	0:	Sequence ignored
Pn	1 … 30:		Hold time in seconds

Shared I/O has to be selected in the printer’s menu. Default = 30 s.

Color Selection (Option) — CSI ... m

Data Structure

ASCII	CSI	n	"m"	[A] set colors
hex.	9B	n	6D	
dec.	155	n	109	

Description

For n the following parameters are available:

n	hex.	dec.	color selection	
0	30	48	reset color	
1	31	49	bold or increased intensity	
2	32	50	alternate color or decreased intensity	❶
22	32 32	50 50	cancels functions of parameters 1 and 2 (normal color or normal intensity)	
30	33 30	51 48	black	
31	33 31	51 49	red	
32	33 32	51 50	green	
33	33 33	51 51	yellow	❷
34	33 34	51 52	blue	
35	33 35	51 53	magenta-red	❷
36	33 36	51 54	cyan-blue	❷
39	33 39	51 57	default color blac	

❶ red or magenta, depending on the ribbon used
❷ only available with multi color ribbon

The available colors depend on the ribbon used and must be matched in the menu. If a ribbon with several black tracks is used, then it is automatically switched every new page between the tracks.

The relationship between tracks and colors is shown in the following table:

track	yellow, cyan, magenta, black (YMCK)	red, green, blue, black (RGBK)	3 ∗ black, red
1	black	black	black
2	cyan	blue	black
3	magenta	green	black
4	yellow	red	red

The relationship between color selection sequences and the resulting colors for several ribbon cassettes is shown in the following table:

selected color	yellow, magenta, cyan, black (YMCK)	red, green, blue, black (RGBK)	3 ∗ black, red
black	black	black	black
red	red *)	red	red
green	green **)	green	****)
yellow	yellow	green	****)
blue	blue ***)	blue	****)
magenta	magenta	red	red
cyan	cyan	blue	****)

* red = yellow + magenta
** green = yellow + cyan
*** blue = magenta + cyan
} two-pass print is automatically executed
**** sequence is ignorered

The sequence **CSI 0 m** resets all CSI...m -sequences! A maximum of 16 CSI...m -sequences can be joined in one sequence: e.g. **CSI n_1;n_2;n_x m**.

Example

```
10 REM    color selection
20 LPRINT CHR$(27);"[31m";
30 LPRINT "this is written with red colour"
40 LPRINT CHR$(27);"[34m";
50 LPRINT "this is written with blue colour"
60 END
```

```
this is written with red colour
this is written with blue colour
```

Input Data Control — CSI ... z

Data Structure

ASCII	CSI	"9"	"z"	[C] set bit 8 = 1	❶
hex.	9B	39	7A		
dec.	155	57	122		
ASCII	CSI	"8"	"z"	[C] set bit 8 = 0	❷
hex.	9B	38	7A		
dec.	155	56	122		
ASCII	CSI	"7"	"z"	[C] bit 8 unchanged	
hex.	9B	37	7A		
dec.	155	55	122		

Description

❶ By means of this MTPL-sequence the eighth data bit is set to “1”.
This results in accordance with the selected character set characters from the national character set or italic characters to be printed.

❷ By means of this MTPL-sequence the eighth data bit is set to “0”.
This means that wrong character printing caused by the eighth data bit set high by the system can be avoided.

CSI 9 z is ignored
– if Extended Character Set is selected,
– if Graphic Mode is selected

STOP

Note that following the sequence **CSI 8 z** all subsequent control sequences must begin with an “**ESC [** ”, because CSI (9B) cannot be represented by a 7 bit code.This remains in effect until the printer receives either the sequence **ESC [9 z** or **ESC [7 z** .

Example

```
10  REM   input data control
20  LPRINT CHR$(27);"[7z";
30  GOSUB 90
40  LPRINT CHR$(27);"[9z";
50  GOSUB 90
60  LPRINT CHR$(27);"[8z";
70  GOSUB 90
80  END
90  FOR I=97 TO 103:LPRINT CHR$(I);:NEXT I
100 FOR I=161 TO 167:LPRINT CHR$(I);:NEXT I
110 LPRINT:LPRINT
120 RETURN
```

```
abcdefgíóúñÑªº

βΓπΣσµτíóúñÑªº

abcdefg!"#$%&'
```

Sheet Feeder/Paperway	CSI ... {
Eject Form	CSI 2 J

Data Structure

ASCII	CSI	n	;	n_1	;	n_2	"{"	[A] set sheed feeder	
hex.	9B	n	3B	n_1	3B	n_2	7B		
dec.	155	n	59	n_1	59	n_2	123		

ASCII	CSI	"2"	"J"	[C] eject form	❶
hex.	9B	32	4A		
dec.	155	50	74		

Description

With the sequence **CSI n ; n_1 ; n_2 {** for n the following parameters are available:

n	hex.	dec.	sheet feeder	
0	30	48	cut sheet paper (manual)	❷
5	35	53	fanfold paper (tractor 1)	❸
6	36	54	fanfold paper (tractor 2)	❹
21	32 31	50 49	bin 1 (front)	❺
22	32 32	50 50	bin 2 (rear)	❺
23	32 33	50 51	bin 3 (reserved)	❻
31	33 31	51 49	stacker 1 of the sorter	❼
50	35 30	53 48	move paper to normal print position (return of view- or tear-off position)	❽
51	35 31	53 49	move paper to tear-off position	❾
52	35 32	53 50	move paper to view position	❿
51	35 31	53 49	cuts fanfold paper at the next perforation	①
54	35 34	53 52	cut is made above/below the current line	②

n_1 [C] specifies the horizontal print head position during the paper loading process (is ignored with new printer models)

n_2 [C] specifies an optional correction value for the vertical load position in $n*1/72$ inches (top of form adjustment or vertical alignment, also see “Positioning the first print line (Phys.adj)”, Operator’s Manual, chapter 4).

❶ Paper is ejected also by FF or the limit, fixed by the form length or the bottom margin.

❷ This sequence selects manual cut sheet paper process.

❸ This sequence selects fanfold paper process for tractor 1.

❹ This sequence selects fanfold paper process for optional tractor 2. If an optional tractor 2 is installed, sequences for selecting the automatic single sheet feeder (ASF) are ignored.

❺ Selection of bin 1 or 2.
Vertical alignment of the first printable line in steps of n$^1/_{72}$ inch downwards is optional.
n is set to $^{12}/_{72}$ inch (one line at 6 lpi) in the default menu.

❻ Depending on the hardware, up to 9 bins can be selected.
(bin 1 - 9 $\triangleq$ CSI 21 - CSI 29).

❼ Depending on the hardware and in the case a sorter has been installed, up to 19 exit ports can be selected.
(stacker 1 - 19 $\triangleq$ CSI 31 - CSI 49).

❽ The page is set from view or tear-off position (perforation) to print position.
This means, that the print head moves to the current position or to the first possible line of the following document.

❾ Independent of the menu settings the paper is set to tear-off position.
This sequence has no affect, if the respective position is just set manual (by means of the “Tear” key) or automatically (printer menu setting: Parameter group “Auto Tear”).**)

❿ Independent of the menu settings the paper is set to view position (last printed line).
This sequence has no affect, if the respective position is just set manual (by means of the “Tear” key) or automatically (printer menu setting: Parameter group “Auto Tear”).

① This sequence is only available if an optional cutter is installed and cuts fanfold paper at the next perforation.*)

② This sequence is only available if an optional cutter is installed; it cuts above the current line, if the text to be printed was ended with a CR (hex. 0D, dec. 13), it cuts below the line, if the text to be printed was ended with a LF (hex. 0A, dec. 10).

The sheet feeder - if installed and connected - has to be selected via the menu of the printer (refer to the Operator’s Manual).

*) If a cutting device is installed.
**) If no cutting device is installed.

For compatibility to former printers a sequence like **CSI 21 ; x ; n {** will also be accepted. In this case, the parameter x will be ignored.

Paper is inserted on receipt of:

- CR (hex.0D; dec.13) if CR = LF is selected
- LF (hex.0A; dec.10)
- VT (hex.0B; dec.11)
- print data

Example

```
10 REM    sheet feeder
15 LPRINT CHR$(27);"[3t";:REM set form length
20 LPRINT CHR$(27);"[21{";
30 LPRINT "This text is on a form"
40 LPRINT "from magazine one."
50 LPRINT CHR$(12):REM form feed
60 LPRINT CHR$(27);"[22{";
70 LPRINT "This text is on a form"
80 LPRINT "from magazine two."
90 END
```

```
This text is on a form
from magazine one.


This text is on a form
from magazine two.
```

Reset to Initial State (RIS) — ESC c / CSI 6 ~

Data Structure

ASCII	ESC	"c"		[A] reset to initial state
hex.	1B	63		
dec.	27	99		
ASCII	CSI	"6"	" ~ "	[C] reset to initial state
hex.	9B	36	7E	
dec.	155	54	126	

Description

The printer is initialized, which means it has the same status as just after power switch-on. All parameters set by CSI-sequences are cleared. The active print position is set on the top position of a page (document).

Example

```
10  REM initialization
20  LPRINT CHR$(27);"[10s";:REM set left margin
30  LPRINT CHR$(27);"[7w";:REM set 17.1 cpi
40  LPRINT CHR$(27);"[1y";:REM set NLQ
50  LPRINT CHR$(27);"[1z";:REM set subscript
60  GOSUB 100
70  LPRINT CHR$(27);"[6~";
80  GOSUB 100
90  END
100 LPRINT "Matrix Printer"
110 LPRINT:LPRINT
120 RETURN
```

Matrix Printer

Matrix Printer

Data Structure

ASCII	CSI	"?"	n	" ~ "	[B]	command set
hex.	9B	3F	n	7E		
dec.	155	63	n	126		

Description

For n the following parameters are available:

n	hex.	dec.	command set
1	31	49	MTPL
2	32	50	MTPL + IBM-Graphics-Printer
3	33	51	MTPL + IBM-Proprinter XL
4	34	52	MTPL + IBM-Proprinter XL 24e
5	35	53	MTPL + EPSON FX
6	36	54	MTPL + EPSON LQ
7	37	55	MTPL + NEC P60/70
10	31 30	49 48	end of special command interpretations and return to standard command set, e.g. end of Barcode interpretation
11	31 31	49 49	start of Barcode interpretation ❶

*) (“Tally Printer Language”)

❶ For more information see the Barcode Programmer’s Manual.

Activation of Character Sets

SI
SO
ESC ...

Data Structure

ASCII	SI		ZG locking shift LS0	❶
hex.	0F			
dec.	15			
ASCII	SO		ZG locking shift LS1	❷
hex.	0E			
dec.	14			
ASCII	ESC	n	ZG locking shift LS2, LS3, LS1R, LS2R, LS3R	❸
hex.	1B	n		
dec.	27	n		

Description

❶ Activation of G0 for codes 21 - 7E

❷ Activation of G1 for codes 21 - 7E

❸ For n the following parameters are available:

n	hex.	dec.	locking shift (LS)		
n	6E	110	[B]	LS2:	Activation of G2 for codes 21 - 7E
o	6F	111	[B]	LS3:	Activation of G3 for codes 21 - 7E
|	7C	124	[B]	LS3R:	Activation of G3 for codes A1 - FE
}	7D	125	[B]	LS2R:	Activation of G2 for codes A1 - FE
~	7E	126	[A]	LS1R:	Activation of G1 for codes A1 - FE

These sequences activate various character sets (also see Invocation of Character Set G0 - G3, page 83). 4 registers - from G0 until G3 - are supported. Into each of these registers a graphic character set by a designation sequence can be placed.

A set of up to 94 characters is made ready for use.

Example

```
10   open "lpt1:" as #1
20   width #1,32000
30                           'Example "Locking Shift"
40   t1$=" French Characters Are Printed"
50   t2$=" German Characters Are Printed"
60   print #1,chr$(27),"+R"  'Designate G3 <— French Substitution
70                           'Table
80   print #1,chr$(27),"o"   'Locking Shift 3
90   print #1,§Öä",t1$       '(French Substitutes)
100  print #1,chr$(27),"+K"  'Designate G3 <— German Substitution
110                          'Table
120  print #1,chr$(27),"o"   'Locking Shift 3
130  print #1,§Öä",t2$       '(German Substitutes)
140  end
```

Designation of Character Set G0 — ESC (...
Designation of Character Set G1 — ESC) ...
Designation of Character Set G2 — ESC * ...
Designation of Character Set G3 — ESC + ...

Data Structure

ASCII	ESC	n	F	designation of character sets
hex.	1B	n	F	
dec.	27	n	F	

Description

Designation of character sets is a process by which a set of graphic symbols is connected to the usable character set registers G0, G1, G2 and G3.

For n and F the following parameters are available:

n	hex.	dec.	character set
(	28	40	[A] G0
)	29	41	[A] G1
*	2A	42	[B] G2
+	2B	43	[B] G3

F	hex.	dec.	character set
A	41	65	US-ASCII
B	42	66	UK-ASCII
H	48	72	Swedish/Finnish
K	4B	75	German
L	4C	76	Portuguese
R	52	82	French
Y	59	89	Italian
Z	5A	90	Spanish
‘	60	96	Norwegian
Ç	80	128	Table 437 (DOS Standard)
é	82	130	Table 850 (DOS Europe)
ç	87	135	Table 852 (DOS Latin 2)
ä	84	132	Table 860 (DOS Portugal)

F	hex.	dec.	character set
à	85	133	Table 863 (DOS French-Canadian)
å	86	134	Table 865 (DOS Norway)
Ä	8E	142	Table 866 Standard (DOS Cyrillic)
Å	8F	143	Table 866 Ukraine
É	90	144	Table 866 Kazakhstan
æ	91	145	Kamenicky (combination of table 437 and 852)
Æ	92	146	Mazovia (code page 437 with polish characters)
p	70	112	(ASCII) Microsoft Windows™ Codepage 1250 (Latin 2)
q	71	113	(ASCII) Microsoft Windows™ Codepage 1251 (Cyrillic)
r	72	114	(ASCII) Microsoft Windows™ Codepage 1252 (Latin 1)
%	25	37	(ASCII) ISO 8859-1 (Latin 1)
&	26	38	(ASCII) ISO 8859-2 (Latin 2)
*	2A	42	(ASCII) ISO 8859-5 (Latin Cyrillic)
<	3C	60	(ASCII) Cro-ASCII character set (Codepage 437 with croatic characters)

Other character sets may be available on request.

If the mode is selected by the printer menu, the MTPL-character set US-ASCII is active (default configuration).

The slashed zero as well as the IBM and Epson character sets can only be selected via the printer control panel.

By default the graphic character set G0 is automatically activated.

This functions do not deal with printing styles like “Italic”, “Helvetica” or “Courier”. Instead, they select character sets defined by the meaning of their symbols. The meaning of the sets may be Greek or French character sets or things like math symbols and line drawing characters.

The available character sets may vary due to used printer type.

Example

```
10  REM    character sets
20  LPRINT CHR$(27);"(B";
30  LPRINT "US-ASCII:  ";:GOSUB 130
40  LPRINT CHR$(27);"(A";
50  LPRINT "UK-ASCII:  ";:GOSUB 130
60  LPRINT CHR$(27);"(K";
70  LPRINT "GERMAN  :  ";:GOSUB 130
80  LPRINT CHR$(27);"(R";
90  LPRINT "FRENCH  :  ";:GOSUB 130
100 LPRINT CHR$(27);"(Y";
110 LPRINT "ITALIAN :  ";:GOSUB 130
120 END
130 LPRINT CHR$(35);CHR$(36);CHR$(64);
140 LPRINT CHR$(91);CHR$(92);CHR$(93);
150 LPRINT CHR$(94);CHR$(96);CHR$(123);
160 LPRINT CHR$(124);CHR$(125);CHR$(126)
170 RETURN
```

```
US-ASCII:  #$@[\]^`{|}~
UK-ASCII:  £$@[\]^`{|}¯
GERMAN  :  #$§ÄÖÜ^`äöüß
FRENCH  :  £$à°ç§^`éùè¨
ITALIAN :  £$§°çé^ùàòèì
```

Activation of Characters

SO
SI

Data Structure

ASCII	SO	[B] ZG single shift SS2	❶
hex.	8E		
dec.	142		
ASCII	SI	[B] ZG single shift SS3	❷
hex.	8F		
dec.	143		

Description

❶ Activation of one single character from G2 (“single shift 2”).

❷ Activation of one single character from G3 (“single shift 3”).

The functions SO and SI each activate one single character of the graphic character set into the codes 21 - 7E.

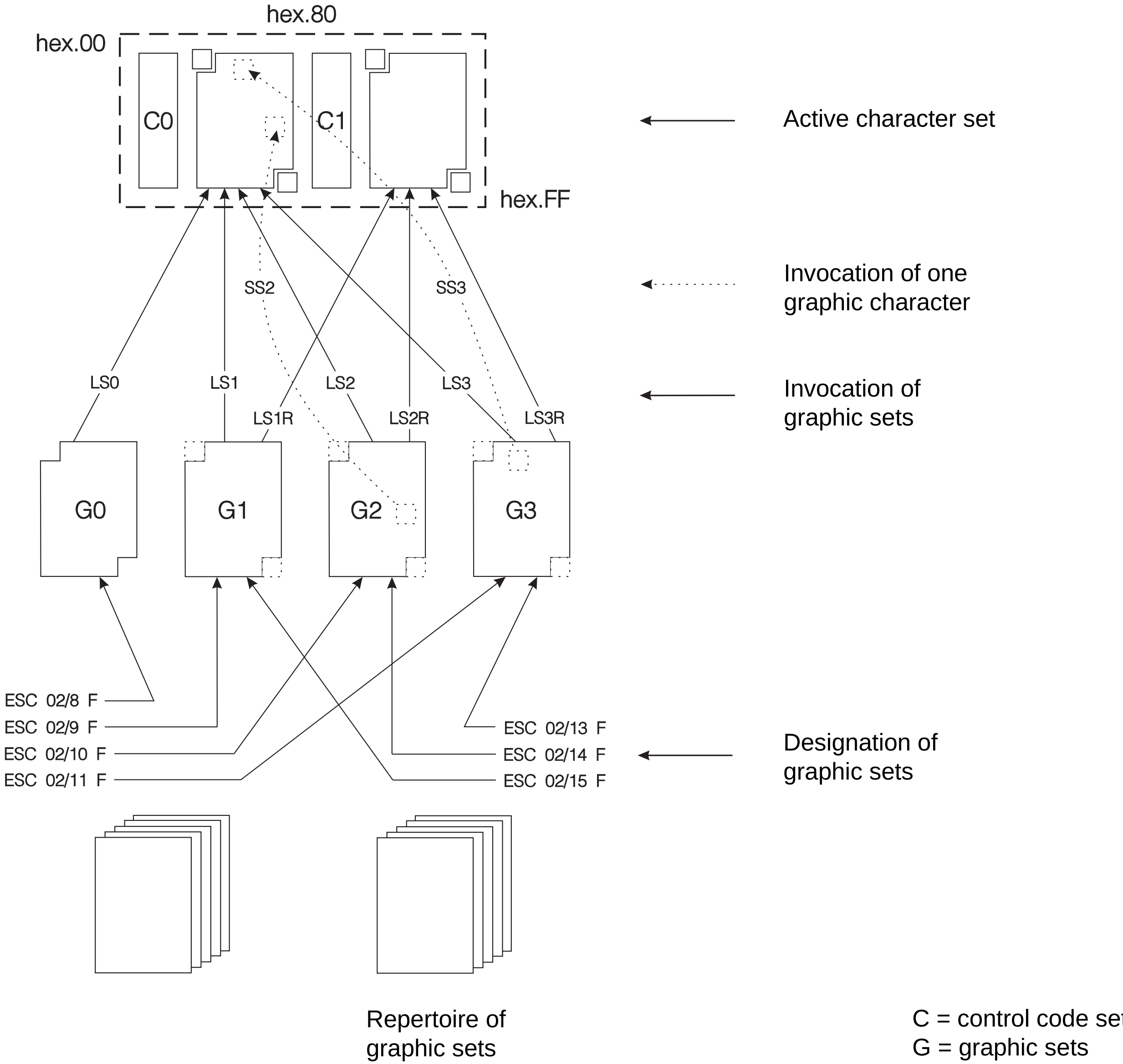
The character following the code for the changeover is replaced once.

The graphic character set activated by the locking shift function (page 81, “Activation of Character Sets”) will not be changed with this function.

Example

```
10   open "lpt1:" as #1
20   width #1,32000
30                           'Example "Single Shift"
40   t1$=" French a accent is printed"
50   t2$=" funny a is printed"
60   print #1,chr$(27),"+R"  'Designate G3 <— French Substitution
70                           'Table
80   print #1,chr$(143);     'Single Shift 3
90   print #1,"@",t1$        'Transfer And Print:
100                          'Code 40hex (French substitute)
110  print #1,"@",t2$        'Transfer And Print:
120                          'Code 40hex (default character)
130  end
```

Graphic description of locking shift/single shift (pages 81 - 86):



Size of Character Set (1 Character Set)	**CSI ? 50 h**
Size of Character Set (2 Character Sets)	**CSI ? 50 ℓ**

Data Structure

ASCII	CSI	"?"	"5"	"0"	"h"	[A] loading a character set	
hex.	9B	3F	35	30	68	with 256 characters	❶
dec.	155	63	53	48	104		
ASCII	CSI	"?"	"5"	"0"	"ℓ"	[A] loading 2 character sets	
hex.	9B	3F	35	30	69	with 94 characters each	❷
dec.	155	63	53	48	105		

Description

In an 8 bit environment, the MTPL standard allows the use of 2 character sets at the same time. In this case each character set contains 94 and 96 characters respectively and can be allocated to the lower or upper half of the code table.
(Example: The control code **SI** activates G0 in the code range 21 - 7E and therefore loads G0 in the lower half of the code table - see also function “Activation of Character Sets” on page 79).

❶ This function causes a character set with 256 characters to be loaded by the control codes **SO**, **SI**, **ESC n** or **ESC o**. Thus, the activation of character sets with the using of the codes **ESC |** , **ESC }** and **ESC ~** becomes ineffective.

This function can be used, for example, to activate a table of PC code page tables.

❷ With this sequence, the function set by **CSI ? 50 h** is turned off. Thus, the activation of character sets with 94 and 96 characters becomes effective again.

STOP

If the setting “Character 8/0 to 9/F printable” is active (default setting) **ESC [? 53 l** has to be sent instead of **CSI ? 53 l**, because the CSI character is not accepted as control code.

Proportional Spacing (Draft) OFF — CSI ? 51 h
Proportional Spacing (Draft) ON — CSI ? 51 ℓ

Data Structure

ASCII	CSI	"?"	"5"	"1"	"h"	[A] disabling proportional spacing in print quality DRAFT ❶
hex.	9B	3F	35	31	68	
dec.	155	63	53	49	104	
ASCII	CSI	"?"	"5"	"1"	"ℓ"	[A] enabling proportional spacing in print quality DRAFT (default) ❷
hex.	9B	3F	35	31	69	
dec.	155	63	53	49	105	

Description

❶ This sequence disables proportional spacing for the print quality DRAFT. The printer switches automatically to NLQ, if DRAFT and proportional spacing are selected.

❷ This sequence enables proportional spacing for the print quality DRAFT. It is the default setting.

Font Selection for NLQ/LQ only — CSI ? 52 h
Font Selection — CSI ? 52 ℓ

Data Structure

ASCII	CSI	"?"	"5"	"2"	"h"	[A] print control ignores the font IDs of the default character generator ❶
hex.	9B	3F	35	32	68	
dec.	155	63	53	50	104	
ASCII	CSI	"?"	"5"	"2"	"ℓ"	[A] print control does not ignore the font IDs of the default character generator (default) ❷
hex.	9B	3F	35	32	69	
dec.	155	63	53	50	105	

Description

❶ The font settings (CSI 10 ... 19 m) are only valid and can only be selected for the print qualities NLQ and LQ.

❷ All print qualities (Draft, NLQ, LQ) are valid, if fonts are selected using sequence CSI 10 ... 19 m. This is the default setting.

Print Code Area Expansion ON — CSI ? 53 h
Print Code Area Expansion OFF — CSI ? 53 ℓ

Data Structure

ASCII	CSI	"?"	"5"	"3"	"h"	[A] print code area expansion on	❶
hex.	9B	3F	35	33	68		
dec.	155	63	53	51	104		
ASCII	CSI	"?"	"5"	"3"	"ℓ"	[A] print code area expansion off	❷
hex.	9B	3F	35	33	69		
dec.	155	63	53	51	105		

Description

❶ This code sequence causes the printout of the higher control codes (ASCII 128 to 159) in the form of characters and symbols instead of the repetition of the functions of ASCII codes 0 to 31.

❷ When the printer receives this code sequence, the ASCII codes 128 to 159 are used as duplicates of the functions of the lower control codes 0 to 31.

Print Transparent Characters — CSI Pn … SP r

Data Structure

ASCII	CSI	Pn	…	SP	"r"
hex.	9B	Pn	…	20	72
dec.	155	Pn	…	32	114

Description

Up to 16 numeric parameters. The parameter represents the decimal code of a printable character. If no parameter is specified, the sequence is ignored.

Example:

CSI 129 SP r: Print “Umlaut u” (if character set is PC table 437).

Printer Identification (DA) — CSI ... c

Data Structure

ASCII	CSI	"0"	"c"		[B] request for printer indentification
hex.	9B	30	63		
dec.	155	48	99		

ASCII	CSI	"?"	...	"c"	Response from printer
hex.	9B	3F	...	63	
dec.	155	63	...	99	

Description

By this functions it is possible to get for example informations of device type and firmware revision or some capabilities like paper supply bins of the connected printer.

This sequence can only be used if the printer is connected to a computer with a serial interface or another bi-directional interface.

Device Status Report (DSR) — CSI ... n

Data Structure

ASCII	CSI	n	"n"	[B] device status report
hex.	9B	n	6E	
dec.	155	n	110	

Description

For n the following parameters are available:

n	hex.	dec.	status report	direction
0	30	48	“Ready, no malfunction”	printer → computer
1	31	49	“Busy, no malfunction”	printer → computer
3	33	51	“Some malfunction detected”	printer → computer
5	35	53	Request of device status report	computer → printer

By these sequences it is possible to request the current printer status.

This sequence can only be used if the printer is connected to the computer with a serial interface or a other bi-directional interface.
The printer sends the complete sequence back to host with parameters 0, 1 or 3.

Enquiry for Status **ENQ**

Data Structure	ASCII	ENQ	enquiry for status
	hex.	05	
	dec.	5	

Description With this code it is possible to get a status byte from the printer:

Status byte:			
	Bit 0	1	= BUSY
	Bit 1	1	= offline
	Bit 2	0	= paper end
	Bit 3	1	= always 1
	Bit 4	1	= cover open
	Bit 5	1	= buffer overflow
	Bit 6	1	= parity or framing error
	Bit 7	0	= always 0

This code concerns only serial data transfer with ENQ/STX protocol.

Fill Character NUL

Data Structure

ASCII	NUL		fill character without further meaning
hex.	80		
dec.	128		

Start of Text Block STX

Data Structure

ASCII	STX		start of text block
hex.	02		
dec.	2		

Description

Start of a text block.

With this code any error message (parity error, memory overflow) is cleared.

This code concerns only serial data transfer with ENQ/STX protocol.

End of Text Block ETX

Data Structure ASCII ETX end of text block
hex. 03
dec. 3

Description End of a text block.

This code concerns only serial data transfer with ACK/NAK protocol.
For a detailed description see the documentation which comes with your printer.

Positive Acknowledge ACK

Data Structure ASCII ACK positive acknowledge
hex. 06
dec. 6

Description By sending the positive acknowledge code ACK (printer ✂ host) it is possible to transmit a data block to the printer and thus transfer it into the printer's line buffer.

This code concerns only serial data transfer with ACK/NAK protocol.
For a detailed description see the documentation which comes with your printer.

Negative Acknowledge **NAK**

Data Structure	ASCII	NAK	negative acknowledge
	hex.	15	
	dec.	21	

Description By sending the negative acknowledge code NAK a parity error (character with the wrong parity) is indicated in the data block to be transmitted and then the data block is deleted.

This code concerns only serial data transfer with ACK/NAK protocol.
For a detailed description see the documentation which comes with your printer.

Clear Print Buffer **CAN**

Data Structure	ASCII	CAN	clear print buffer
	hex.	18	
	dec.	24	

Description When the printer receives this code, the print buffer is completely cleared.

This code only clears the contents of the print buffer. It does not affect any control codes which have previously been sent to the printer.

Delete DEL

Data Structure	ASCII	DEL	[A] delete
	hex.	7F	
	dec.	127	

Description With this code the line buffer is cleared.

Acoustic Alarm BEL

Data Structure	ASCII	BEL	[A] acoustic alarm
	hex.	07	
	dec.	7	

Description This code causes an acoustic alarm.

Start Character “Escape” ESC

Data Structure	ASCII	ESC	[A] start character ESC
	hex.	1B	
	dec.	27	

Description ESC is the start character for a subsequent program sequence (the string $$ can be used instead of ESC if option is selected in the printer setup).

MTPL-Start Character “Control Sequence Introducer”

Data Structure	ASCII	CSI	[A] MTPL-start character CSI
	hex.	9B	
	dec.	155	

Description MTPL-start character for a subsequent program sequence.

STOP Do not use “CSI” in any other printer command language than MTPL. Use instead “ESC [” if using other emulations.

Appendix A

Summary of Possible Codes

Sorted by Sequences

Function	Sequence	Page
ACK	control code ACK (Positive Acknowledge)	95
BEL	acoustic alarm	97
BS	control code BS (Backspace)	61
CAN	control code CAN (Cancel)	96
CR	control code CR (Carriage Return)	61
CSI	MTPL-start character for program sequence	98
CSI % 3 n_1 ; n_2 ; DATA	graphics 60 dpi	65
CSI % 4 n_1 ; n_2 ; DATA	graphics 120 dpi	65
CSI % 5 n_1 ; n_2 ; DATA	graphics 80 dpi	65
CSI % 6 n_1 ; n_2 ; DATA	graphics 240 dpi	65
CSI % 7 n_1 ; n_2 ; DATA	graphics 240 dpi	65
CSI % 8 n_1 ; n_2 ; DATA	graphics 72 dpi	65
CSI % 9 n_1 ; n_2 ; DATA	graphics 90 dpi	65
CSI : y	right justification and centering OFF	14
CSI = z	emphasized mode ON	35
CSI > z	emphasized mode ON	35
CSI ? 1 ~	MTPL command set	80
CSI ? 10 ~	end of command interpretations	80
CSI ? 11 ~	start of Barcode interpretation	80
CSI ? 2 ~	MTPL + IBM-Graphics-Printer command set	80
CSI ? 3 ~	MTPL + IBM-Proprinter XL command set	80
CSI ? 4 ~	MTPL + IBM-Proprinter XL 24e command set	80
CSI ? 5 ~	MTPL + EPSON FX command set	80
CSI ? 50 l	size of character set (2 character sets)	88
CSI ? 50 h	size of character set (1 character set)	88
CSI ? 51 h	proportional spacing (Draft) off	89
CSI ? 51 l	proportional spacing (Draft) on	89
CSI ? 52 h	font selection for NLQ/LQ	89
CSI ? 52 l	font selection	89
CSI ? 53 h	print code area expansion on	90
CSI ? 53 l	print code area expansion off	90
CSI ? 6 ~	MTPL + EPSON LQ command set	80
CSI ? 7 ~	MTPL + NEC P60/70 command set	80
CSI 0 + y	direct setting of the print head distance (default value)	70
CSI 0 + z	automatic gap adjustment OFF	70

Sequence	Function	Page
CSI 0 c	request for printer identification	91
CSI 0 g	clear horizontal tab stop at current position	46
CSI 0 m	clear all selected fonts	33
CSI 0 m	reset color	72
CSI 0 n	device status report: “Ready, no malfunction”	92
CSI 0 n_1 ; n_2 {	cut sheet paper	76
CSI 0 SP I	size unit Decipoints	8
CSI 0 SP K	horizontal character spacing 10 cpi	18
CSI 0 SP L	vertical spacing 6 lpi	24
CSI 0 SP q	load default menu	69
CSI 0 SP X	high print quality	31
CSI 0 w	horizontal spacing 5 cpi	16
CSI 0 y	draft print quality (DPQ)	31
CSI 0 z	superscript ON	27
CSI 1 + z	automatic gap adjustment ON	70
CSI 1 g	clear all vertical tab stops	52
CSI 1 m	bold or increased intensity	72
CSI 1 n	device status report: “Busy, no malfunction”	92
CSI 1 SP I	size unit Millimeters	8
CSI 1 SP K	horizontal character spacing 6 cpi	18
CSI 1 SP L	vertical spacing 4 lpi	24
CSI 1 SP q	load menu No. 1	69
CSI 1 SP X	medium print quality	31
CSI 1 w	horizontal spacing 6 cpi	16
CSI 1 y	near letter quality (NLQ)	31
CSI 1 z	subscript ON	27
CSI 10 m	font 0 (DPQ)	33
CSI 10 y	right justification and centering OFF	14
CSI 10 z	mikroscript ON	27
CSI 11 l	turn OFF positioning unit mode	7
CSI 11 h	turn ON positioning unit mode	7
CSI 11 m	font 1	33
CSI 11 w	horizontal spacing 20 cpi	16
CSI 11 y	fast draft print quality	31
CSI 12 m	font 2	33
CSI 12 w	horizontal spacing 10 cpi	16
CSI 12 y	letter print quality (LQ)	31

Sequence	Function	Page
CSI 12 z	double height ON (upper half)	29
CSI 13 m	font 3	33
CSI 13 y	high resolution print quality	31
CSI 13 z	double height ON (lower half)	29
CSI 14 m	font 4	33
CSI 15 m	font 5	33
CSI 16 m	font 6	33
CSI 17 m	font 7	33
CSI 18 m	font 8	33
CSI 19 m	font 9	33
CSI 2 g	clear all horizontal tab stops at current line	46
CSI 2 J	eject form	76
CSI 2 m	change color	72
CSI 2 SP I	size unit Decipoints	8
CSI 2 SP K	horizontal character spacing 15 cpi	18
CSI 2 SP L	vertical spacing 3 lpi	24
CSI 2 SP q	load menu No. 2	69
CSI 2 SP X	low print quality	31
CSI 2 w	horizontal spacing 7,5 cpi	16
CSI 2 y	proportional mode ON	37
CSI 2 z	double height OFF	29
CSI 2 z	super-/sub- and microscript OFF	27
CSI 21 m	double underline mode	39
CSI 21 n_1 ; n_2 {	bin 1 (front)	76
CSI 22 m	normal color or normal intensity	72
CSI 22 n_1 ; n_2 {	bin 2 (rear)	76
CSI 23 m	italic mode OFF	36
CSI 23 n_1 ; n_2 {	bin 3 (option)	76
CSI 24 m	underline mode OFF	39
CSI 26 m	proportional mode ON	37
CSI 3 g	clear all horizontal tab stops	46
CSI 3 m	italic mode ON	36
CSI 3 n	device status report: “Some malfunction detected”	92
CSI 3 SP I	size unit Decididots	8
CSI 3 SP K	horizontal character spacing 6 cpi	18
CSI 3 SP L	vertical spacing 12 lpi	24
CSI 3 SP q	load menu No. 3	69

Sequence	Function	Page
CSI 3 w	horizontal spacing 8,6 cpi	16
CSI 3 z	line density 6 lpi	25
CSI 30 m	color black	72
CSI 31 m	color red	72
CSI 31 n_1 ; n_2 {	output paper to stacker 1 (sorter)	76
CSI 32 m	color green	72
CSI 33 m	color yellow	72
CSI 34 m	color blue	72
CSI 35 m	color magenta red	72
CSI 36 m	color cyan-blue	72
CSI 37 m	color white	72
CSI 39 m	default color black	72
CSI 4 g	clear all vertical tab stops	52
CSI 4 m	underline mode ON	39
CSI 4 SP I	size unit Mils	8
CSI 4 SP L	vertical spacing 8 lpi	24
CSI 4 SP q	load menu No. 4	69
CSI 4 w	horizontal spacing 10 cpi	16
CSI 4 y	NLQ at 10 cpi	31
CSI 4 z	line density 8 lpi	25
CSI 5 n	request for device status report	92
CSI 5 n_1 ; n_2 {	fanfold paper	76
CSI 5 SP I	Basic Measuring Unit BMU	8
CSI 5 w	horizontal spacing 12 cpi	16
CSI 5 y	NLQ at 12 cpi	31
CSI 5 z	graphics normal	65
CSI 50 m	proportional mode OFF	37
CSI 50 n_1 ; n_2 {	move paper to normal print position	76
CSI 51 n_1 ; n_2 {	move paper to tear-off position	76
CSI 52 n_1 ; n_2 {	move paper to view position	76
CSI 53 m	overline mode ON	40
CSI 55 m	overline mode OFF	40
CSI 6 ~	reset to initial state	79
CSI 6 SP I	size unit Micrometers	8
CSI 6 w	horizontal spacing 15 cpi	16
CSI 6 z	graphics invers	65
CSI 7 SP I	size unit Pixels	8

Sequence	Function	Page
CSI 7 w	horizontal spacing 17,1 cpi	16
CSI 7 y	proportional mode OFF	37
CSI 7 z	set bit 8 = unchanged	74
CSI 8 w	double strike OFF	41
CSI 8 y	right justification ON	14
CSI 8 z	set bit 8 = 0	74
CSI 9 SP L	vertical spacing 2 lpi	24
CSI 9 w	double strike ON	41
CSI 9 y	centering ON	14
CSI 9 z	set bit 8 = 1	74
CSI n ’	horizontal position absolute	54
CSI n + y	direct setting of the print head distance (in n -steps)	70
CSI n a	horizontal position relative	55
CSI n d	vertical position absolute	57
CSI n e	vertical position relative	58
CSI n j	horizontal position backward	56
CSI n k	vertical position backward	59
CSI n p	Horizontal Motion Index	22
CSI n q	horizontal step	23
CSI n SP \	additional character spacing	20
CSI n SP C	graphic size selection	42
CSI n SP f	reduced character spacing	21
CSI n SP g	character spacing	19
CSI n SP h	line spacing	26
CSI n SP U	line home	13
CSI n SP V	line limit	13
CSI n t	form lenght in lines	5
CSI n v	vertical tab stop	52
CSI n_1 ; n_2 SP B	graphic size modification	42
CSI n_1; n_2 f	horizontal- and vertical position absolute	60
CSI n_1 ; n_2 r	top and bottom margin	9
CSI n_1 ; n_2 s	left and right margin	11
CSI n_1 ; n_2 SP D	select font	43
CSI n_1 ; n_2 SP G	spacing increment	15
CSI n1 u	set horizontal tab stop	46
CSI Pn … SP r	print transparent characters	90
CSI Pn + w	shared interface	71
CSI Ps1 Ps2 Ps3 Pn4 Pn5 + v	paper handling	71

Sorted by Functions

Function	Sequence	Page
acoustic alarm	BEL	97
additional character spacing	CSI n SP \	20
automatic gap adjustment OFF	CSI 0 + z	70
automatic gap adjustment ON	CSI 1 + z	70
Basic Measuring Unit BMU	CSI 5 SP I	8
bin 1 (front)	CSI 21 n_1 ; n_2 {	76
bin 2 (rear)	CSI 22 n_1 ; n_2 {	76
bin 3 (option)	CSI 23 n_1 ; n_2 {	76
bold or increased intensity	CSI 1 m	72
centering ON	CSI 9 y	14
change color	CSI 2 m	72
character spacing	CSI n SP g	19
clear all horizontal tab stops	CSI 3 g	46
clear all horizontal tab stops at current line	CSI 2 g	46
clear all selected fonts	CSI 0 m	33
clear all vertical tab stops	CSI 1 g	52
clear all vertical tab stops	CSI 4 g	52
clear horizontal tab stop at current position	CSI 0 g	46
color black	CSI 30 m	72
color blue	CSI 34 m	72
color cyan-blue	CSI 36 m	72
color green	CSI 32 m	72
color magenta red	CSI 35 m	72
color red	CSI 31 m	72
color white	CSI 37 m	72
color yellow	CSI 33 m	72
control code ACK (Positive Acknowledge)	ACK	95
control code BS (Backspace)	BS	61
control code CAN (Cancel)	CAN	96
control code CR (Carriage Return)	CR	61
control code DEL (Delete)	DEL	97
control code ENQ (Enquiry for Status)	ENQ	93
control code ETX (End of Text Block)	ETX	95
control code FF (Form Feed)	FF	62
control code HT (horizontal step)	HT	45

Function	Sequence	Page
control code HTS	HTS	45
control code IND (Vertical Index)	IND	51
control code LF (Line Feed)	LF	49
control code NAK (Negative Acknowledge)	NAK	95
control code NEL (Next Line)	NEL	64
control code NUL (fill character)	NUL	94
control code PLD (Partial Line Down)	PLD	63
control code PLU (Partial Line Up)	PLU	63
control code RI (Reverse Index)	RI	64
control code STX (Start of Text Block)	STX	94
control code VT (Vertical Tab Stop)	VT	50
control code VTS	VTS	51
cut sheet paper	CSI 0 n_1 ; n_2 {	76
default color black	CSI 39 m	72
designate character set G0	ESC (F	83
designate character set G1	ESC) F	83
designate character set G2	ESC * F	83
designate character set G3	ESC + F	83
device status report: “Busy, no malfunction”	CSI 1 n	92
device status report: “Ready, no malfunction”	CSI 0 n	92
device status report: “Some malfunction detected”	CSI 3 n	92
direct setting of the print head distance (default value)	CSI 0 + y	70
direct setting of the print head distance (in n -steps)	CSI n + y	70
double height OFF	CSI 2 z	29
double height ON (lower half)	CSI 13 z	29
double height ON (upper half)	CSI 12 z	29
double strike OFF	CSI 8 w	41
double strike ON	CSI 9 w	41
double underline mode	CSI 21 m	39
draft print quality (DPQ)	CSI 0 y	31
eject form	CSI 2 J	76
emphasized mode ON	CSI = z	35
emphasized mode ON	CSI > z	35
end of command interpretations	CSI ? 10 ~	80
fanfold paper	CSI 5 n_1 ; n_2 {	76
fast draft print quality	CSI 11 y	31
font 0 (DPQ)	CSI 10 m	33

Function	Sequence	Page
font 1	CSI 11 m	33
font 2	CSI 12 m	33
font 3	CSI 13 m	33
font 4	CSI 14 m	33
font 5	CSI 15 m	33
font 6	CSI 16 m	33
font 7	CSI 17 m	33
font 8	CSI 18 m	33
font 9	CSI 19 m	33
font selection for NLQ/LQ	CSI ? 52 h	89
font selection	CSI ? 52 l	89
form lenght in lines	CSI n t	5
graphic size modification	CSI n_1 ; n_2 SP B	42
graphic size selection	CSI n SP C	42
graphics 120 dpi	CSI % 4 n_1 ; n_2 ; DATA	65
graphics 240 dpi	CSI % 6 n_1 ; n_2 ; DATA	65
graphics 240 dpi	CSI % 7 n_1 ; n_2 ; DATA	65
graphics 60 dpi	CSI % 3 n_1 ; n_2 ; DATA	65
graphics 72 dpi	CSI % 8 n_1 ; n_2 ; DATA	65
graphics 80 dpi	CSI % 5 n_1 ; n_2 ; DATA	65
graphics 90 dpi	CSI % 9 n_1 ; n_2 ; DATA	65
graphics invers	CSI 6 z	65
graphics normal	CSI 5 z	65
high print quality	CSI 0 SP X	31
high resolution print quality	CSI 13 y	31
horizontal character spacing 10 cpi	CSI 0 SP K	18
horizontal character spacing 15 cpi	CSI 2 SP K	18
horizontal character spacing 6 cpi	CSI 1 SP K	18
horizontal character spacing 6 cpi	CSI 3 SP K	18
Horizontal Motion Index	CSI n p	22
horizontal position absolute	CSI n ’	54
horizontal position backward	CSI n j	56
horizontal position relative	CSI n a	55
horizontal spacing 10 cpi	CSI 12 w	16
horizontal spacing 10 cpi	CSI 4 w	16
horizontal spacing 12 cpi	CSI 5 w	16
horizontal spacing 15 cpi	CSI 6 w	16

Function	Sequence	Page
horizontal spacing 17,1 cpi	CSI 7 w	16
horizontal spacing 20 cpi	CSI 11 w	16
horizontal spacing 5 cpi	CSI 0 w	16
horizontal spacing 6 cpi	CSI 1 w	16
horizontal spacing 7,5 cpi	CSI 2 w	16
horizontal spacing 8,6 cpi	CSI 3 w	16
horizontal step	CSI n q	23
horizontal- and vertical position absolute	CSI n_1; n_2 f	60
italic mode OFF	CSI 23 m	36
italic mode ON	CSI 3 m	36
left and right margin	CSI n_1 ; n_2 s	11
letter print quality (LQ)	CSI 12 y	31
line density 6 lpi	CSI 3 z	25
line density 8 lpi	CSI 4 z	25
line home	CSI n SP U	13
line limit	CSI n SP V	13
line spacing	CSI n SP h	26
load default menu	CSI 0 SP q	69
load menu No. 1	CSI 1 SP q	69
load menu No. 2	CSI 2 SP q	69
load menu No. 3	CSI 3 SP q	69
load menu No. 4	CSI 4 SP q	69
low print quality	CSI 2 SP X	31
LS1R: Activation of G1 for codes A1 - FE	ESC ~	81
LS2: Activation of G2 for codes 21 - 7E	ESC n	81
LS2: Activation of G3 for codes 21 - 7E	ESC o	81
LS2R: Activation of G2 for codes A1 - FE	ESC }	81
LS3R: Activation of G3 for codes A1 - FE	ESC |	81
medium print quality	CSI 1 SP X	31
mikroscript ON	CSI 10 z	27
move paper to normal print position	CSI 50 n_1 ; n_2 {	76
move paper to tear-off position	CSI 51 n_1 ; n_2 {	76
move paper to view position	CSI 52 n_1 ; n_2 {	76
MTPL + EPSON FX command set	CSI ? 5 ~	80
MTPL + EPSON LQ command set	CSI ? 6 ~	80
MTPL + IBM-Graphics-Printer command set	CSI ? 2 ~	80
MTPL + IBM-Proprinter XL 24e command set	CSI ? 4 ~	80

Function	Sequence	Page
MTPL + IBM-Proprinter XL command set	CSI ? 3 ~	80
MTPL + NEC P60/70 command set	CSI ? 7 ~	80
MTPL command set	CSI ? 1 ~	80
MTPL-start character for program sequence	CSI	98
MTPL-start character for program sequence	ESC	98
near letter quality (NLQ)	CSI 1 y	31
NLQ at 10 cpi	CSI 4 y	31
NLQ at 12 cpi	CSI 5 y	31
normal color or normal intensity	CSI 22 m	72
output paper to stacker 1 (sorter)	CSI 31 n_1 ; n_2 {	76
overline mode OFF	CSI 55 m	40
overline mode ON	CSI 53 m	40
paper handling	CSI Ps1 Ps2 Ps3 Pn4 Pn5 + v	71
print code area expansion ON	CSI ? 53 h	90
print code area expansion OFF	CSI ? 53 l	90
print transparent characters	CSI Pn … SP r	90
proportional mode OFF	CSI 50 m	37
proportional mode OFF	CSI 7 y	37
proportional mode ON	CSI 2 y	37
proportional mode ON	CSI 26 m	37
proportional spacing (Draft) off	CSI ? 51 h	89
proportional spacing (Draft) on	CSI ? 51 l	89
reduced character spacing	CSI n SP f	21
request for device status report	CSI 5 n	92
request for printer identification	CSI 0 c	91
reset color	CSI 0 m	72
reset to initial state	CSI 6 ~	79
reset to initial state	ESC c	79
right justification and centering OFF	CSI : y	14
right justification and centering OFF	CSI 10 y	14
right justification ON	CSI 8 y	14
select font	CSI n_1 ; n_2 SP D	43
set bit 8 = 0	CSI 8 z	74
set bit 8 = 1	CSI 9 z	74
set bit 8 = unchanged	CSI 7 z	74
set horizontal tab stop	CSI n1 u	46
set horizontal tab stop	ESC H	46

Function	Sequence	Page
shared interface	CSI Pn + w	71
single shift 2 (SS2)	SO	86
single shift 3 (SS3)	SI	86
size of character set (1 character set)	CSI ? 50 h	88
size of character set (2 character sets)	CSI ? 50 l	88
size unit Decididots	CSI 3 SP I	8
size unit Decipoints	CSI 0 SP I	8
size unit Decipoints	CSI 2 SP I	8
size unit Micrometers	CSI 6 SP I	8
size unit Millimeters	CSI 1 SP I	8
size unit Mils	CSI 4 SP I	8
size unit Pixels	CSI 7 SP I	8
spacing increment	CSI n_1 ; n_2 SP G	15
start of Barcode interpretation	CSI ? 11 ~	80
subscript ON	CSI 1 z	27
super-/sub- and microscript OFF	CSI 2 z	27
superscript ON	CSI 0 z	27
top and bottom margin	CSI n_1 ; n_2 r	9
turn OFF positioning unit mode	CSI 11 l	7
turn ON positioning unit mode	CSI 11 h	7
underline mode OFF	CSI 24 m	39
underline mode ON	CSI 4 m	39
vertical position absolute	CSI n d	57
vertical position backward	CSI n k	59
vertical position relative	CSI n e	58
vertical spacing 12 lpi	CSI 3 SP L	24
vertical spacing 2 lpi	CSI 9 SP L	24
vertical spacing 3 lpi	CSI 2 SP L	24
vertical spacing 4 lpi	CSI 1 SP L	24
vertical spacing 6 lpi	CSI 0 SP L	24
vertical spacing 8 lpi	CSI 4 SP L	24
vertical tab stop	CSI n v	52

Appendix B

Character Sets

Standard Character Set

hex / dec	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL 0	16	SP 32	0 48	64	P 80	96	p 112	NUL 128	144	á 160	░ 176	192	╨ 208	224	≡ 240
1	1	DC1 17	! 33	1 49	A 65	Q 81	a 97	q 113	129	145	í 161	▒ 177	┴ 193	╤ 209	β 225	± 241
2	STX 2	DC2 18	" 34	2 50	B 66	R 82	b 98	r 114	130	146	ó 162	▓ 178	┬ 194	╥ 210	Γ 226	≥ 242
3	ETX 3	DC3 19	35	3 51	C 67	S 83	c 99	s 115	131	147	163	│ 179	├ 195	╙ 211	π 227	≤ 243
4	4	DC4 20	36	4 52	D 68	T 84	d 100	t 116	IND 132	148	164	┤ 180	─ 196	╘ 212	Σ 228	⌠ 244
5	ENQ 5	NAK 21	% 37	5 53	E 69	U 85	e 101	u 117	NEL 133	149	Ñ 165	╡ 181	┼ 197	╒ 213	σ 229	⌡ 245
6	ACK 6	22	& 38	6 54	F 70	V 86	f 102	v 118	134	150	ª 166	╢ 182	╞ 298	╓ 214	µ 230	÷ 246
7	BEL 7	23	' 39	7 55	G 71	W 87	g 103	w 119	BEL 135	151	º 167	╖ 183	╟ 199	╫ 215	τ 231	≈ 247
8	BS 8	CAN 24	(40	8 56	H 72	X 88	h 104	x 120	HTS 136	152	¿ 168	╕ 184	╚ 200	╪ 216	Φ 232	° 248
9	HT 9	25	) 41	9 57	I 73	Y 89	i 105	y 121	HT 137	153	⌐ 169	╣ 185	╔ 201	┘ 217	Θ 233	• 249
A	LF 10	26	* 42	: 58	J 74	Z 90	j 106	z 122	VTS 138	154	¬ 170	║ 186	╩ 202	┌ 218	Ω 234	· 250
B	VT 11	ESC 27	+ 43	; 59	K 75	91	k 107	123	PLD 139	CSI 155	½ 171	╗ 187	╦ 203	219	δ 235	251
C	FF 12	28	, 44	< 60	L 76	92	l 108	124	PLU 140	156	¼ 172	╝ 188	╠ 204	220	∞ 236	252
D	CR 13	29	- 45	= 61	M 77	93	m 109	125	RI 141	157	¡ 173	╜ 189	═ 205	221	φ 237	253
E	SO 14	30	. 46	> 62	N 78	94	n 110	126	SO 142	158	« 174	╛ 190	╬ 206	222	ϵ 238	254
F	15	31	/ 47	? 63	O 79	_ 95	o 111	DEL 127	SI 143	159	» 175	┐ 191	╧ 207	▀ 223	∩ 239	DEL 255

For the contents of the shaded areas, see table at page B-3.

International Substitution Table - normal font

	dec	35	36	64	91	92	93	94	96	123	124	125	126
	hex	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
USA		#	$	@	[	\	]	^	`	{	|	}	~
Great Britain		£	$	@	[	\	]	^	`	{	|	}	‾
Sweden		#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Germany		#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
Portugal		#	$	§	Ã	Ç	Õ	^	`	ã	ç	õ	°
France		£	$	à	°	ç	§	^	`	é	ù	è	¨
Italy		£	$	§	°	ç	é	^	ù	à	ò	è	ì
Spain		£	$	§	¡	Ñ	¿	^	`	°	ñ	ç	~
Norway		#	$	@	Æ	Ø	Å	^	`	æ	ø	å	‾

The contents of the columns 8 and 9 depend on the settings in the printer’s menue.

Programmer's Reference

Tally ANSI Barcode and LCP

Table of Contents

Barcode and LCP Printing

Introduction

Secured/Unsecured Mode

Barcode Printing

Data Formats of Barcode Types

Appendix

Character Sets

Barcode and LCP Printing

Introduction

This User’s Manual describes all barcode and LCP sequences and control codes, regardless of your printer’s special barcode and LCP implementation (LCP = Large Character Printing).

Be sure to observe the notes in the operator’s manual regarding the special barcode/LCP Types implemented in your printer and which of the sequences described here are not available as a result.

Select the Barcode Mode as described in the operator’s manual. Selecting this emulation mode automatically selects the corresponding character set.

To print barcode or LCP characters, the following steps must be carried out in most cases:

- Activate barcode
- Activate secured/unsecured mode
  (see section "Secured/Unsecured Mode")
- Transfer barcode and/or LCP header
  (see sections "Barcode Header" and "LCP Header")
- Calculate check number
  (barcode only, see section "Calculate Check Number")
- Transfer barcode and/or LCP data
  (see sections "Barcode Typees" and "LCP Header")
- Deactivate barcode (if necessary)

The following commands are used to print barcode and LCP characters:

SUB	Start Character Barcode header
DLE	Start Character LCP header
EM	Stop Character Barcode and LCP header
DC4	Barcode brackets (start and end characters for barcode data)
SI	LCP brackets (start and end characters for LCP data)
ESC P ... ESC \	Settings for Barcode and LCP

This may cause conflict with other emulations, since the above commands may have different functions in these emulations, for example:

SI	Condensed print
DC4	Reset expanded print
ESC P <n>	Proportional spacing ON/OFF
ESC P	Pica

In barcode mode the barcode sequences have priority.

The barcode interpreter can be switched on or off with the MTPL sequences ESC [? 11 ~ (Barcode ON) and ESC [? 10 ~ (Barcode OFF). The typical transmission procedure **should** take place as follows:

- Barcode Interpreter ON
- Transmit mode specification (barcode header, LCP header, secured/unsecured mode, ...)
- Transmit barcode / LCP
- Barcode OFF

The mode specifications is saved temporarily and must only be sent to the printer once. The specification remains valid until the printer is switched off.

Please note the following explanatory information:

Every sequence description begins with a header in which the function and short form of the sequence of the barcode Types are listed, e.g.:

2/5 Matrix	Type A

The header is followed by the **data structure** in ASCII, hexadecimal and decimal representation with the necessary parameters, e.g.:

ASCII	DC4	start code	n ... n	stop code	DC4
hex.	14	start code	n ... n	stop code	14
dec.	20	start code	n ... n	stop code	20

The **syntax** for the parameters, the start, separate and stop code is represented as follows:

	Figures (n)	Start code	Stop code
ASCII	"0" to "9"	":"	":"
hex.	30 to 39	3A	3A
dec.	48 to 57	58	58

For the parameter (here **n**), a distinction must be made between two types of representation:

- if the parameter is in pointed parentheses, the decimal value must be transmitted
- if the parameter is not in pointed parentheses, the ASCII value must be transmitted

Example:

Parameter representation: **<n>**, with n=0
to be transmitted: dec.0 (hex.00)

Parameter representation: **n**, with n=0
to be transmitted: ASCII "0" (hex.30, dec.48)

Character explanation and symbol descriptions

i	Information/important notes
[ ]	Optional, must be not necessarily be transmitted
9	9-needle printer
24	24-needle printer

Special Feature

If you not get the required control codes for Barcode of ASCII code table columns 0 and 1 out of your computer you can use in addition to these Barcode control codes a set of MTPL sequences to generate these control codes by printable ASCII characters. By this measure the Barcode programs will remain fully compatible in spite of the additional MTPL sequences.

Following control codes in Barcode strings can be substituted by the appropriate ANSI sequences:

Control Code	ANSI Sequence (CSI = Hex 9B *or* Hex 1B 5B *or* ESC[ )	Example (Hex)
SUB	CSI 26 Space s	1B 5B 32 36 20 73
EM	CSI 25 Space s	1B 5B 32 35 20 73
DC4	CSI 20 Space s	1B 5B 32 30 20 73
DLE	CSI 16 Space s	1B 5B 31 36 20 73
SI	CSI 15 Space s	1B 5B 31 35 20 73
ANSI sequences with not matching parameters will be ignored.		

These feature is not available with all printers with MTPL emulation.
Special firmware is possibly necessary. Please ask your dealer or representative!

Secured/Unsecured Mode

Unsecured mode	**ESC PSC0 ESC **
Secured mode	**ESC PSC1 ESC **

Data Structure

ASCII	ESC	"P"	"S"	"C"	"0"	ESC	"\\"	Unsecured mode
hex.	1B	50	53	43	30	1B	5C	
dec.	27	80	83	67	48	27	92	
ASCII	ESC	"P"	"S"	"C"	"1"	ESC	"\\"	Secured mode
hex.	1B	50	53	43	31	1B	5C	
dec.	27	80	83	67	49	27	92	

Description

In **secured mode**, the amount of space the barcode or LCP character requires is "secured". In each line, additional barcode and normal characters can be printed.

These additional characters are printed in the current line and in the following lines without influencing the barcode or LCP character. As a result normal characters can be printed to the right or left of the barcode or LCP character in each line.

In order to guarantee successful barcode and LCP character printing, it is important to insert the correct paper feed commands, so that paper feed is ensured to the end of the barcode and LCP height.

In **unsecured mode**, the paper feed necessary for barcode and LCP printing is automatic and it is not possible to print more than one line with normal characters in the barcode and LCP line.

All characters in the mixed line are printed, so that the bottom edges are aligned in a straight line.

Example 1

```
10  REM LCP unsecured mode
20  LPRINT CHR$(27); "[?11~";
30  REM select unsecured mode
40  LPRINT CHR$(27); "PSC0"; CHR$(27); "\";
50  REM set character size to 5
60  LPRINT CHR$(16); "5"; CHR$(25);
70  LPRINT "Example for "
80  LPRINT CHR$(15); "LCP"; CHR$(15); " unsecured mode"
```

```
Example for
LCP unsecured mode
```

Example 2

```
10  REM LCP secured mode
20  LPRINT CHR$(27); "[?11~";
30  REM select secured mode
40  LPRINT CHR$(27); "PSC1"; CHR$(27); "\";
50  REM set character size to 5
60  LPRINT CHR$(16); "5"; CHR$(25);
70  LPRINT "This is "; CHR$(15); "LCP"; CHR$(15);
80  LPRINT " an ex-"
90  LPRINT "ample    of an"
100 LPRINT "expres-  sion"
110 LPRINT "in the secured mode"
```

```
This is        an ex-
ample    LCP   of an
expres-        sion
in the secured mode
```

Barcode Printing

Barcode Header

Before the data, which contains the barcode information, are transmitted to the printer, the barcode header must be sent. Otherwise the standard parameter values are used (see section "Header Format"). In the header, the printing parameters, the barcode size and the barcode type are defined. This header only needs to be transferred once, unless settings are to be changed or the printer has been turned off.

Header Format

Format: **SUB [F] a [n] [;xyz] EM**

[ ]	Specification is optional
x, y	unregarded at EAN/UPC-Barcode! For Code 128 and EAN 128 (Type S+T) only the X parameter is valid. This is automatically used for the Y parameter. The Z parameter is not evaluated.

Meaning of the characters:

SUB	(hex.1A, dec.26)		Start header
F			Print feature (see section "Barcode Print Feature F" to select the F codes, page 10)
a	ASCII	a = "A"..."S"	Barcode Types (see section "Barcode Types")
n	ASCII	n = "0"..."90"	Barcode height in n/6 inch. At n="0" the barcode height equals to 1/12 inch.
;	ASCII		Separation character
x	ASCII	x = "0"..."3"	Width of the narrow bar (see section "Barcode width")
y	ASCII	y = "0"..."3"	Width of the narrow space (see section "Barcode width")
z	ASCII	z = "0"..."3"	Ratio of wide to narrow (see section "Barcode width")
EM	(hex.19, dec.25)		End of header

For encoding ASCII values to decimal or hexadecimal values refer to the appendix, "Character Sets".

The default parameter values are the following:

- Unsecured mode (see section "Secured/Unsecured mode")
- HRI OFF, Normal Print, Double Pass (F = SP)
- Barcode Type 2/5 matrix (a = "A")
- Barcode height 1/6 inch (n = "1")
- Narrow bar (x = "0")
- Narrow space (y = "0")
- Ratio of wide to narrow 2 to 1 (z = "0")

When only parts of the header are to be changed, copy the header up to the parameter which must be changed, and then close the header with the end-of-header character. If a header error was detected the total previous features are still active.

The "Barcode brackets" (hex.14, dec.20), initiate and terminate the printing of the barcodes.

If the printer is switched OFFLINE, all defined barcodes are printed out completely. Please not that the barcode remains resident in the background and can be activated again by the barcode bracket. Text justification and centering are both permitted. With activated barcodes these function are not carried out, since they lead to conflicts with the barcodes.

Transparent Barcode / LCP commands

Following control codes in Barcode strings can be substituted by the appropriate MTPL sequences (see also section "Special Feature" on page 5):

Control Code	ANSI Sequence (CSI = Hex 9B or ESC[ )
SUB	CSI 26 Space s
EM	CSI 25 Space s
DC4	CSI 20 Space s
DLE	CSI 16 Space s
SI	CSI 15 Space s
MTPL sequences with not matching parameters will be ignored.	

This sequences may only be used with activated BARCODE.

Barcode Print Feature F for Selection of F-Code

HRI or normal/compressed as well as single or double pass is switched via character F according to the following table.

ASCII Char.	Hex-Value	HRI [2])		Print		Pass [3])		Direction [2])	
		On	Off	Normal	Compr.	Double	Single	Unidir.	Bidir.
SP	20	–	x	x	–	x	–	x	–
!	21	–	x	x	–	x	–	–	x
"	22	x	–	x	–	x	–	x	–
#	23	x	–	x	–	x	–	–	x
$	24	–	x	–	x	x	–	x	–
%	25	–	x	–	x	x	–	–	x
&	26	x	–	–	x	x	–	x	–
‘	27	x	–	–	x	x	–	–	x
(	28	–	x	x	–	–	x	x	–
)	29	–	x	x	–	–	x	–	x
*	2A	x	–	x	–	–	x	x	–
+	2B	x	–	x	–	–	x	–	x
,	2C	–	x	–	x	–	x	x	–
-	2D	–	x	–	x	–	x	–	x
.	2E	x	–	–	x	–	x	x	–
/	2F	x	–	–	x	–	x	–	x
0 [1])	30	–	x	x	–	x	–	x	–
1 [1])	31	x	–	x	–	x	–	x	–

[1]) It is recommended, to avoid using of ASCII Characters 0 and 1 when possible, since they are reserved for future functions.

[2]) **H**uman **R**eadable **I**ndex

[3]) It depends on the used printer type whether the printer performs "Double Pass" with two physical print passes or special print modes (i.e. emphasized).

Barcode Types

A = 2/5 matrix (default)
B = 2/5 industrial
C = 2/5 interleaved
D = Code 11
E = Code BCD matrix
F = Code 39
G = Codabar
H = EAN 8 with HRI
I = EAN 8 without HRI
J = 2/5 matrix (default)
K = EAN 13 with HRI
L = EAN 13 without HRI
M = MSI/modified Plessey
N = UPC A with HRI
O = UPC A without HRI
P = UPC E with HRI
Q = UPC E without HRI
R = Delta Distance (IBM)
S = Code 128
T = EAN 128

All commercial barcodes (for labeling systems) of the H, I, K, L, N, O, P, Q Types can be extended using the barcodes Add-On 2 or Add-On 5 (see section "Add-On Barcodes").

HRI

HRI = **H**uman **R**eadable **I**ndex

HRI characters are centered if enough space is left. If the barcode-printout is smaller than the HRI character field, smaller character density (CPI) is used. Start and stop codes are not printed as HRI; a space character (SP) will be stored.

Barcode Width By specifying an ASCII value from 0 to 3, the barcode width can be defined. This allows ideal adaption to the scanner specifications, particularly for long-range scanners.

Table 1

	Header Parameter	Normal			Compressed 1)		
		9	24	older printer types (e.g.MT230)	9	24	older printer types (e.g.MT230)
Width of the narrow bar	x = 0 x = 1 x = 2 x = 3	0.53 mm 0.74 mm 1.16 mm 1.38 mm	0.54 mm 0.67 mm 1.09 mm 1.30 mm	0.48 mm 0.69 mm 1.16 mm 1.33 mm	0.32 mm 0.53 mm 0.74 mm 0.95 mm	0.33 mm 0.43 mm 0.65 mm 0.88 mm	0.27 mm 3) 0.37 mm 3) 0.59 mm 3) 0.90 mm 3)
Width of the narrow space	y = 0 y = 1 y = 2 y = 3	0.53 mm 0.74 mm 1.16 mm 1.38 mm	0.54 mm 0.67 mm 1.09 mm 1.30 mm	0.48 mm 0.69 mm 1.16 mm 1.33 mm	0.32 mm 0.53 mm 0.74 mm 0.95 mm	0.33 mm 0.43 mm 0.65 mm 0.88 mm	0.27 mm 3) 0.37 mm 3) 0.59 mm 3) 0.90 mm 3)
Enlarge-ment factor	z = 0 z = 1 z = 2 z = 3			2.0 : 1 2.5 : 1 3.0 : 1 3.5 : 1			2.0 : 1 2) 2.5 : 1 2) 3.0 : 1 2) 3.5 : 1 2)

EAN/UPC Barcode
(X, Y = unregarded):

Table 2	Header Parameter	Normal	Compressed 1)
Enlargement factor	z = 0 z = 1	1.95 : 1 1.60 : 1	1.30 : 1 0.95 : 1

1) These values are true, if "Compressed Print" is selected in the menu (see print feature [F]).
2) Note: It is recommended to set the bar width equal to the space width (x=y).
3) Printer-dependent reference value.

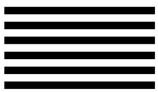
All values are only valid when a new colour ribbon is used. They change depending on the degree of wear:

- narrow bar: approx. -0.05 mm
- narrow space: approx. +0.05 mm.

The **Code EAN 128**, as the **EAN/UPC**, is based on module widths. Therefore only the X parameter is valid for this type. This parameter is also used for the Y parameter (narrow bar width). The Z parameter has no meaning. In combination with the normal/condensed feature, 8 widths result (see table 1).

Error Code

Wrong characters in a control code or in a barcode test (e.g. an undefined character in a certain barcode Type) cause the barcode error sign to be printed.



Code 2/5 Matrix — Type A

Syntax

	Figures (n)	Start code	Stop code
ASCII	"0" to "9"	":"	":"
hex.	30 to 39	3A	3A
dec.	48 to 57	58	58

Data Structure

ASCII	DC4	Start code	n ... n	Stop code	DC4
hex.	14	Start code	n ... n	Stop code	14
dec.	20	Start code	n ... n	Stop code	20

Example

```
10 REM  code 2/5 matrix barcode
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" A3;111";CHR$(25);
40 LPRINT CHR$(20);":123:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code 2/5 Industrial — Type B

Syntax

	Figures (n)	Start code	Stop code
ASCII	"0" to "9"	":", "<" or ">"	";", "=" or "?"
hex.	30 to 39	3A, 3C or 3E	3B, 3D or 3F
dec.	48 to 57	58, 60 or 62	59, 61 or 63

Data Structure

ASCII	DC4	Start code	n ... n	Stop code	DC4
hex.	14	Start code	n ... n	Stop code	14
dec.	20	Start code	n ... n	Stop code	20

Example

```
10 REM  code 2/5 industrial barcode
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" B3;111";CHR$(25);
40 LPRINT CHR$(20);":123;";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code 2/5 Interleaved — Type C

Syntax

	Figures (n)	Start code	Stop code
ASCII	"0" to "9"	":" or "<"	";" or "="
hex.	30 to 39	3A or 3C	3B or 3D
dec.	48 to 57	58 or 60	59 or 61

Data Structure

ASCII	DC4	Start code	n ... n	Stop code	DC4
hex.	14	Start code	n ... n	Stop code	14
dec.	20	Start code	n ... n	Stop code	20

Example

```
10 REM  code 2/5 interleaved barcode
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" C3;111";CHR$(25);
40 LPRINT CHR$(20);":123;";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Due to the "interleaved mechanism", data stream (n ... n) should only be transferred as even number e.g.:
not: 398 **but:** 0398

If odd count of numbers are transferred the printer adds a leading zero to the printed barcode.

Code 11 Type D

Syntax

	Figures/Characters (n)	Start code	Stop code
ASCII hex. dec.	"0" to "9" and "-" 30 to 39 and 2D 48 to 57 and 45	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	n ... n	Stop code	DC4
hex.	14	Start code	n ... n	Stop code	14
dec.	20	Start code	n ... n	Stop code	20

Example

```
10 REM  code 11 barcode
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" D3;111";CHR$(25);
40 LPRINT CHR$(20);":123:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code BCD Matrix Type E

Syntax

	Figures (n)	Start code	Stop code
ASCII hex. dec.	"0" to "9" 30 to 39 48 to 57	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	n ... n	Stop code	DC4
hex.	14	Start code	n ... n	Stop code	14
dec.	20	Start code	n ... n	Stop code	20

Example

```
10 REM  BCD-matrix-code barcode
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" E3;111";CHR$(25);
40 LPRINT CHR$(20);":123:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code 39 — Type F

Syntax

	Numbers/Character (n)	Start code	Stop code
ASCII	"0" to "9", "A" to "Z" and "$", "%", "*", "+", "-", ".", "/"	not fixed, recommended: "*"	not fixed, recommended: "*"
hex.	30 to 39, 41 to 5A and 24, 25, 2A, 2B, 2D, 2E, 2F	2A	2A
dec.	48 to 57, 65 to 90 and 36, 37, 42, 43, 45, 46, 47	42	42

Data Structure

ASCII	DC4	Start code	n ... n	Stop code	DC4
hex.	14	Start code	n ... n	Stop code	14
dec.	20	Start code	n ... n	Stop code	20

Example

```
10 REM  code 39 barcode
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" F3;111";CHR$(25);
40 LPRINT CHR$(20);"*123*";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

CODABAR	**Type G**

Syntax

	Numbers/Character (n)	Start code	Stop code
		not fixed, recommended:	not fixed, recommended:
ASCII	"0" to "9", "A" to "D" and "$", "+", "-", ".", "/", ":"	"a" to "e" and "n", "t", "*"	"a" to "e" and "n", "t", "*"
hex.	30 to 39 and 24, 2B, 2D, 2E, 2F, 3A	61 to 65 and 6E, 74, 2A	61 to 65 and 6E, 74, 2A
dec.	48 to 57 and 36, 43, 45, 46, 47, 58	97 to 101 and 110, 116,42	97 to 101 and 110, 116,42

Data Structure

ASCII	DC4	Start code	n ... n	Stop code	DC4
hex.	14	Start code	n ... n	Stop code	14
dec.	20	Start code	n ... n	Stop code	20

Example

```
10 REM  codabar barcode
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" G3;111";CHR$(25);
40 LPRINT CHR$(20);"*123*";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code EAN 8 with HRI — Type H

Syntax

	Figures (n)	Check Number (c)	Start code	Separation code	Stop code
ASCII hex. dec.	"0" to "9" 30 to 39 48 to 57	Calculation see section "Check Number Calculation"	":" 3A 58	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	nnnn	Separation code	nnnc	Stop code	DC4
hex.	14	Start code	nnnn	Separation code	nnnc	Stop code	14
dec.	20	Start code	nnnn	Separation code	nnnc	Stop code	20

Example

```
10 REM  code EAN 8 with HRI
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" H3;111";CHR$(25);
40 LPRINT CHR$(20);":0123:4567:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

0123 4567

Code EAN 8 without HRI — Type I

Syntax

	Figures (n)	Check Number (c)	Start code	Separation code	Stop code
ASCII hex. dec.	"0" to "9" 30 to 39 48 to 57	Calculation see section "Check Number Calculation"	":" 3A 58	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	nnnn	Separation code	nnnc	Stop code	DC4
hex.	14	Start code	nnnn	Separation code	nnnc	Stop code	14
dec.	20	Start code	nnnn	Separation code	nnnc	Stop code	20

Example

```
10 REM  code EAN 8 without HRI
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" I3;111";CHR$(25);
40 LPRINT CHR$(20);":0123:4567:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code EAN 13 with HRI — Type K

Syntax

	Figures (n)	Check Number (c)	Start code	Separation code	Stop code
ASCII hex. dec.	"0" to "9" 30 to 39 48 to 57	Calculation see section "Check Number Calculation"	":" 3A 58	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	nnnnnnn	Separation code	nnnnnc	Stop code	DC4
hex.	14	Start code	nnnnnnn	Separation code	nnnnnc	Stop code	14
dec.	20	Start code	nnnnnnn	Separation code	nnnnnc	Stop code	20

Example

```
10 REM  code EAN 13 with HRI
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" K3;111";CHR$(25);
40 LPRINT CHR$(20);":0123456:789012:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

0 123456 789012

Code EAN 13 without HRI — Type L

Syntax

	Figures (n)	Check Number (c)	Start code	Separation code	Stop code
ASCII hex. dec.	"0" to "9" 30 to 39 48 to 57	Calculation see section "Check Number Calculation"	":" 3A 58	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	nnnnnnn	Separation code	nnnnnc	Stop code	DC4
hex.	14	Start code	nnnnnnn	Separation code	nnnnnc	Stop code	14
dec.	20	Start code	nnnnnnn	Separation code	nnnnnc	Stop code	20

Example

```
10 REM  code EAN 13 without HRI
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" L3;111";CHR$(25);
40 LPRINT CHR$(20);":0123456:789012:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code MSI / modified Plessey — Type M

Syntax

	Figures (n)	Start code	Stop code
ASCII	"0" to "9"	":"	";"
hex.	30 to 39	3A	3B
dec.	48 to 57	58	59

Data Structure

ASCII	DC4	Start code	n ... n	Stop code	DC4
hex.	14	Start code	n ... n	Stop code	14
dec.	20	Start code	n ... n	Stop code	20

Example

```
10 REM  code MSI/plessey modified
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" M3;111";CHR$(25);
40 LPRINT CHR$(20);":0123;";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code UPC A with HRI — Type N

Syntax

	Figures (n)	Check Number (c)	Start code	Separation code	Stop code
ASCII hex. dec.	"0" to "9" 30 to 39 48 to 57	Calculation see section "Check Number Calculation"	":" 3A 58	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	nnnnnn	Separation code	nnnnnc	Stop code	DC4
hex.	14	Start code	nnnnnn	Separation code	nnnnnc	Stop code	14
dec.	20	Start code	nnnnnn	Separation code	nnnnnc	Stop code	20

Example

```
10 REM  code UPC A with HRI
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" N3;111";CHR$(25);
40 LPRINT CHR$(20);":012345:678901:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

0 12345 67890

Code UPC A without HRI Type O

Syntax

	Figures (n)	Check Number (c)	Start code	Separation code	Stop code
ASCII hex. dec.	"0" to "9" 30 to 39 48 to 57	Calculation see section "Check Number Calculation"	":" 3A 58	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	nnnnnn	Separation code	nnnnnc	Stop code	DC4
hex.	14	Start code	nnnnnn	Separation code	nnnnnc	Stop code	14
dec.	20	Start code	nnnnnn	Separation code	nnnnnc	Stop code	20

Example

```
10 REM  code UPC A without HRI
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" O3;111";CHR$(25);
40 LPRINT CHR$(20);":012345:678901:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code UPC E with HRI — Type P

Syntax

	Figures (n)	Check Number (c)	Start code	Stop code
ASCII hex. dec.	"0" to "9" 30 to 39 48 to 57	Calculation see section "Check Number Calculation"	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	nnnnnnnc	Stop code	DC4
hex.	14	Start code	nnnnnnnc	Stop code	14
dec.	20	Start code	nnnnnnnc	Stop code	20

Example

```
10 REM  code UPC E with HRI
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" P3;111";CHR$(25);
40 LPRINT CHR$(20);":01234567:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

0 123456

Code UPC E without HRI — Type Q

Syntax

	Figures (n)	Check Number (c)	Start code	Stop code
ASCII hex. dec.	"0" to "9" 30 to 39 48 to 57	Calculation see section "Check Number Calculation"	":" 3A 58	":" 3A 58

Data Structure

ASCII	DC4	Start code	nnnnnnnc	Stop code	DC4
hex.	14	Start code	nnnnnnnc	Stop code	14
dec.	20	Start code	nnnnnnnc	Stop code	20

Example

```
10 REM  code UPC E without HRI
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" Q3;111";CHR$(25);
40 LPRINT CHR$(20);":01234567:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code Delta distance (IBM) Type R

Syntax

	Figures/Characters (n)	Start code	Stop code
ASCII hex. dec.	"0" to "9" and "A" to "F" 30 to 39 and 41 to 46 48 to 57 and 65 to 70	"F" 46 70	"D" 44 68

Data Structure

ASCII	DC4	Start code	n ... n	Stop code	DC4
hex.	14	Start code	n ... n	Stop code	14
dec.	20	Start code	n ... n	Stop code	20

Example

```
10 REM  code delta distance (IBM) without HRI
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" R3;111";CHR$(25);
40 LPRINT CHR$(20);"F0123D";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

Code 128	Type S

Syntax

	Figures/Characters (n)	Start code	Stop code
ASCII hex. dec.	see table section "Description"	none	none

Data Structure

ASCII	DC4	n ... n	DC4
hex.	14	n ... n	14
dec.	20	n ... n	20

Description

While many barcode styles are limited to numbers, Code 128 can encode numbers, letters, and other symbols commonly found on keyboards, such as @, # and %. The Code 128 barcode is like three barcodes in one. The three barcode styles are called Code A, Code B, and Code C. Each barcode style is designed to encode certain types of data in the most compact way.

Code A encodes uppercase alpha, numeric and control codes.

Code B encodes upper and lower case alpha and numeric codes.

Code C encodes digits in pairs.

A single *Code 128* barcode may consist of all three Code 128 barcode styles A, B and C. The printer chooses the styles which result in the most compact Code 128 barcode for the data to be encoded. Thus, the user needn’t be concerned about choosing the correct barcode styles as this is done automatically.

The ">" (greater than) symbol is a special character prefix. If a character with decimal value less then 32 needs to be encoded (i.e., a control character), then send a ">" followed by the character that is decimal 64 higher then the control code character.

Control characters are conflicting with DC4, SUB ... these characters are encoded by using the ">" sign following a number that is 64 dec. higher than the control character.

Example

If you need to print a Carriage Return (hex.0D, dec.13), as part of a barcode, you need to send ">M" (13 + 64 = 77, see character set table to verify that decimal 77 equals the capital letter "M"). This substitution process can be carried out throughout the range of decimal values, i.e., ">d" (decimal 100) equates to the dollar symbol (decimal 36). If the greater-than symbol itself must be transmitted, send ">0" (the 0 will not be part of the printed data).

The Code 128 barcode style selection can also be done manually. This is done by adding a style selection character to the beginning of the barcode data. Choose code A, B, or C from the table of special characters below:

Special Character	Code 128
">0"	30 (">")
">1"	95
">2"	96
">3"	97
">4"	98
">5"	99 (Code C)
">6"	100 (Code B)
">7"	101 (Code A)
">8"	102

When a style selection has been made, character data will be translated from the selected code style to the *Code 128* repesentation. The translation table on the following page summarizes this:

Code 128 translation table

Wert	Code A	Code B	Code C
0	Space	Space	00
1	!	!	01
2	"	"	02
3	#	#	03
4	$	$	04
5	%	%	05
6	&	&	06
7	‘	‘	07
8	(	(	08
9	)	)	09
10	*	*	10
11	+	+	11
12	‘	‘	12
13	hyphen	hyphen	13
14	period	period	14
15	/	/	15
16	0	0	16
17	1	1	17
18	2	2	18
19	3	3	19
20	4	4	20
21	5	5	21
22	6	6	22
23	7	7	23
24	8	8	24
25	9	9	25
26	:	:	26
27	;	;	27
28	<	<	28
29	=	=	29
30	>	>	30
31	?	?	31
32	@	@	32
33	A	A	33
34	B	B	34
35	C	C	35

Wert	Code A	Code B	Code C
36	D	D	36
37	E	E	37
38	F	F	38
39	G	G	39
40	H	H	40
41	I	I	41
42	J	J	42
43	K	K	43
44	L	L	44
45	M	M	45
46	N	N	46
47	O	O	47
48	P	P	48
49	Q	Q	49
50	R	R	50
51	S	S	51
52	T	T	52
53	U	U	53
54	V	V	54
55	W	W	55
56	X	X	56
57	Y	Y	57
58	Z	Z	58
59	[	[	59
60	\	\	60
61	]	]	61
62	^	^	62
63	_	_	63
64	NUL	‘	64
65	SOH	a	65
66	STX	b	66
67	ETX	c	67
68	EOT	d	68
69	ENQ	e	69
70	ACK	f	70
71	BEL	g	71

Wert	Code A	Code B	Code C
72	BS	h	72
73	HT	i	73
74	LF	j	74
75	VT	k	75
76	FF	l	76
77	CR	m	77
78	SO	n	78
79	SI	o	79
80	DLE	p	80
81	DC1	q	81
82	DC2	r	82
83	DC3	s	83
84	DC4	t	84
85	NAK	u	85
86	SYN	v	86
87	ETB	w	87
88	CAN	x	88
89	EM	y	89
90	SUB	z	90
91	ESC	{	91
92	FS	|	92
93	GS	}	93
94	RS	~	94
95	US	DEL	95
96	FNC3	FNC3	96
97	FNC2	FNC2	97
98	SHIFT	SHIFT	98
99	CODEC	CODEC	99
100	CODEB	FUNC4	CODEB
101	FNC4	CODEA	CODEA
102	FNC1	FNC1	FNC1

Codes 96 through 102 do not have corresponding ASCII character translations; these may be encoded using the special character table above.

Example

```
10 REM  code 128
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" S3;111";CHR$(25);
40 LPRINT CHR$(20);"ABCD0123";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

EAN 128 Type T

Syntax

	Figures/Characters (n)	Start code	Stop code
ASCII hex. dec.	see Code 128	none	none

Data Structure

ASCII	DC4	n ... n	DC4
hex.	14	n ... n	14
dec.	20	n ... n	20

Description

The only difference between both types of barcodes is the initial sequence.
Code 128 starts with Code A, Code B or Code C followed by character information.
Barcode EAN 128 requires the code FNC1 between Startcode and character information.
The Code 128 translation table remains valid. The checksum of EAN 128 is calculated using Code 128 algorythm.

Example

```
10 LPRINT CHR$(27);"[?11~";
20 LPRINT CHR$(26);" T3;111";CHR$(25);
30 LPRINT CHR$(20);"1234ABCD";CHR$(20);
40 LPRINT CHR$(27);"[?10~"
```

Add-On Barcodes

UPC and EAN barcodes (commercial barcodes) can be extended with Add-On barcodes.

The following barcodes can be extended with Add-On barcodes:
Barcode Types H, I, K, L, N, O, P and Q

In the data formats the numbers are generally given in the sequence they are printed from left to right in the barcode, i.e. in the case of the EAN13 code, the 13. figure is transferred first and the 1. figure last. For the formats for the Add-On barcodes, the printer expects the check number as the first figure. This is not printed in the barcode. The following numbers (2 or 5) are printed from left to right in Add-On barcode.

Example 1

EAN13 barcode with HRI and Add On 2 extention

Format: DC4:nnnnnnn:nnnnnn**:cnn**:DC4

```
10 REM  code EAN 13 with add-on-2 extention
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" K3;111";CHR$(25);
40 LPRINT CHR$(20);":0123456:789012:012:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

0 123456 789012 12

Example 2 EAN13 barcode with HRI and Add On 5 extention

Format: DC4:nnnnnnn:nnnnnn**:cnnnnn**:DC4

```
10 REM  code EAN 13 with add-on-5 extention
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(26);" K3;111";CHR$(25);
40 LPRINT CHR$(20);":0123456:789012:012345:";CHR$(20);
50 LPRINT CHR$(27);"[?10~"
```

0 123456 789012 12345

Check Number Calculation

The following barcode types must be given a check number for transfer to the printer:

- Type H and I (EAN 8)
- Type K and L (EAN 13)
- Type N and O (UPC A)
- Type P and Q (UPC E)

The check number (c) is transferred after the barcode information (n). For the "EAN" type, this is printed as the last figure in the HRI data line. For the "UPC" type, the check number is not printed in the HRI data line.

The commercial barcodes can be extended with the following user-specific additional barcodes (see also section "Add-On Barcodes"):

- Add-On 2 barcode
- Add-On 5 barcode

Here the check number (c) is transferred before the barcode information (n).

EAN 13

12 numbers are transferred for the EAN 13 code. The 13. digit represents the check number. The even figures of the information have a factor of "3", the uneven figures a factor of "1". Counting begins from the left. The resulting cross sum is divided by 10. The remaining figure is subtracted from the modul (10). If the remainder is "0", the check number is also "0".

Example

Information	4		0		1		2		3		4		5		6		7		8		9		0	1
Factor	1		3		1		3		1		3		1		3		1		3		1		3	
Product	4	+	0	+	1	+	6	+	3	+	12	+	5	+	18	+	7	+	24	+	9	+	0	
Cross sum	89																							

10 (modul) - 9 (remainder) = 1 (check number)

For remainder "0" the check number is also "0".

EAN 8

7 numbers are transferred for the EAN 8 code. The 8. digit represents the check number. The even figures of the information have a factor of "3", the uneven figures a factor of "1". Counting begins from the left. The resulting cross sum is divided by 10. The remaining figure is subtracted from the modul (10). If the remainder is "0", the check number is also "0".

Example

Information	4	0	1	2	3	4	6	2
Factor	3	1	3	1	3	1	3	
Product	12 +	0 +	3 +	2 +	9 +	4 +	18	
Cross sum	48							

10 (modul) - 8 (remainder) = 2 (check number)

For remainder "0" the check number is also "0".

UPCA

11 numbers are transferred for the UPCA code. The 12. digit represents the check number. The even figures of the information have a factor of "3", the uneven figures a factor of "1". Counting begins from the left. The resulting cross sum is divided by 10. The remaining figure is subtracted from the modul (10). If the remainder is "0", the check number is also "0".

Example

Information	4	0	1	2	3	4	5	6	7	8	9	3
Factor	3	1	3	1	3	1	3	1	3	1	3	
Product	12 +	0 +	3 +	2 +	9 +	4 +	15 +	6 +	21 +	8 +	27	
Cross sum	107											

10 (modul) - 7 (remainder) = 3 (check number)

For remainder "0" the check number is also "0".

UPCE

For the UPCE barcode type, 7 numbers are transmitted. The 8. digit is the check number. The uneven figures of the information have a factor of "3", the even figures a factor of "1". Counting begins from the left. The resulting cross sum is divided by 10. The remaining integer value is subtracted from the modul (10). The result is transmitted as the check number (c).

Example

Information	0	1	2	3	4	5	6	5
Factor	3	1	3	1	3	1	3	
Product	0 +	1 +	6 +	3 +	12 +	5 +	18	
Cross sum	45							

10 (modul) - 5 (remainder) = 5 (check number)

For remainder "0" the check number is also "0".

Add-On 5

6 numbers are transmitted: check number (c) + 5 information items (n). The uneven figures of the information have a factor of "3", the even figures a factor of "1". Counting begins from the right. The resulting cross sum is divided by 10. The remaining integer value is subtracted from the modul (10). The result is transmitted as the check number (c).

Example

Information	3	8	6	1	0	4
Factor		3	9	3	9	3
Product		24 +	54 +	3 +	0 +	12
Cross sum		93				

3 (remainder) = 3 (check number)

The check number is not printed in the clear data line (HRI).

Add-On 2

3 numbers are transmitted: check number (c) and 2 informations (n).
The check number results from the remaining integer value of modul (4).

Example 1

Information	0	0	4
Remainder (0)		0	

Example 2

Information	2	0	6
Remainder (2)		2	

Example 3

Information	3	9	9
Remainder (3)		3	

Example 4

Information	1	0	9
Remainder (1)		1	

The check number always lies between "0" and "3"; it is not printed in the HRI line.

US Postnet Barcode — ESC [1 SP p

Data Structure

ASCII	ESC	"["	"1"	SP	"p"
hex.	1B	5B	31	20	70
dec.	27	91	49	32	112

Description

After receiving this sequence numeric characters from 0 (hex. 30) to 9 (hex. 39) are interpreted as barcode figures. Other Alpha characters are ignored. Control characters from hex.00 (dec.9) to hex.1F (dec.31) terminates this barcode mode.

You may use this sequence in every emulation. The printer can print barcode in NLQ and LQ. If draft print quality (DPQ) is selected the printout is performed in NLQ.

The character (e.g. CR = carriage return, hex. 0D), which terminates barcode mode will neither be printed nor carried out. A tab command (hex.09, dec.9) is carried out in this barcode.

```
10 REM  US postnet barcode
20 LPRINT CHR$(27);"[?11~";
30 LPRINT CHR$(27);"[1 p";"1234567";CHR$(13);
40 LPRINT CHR$(27);"[?10~"
```

Large Character Printing (LCP)

Before you can transfer the LCP data (Large Character Printing) data to the printer, the LCP header must be sent. The LCP header is made up of a series of max. 5 characters. In the header, the printing parameters and the LCP character size are defined. This header only needs to be transferred once, unless settings are to be changed or the printer has been turned off.

LCP Header

Format: DLE [!] n EM

[ ] optional, does not need to be transmitted

Character meaning:

DLE	(hex.10, dec.16)		start character
!	ASCII		bidirectional printing (if transferred)
n	ASCII	n = "2"..."99"	enlargement factor
EM	(hex.19, dec.25)		end character

with the help of the header, the enlargement factor is fixed to the original size. A character in LCP mode with the size n occupies a horizontal space for n normal character, depending on the selected character distance. At a character distance of 10 characters/inch and an enlargement factor of 6, the LCP font width is 6/10 inch.

The height of a LCP character is n times 1/12 inch. It is independent of the selected line spacing. The LCP characters are printed, so that their lower edges lie flush with the next available ground line. The result is that only the upper edges of characters with an uneven enlargement factor can lie flush with the line.

LCP Data

The existence of a LCP header does not mean that all subsequent characters are printed in LCP size. The LCP mode must be begun and it must end with the LCP brackets SI (hex.0F, dec.15). All characters inside these brackets are printed as LCP characters. The LCP mode is ended by all characters from hex.00 to hex.1F.

LCP characters can be printed with character densities 10, 12, 15, 17.1 and 20 cpi. If there is a LCP line overflow, the printing procedure is automatically started. All characters which caused the overflow are then printed as normal characters.

Refer to the section "Secured/Unsecured Mode" at the front of the Manual for detailed description of the secured/unsecured mode.

Data Structure

ASCII	SI	n ... n	SI
hex.	0F	n ... n	0F
dec.	15	n ... n	15

n = all printable characters (> hex.1F, dec.31)

Example

```
REM Example Character Densities
OPEN "lpt1:" FOR RANDOM AS #1
WIDTH #1,  255
PRINT #1, CHR$(27); "[?11~";
PRINT #1,  CHR$(16); "4"; CHR$(25);
PRINT #1, CHR$(27); "[6w"; :REM 15CPI
PRINT #1, CHR$(15); "LCP ";CHR$(15);
PRINT #1, CHR$(27); "[5w"; :REM 12CPI
PRINT #1, CHR$(15); "LCP ";CHR$(15);
PRINT #1, CHR$(27); "[4w"; :REM 10CPI
PRINT #1, CHR$(15); "LCP ";CHR$(15);

PRINT #1, STRING$ (5, 10);

END
```

LCP LCP LCP

LCP Character Set US-ASCII	**ESC P L S 0 1 ESC **
LCP Character Set German	**ESC P L S 0 2 ESC **

Data Structure	ASCII	ESC	"P"	"L"	"S"	"0"	"1"	ESC	"\\"	character set US-ASCII
	hex.	1B	50	4C	53	30	31	1B	5C	
	dec.	27	80	76	83	48	49	27	92	
	ASCII	ESC	"P"	"L"	"S"	"0"	"2"	ESC	"\\"	character set german
	hex.	1B	50	4C	53	30	32	1B	5C	
	dec.	27	80	76	83	48	50	27	92	

Various Examples

Example 1

```
10  LPRINT "Example for different LCP character sizes"
20  LPRINT CHR$(27); "[?11~";
30  LPRINT CHR$(27); "PSC0"; CHR$(27); "\";
40  REM set character size to 3
50  LPRINT CHR$(16); "3"; CHR$(25);
60  LPRINT CHR$(15); "3"; CHR$(15);
70  REM set character size to 4
80  LPRINT CHR$(16); "4"; CHR$(25);
90  LPRINT CHR$(15); "4"; CHR$(15);
100 REM set character size to 5
110 LPRINT CHR$(16); "5"; CHR$(25);
120 LPRINT CHR$(15); "5"; CHR$(15);
130 REM set character size to 6
140 LPRINT CHR$(16); "6"; CHR$(25);
150 LPRINT CHR$(15); "6"; CHR$(15);
160 LPRINT CHR$(27); "[?10~"
170 END
```

```
Example for different LCP character sizes

3456
```

Example 2

```
REM Example 2 for unsecured printing
OPEN "lpt1:" FOR RANDOM AS #1
WIDTH #1, 255
PRINT #1, CHR$(27); "[?11~";
PRINT #1, CHR$(27); "PSCO"; CHR$(27); "\";
PRINT #1, CHR$(16); "5"; CHR$(25);
PRINT #1, "Example for ";
PRINT #1, CHR$(15); "LCP"; CHR$(15);" unsecured mode";
PRINT #1, CHR$(10); CHR$(13);
END
```

Example for **LCP** unsecured mode

Example 3

```
REM Example 3 for secured printing
OPEN "lpt1:" FOR RANDOM AS #1
WIDTH #1, 255
PRINT #1,  CHR$(27); "[?11~";
PRINT #1, CHR$(27); "PSC1"; CHR$(27); "\";
PRINT #1, CHR$(16); "5"; CHR$(25);
PRINT #1, "Example for ";
PRINT #1,  CHR$(15); "LCP"; CHR$(15);
PRINT #1, " secured"; CHR$(10); "mode";
PRINT #1, STRING$(5, 10);
END
```

Example for **LCP** secured
mode

Example 4 Unsecured Mode, vertical spacing with various enlargement factors

Legend

A – a single line Feed (1/6")

B – Factor 6 character; heigth $6\times\frac{1}{12} \widehat{=} 3\times\frac{1}{6}$ LF

C – Factor 7 character; height $7\times\frac{1}{12}$ additional space is added to gain a full Line Feed

D – Factor 4 character; height $4\times\frac{1}{12} \widehat{=} 2\times\frac{1}{6}$ LF

E – Factor 5 character; height $5\times\frac{1}{12}$ additional space is added to gain a full Line Feed

▲ – start, actual print (cursor)-position

● – end, actual print-position

Description The room will be occupied to fit the highest character in one line. Characters which will not meet the 1/6" grid *) are adding additional space to fill the room to the next possible line.

*) this value may change due to actual line density setting.

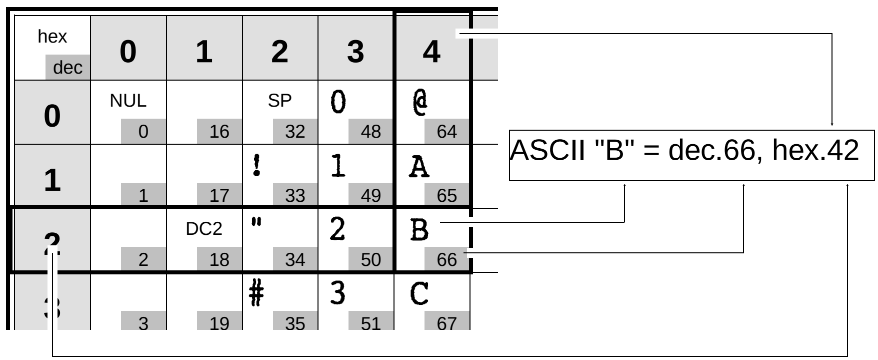
Appendix

Character Sets

The following symbol sets are available in the barcode mode.

The LCP symbol sets can only be selected via Escape sequences. The OCR-A and OCR-B fonts can be selected via control panel or by Escape sequences, which correspond to the selected emulation mode. The codes hex.A0 up to hex.FE correspond to the selected character set.

The following example shows you how to find the hexadecimal value for a character from the character set table.



LCP Character Set, german

hex / dec	0	1	2	3	4	5	6	7
0	NUL 0	16	32	0 48	§ 64	P 80	` 96	p 112
1	1	17	! 33	1 49	A 65	Q 81	a 97	q 113
2	2	18	" 34	2 50	B 66	R 82	b 98	r 114
3	3	19	# 35	3 51	C 67	S 83	c 99	s 115
4	4	20	$ 36	4 52	D 68	T 84	d 100	t 116
5	5	21	% 37	5 53	E 69	U 85	e 101	u 117
6	6	22	& 38	6 54	F 70	V 86	f 102	v 118
7	7	23	' 39	7 55	G 71	W 87	g 103	w 119
8	8	24	(40	8 56	H 72	X 88	h 104	x 120
9	9	25	) 41	9 57	I 73	Y 89	i 105	y 121
A	10	26	* 42	: 58	J 74	Z 90	j 106	z 122
B	11	27	+ 43	; 59	K 75	Ä 91	k 107	ä 123
C	12	28	, 44	< 60	L 76	Ö 92	l 108	ö 124
D	13	29	- 45	= 61	M 77	Ü 93	m 109	ü 125
E	14	30	. 46	> 62	N 78	↑ 94	n 110	ß 126
F	15	31	/ 47	? 63	O 79	← 95	o 111	ä 127

LCP Character Set, US-ASCII

hex / dec	0	1	2	3	4	5	6	7
0	NUL 0	16	32	0 48	@ 64	P 80	` 96	p 112
1	1	17	! 33	1 49	A 65	Q 81	a 97	q 113
2	2	18	" 34	2 50	B 66	R 82	b 98	r 114
3	3	19	# 35	3 51	C 67	S 83	c 99	s 115
4	4	20	$ 36	4 52	D 68	T 84	d 100	t 116
5	5	21	% 37	5 53	E 69	U 85	e 101	u 117
6	6	22	& 38	6 54	F 70	V 86	f 102	v 118
7	7	23	' 39	7 55	G 71	W 87	g 103	w 119
8	8	24	(40	8 56	H 72	X 88	h 104	x 120
9	9	25	) 41	9 57	I 73	Y 89	i 105	y 121
A	10	26	* 42	: 58	J 74	Z 90	j 106	z 122
B	11	27	+ 43	; 59	K 75	[91	k 107	{ 123
C	12	28	, 44	< 60	L 76	\ 92	l 108	| 124
D	13	29	- 45	= 61	M 77	] 93	m 109	} 125
E	14	30	. 46	> 62	N 78	↑ 94	n 110	‾ 126
F	15	31	/ 47	? 63	O 79	← 95	o 111	ä 127

OCR-A Character Set (Code Page 437)

hex / dec	0	1	2	3	4	5	6	7
0	NUL 0	16	32	0 48	@ 64	P 80	` 96	p 112
1	1	17	! 33	1 49	A 65	Q 81	a 97	q 113
2	2	18	" 34	2 50	B 66	R 82	b 98	r 114
3	3	19	# 35	3 51	C 67	S 83	c 99	s 115
4	4	20	$ 36	4 52	D 68	T 84	d 100	t 116
5	5	21	% 37	5 53	E 69	U 85	e 101	u 117
6	6	22	& 38	6 54	F 70	V 86	f 102	v 118
7	7	23	' 39	7 55	G 71	W 87	g 103	w 119
8	8	24	(40	8 56	H 72	X 88	h 104	x 120
9	9	25	) 41	9 57	I 73	Y 89	i 105	y 121
A	10	26	* 42	: 58	J 74	Z 90	j 106	z 122
B	11	27	+ 43	; 59	K 75	[91	k 107	{ 123
C	12	28	, 44	< 60	L 76	\ 92	l 108	| 124
D	13	29	- 45	= 61	M 77	] 93	m 109	} 125
E	14	30	. 46	> 62	N 78	^ 94	n 110	~ 126
F	15	31	/ 47	? 63	O 79	_ 95	o 111	127

OCR-B Character Set (Code Page 437)

hex / dec	0	1	2	3	4	5	6	7
0	NUL 0	16	32	0 48	@ 64	P 80	` 96	p 112
1	1	17	! 33	1 49	A 65	Q 81	a 97	q 113
2	2	18	" 34	2 50	B 66	R 82	b 98	r 114
3	3	19	# 35	3 51	C 67	S 83	c 99	s 115
4	4	20	$ 36	4 52	D 68	T 84	d 100	t 116
5	5	21	% 37	5 53	E 69	U 85	e 101	u 117
6	6	22	& 38	6 54	F 70	V 86	f 102	v 118
7	7	23	' 39	7 55	G 71	W 87	g 103	w 119
8	8	24	(40	8 56	H 72	X 88	h 104	x 120
9	9	25	) 41	9 57	I 73	Y 89	i 105	y 121
A	10	26	* 42	: 58	J 74	Z 90	j 106	z 122
B	11	27	+ 43	; 59	K 75	[91	k 107	{ 123
C	12	28	, 44	< 60	L 76	\ 92	l 108	| 124
D	13	29	- 45	= 61	M 77	] 93	m 109	} 125
E	14	30	. 46	> 62	N 78	^ 94	n 110	~ 126
F	15	31	/ 47	? 63	O 79	_ 95	o 111	127