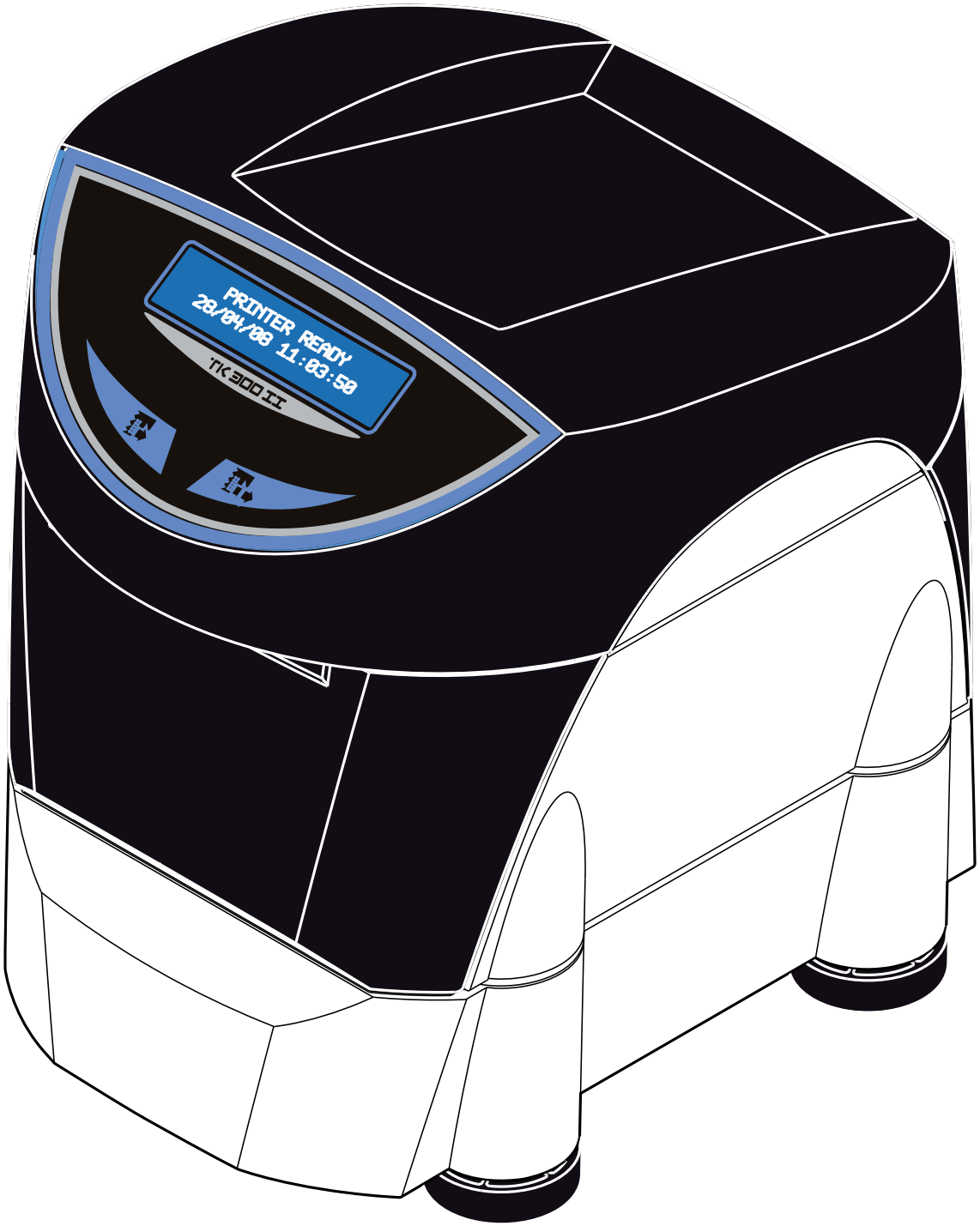
TK300II

COMMAND REFERENCE

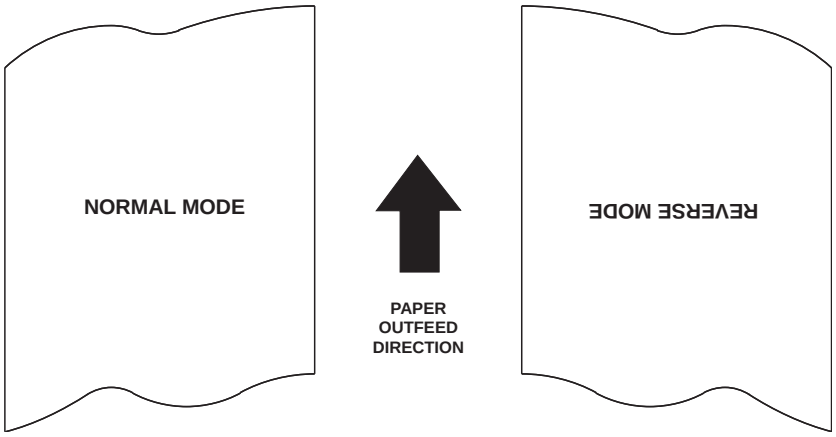


Documentazione redatta da:

CUSTOM ENGINEERING S.p.A.
Str. Berettine 2 - 43010 Fontevivo (PARMA) - Italy
http: www.custom.it

1.1 PRINT DIRECTION

The printer has two print modes, selectable through the control characters: normal and reverse.



(Fig.1.1)

1.2 COMMAND DESCRIPTIONS

The table 1.1 shows the commmands list, ordered by their hexadecimal value.

LEGEND :

Symbol	Function
$	indicates the representation of the command hexadecimal value (for example $40 means HEX 40).
{ }	indicates an ASCII character not performable.
n, m, t, x, y	are optional parameters that can have different values.

1.2.1 ESC/POS Emulation

The following table lists all the commands for function management in ESC/POSä emulation of the printer. The commands can be transmitted to the printer at any moment, but they will only be carried out when the commands ahead of them have been executed. The commands are carried out when the circular buffer is free to do so.

(Tab.1.1)

COMMAND DESCRIPTION TABLE

HEX	ASCII	Description
$08	BS	Back space
$09	HT	Horizontal tab
$0A	LF	Print and line feed
$0D	CR	Print and carriage return
$10 $04 n	DLE EOT n	Real-time status transmission
$18	CAN	Cancel current line transmitted
$1B $20 n	ESC SP n	Set character right-side spacing
$1B $21 n	ESC ! n	Set print mode
$1B $24 nL nH	ESC $ nL nH	Set absolute position
$1B $25 n	ESC % n	Select/cancel user-defined character set
$1B $26 y c1 c2	ESC & y c1 c2	Define user-defined characters
$1B $28 $76 nL nH	ESC (v nL nH	Set relative vertical print position

$1B $2A m nL nH d1...dk	ESC * m nL nH d1...dk	Select image print mode	
$1B $2D n	ESC - n	Turn underline mode on/off	
$1B $30	ESC 0	Select 1/8-inch line spacing	
$1B $32	ESC 2	Select 1/6-inch line spacing	
$1B $33 n	ESC 3 n	Set line spacing using minimum units	
$1B $3D n	ESC = n	Select device	
$1B $3F n	ESC ? n	Cancel user-defined characters	
$1B $40	ESC @	Initialize printer	
$1B $44 n1...nk 00	ESC D n1...nk NUL	Set horizontal tab positions	
$1B $45 n	ESC E n	Select emphasized mode	
$1B $47 n	ESC G n	Select double-strike mode	
$1B $4A n	ESC J n	Print and feed the paper	
$1B $4D n	ESC M n	Select character font	
$1B $52 n	ESC R n	Select international character set	
$1B $56 n	ESC V n	Select print mode 90° turned	
$1B $5C nL nH	ESC \ nL nH	Set relative print position	
$1B $61 n	ESC a n	Select justification	
$1B $64 n	ESC d n	Print and feed paper n lines	
$1B $69	ESC i	Total cut	
$1B $74 n	ESC t n	Select character code table	
$1B $76	ESC v	Transmit printer status	
$1B $7B n	ESC { } n	Set/cancel upside-down character printing	
$1B $C1 n	ESC { } n	Set/cancel cpi mode	
$1C $3C n	FS < n	Change printer emulation to SVELTA	
$1C $80	FS { }	Read date/time of the real time clock	
$1C $81 m n d0...dn	FS { } m n d0...dn	Set date/time of the real time clock	
$1C $82	FS { }	Print date	
$1C $83	FS { }	Print time	
$1C $84 n d1...dk 00	FS { } n d1...dk NUL	Set User-Defined Date/Time Formats	
$1C $90	FS { }	Get number of stored logo	
$1C $91	FS { }	Get pictures header list	
$1C $92 nH nL	FS { } nH nL	Get pictures header info	
$1C $93 nH nL	FS { } nH nL	Print logo	
$1C $94 nH nL xDimH xDimL yDimH yDimL TbdH RbdL ld0...ldn d0...dn	FS { } nH nL xDimH xDimL yDimH yDimL TbdH RbdL ld0...ldn d0...dn	Save the image received from serial port into the flash	
$1C $B0 n	FS { } n	Sets the barcode reader status	*Only in the version with barcode scanner*
$1C $B1 n	FS { } n	Get barcode reader status	
$1C $B2	FS { }	Barcode reader Trigger	
$1D $21 n	GS ! n	Select character size	
$1D $2A x y d1...d (x x y x 8)	GS * x y d1...d(x x y x 8)	Define downloaded bit image	
$1D $2F m	GS / m	Print downloaded bit image	
$1D $3A	GS :	Set start/end of macro definition	

$1D $42 n	GS B n	Turn white/black reverse printing mode on/off
$1D $48 n	GS H n	Select printing position of HRI characters
$1D $49 n	GS I n	Transmit printer ID
$1D $4C nL nH	GS L nL nH	Set left margin
$1D $50 x y	GS P x y	Set horizontal and vertical motion unit
$1D $56 m	GS V m	Select cut mode
$1D $57 nL nH	GS W nL nH	Set printing area width
$1D $5E r t m	GS ^ r t m	Execute macro
$1D $66 n	GS f n	Select font for HRI characters
$1D $68 n	GS h n	Select height of bar code
$1D $6B m 00	GS k m NUL	Print bar code
$1D $72 n	GS r n	Transmit status
$1D $76 $30 m xL xH yL yH d1...dk	GS v 0 m xL xH yL yH d1...dk	Select horizontal side (enlargement) of bar code
$1D $77 n	GS w n	Set barcode width
$1D $7C n	GS { } n	Set printing density
$1D $E0 n	GS { } n	Enable/disable automatic full status back
$1D $E1	GS { }	Reading of length paper (cm) available before virtual paper end
$1D $E2	GS { }	Reading number of cuts performed from the printer
$1D $E3	GS { }	Reading of length (cm) of printed paper
$1D $E5	GS { }	Reading number of power on
$1D $E6 nH nL	GS { } nH nL	Set virtual paper end limit
$1D $E7 nH nL	GS { } nH nL	Set notch distance
$1D $F0 n	GS { } n	Set printing speed
$1D $F6	GS { }	Ticket align at first printing line
$1D $F8	GS { }	Ticket align at cut

Given below are more detailed descriptions of each command.

$08

[Name] **Back space**
[Format] ASCII BS
Hex 08
Decimal 8
[Description] Moves print position to previous character.
[Notes] Can be used to put two characters at the same position.
[Default]
[Reference]
[Example]

$09

[Name] **Horizontal tab**
[Format] ASCII HT
Hex 09
Decimal 9
[Description] Moves the print position to the next horizontal tab position.
[Notes] • Ignored unless the next horizontal tab position has been set.
• If the command is received when the printing position is at the right margin, the printer

executes print buffer full printing and horizontal tab processing from the beginning of the next line.
• Horizontal tab positions are set using **$1B $44**.

[Default]
[Reference] $1B $44
[Example]

$0A

[Name] **Print and line feed**
[Format]

ASCII	LF
Hex	0A
Decimal	10

[Range]
[Description] Prints the data in the buffer and feeds one line based on the current line spacing.
[Notes] • Sets the print position to the beginning of the line.
• If the buffer is empty, the printing feeds of (character height + spacing gap) dot.

[Default]
[Reference] $1B $32, $1B $33, $0D
[Example]

$0D

[Name] **Print and carriage return**
[Format]

ASCII	CR
Hex	0D
Decimal	13

[Description] When autofeed is “CR enabled”, this command functions in the same way as $0A, otherwise it is disregarded.
[Notes] • Sets the print position to the beginning of the line.
[Default] See “Autofeed in setup” parameter.
[Reference] $0A
[Example]

$10 $04 n

[Name] **Real-time status transmission**
[Format]

ASCII	DLE	EOT	n
Hex	10	04	n
Decimal	16	4	n

[Range] 1 £ n £ 4, 17, 20, 21
[Description] Transmits the selected printer status specified by n in real time according to the following parameters:

n = 1	transmit printer status
n = 2	transmit off-line status
n = 3	transmit error status
n = 4	transmit paper roll sensor status
n = 17	transmit print status
n = 20	transmit FULL STATUS
n = 21	transmit printer ID

[Notes] • Immediately executed even when the data buffer is full.
• This status is transmitted whenever data sequence $10 $04 n is received.

[Default]
[Reference] See tables below.
[Example]

n=1: Printer status

Bit	Off/On	Hex	Decimal	Function
0	Off	00	0	Not used. Fixed to Off.
1	ON	02	2	Not used. Fixed to On.
2	-	-	-	RESERVED.
3	Off	00	0	On-line.
	On	08	8	Off-line.
4	On	10	16	Not used. Fixed to On.
5	-	-	-	RESERVED
6	-	-	-	RESERVED
7	Off	00	0	Not used. Fixed to Off.

n=2: Off-line status

Bit	Off/On	Hex	Decimal	Function
0	Off	00	0	Not used. Fixed to Off.
1	On	02	2	Not used. Fixed to On
2	Off	00	0	Cover closed.
	On	04	4	Cover opened.
3	Off	00	0	Paper isn't feeded by LINE FEED button
	On	08	8	Paper is feeded by LINE FEED button.
4	On	10	16	Not used. Fixed to On.
5	Off	00	0	aper present.
	On	20	32	Printing stop due to paper end.
6	Off	00	0	No error.
	On	40	64	Error
7	Off	00	0	Not used. Fixed to Off.

n=3: Stato di errore

Bit	Off/On	Hex	Decimal	Function
0	-	-	-	Not used. Fixed to Off.
1	-	-	-	Not used. Fixed to On.
2	-	-	-	RESERVED
3	Off	00	0	Cutter ok.
	On	08	8	Cutter error.
4	-	-	-	Not used. Fixed to On.
5	Off	00	0	No unrecoverable error.
	On	20	32	Unrecoverable error.
6	Off	00	0	No auto-recoverable error.
	On	40	64	Auto-recoverable error.
7	-	-	-	Not used. Fixed to Off.

n=4: Paper roll sensor status

Bit	Off/On	Hex	Decimal	Function
0	-	-	-	Not used. Fixed to Off.
1	-	-	-	Not used. Fixed to On.
2,3	Off	00	0	Paper present in abundance.
	On	0C	12	Near paper end.
4	-	-	-	Not used. Fixed to On.
5, 6	Off	00	0	Paper present.
	On	60	96	Paper not present.
7	-	-	-	Not used. Fixed to Off.

n=17: Print status

Bit	Off/On	Hex	Decimal	Function
0	-	-	-	Not used. Fixed to Off.
1	-	-	-	Not used. Fixed to On.
2	Off	00	0	Paper drag motor off.
	On	04	4	Paper drag motor on.
3	-	-	-	RESERVED
4	-	-	-	Not used. Fixed to On.
5	Off	00	0	Paper present.
	On	20	32	Paper absent.
6	-	-	-	RESERVED
7	-	-	-	Not used. Fixed to Off.

n=20: FULL status (6 bytes)

1° byte = $10 (DLE); 2° byte = $0F; 3° byte = Paper status

Bit	Off/On	Hex	Decimal	Function
0	Off	00	0	Paper present.
	On	01	1	Paper not present.
1	-	-	-	RESERVED
2	Off	00	0	Paper present in abundance.
	On	04	4	Near paper end.
3	-	-	-	RESERVED
4	-	-	-	RESERVED
5	Off	00	0	Ticket not present in output.
	On	20	32	Ticket present in output.
6	Off	00	0	Not virtual paper end (*).
	On	40	64	Virtual paper end (*).
7	Off	00	0	Notch not found
	On	80	128	Found Notch

(*) Virtual paper end is set when the paper length available, readed by $1D $E1, is 0.

4° byte = USER STATUS

Bit	Off/On	Hex	Decimal	Function
0	Off	00	0	Printing head down.
	On	01	1	Printing head up error.
1	Off	00	0	Cover closed.
	On	02	2	Cover opened.
2	Off	00	0	No spooling.
	On	04	4	Spooling.
3	Off	00	0	Drag paper motor off.
	On	08	8	Drag paper motor on.
4	-	-	-	RESERVED
5	Off	00	0	LF key released.
	On	20	32	LF key pressed.
6	Off	00	0	FF key released.
	On	40	64	FF key pressed.
7	-	-	-	RESERVED

5° byte = Recoverable error Status

Bit	Off/On	Hex	Decimal	Function
0	Off	00	0	Head temperature ok.
	On	01	1	Head temperature error.
1	Off	00	0	No COM error.
	On	02	2	RS232 COM error.
2	-	-	-	RESERVED
3	Off	00	0	Power supply voltage ok.
	On	08	8	Power supply voltage error.
4	-	-	-	RESERVED
5	Off	00	0	Acknowledge command.
	On	20	32	Not acknowledge command error.
6	Off	00	0	Free paper path.
	On	40	64	Paper jam.
7	Off	00	0	Notch search ok
	On	80	128	Error in Notch search

6° byte = Unrecoverable error Status

Bit	Off/On	Hex	Decimal	Function
0	Off	00	0	Cutter ok.
	On	01	1	Cutter error.
1	Off	00	0	Cutter cover ok
	On	02	2	Cutter cover open
2	Off	00	0	RAM ok.
	On	04	4	RAM error.
3	Off	00	0	EEPROM ok.
	On	08	8	EEPROM error.
4	-	-	-	RESERVED
5	-	-	-	RESERVED
6	-	-	-	RESERVED
7	-	-	-	RESERVED

n=21: transmit printer ID 1° byte = $75 (refer to command $1D $49)

$18

[Name] **Cancel current line transmitted**

[Format]

ASCII	CAN	
Hex	18	
Decimal	24	

[Range]

[Description] Deletes current line transmitted.

[Notes]
- Sets the print position to the beginning of the line.
- However, this command does not clear the receive buffer.

[Default]

[Reference]

[Example]

$1B $20 n

[Name] **Set right-side character spacing**

[Format]

ASCII	ESC	SP	n
Hex	1B	20	n
Decimal	27	32	n

[Range] 0 ≤ n ≤ 255

[Description] Sets the character spacing for the right side of the character to [n x horizontal or vertical motion units].

[Notes]
- The right character spacing for double-width mode is twice the normal value. When the characters are enlarged, the right side character spacing is m (2 or 4) times the normal value.
- The horizontal and vertical motion units are specified by $1D $50. Changing the horizontal or vertical motion units does not affect the current right side spacing.
- The $1D $50 command can change the horizontal (and vertical) motion unit. However, the value cannot be less than the minimum horizontal movement amount.
- In standard mode, the horizontal motion unit is used.
- The maximum right side character spacing is 32 mm.

[Default] n = 0

[Reference] $1D $50

[Example]

$1B $21 n

[Name] **Select print modes**

[Format]

ASCII	ESC	!	n
Hex	1B	21	n
Decimal	27	33	n

[Range] 0 ≤ n ≤ 255

[Description] Selects print modes using n (see table below):

Bit	Off/On	Hex	Decimal	Function	11/15 cpi	15/20 cpi
0	Off	00	0	Character font A selected.	18 x 24	14 x 24
	On	01	1	Character font B selected.	14 x 24	10 x 24
1	-	-	-	Undefined.		
2	-	-	-	Undefined.		
3	Off	00	0	Expanded mode not selected.		
	On	08	8	Expanded mode selected.		
4	Off	00	0	Double-height mode not selected.		
	On	10	16	Double-height mode selected.		
5	Off	00	0	Double-width mode not selected.		
	On	20	32	Double-width mode selected.		
6	Off	00	0	Italic mode not selected.		
	On	40	64	Italic mode selected.		
7	Off	00	0	Underline mode not selected.		
	On	80	128	Underline mode selected		

[Notes]
- The printer can underline all characters, but cannot underline the spaces set by $09, $1B $24, $1B $5C and 90°/270° rotated characters.
- This command resets the left and right margin at default value (see $1D $4C, $1D $57).
- $1B $45 can also be used to turn the emphasized mode on/off. However, the last-received setting command is the effective one.
- $1B $2D can also be used to turn the underlining mode on/off. However, the last-received setting command is the effective one.
- $1D $21 can also be used to select character height/width. However, the last-received setting command is the effective one.

[Default] n = 0

[Reference] $1B $2D, $1B $45, $1D $21

[Example]

$1B $24 nL nH

[Name] **Set absolute print position**

[Format]

ASCII	ESC	$	nL	nH
Hex	1B	24	nL	nH
Decimal	27	36	nL	nH

[Range]
0 ≤ nL ≤ 255
0 ≤ nH ≤ 255

[Description]
Sets the distance from the beginning of the line to the position at which subsequent characters are to be printed.
The distance from the beginning of the line to the print position is [(nL + nH ´ 256) ´ (vertical or horizontal motion unit)] inches.

[Notes]
• Settings outside the specified printable area are ignored.
• The horizontal and vertical motion unit are specified by $1D $50.
• $1D $50 can change the horizontal (and vertical) motion unit. However, the value cannot be less than the minimum horizontal movement amount.
• In standard mode, the horizontal motion unit (x) is used.
• If the setting is outside the printing area width, it sets the absolute print position, but the left or right margin is set at default value.

[Default]

[Reference] $1B $5C, $1D $50

[Example]

$1B $25 n

[Name] **Select/cancel user-defined characters**

[Format]

ASCII	ESC	%	n
Hex	1B	25	n
Decimal	27	37	n

[Range]
0 ≤ n ≤ 255

[Description]
Selects or cancels the user-defined character set.
When the Least Significant Bit (LSB) of n is 0, the user-defined character set is canceled.
When the LSB of n is 1, the user-defined character set is selected.

[Notes]
• Only the LSB of n is applicable.
• When the user-defined character set is canceled, the internal character set is automatically selected.

[Default] n=0

[Reference] $1B $26, $1B $3F

[Example]

$1B $26 y c1 c2

[Name] **Defines user-defined characters**

[Format]

ASCII	ESC	&	y	c1	c2
Hex	1B	26	y	c1	c2
Decimal	27	37	y	c1	c2

[Range]
y = 3
32 ≤ c1 ≤ c2 ≤ 126
0 ≤ x ≤ 16 (Font (18 x 24))
0 ≤ x ≤ 13 (Font (14 x 24))
0 ≤ x ≤ 10 (Font 10 x 24)
0 ≤ d1 … d (y x xk) ≤ 255
k = c2 – c1 + 1

[Description]
Defines user-defined characters.
Y specifies the number of bytes in the vertical direction.
C1 specifies the beginning character code for the definition, and C2 specifies the final code.

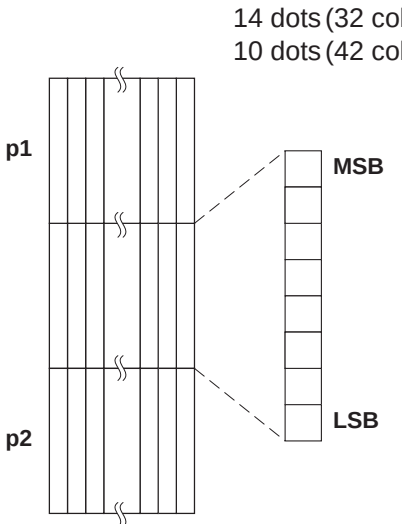
X specifies the number of dots in the horizontal direction.

[Notes]
- The allowable character code range is from ASCII $20 (32) to $7E (126) (95 characters).
- It is possible to define multiple characters for consecutive character codes. If only one character is desired, use c1 = c2.
- If c2 < c1, the command is not executed.
- d is the dot data for the characters. The dot pattern is in the horizontal direction starting from the left. Any remaining dots on the right remain blank.
- The data to define a user-defined character is (x x y) bytes.
- To print a dot, set the corresponding bit to 1; to not have it print, set to 0.
- This command can define different user-defined character patterns for each font. To select the font, use $1B $21.
- A user-defined character and a downloaded bit image cannot be defined simultaneously. When this command is executed, the downloaded bit image is cleared.
- The user-defined character definitions are cleared when $1B $40 or $1D $2A or $1B $3F are executed or the printer is reset or the power shut off.

[Default] Internal character set.

[Reference] $1B $25, $1B $3F

[Example]



$1B $28 $76 nL nH

[Name] **Set relative vertical print position**

[Format]

ASCII	ESC	(	v	nL	nH
Hex	1B	28	76	nL	nH
Decimal	27	10	118	nL	nH

[Range]
0 ≤ nL ≤ 255
0 ≤ nH ≤ 255

[Description] Sets the print vertical position based on the current position by using the horizontal or vertical motion unit.
- This command sets the distance from the current position to [(nL + nH x 256) x (horizontal or vertical motion unit)].

[Notes]
- When the starting position is specified by N motion unit to the bottom :
nL + nH x 256 = N
When the starting position is specified by N motion unit to the top (negative direction), use the complement of 65536 :
nL + nH x 256 = 65536 - N
- The horizzontal and vertical motion unit are specified by $1D $50.
- The $1D $50 command can change the horizontal (and vertical) motion unit. However, the value cannot be less than the minimum horizontal movement amount.
- In standard mode, the vertical motion unit is used.

[Default]

[Reference] $1D $50
[Example]

$1B $2A m nL nH d1...dk

[Name] **Select bit image mode**

[Format]

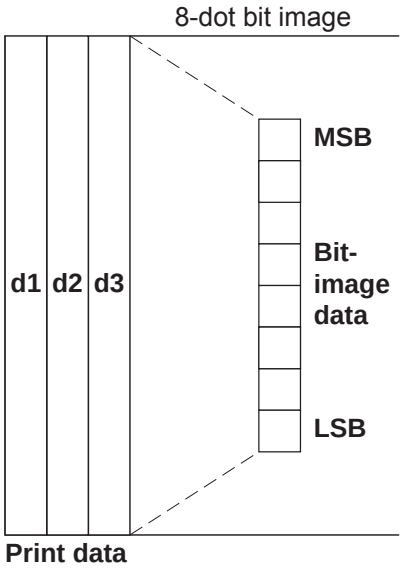
ASCII	ESC	*	m	nL	nH	d1...dk
Hex	1B	2A	m	nL	nH	d1...dk
Decimal	27	42	m	nL	nH	d1...dk

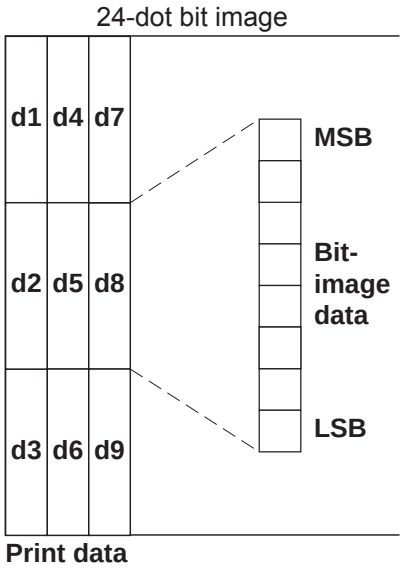
[Range] m = 0, 1, 32, 33
0 ≤ nL ≤ 255
0 ≤ nH ≤ 3
0 ≤ d ≤ 255

[Description] Selects a bit image mode using m for the number of dots specified by nL and nH, as follows:

m	Mode	Vertical direction		Horizontal direction	
		N. dots	DPI	DPI	N. of Data (k)
0	8 dot single density	8	67	100	nL + nH * 256
1	8 dot double density	8	67	200	nL + nH * 256
32	24 dot single density	24	200	100	(nL + nH * 256) * 3
33	24 dot double density	24	200	200	(nL + nH * 256) * 3

[Notes]
• The nL and nH commands indicate the number of dots of the bit image in the horizontal direction. The number of dots is calculated using: nL + nH * 256.
• If the bit image data input exceeds the number of dots to be printed on a line, the excess data is ignored.
• d indicates the bit image data. Set a corresponding bit to 1 to print a dot, or to 0 to not print the dot.
• If the value of m is outside the specified range, nL and data following it are processed as normal data.
• If the width of the printing area set by $1D $4C and $1D $57 is less than the width required by the data set using $1B $2A , the excess data are ignored.
• To print the bit image use $1B $4A or $1B $64.
• After printing a bit image, the printer returns to normal data processing mode.
• This command is not affected by the emphasized, double-strike, underline (etc.) print modes, except for the upside-down mode.
• The relationship between the image data and the dots to be printed is as follows:





[Default]
[Reference]
[Example]

$1B $2D n

[Name] **Turn underline mode on/off**

[Format]

ASCII	ESC	-	n
Hex	1B	2D	n
Decimal	27	45	n

[Range] 0 ≤ n ≤ 2, 48 ≤ n ≤ 50

[Description] Turns underline mode on or off, based on the following values of n:

n = 0, 48	Turns off underline mode
n = 1, 49	Turns on underline mode (1-dot thick)
n = 2, 50	Turns on underline mode (2-dot thick)

[Notes]
• The printer can underline all characters, but cannot underline the space and right-side character spacing.
• The printer cannot underline 90°/270° rotated characters and white/black inverted characters.
• When underline mode is turned off by setting the value of n to 0 or 48, the data which follows is not underlined.
• Underline mode can also be turned on or off by using $1B $21. Note, however, that the last received command is the effective one.

[Default] n=0
[Reference] $1B $21
[Example]

$1B $30

[Name] **Select 1/8-inch line spacing**

[Format]

ASCII	ESC	0
Hex	1B	30
Decimal	27	48

[Description] Selects 1/8-inch line spacing.
[Notes]
[Default]
[Reference] $1B $33
[Example]

$1B $32

[Name] **Select 1/6-inch line spacing**

[Format]

ASCII	ESC	2
Hex	1B	32
Decimal	27	50

[Description] Selects 1/6-inch line spacing.
[Notes]
[Default]
[Reference] $1B $33
[Example]

$1B $33 n

[Name] **Set line spacing**

[Format]

ASCII	ESC	3	n
Hex	1B	33	n
Decimal	27	51	n

[Range] 0 ≤ n ≤ 255

[Description] Sets line spacing to [n ´ (vertical or horizontal motion unit)] inches.

[Notes] • The horizontal and vertical motion unit are specified by $1D $50. Changing the horizontal or vertical motion unit does not affect the current line spacing.
• The $1D $50 command can change the horizontal (and vertical) motion unit. However, the value cannot be less than the minimum vertical movement amount.
• In standard mode, the vertical motion unit is used.
• The maximum spacing is 32,5 mm.

[Default] n = 64 (1/6 inch)

[Reference] $1B $32, $1D $50

[Example]

$1B $3D n

[Name] **Select peripheral device**

[Format]

ASCII	ESC	=	n
Hex	1B	3D	n
Decimal	27	61	n

[Range] 0 ≤ n ≤ 255

[Description] Select the device to which the host computer sends data, using n as follows:

n = 1, n = 3	Printer Enabled
n = 2	Printer Disabled
n = 5 or n = '5'	Select Pass-Through toward RFID module

[Notes] • When the printer is disabled, it ignores all transmitted data until the printer is enabled through this command.
• When the Pass-Through function is enabled, all transmitted data are sent on 2nd Serial.

[Default] n = 1

[Reference]

[Example]

$1B $3F n

[Name] **Cancel user-defined characters**

[Format]

ASCII	ESC	?	n
Hex	1B	3F	n
Decimal	27	63	n

[Range] 32 ≤ n ≤ 126

[Description] Cancels user-defined characters.

[Notes] • This command cancels the pattern defined for the character code specified by n. After the user-defined character is cancelled, the corresponding pattern for the internal character is printed.
• This command deletes the pattern defined for the specified character code in the font selected by $1B $21.
• If the user-defined character has not been defined for the specified character code, the printer ignores this command.

[Default]

[Reference] $1B $26, $1B $25

[Example]

$1B $40

[Name] **Initialize printer**

[Format]

ASCII	ESC	@
Hex	1B	40
Decimal	27	64

[Description] Clears the data in the print buffer and resets the printer mode to that in effect when power was turned on.

[Notes] • The data in the receiver buffer is not cleared.

• The macro definitions are not cleared.

[Default]
[Reference]
[Example]

$1B $44 [n1...nk] 00

[Name] **Set horizontal tab positions**

[Format]

ASCII	ESC	D	n1...nk	NUL
Hex	1B	44	n1...nk	00
Decimal	27	68	n1...nk	0

[Range] 1 ≤ n ≤ 255
0 ≤ k ≤ 32

[Description] Sets horizontal tab positions
• n specifies the column number for setting a horizontal tab position calculated from the beginning of the line.
• k indicates the total number of horizontal tab positions to be set.

[Notes]
• The horizontal tab position is stored as a value of [character width x n] measured from the beginning of the line. The character width includes the right-side character spacing and double-width characters are set with twice the width of normal characters.
• This command cancels previous tab settings.
• When setting n = 8, the print position is moved to column 9.
• Up to 32 tab positions (k = 32) can be set. Data exceeding 32 tab positions is processed as normal data.
• Send [n] k in ascending order and place a 0 NUL code at the end. When [n] k is less than or equal to the preceding value [n] k-1, the setting is complete and the data which follows is processed as normal data.
• $1B $44 00 cancels all horizontal tab positions.
• The previously specified horizontal tab position does not change, even if the character width is modified.

[Default] Default tab positions are set at intervals of 8 characters (columns 9, 17, 25, …) for Font A when the right-side character spacing is 0.

[Reference] $09
[Example]

$1B $45 n

[Name] **Turn emphasized mode on/off**

[Format]

ASCII	ESC	E	n
Hex	1B	45	n
Decimal	27	69	n

[Range] 0 ≤ n ≤ 255

[Description] Turns emphasized mode on/off.
• When the LSB of n is 0, the emphasized mode is off.
• When the LSB of n is 1, the emphasized mode is on.

[Notes]
• Only the LSB of n is effective.
• $1B $21 also turns on and off the emphasized mode. However, the last received command is the effective one.

[Default] n = 0
[Reference] $1B $21
[Example]

$1B $47 n

[Name] **Turn double-strike mode on/off**

[Format]

ASCII	ESC	G	n
Hex	1B	47	n
Decimal	27	71	n

[Range] 0 ≤ n ≤ 255
[Description] Turns double-strike mode on or off.
• When the LSB of n is 0, the double-strike mode is off.
• When the LSB of n is 1, the double-strike mode is on.
[Notes] • Only the LSB of n is effective.
• Printer output is the same in double-strike and emphasized mode.
[Default] n = 0
[Reference] $1B $45
[Example]

$1B $4A n

[Name] **Print and paper feed**
[Format]

ASCII	ESC	J	n
Hex	1B	4A	n
Decimal	27	74	n

[Range] 0 ≤ n ≤ 255
[Description] Prints the data in the print buffer and feeds the paper [n ´ (vertical or horizontal motion unit)] inches.
[Notes] • After printing has been completed, this command sets the print starting position to the beginning of the line.
• The paper feed amount set by this command does not affect the values set by $1B $32 or $1B $33.
• The horizontal and vertical motion units are specified by $1D $50.
• $1D $50 can change the vertical (and horizontal) motion unit. However, the value cannot be less than the minimum vertical movement amount.
• In standard mode, the vertical motion unit is used.
• The maximum paper feed amount is 520 mm.
[Default]
[Reference] $1D $50
[Example]

$1B $4D n

[Name] **Select character font**
[Format]

ASCII	ESC	M	n
Hex	1B	4D	n
Decimal	27	77	n

[Range] n = 0, 1, 48, 49
[Description] Selects characters font depending of cpi value set (Char/Inch) as follows :

Char/Inch.	n	Function
A=11 cpi B=15 cpi	0, 48	Font 11 cpi (18 x 24)
	1, 49	Font 15 cpi (14 x 24)
A=15 cpi B=20 cpi	0, 48	Font 15 cpi (14 x 24)
	1, 49	Font 20 cpi (10 x 24)

[Notes]
[Default]
[Reference] $1B $C1
[Example]

$1B $52 n

[Name] **Select an international character set**
[Format]

ASCII	ESC	R	n
Hex	1B	52	n

Decimal 27 82 n

[Range] 0 ≤ n ≤ 10

[Description] Selects the international character set n according to the table below:

	HEX	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
n	CHARACTER SET												
0	U.S.A.	#	$	@	[	\	]	^	`	{	|	}	~
1	France	#	$	à	°	ç	§	^	`	é	ù	è	“
2	Germany	#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	b
3	United Kingdom	£	$	@	[	\	]	^	`	{	|	}	~
4	Denmark I	#	$	@	Æ	Æ	Å	^	`	æ	f	å	~
5	Sweden	#	¤	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
6	Italy	#	$	@	°	\	é	^	ù	à	ò	è	ì
7	Spain I	Pt	$	@	i	Ñ	¿	^	`	“	ñ	}	~
8	Japan	#	$	@	[	¥	]	^	`	{	|	}	~
9	Norway	#	¤	É	Æ	Æ	Å	Ü	é	æ	f	å	ü
10	Denmark II	#	$	É	Æ	Æ	Å	Ü	é	æ	f	å	ü

[Note]

[Default] n = 0

[Reference]

[Example]

$1B $56 n

[Name] **Set 90° rotated print mode**

[Format]

ASCII	ESC	V	n
Hex	1B	56	n
Decimal	27	86	n

[Range] 0 ≤ n ≤ 1
48 ≤ n ≤ 49

[Description] Turns 90° rotation mode on/off. n is used as follows :

n	Function
0, 48	Turns off 90° rotation mode
0,49	Turns on 90° rotation mode

[Notes]
- When underlined mode is turned on, the printer does not underline 90° rotated characters. All the same it’s possible select the underline mode.
- Double-width and double-height commands in 90° rotation mode enlarge characters in the opposite directions from double-height and double-width commands in normal mode.
- This command is not available in Page mode.
- If this command is entered in Page mode, the printer all the same save the setting.

Default] n = 0

[Reference] $1B $21, $1B $2D

[Example]

$1B $5C nL nH

[Name] **Set relative print position**

[Format]

ASCII	ESC	\	nL	nH
Hex	1B	5C	nL	nH
Decimal	27	92	nL	nH

[Range] 0 ≤ nL ≤ 255
0 ≤ nH ≤ 255

[Description] Sets the print starting position based on the current position by using the horizontal or vertical motion unit.
Sets the distance from the current position to [(nL+ nH * 256) * (horizontal or vertical motion unit)].

[Notes]
• It's possible to print further on the right margin set for every font. In this case the printing continues up to the maximum border of the printer mechanism and then begins a new row.
• Any setting that exceeds the printable area is ignored.
• When the starting position is specified by n motion units to the right:
nL + nH * 256 = N
When the starting position is specified by n motion units to the left (negative direction), use the complement of 65536: nL + nH * 256 = 65536 – N
• If setting exceeds the printing area width, the left or right margin is set to the default value.
• The horizontal and vertical motion unit are specified by $1D $50.
• $1D $50 can change the horizontal (and vertical) motion units. However, the value cannot be less than the minimum horizontal movement amount.
• In standard mode, the horizontal motion unit is used.
• Setting the right value, it's possible to print characters over the right edge.

[Default]

[Reference] $1B $24, $1D $50

[Example]

$1B $61 n

[Name] **Select justification**

[Format]

ASCII	ESC	a	n
Hex	1B	61	n
Decimal	27	97	n

[Range] 0 ≤ n ≤ 2, 48 ≤ n ≤ 50

[Description] Aligns all data in one line to the specified position; n selects the type of justification as follows:

n	Justification
0, 48	Flush left
1, 49	Centered
2, 50	Flush right

[Notes]
• This command is only enabled when inserted at the beginning of a line.
• Lines are justified within the specified printing area.
• Spaces set by $09, $1B $24 and $1B $5C will be justified according to the previously-entered mode.

[Default] n = 0

[Reference]

[Example]

Flush left

```
ABC
ABCD
ABCDE
```

Centered

```
 ABC
ABCD
ABCDE
```

Flush right

```
  ABC
 ABCD
ABCDE
```

$1B $64 n

[Name] **Print and feed paper n rows**

[Format]

ASCII	ESC	d	n
Hex	1B	64	n
Decimal	27	100	n

[Range] 0 ≤ n ≤ 255

[Description] Prints the data in the print buffer and feeds the paper n rows.

[Notes]
- n rows paper feed is equivalent to (n x char height + line spacing set).
- Sets the print starting position at the beginning of the line.
- This command does not affect the line spacing set by $1B $32 or $1B $33.
- The maximum paper feed amount is 254 rows. Even if a paper feed amount of more than 254 rows is set, the printer feeds the paper only 254 rows.

[Default]

[Reference] $1B $32, $1B $33

[Example]

$1B $69

[Name] **Total cut**

[Format]

ASCII	ESC	i
Hex	1B	69
Decimal	27	105

[Description] This command enables cutter operation. If there is no cutter, a disabling flag is set and any subsequent cut commands will be ignored.

[Notes]
- The printer waits to complete all paper movement commands before it executes a total cut.

[Default]

[Reference]

[Example]

$1B $74 n

[Name] **Select character code table**

[Format]

ASCII	ESC	t	n
Hex	1B	74	n
Decimal	27	116	n

[Range] n = 0, 2, 3, 4, 5, 19, 255

[Description] Selects a page n from the character code table, as follows:

n	Page
0	0 (PC437 [U.S.A., Standard Europe])
2	2 (PC850 [Multilingual])
3	3 (PC860 [Portuguesel])
4	4 (PC863 [Canadian-French])
5	5 (PC865 [Nordic])
19	19 (PC858 for Euro symbol at position 213)
255	Space page

[Notes]

[Default] n = 0

[Reference] See character code tables

[Example] For printing Euro symbol (€), the command sequence is: 1B, 74, 13, D5

$1B $76

[Name] **Transmit paper sensor status**

[Format]

ASCII	ESC	v
Hex	1B	76
Decimal	27	118

[Description] When this command is received, transmit the current status of the paper sensor.

Bit	Off/On	Hex	Decimal	Function
0,1	Off	00	0	Near paper-end sensor: Paper present
	On	03	3	Near paper-end sensor: Paper not present
2,3	Off	00	0	Paper-end sensor: Paper present
	On	(0C)	(12)	Paper-end sensor:Paper not present
4	Off	00	0	Not used. Fixed to Off.
5	-	-	-	Undefined
6	-	-	-	Undefined
7	Off	00	0	Not used. Fixed to Off.

[Notes] • This command is executed immediately, even when the data buffer is full (Busy).
The status to be transmitted is shown in the table below:

[Default]

[Reference] **$10 $04**

[Example]

$1B $7B n

[Name] **Turn upside-down printing mode on/off**

[Format]

ASCII	ESC	{.}	n
Hex	1B	7B	n
Decimal	27	123	n

[Range] 0 ≤ n ≤ 255

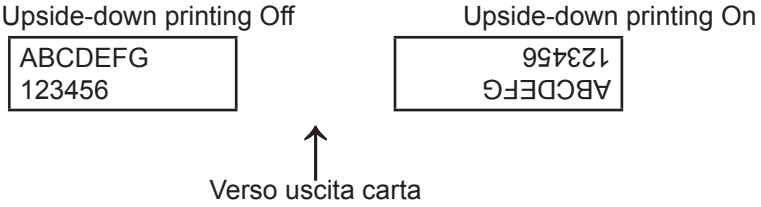
[Description] Turns upside-down printing mode on or off.
• When the LSB of n is 0, the upside-down printing mode is off.
• When the LSB of n is 1, the upside-down printing mode is on.

[Notes] • Only the LSB of n is effective.
• This command is valid only if entered at the beginning of a line.
• In upside-down printing mode, the printer rotates the line to be printed 180° and then prints it.

[Default] n = 0

[Reference]

[Example]



$1B $C1 n

[Name] **Set/cancel cpi mode**

[Format]

ASCII	ESC	{}	n
Hex	1B	C1	n
Decimal	27	193	n

[Range] 0 ≤ n ≤ 1, 48 ≤ n ≤ 49

[Description] Sets cpi mode based on the following values of n:

n	Function
0,48	Font A= 11 cpi
	Font B= 15 cpi
1,49	Font A= 15 cpi
	Font B= 20 cpi

[Notes]
[Default] n = 0
[Reference] $1B $21
[Example]

$1C $3C n

[Name] Change printer emulation to SVELTA.
[Format]

ASCII	FS	<	S	V	E	L	>
Hex	1C	3C	53	56	45	4C	3E
Decimal	28	60	83	86	69	76	62

[Description] Change the printer emulation to SVELTA emulation.
[Note]
[Default]
[Reference]
[Example]

$1C $80

[Name] **Read date/time of the real time clock.**
[Format]

ASCII	FS	{ }	m
Hex	1C	80	m
Decimal	28	128	m

[Range] 0 ≤ m ≤ 3
[Description] Read date/time of the real time clock in the format specified by m values as follows :

m	FORMAT
0	DD/MM/YY hh:mm:ss
1	DDMMYYhhmmss
2	YYMMDDhhmmss
3	YYMMDDhhmmssd

where :
DD = represents the dayof the date
MM = represents the month of the date
YY = represents year of the date
hh = represents the hour of the time
mm = represents the minutes of the time
ss = represents the seconds of the time
d = indicates the day of the week

[Note] • Before send the date/time, if the m parameter is valid the printer transmits the ACK ($06), otherwise return NACK ($015).
[Default]
[Reference]
[Example] To read date/time in the “DDMMYYhhmmss” format, transmit :
Host

Hex	$1C	$80	$01
ASCII	FS	{ }	m

For example if the current date/time are “15 September 2006 at 10:56:20 (AM)” the printer’s answer is as follows :

Hex	$06	$31	$35	$30	$39	$30	$36	$31	$30	$35	$36	$32	$30
ASCII	ACK	1	5	0	9	0	6	1	0	5	6	2	0

$1C $81 m n d0...dn

[Name] **Set date/time of the real time clock.**

[Format]

ASCII	FS	{ }	m	n	d0....dn
Hex	1C	81	m	n	d0....dn
Decimal	28	129	m	n	d0....dn

[Range] 0 ≤ m ≤ 3
0 ≤ d0, dn ≤ 255

[Description] Set the date/time of the real time clock in the format specified by m values as follows:

m	FORMAT
0	DD/MM/YY hh:mm:ss
1	DDMMYYhhmmss
2	YYMMDDhhmmss
3	YYMMDDhhmmssd

where :
DD = represents the dayof the date
MM = represents the month of the date
YY = represents year of the date
hh = represents the hour of the time
mm = represents the minutes of the time
ss = represents the seconds of the time
d = indicates the day of the week
• n specifies the number of charecters to send.
• d0..dn are the ASCII characters relative to the date and time to set .

[Note]
• if the transmission has been received correctly and the command is valid, the printer returns the ACK ($06), otherwise returns NACK ($015).
• the day of the week is calculated automatically from the printer and then it’s possible that the returned value is different from the one transmitted.

[Default]
[Reference]

[Example] For example to set the date and time to “29 September 2006 at 13:51:00 (PM)” in the “YYMMDDhhmmss” format transmit:

Host

Hex	$1C	$81	$02	$0C	$30	$36	$30	$39	$32	$39	$31	$33	$35	$31	$30	$30
ASCII	FS	{ }	STX	FF	0	6	0	9	2	9	1	3	5	1	0	0

The printer’s answer ACK ($06) if the transmission is OK otherwise NACK($15).

$1C $82

[Name] **Print date**

[Format]

ASCII	FS	{ }
Hex	1C	82
Decimal	28	130

[Description] Prints date in the format specified by the command $1C$84 with the parameter n = ‘D’.
[Note]

[Default] “dd/mm/yy”
[Reference] $1C $83, $1C $84
[Example]

$1C $83

[Name] **Print time**
[Format]

ASCII	FS	{ }
Hex	1C	83
Decimal	28	131

[Description] Prints time with the format specified by the command $1C$84 with the parameter n = ‘T’.
[Note]
[Default] “hh:nn:ss”
[Reference] $1C $82, $1C $84
[Example]

$1C $84 n d1...dk 00

[Name] **Set user defined date/Time formats**
[Format]

ASCII	FS	{ }	n	d1...dk NUL
Hex	1C	84	n	d1...dk $00
Decimal	28	132	n	d1...dk 0

[Range] n = ‘D’, n = ‘T’
0 ≤ d0, dK ≤ 255
[Description] Sets the format string for date and time used to printing ($1C $83, $1C $84).
• n specifies wich user-defined string format is set D for date and T for time
• d0..dk are the ASCII characters relative to user-defined date/time formats.
• the maximum length of fthe user-defined date/time format string is 64 chars.
The following table shows characters used to create user-defined date/time formats :

Character	Description
l	Selects Italian language
E	Selects English language (is the default language)
c	Selects default date/time
d	Displays the day as a number without a leading zero (1-31).
dd	Displays the day as a number with a leading zero (01-31).
ddd	Displays the day as an abbreviation (for example, Sun).
dddd	Displays the day as a full name (for example, Sunday).
ddddd	Displays the date as a complete date in the short format where date values are formatted with day, month and year (the short date format is dd/mm/yy).
dddddd	Displays the date as a complete date in the extended format where date values are formatted with day, month and year (the extended date format is dd mmmm, yyyy).
m	Displays the month as a number without a leading zero (1-12). If the character m is immediately after the character h or hh , displays the minutes instead of month (see also the n character formatting).
mm	Displays the month as a number with leading zeros (01-12). If the character m is immediately after the character h or hh , displays the minutes instead of month (see also the nn character formatting).
mmm	Displays the month as an abbreviation (for example, Jan).
mmmm	Displays the month as a full month name (for example, January).
yy	Displays the year in two-digit numeric format with a leading zero.
yyyy	Displays the year in four digit numeric format.

Character	Description
h	Displays the hour as a number without leading zeros (0-23)
hh	Displays the hour as a number with leading zeros (00-23)
n	Displays the minutes as a number without leading zeros (0-59)
nn	Displays the minutes as a number with leading zeros (00-59)
s	Displays the seconds as a number without leading zeros (0-59)
ss	Displays the seconds as a number with leading zeros (00-59)
ttttt	Displays the time in the extended format where time values are formatted with hour, minutes and seconds (the extended time format is h:mm:ss).
AM/PM	Using the 12-hour clock and displays the AM prefix in uppercase next to the hours that preceding midday and the PM prefix in uppercase next to the hours between midday and midnight.
am/pm	Using the 12-hour clock and displays the am prefix in lowercase next to the hours that preceding midday and the pm prefix in lowercase next to the hours between midday and midnight.
A/P	Using the 12-hour clock and displays the A prefix in uppercase next to the hours that preceding midday and the a prefix in uppercase next to the hours between midday and midnight.
a/p	Using the 12-hour clock and displays the a prefix in lowercase next to the hours that preceding midday and the a prefix in lowercase next to the hours between midday and midnight.

[Note]
[Default]
[Reference]
[Example] For example to print the current time with the string format 'yy/mm/dd hh:mm:ss' follow these steps :
1. Send the following command to define the user-defined Time string format:

Hex	$1C	$84	$54	$79	$79	$2F	$6D	$6D	$2F	$64	$64	$20	$68	$68	$3A	$6E	$6E	$3A	$73	$73	$00
ASCII	FS	{ }	T	y	y	/	m	m	/	d	d		h	h	:	n	n	:	s	s	NUL

The printer's answer ACK ($06) if the transmission is OK otherwise NACK($15).
2. Send the following command to print the time :

Hex	$1C	$83	$0A
ASCII	FS	{ }	LF

Note : The character $0A feeds one line based on the current line spacing.
If the date and time is 22 October 2006 at 17:35:27 (PM) the output string printed will be: 06/10/22 17:35:27

$1C $90

[Name] **Get number of stored logo**
[Format] ASCII FS { }
Hex 1C 90
Decimal 28 144
[Description] This command sends to the printer the request of number of stored logo; the printer returns a bytes sequence as follows :
<PNn>
where n (in ASCII format) indicates the number of stored images.
[Note]
[Default]
[Reference]
[Example] If in the flash memory are stored 10 logos send this command :

Hex	$1C	$90
ASCII	FS	{ }

The printer's answer will be :

Hex	$3C	$50	$4E	$31	$30	$3E
ASCII	<	P	N	1	0	>

$1C $91

[Name] **Get pictures header list**

[Format]

ASCII	FS	{ }
Hex	1C	91
Decimal	28	145

[Description] This command requests to the printer the list of stored logo. The printer returns a bytes sequence as follows :
<PL ***CrLf*** [***N-ID CrLf***]>
where
• ***CrLf*** indicates the two characters $0D (Carriage return) and $0A (Line Feed);
• ***N*** is the number of stored logo;
• ***[ID]*** indicates the logo identifier, a sequence of 16 bytes that was defined when the logo is stored.This field is optional because it's returned only if the logo has been found.

[Note]
[Default]
[Reference] $1C $92, $1C $94
[Example]

$1C $92 nH nL

[Name] **Get pictures header info**

[Format]

ASCII	FS	{ }	nH	nL
Hex	1C	92	nH	nL
Decimal	28	146	nH	nL

[Range] 0 ≤ nH, nL ≤ 255
[Description] Gets the logo header info stored specified by n (express in ASCII).
• n is the number of stored logo;
The printer returns a byte sequence as follows :
<Pl***e***[***ID***]>
where
• ***e*** indicates the search result
e = 0 picture not found
e = 1 picture found
• ***[ID]*** indicates the logo identifier, a sequence of 16 bytes that was defined when the logo is stored. This fiedl is optional because it's returned only if the logo has been found.

[Note]
[Default]
[Reference]
[Example]

$1C $93 nH nL

[Name] **Print logo**

[Format]

ASCII	FS	{ }	nH	nL	opt	sp	posH	posL
Hex	1C	93	nH	nL	opt	sp	posH	posL
Decimal	28	147	nH	nL	opt	sp	posH	posL

[Range] 0 ≤ nH, nL ≤ 255
[Description] Prints logo defined by n.
• n is the number of image to print;
• opt is the option byte that specifies justification and rotation as shown in the following table:

Bit	Description	BIN	Function
0,1	Justification	00	Left
		01	Center
		10	Right
		11	User Define (on the basis of position specified by posH and posW)
2, 3	N.U	00	Not used.
4, 6	N.U.	00	Not used.
7	Print Rotate	0	Print normal.
		1	Print rotate.

• sp specifies the thickness of the image border.
• posH, posL specifies the logo's horizontal position (from the left border); used only with user-defined justification.

[Note]
[Default]
[Reference]
[Example]

Example 1: To print logo no.10 centered and rotated transmits :

```
$1C $93 $00 $0A $81 $01 $00 $00
where
$1C $93          //print logo command
$00 $0A          //Logo no. 10
$81                    //printing rotated and centered
$01                    //1 pixel of image border
$00 $00          //Positioning not used
```

Example 2: To print logo no.10 not rotated and with a user-defined printing position transmits :

```
$1C $93 $00 $0A $03 $01 $00 $50
where
$1C $93          //print logo command
$00 $0A          //Logo no. 10
$03                    //printing with a user define positioning and not rotated
$01                    //1 pixel of image border
$00 $50          //Printing 10 mm from the left border
```

$1C $94

[Name] **Save the image received from serial port into the flash**

[Format]

ASCII	FS	{ }	nH	nL	xDimH	xDimL	yDimH	yDimL	TbdH	TbdL	Id0..Idn	d0..dn	>
Hex	1C	94	nH	nL	xDimH	xDimL	yDimH	yDimL	TbdH	TbdL	Id0..Idn	d0..dn	3E
Decimal	28	148	nH	nL	xDimH	xDimL	yDimH	yDimL	TbdH	TbdL	Id0..Idn	d0..dn	62

[Range]
0 ≤ nH, nL ≤ 255,
0 ≤ xDimH, xDimL ≤ 255,
0 ≤ yDimH, yDimL ≤ 255,
0 ≤ d0, dn ≤ 255

[Description] Saves the image received from serial port into the printer flash; if the number used to store logo is not already present inside the printer, the new logo is appended to stored logos. Otherwise the new logo is updated.
• ***nH*** and ***nL*** indicates the number of logo (2 bytes expressed in hexadecimal notation).
• ***xDimH*** and ***xDimL*** indicate the logo horizontal dimension in pixel (2 bytes expressed in hexadecimal notation); the value must be multiple of 16.
• ***yDimH*** and ***yDimL*** indicates the logo vertical dimension in pixel (2 bytes expressed in hexadecimal notation).
• ***TbdH*** and ***TbdL*** 2 bytes fixed to $00 (RESERVED)
• ***Id0..Idn*** indicates the logo Id, a sequence of 16 bytes to identify univocally the logo.

• ***d0 ...dn*** are the image data. The size of image is defined as follows :
xSize = xDim /16; number of WORD (16 bit) in a horizontal image line
Total Size = (xSize * yDim) *2;
• ***'>'*** is the character terminator (in ASCII) of this command.
The printer returns a sequence of bytes as follows :
<PC0> if the saving include an incorrect syntax or the memory in flash available for logos is finished (128Kbyte);
<PC1n> if the syntax command is correct and there's memory enough in flash for saving logos; n returns the status of the flash programming :
$88 sector not erased
$77 error during programming
$AA Programming done.

[Note]
[Default]
[Reference]
[Example] The following example shows the bytes sequence received from serial port to store a logo into the printer flash :

```
Offset          Hexadecimal                                     ASCII

00000000:  1C 94 00-08 01 C0 02-49 00 00 50-69 63 2D 32 36   ° ° ° '+I ^ Pic-26
00000010:  20 20 32-32 2F 30 39-2F 30 34 00-00 00 00 00 00   22/09/04
00000020:  00 00 00-00 00 00 00-00 00 00 00-00 00 00 00 00
  ....
  ....                                                       Image data
  ....

00008000:  00 00 00 00-00 00 00 00-00 00 00 00-00 00 00 00
00008010:  00 00 3E
>
```

If the programming is successful, the printer's answer will be :

Hex	$3C	$50	$43	$31	$AA	$3E
ASCII	<	P	C	1	{ }	>

$1C $B0 n

[Name] **Sets the barcode reader status.**
[Format]

ASCII	FS	{ }	n
Hex	1C	B0	n
Decimal	28	176	n

[Range] $30 ≤ n ≤ $36
[Description] This command sets the operating status of the barcode reader; n identifies the status of the barcode setting as follows :

$30 TRIGGER ON/OFF: Every trigger the barcode reader toggle the previous status. After a correct reading the barcode reader automatically turn off.
$31 GOOD READ OFF: Every trigger the barcode reader is turn ON and switch off after a timeout (standard) or after a correct reading.
$32 CONTINUOUS TRIGGER OFF: Every trigger the barcode reader toggle the previous status.
$33 CONTINUOUS / AUTO POWER ON: The barcode reader remains power on.
$34 FLASH: Every trigger the barcode reader switches between a power off condition and continuous flashing. During the reading phase the flashing condition is changed in a continuous lighting, and then return to flashing condition after a timeout or after a reading operation.

$35 TESTING: Every trigger the barcode reader is turn ON and switch off or whereas. If the barcode reader recognize a correct barcode the reading operation is not single, like the trigger on/off state, but is made permanent until the barcode is removed.
$36 FLASH/AUTO POWER ON: The barcode reader remains in a continuous flashing condition, when occurs a reading the barcode reader is turned ON. This condition still stays for a standard timeout, then the barcode reader returns in a flashing condition.

[Note]
• After the barcode reader executes the command, a beep signal is emitted.
• The execution of this command clears the ouput buffer of barcode reader; if a scansion is executed without data acquisition by the host, all data read are deleted.

The printer returns a byte :

ACK ($06)	The command is executed successfully.
NACK($15)	The command is not executed successfully.
$FF	The n parameter send is not valid
$FE	The barcode reader is not working or it not installed on the printer.

[Default]
[Reference]
[Example]

$1C $B1 n

[Name] **Get barcode reader status.**

[Format]

ASCII	FS	{ }	n
Hex	1C	B1	n
Decimal	28	177	n

[Range] 30 ≤ n ≤ $34

[Description] Reads the barcode reader parameters in base of n value :

n = $30 STATUS:

Reads the barcode reader status. It returns :
• NACK ($15) character if the command is not successful
• $FE character if the barcode reader is not working or it not installed on the printer.
• ACK ($06) character, followed by a status byte; the status to be transmitted is shown in the table below:

Bit	Value	Function
0, 1, 2	$00	TRIGGER ON/OFF
	$01	GOOD READ OFF
	$02	CONTINUOUS TRIGGER OFF
	$03	CONTINUOUS / AUTO POWER ON
	$04	FLASH
	$05	TESTING
	$06	FLASH / AUTO POWER ON
	$07	RESERVED
3	0	PE Off
	1	PE On
4	0	TG Off
	1	TG On
5	0	Decode OK
	1	Decode ERROR
6, 7	-	RESERVED

The execution of this command clears the ouput buffer of barcode reader; if a scansion is executed without data acquisition by the host, all data read are deleted.

n = $31 BYTES ON RECEPTION BUFFER:

Indicates the number of bytes sent from barcode reader. It returns :
• NACK ($15) character if the command is not successful or the buffer is empty

• $FE character if the barcode reader is not working or it not installed on the printer.
• ACK ($06) character, followed by one byte that Indicates the number of bytes send from barcode reader.

n = $32 BYTES READING ON OUTPUT FROM BARCODE READER
Indicates the number of bytes sent from barcode reader. It returns :
• NACK ($15) character if the command is not successful or the buffer is empty
• $FE character if the barcode reader is not working or it not installed on the printer.
• ACK ($06) character, followed by a bytes sequence B1, B2, ...Bn where n are the bytes on output from barcode reader.

n = $33 DELETE BYTES ON OUTPUT
This command deletes all bytes on the output buffer from the barcode reader. It returns
• NACK ($15) character if the command is not successful.
• $FE character if the barcode reader is not working or it not installed on the printer.
• ACK ($06) character if the command is successful.

n = $34 READING OF ONE BYTE ON OUTPUT FROM BARCODE READER
This command reads one byte on output from barcode reader. It returns :
• NACK ($15) character if there are no bytes on output from barcode reader.
• $FE character if the barcode reader is not working or it not installed on the printer.
• ACK ($06) character, followed by one byte that is the first byte present on the output FIFO from barcode reader.

[Note] • with n = $30 after the barcode reader executes this command, emits a beep as acoustic signalling.

[Default]
[Reference] $FS $B0
[Example]

$1C $B2

[Name] **Barcode reader Trigger.**

[Format]

ASCII	FS	{ }	
Hex	1C	B2	
Decimal	28	178	

[Description] This command execution forces a trigger of barcode reader. It returns:
• NACK ($15) character if the command is successful.
• $FE character if the barcode reader is not working or it not installed on the printer.
• ACK ($06) character, if the command is successful.

[Note] • A trigger event may be effect on barcode reader setting, depending on the barcode reader status.
• The execution of this command clears the ouput buffer of barcode reader; if a scansion is executed without data acquisition by the host, all data read are deleted.

[Default]
[Reference] $FS $B0
[Example]

$1D $21 n

[Name] Select character size

[Format]

ASCII	GS	!	n
Hex	1D	21	n
Decimal	29	33	n

[Intervallo] 0 ≤ n ≤ 255

[Description] Selects character height and width, as follows:
• Bits 0 to 3: to select character height (see table 2).
• Bits 4 to 7: to select character width (see table 1).

Table 1 Select Character Width				Table 2 Select character height		
Hex	Decimal	Width		Hex	Decimal	Height
00	0	1 (normal)		00	0	1 (normal)
10	16	2 (width = 2x)		01	1	2 (height = 2x)
20	32	3 (width = 3x)		02	2	3 (height = 3x)
30	48	4 (width = 4x)		03	3	4 (height = 4x)
40	64	5 (width = 5x)		04	4	5 (height = 5x)
50	80	6 (width = 6x)		05	5	6 (height = 6x)
60	96	7 (width = 7x)		06	6	7 (height = 7x)
70	112	8 (width = 8x)		07	7	8 (height = 8x)

[Notes] • This command is effective for all characters (except HRI characters).
• If n falls outside the defined range, this command is ignored.
• Characters enlarged to different heights on the same line are aligned at the baseline or topline.
• $1B $21 can also be used to select character size. However, the setting of the last received command is the effective one.

[Default] n = 0

[Reference] $1B $21

[Example]

$1D $2A x y d1...d (x x y x 8)

[Name] **Define dowloaded bit image**

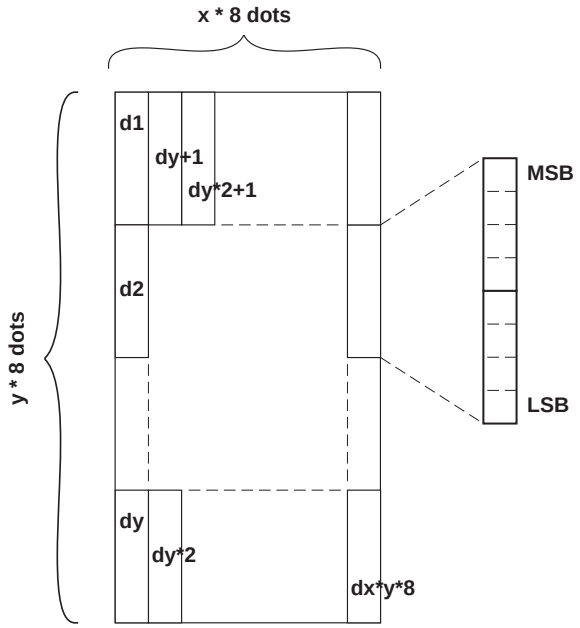
[Format]

ASCII	GS	*	x	y	d1...d(x x y x 8)
Hex	1D	2A	x	y	d1...d(x x y x 8)
Decimal	29	42	x	y	d1...d(x x y x 8)

[Range] 1 ≤ x ≤ 255
1 ≤ y ≤ 48
x * y ≤ 1536
0 ≤ d ≤ 255

[Description] Defines a downloaded bit image using the number of dots specified by x and y.
• x specifies the number of dots in the horizontal direction.
• y specifies the number of dots in the vertical direction.

[Notes] • The number of dots in the horizontal direction is x * 8, in the vertical direction it is y *8.
• If x x y is out of the specified range, this command is disabled.
• The d indicates bit-image data. Data (d) specifies a bit printed to 1 and not printed to 0.
• The downloaded bit image definition is cleared when:
1) $1B $40 is executed.
2) $1B $26 is executed.
Printer is reset or the power is turned off.
• The following figure shows the relationship between the downloaded bit image and the printed data.



[Reference] $1D $5C
[Example]

$1D $2F m

[Name] **Print dowloaded bit image**

[Format]

ASCII	GS	/	m
Hex	1D	2F	m
Decimal	29	47	m

[Description] Prints a downloaded bit image using the mode specified by m. m selects a mode from the table below :

m	Mode
0,48	Normal
1, 49	Double-width
2, 50	Double-height
3, 51	Quadruple

[Notes]
• This command is ignored if a downloaded bit image has not been defined.
• In standard mode, this command is effective only when there is no data in the print buffer.
• This command has no effect in the print modes (emphasized, underline, character size, or white/black reverse printing), except for upside-down printing mode.
• If the downloaded bit-image to be printed exceeds the printable area, the excess data is not printed.
• If the printing area width set by $1D $4C and $1D $57 is less than the bit image horizontal size, the following processing is performed:

1) The printing area width is extended toward the right side up to hold the bit image. In this case, printing does not exceed the printable area.
2) If the printing area width cannot be extended toward the right side, because there's no more printing area, the left margin is reduced to accommodate the bit image.

[Reference] $1D $2A
[Example]

$1D $3A

[Name] **Start/end macro definition**

[Format]

ASCII	GS	:
Hex	1D	3A
Decimal	29	58

[Description] Starts or ends macro definition.

[Notes]
- Macro definition starts when this command is received during normal operation.
- When $1D $5E is received during macro definition, the printer ends macro definition and clears all definitions.
- Macros are not defined when power is turned on to the machine.
- Macro content is not cancelled by the $1B $40 command. Therefore, $1B $40 may be included in the content of macro definitions.
- If the printer receives $1D $3A a second time after previously receiving $1D $3A, the printer remains in macro undefined status.
- The contents of the macro can be defined up to 1024 bytes. If the macro definition exceeds 1024 bytes, excess data is not stored.

[Default]

[Reference] $1D $5E

[Example]

[Name] **Turn white/black reverse printing mode on/off**

[Format]

ASCII	GS	B	n
Hex	1D	42	n
Decimal	29	66	n

[Range] 0 ≤ n ≤ 255

[Description] Turns white/black reverse printing mode on or off.
- When the LSB of n is 0, white/black reverse printing is turned off.
- When the LSB of n is 1, white/black reverse printing is turned on.

[Notes]
- Only the LSB di n is effective.
- This command is available for both built-in and user-defined characters.
- This command does not affect bit image, downloaded bit image, bar code, HRI characters and spacing skipped by $09, $1B $24 and $1B $5C.
- This command does not affect white space between lines.
- White/black reverse mode has a higher priority than underline mode. Even if underline mode is on, it will be disabled (but not cancelled) when white/black reverse mode is selected.

[Default] n = 0

[Reference]

[Example]

$1D $48 n

[Name] **Select printing position of Human Readable Interpretation (HRI) characters**

[Format]

ASCII	GS	H	n
Hex	1D	48	n
Decimal	29	72	n

[Range] 0 ≤ n ≤ 3, 48 ≤ n ≤ 51

[Description] Selects the printing position of HRI characters when printing bar codes; n selects the printing positions as follows:

n	Function
0, 48	Not printed
1, 49	Above the bar code
2, 50	Below the bar code
3, 51	Both above the below the bar code

[Notes] • HRI characters are printed using the font specified by $1D $66.
[Default] n = 0
[Reference] $1D $66, $1D $6B
[Example]

$1D $49 n

[Name] **Transmit printer ID**
[Format]

ASCII	GS	I	n
Hex	1D	49	n
Decimal	29	73	n

[Range] 1 ≤ n ≤ 3, 49 ≤ n ≤ 51
[Description] Transmits the printer ID specified by n follows:

n	Printer ID	Specification
1, 49	Printer model ID	$75
2, 50	Type ID	See table below
3, 51	ROM version ID	Depends on ROM version (4 character)

n = 2, 50 Type ID

Bit	Off/On	Hex	Decimal	Function
0	Off	00	0	2-byte character codes not supported
1	Off	00	0	Autocutter not supplied
	On	02	2	Autocutter supplied
2	Off	00	0	Thermal paper w/o label
	On	04	4	Thermal paper w/label
3	-	-	-	Undefined
4	Off	00	0	Not used. Fixed to Off.
5	-	-	-	Undefined
6	-	-	-	Undefined
7	Off	00	0	Not used. Fixed to Off.

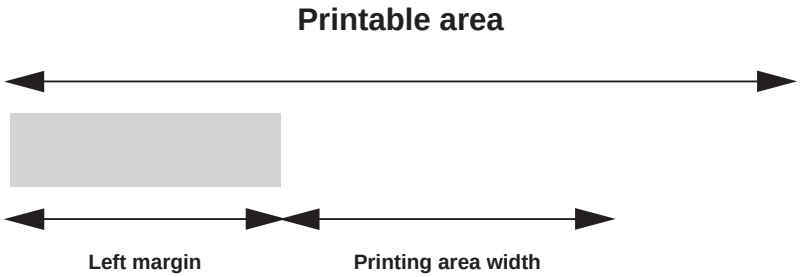
[Notes] • The printer only transmits 1 byte (printer ID) without confirmation that the host is ready to receive data.
• This command is executed when the data is processed in the data buffer. Therefore, there could be a time lag between command reception and data transmission, depending on data buffer status.
[Default]
[Reference]
[Example]

$1D $4C nL nH

[Name] **Set left margin**
[Format]

ASCII	GS	L	nL	nH
Hex	1D	4C	nL	nH
Decimal	29	76	nL	nH

[Range] 0 ≤ nL, nH ≤ 255
[Description] Sets the left margin.
• The left margin is set to [(nL + nH * 256) * (horizontal motion unit)] inches.



[Notes]
- This command is enabled only if set at the beginning of the line.
- If the setting exceeds the printable area, the maximum value of the printable area is used.
- If the left margin + printing area width is greater than the printable area, the printing area width is set at maximum value.
- The horizontal and vertical motion unit are specified by $1D $50. Changing the horizontal or vertical motion unit does not affect the current left margin.
- The $1D $50 command can change the horizontal (and vertical) motion unit.
- However, the value cannot be less than the minimum horizontal movement amount and it must be in even units of the minimum horizontal movement amount.

[Default]
[Reference] $1D $50, $1D $57
[Example]

$1D $50 x y

[Name] **Set horizontal and vertical motion units**

[Format]

ASCII	GS	P	x	y
Hex	1D	50	x	y
Decimal	29	80	x	y

[Range] 0 ≤ x, y ≤ 255

[Description] Sets the horizontal and vertical motion units to 1/x inch and 1/y inch respectively.
When x is set to 0, the default setting value is used.
When y is set to 0, the default setting value is used.

[Notes]
- The horizontal direction is perpendicular to the paper feed direction.
- In standard mode, the following commands use x or y, regardless of character rotation (upside-down or 90° clockwise rotation):

① Commands using x : $1B $20, $1B $24, $1B $5C, $1D $4C, $1D $57.
② Commands using y : $1B $33, $1B $4A.

,
- This command does not affect the previously specified values.
- The calculated result from combining this command with others is truncated to the minimum value of the mechanical pitch or an exact multiple of that value.

[Default] x = 204, y = 408
[Reference] $1B $20, $1B $24, $1B $5C, $1B $33, $1B $4A, $1D $4C, $1D $57
[Example]

① $1D $56 m, ② $1D $56 m n

[Name] Select cut mode

[Format]

①	ASCII	GS	V	m		
	Hex		1D	56	m	
	Decimal		29	86	m	
②	ASCII	GS	V	m	n	
	Hex		1D	56	m	n
	Decimal		29	86	m	n

[Range]
① m = 0, 48
② m = 65, 0 ≤ n ≤ 255

[Description] Selects cut mode and executes the cut command. m selects cut mode as follows:

m	Function
0, 48	Total cut.
65, 66	Form feed (cut position + [n x vertical motion unit]) and total cut

[Notes]
- This command is only enabled if set at the beginning of the line.
- The horizontal and vertical motion units are specified by $1D $50.

[Default]
[Reference] $1B $69, $1B $6D
[Example]

$1D $57 nL nH

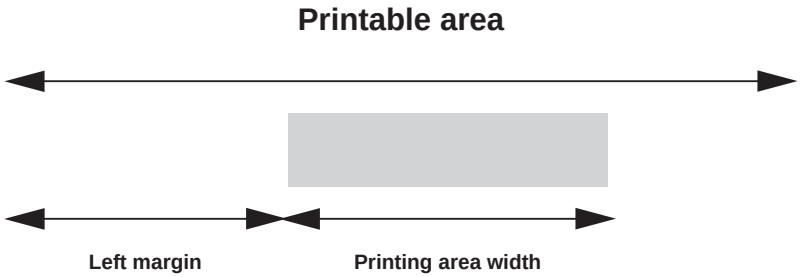
[Name] **Set printing area width**

[Format]

ASCII	GS	W	nL	nH
Hex	1D	57	nL	nH
Decimal	29	87	nL	nH

[Range] 0 ≤ nL, nH ≤ 255
0 ≤ (nL + nH * 256) ≤ 832

[Description] Sets the printing area width to the area specified by nL and nH.
- The left margin is set to [(nL + nH ´ 256) ´ (horizontal motion unit)] inches.



[Notes]
- This command is only enabled if set at the beginning of the line.
- If the right margin is greater than the printable area, the printing area width is set at maximum value.
- If the printing area width = 0, it is set at the maximum value.
- The horizontal and vertical motion units are specified by $1D $50. Changing the horizontal or vertical motion unit does not affect the current left margin.
- The $1D $50 command can change the horizontal (and vertical) motion unit.
- However, the value cannot be less than the minimum horizontal movement amount and it must be in even units of the minimum horizontal movement amount.

[Default]
[Reference] $1D $4C, $1D $50
[Example]

$1D $5E r t m

[Name] **Execute macro**

[Format]

ASCII	GS	{ }	r	t	m
Hex	1D	5E	r	t	m
Decimal	29	94	r	t	m

[Range] 0 ≤ r, t ≤ 255
0 ≤ m ≤ 1

[Description] Executes a macro.
- r specifies the number of times to execute the macro.
- t specifies the waiting time for executing the macro. The waiting time is t * 100 msec. for each macro execution.
- m specifies macro executing mode:

When the LSB of m = 0, the macro is executed r times continuously at the interval specified by t.
When the LSB of m = 1, after waiting for the period specified by t, the LED indicator blinks and the printer waits for the FEED button to be pressed. After the button is pressed, the printer executes the macro once. The printer repeats the operation r times.

[Notes]
- This command has an interval of (t * 100 msec.) after a macro is executed by t.
- If this command is received while a macro is being defined, the macro definition is aborted and the definition is cleared.
- If the macro is not defined or if r is 0, nothing is executed.
- When the macro is executed by pressing the FEED button (m=1), the paper cannot be fed using the FEED button.

[Default]
[Reference] $1D $3A
[Example]

$1D $66 n

[Name] **Select font for HRI characters**

[Format]

ASCII	GS	f	n
Hex	1D	66	n
Decimal	29	102	n

[Range] n = 0, 1, 48, 49

[Description] Selects a font for the HRI characters used when printing a bar code. n selects a font from the following table:

n	FUNCTION
0, 48	Font A
1,49	Font B

[Notes] HRI characters are printed at the position specified by $1D $48.
[Default] n = 0
[Reference] $1D $48, $1D $6B
[Example]

$1D $68 n

[Name] **Set bar code height**

[Format]

ASCII	GS	h	n
Hex	1D	68	n
Decimal	29	104	n

[Range] 1 ≤ n ≤ 255
[Description] Sets the height of the bar code; n specifies the number of vertical dots.
[Notes]
[Default] n = 162 (20.25 mm)
[Reference] $1D $6B
[Example]

① $1D $6B m [d1...dk] $00, ② $1D $6B m [d1...dn]

[Name] **Print bar code**

[Format]

①	ASCII	GS	k	m	NUL
	Hex	1D	6B	m	00
	Decimal	29	107	m	0
②	ASCII	GS	k	m	n
	Hex	1D	6B	m	n
	Decimal	29	107	m	n

[Range] ① 0 ≤ m ≤ 20
② 65 ≤ m ≤ 90

[Description] Selects a bar code system and prints the bar code. m selects a bar code system as follows:

	m	Barcode system	No. of characters	Remarxs
①	0	UPC-A	11 ≤ k ≤ 12	48 ≤ d ≤ 57
	1	UPC-E	11 ≤ k ≤ 12	48 ≤ d ≤ 57
	2	EAN13 (JAN)	12 ≤ k ≤ 13	48 ≤ d ≤ 57
	3	EAN8 (JAN)	7 ≤ k ≤ 8	48 ≤ d ≤ 57
	4	CODE39	1 ≤ k	48 ≤ d ≤ 57, 65 ≤ d ≤ 90, 32, 36, 37, 43, 45, 46, 47
	5	ITF	1 ≤ k (even number)	48 ≤ d ≤ 57
	6	CODABAR	1 ≤ k	48 ≤ d ≤ 57, 65 ≤ d1 ≤ 68, 36, 43, 45, 46, 47, 58
	7	CODE93	1 ≤ k ≤ 255	1 ≤ d ≤ 127
	8	CODE128	2 ≤ k ≤ 255	1 ≤ d ≤ 127
	20	CODE32	8 ≤ k ≤ 9	48 ≤ d ≤ 57

②	65	UPC-A	11 ≤ n ≤ 12	48 ≤ d ≤ 57
	66	UPC-E	11 ≤ n ≤ 12	48 ≤ d ≤ 57
	67	EAN13 (JAN)	12 ≤ n ≤ 13	48 ≤ d ≤ 57
	68	EAN8 (JAN)	7 ≤ n ≤ 8	48 ≤ d ≤ 57
	69	CODE39	1 ≤ n ≤ 255	48 ≤ d ≤ 57, 65 ≤ d ≤ 90, 32, 36, 37, 43, 45, 46, 47
	70	ITF	1 ≤ n ≤ 255	48 ≤ d ≤ 57
	71	CODABAR	1 ≤ n ≤ 255	48 ≤ d ≤ 57, 65 ≤ d1 ≤ 68, 36, 43, 45, 46, 47, 58
	72	CODE93	1 ≤ n ≤ 255	1 ≤ d ≤ 127
	73	CODE128	2 ≤ n ≤ 255	1 ≤ d ≤ 127
	90	CODE32	8 ≤ n ≤ 9	48 ≤ d ≤ 57

[Notes]
- If d is outside of the specified range, the printer prints the following message: “BAR CODE GENERATOR IS NOT OK!” and processes the data which follows as normal data.
- If the horizontal size exceeds the printing area, the printer only feeds the paper.
- This command feeds as much paper as is required to print the bar code, regardless of the line spacing specified by $1B $32 or $1B $33.
- After printing the bar code, this command sets the print position to the beginning of the line.
- This command is not affected by print modes (emphasized, double-strike, underline or character size), except for upside-down and justification mode.

[Notes per ①]
- This command ends with a NUL code.
- When the bar code system used is UPC-A or UPC-E, the printer prints the bar code data after receiving 11 (without check digit) or 12 (with check digit) bytes bar code data.
- When the bar code system used is EAN13, the printer prints the bar code data after receiving 12 (without check digit) or 13 (with check digit) bytes bar code data.
- When the bar code system used is EAN8, the printer prints the bar code data after receiving 7 (without check digit) or 8 (with check digit) bytes bar code data.
- The number of data for ITF bar code must be even numbers. When an odd number of data is input, the printer ignores the last received data.

[Notes per ②]
- If n is outside of the specified range, the printer stops command processing and processes the following data as normal data.

When CODE93 is used the printer:
- prints an HRI character (o) as a start character at the beginning of the HRI character string.
- prints an HRI character (o) as a stop character at the end of the HRI character string.
- The printer prints an HRI character (n) as a control character (00H to 1FH and 7FH).

When CODE128 is used:
- When using CODE128 in this printer, please note the following regarding data transmission:
- The top part of the bar code data string must be a code set selection character (CODE A, CODE

B or CODE C) which selects the first code set.
• Special characters are defined by combining two characters "{" and one character. ASCII character "{" is defined by transmitting "{" twice, consecutively.

Specific character	Data transmission		
	ASCII	Hex	Decimal
SHIFT	{S	7B, 53	123, 83
CODE A	{A	7B, 41	123, 65
CODE B	{B	7B, 42	123, 66
CODE C	{C	7B, 43	123, 67
FNC1	{1	7B, 31	123, 49
FNC2	{2	7B, 32	123, 50
FNC3	{3	7B, 33	123, 51
FNC4	{4	7B, 34	123, 52
'{'	{{	7B, 7B	123, 123

Quando si utilizza UPC-E, introducendo i caratteri barcode, la stampante stampa.

Transmitted data											Printing data					
d1	**d2**	**d3**	**d4**	**d5**	**d6**	**d7**	**d8**	**d9**	**d10**	**d11**						
0	0-9	0-9	0	0	0	0	0	0-9	0-9	0-9	d2	d3	d9	d10	d11	0
0	0-9	0-9	1	0	0	0	0	0-9	0-9	0-9	d2	d3	d9	d10	d11	1
0	0-9	0-9	2	0	0	0	0	0-9	0-9	0-9	d2	d3	d9	d10	d11	2
0	0-9	0-9	3-9	0	0	0	0	0	0-9	0-9	d2	d3	d4	d10	d11	3
0	0-9	0-9	0-9	1-9	0	0	0	0	0	0-9	d2	d3	d4	d5	d11	4
0	0-9	0-9	0-9	0-9	1-9	0	0	0	0	5-9	d2	d3	d4	d5	d6	d11

[Default]
[Reference] $1D $48, $1D $66, $1D $68, $1D $77
[Example] ① Example of print the Bar Code 39
1D 6B 04 54 45 53 54 00
② Example of print the Bar Code 39
1D 6B 45 04 54 45 53 54

$1D $72 n

[Name] **Transmit status**
[Format]

ASCII	GS	r	n
Hex	1D	72	n
Decimal	29	114	n

[Range] n = 1, 49
[Description] Transmits the status specified by n as follows:

n	FUNCTION
1, 49	Transmits paper sensor status (as for $1B $76.

Paper sensor status (n = 1, 49)

Bit	Off/On	Hex	Decimal	Function
0,1	Off	00	0	Near paper-end sensor: Paper present
	On	03	3	Near paper-end sensor: Paper not present
2,3	Off	00	0	Paper-end sensor: Paper present
	On	(0C)	(12)	Paper-end sensor: Paper not present
4	Off	00	0	Not used. Fixed to Off.
5	-	-	-	Undefined
6	-	-	-	Undefined
7	Off	00	0	Not used. Fixed to Off.

[Notes] • This command is executed wen the data is processed in the data buffer. Therefore, there may be a time lag between receiving the command and transmitting the status, depending on data buffer status.

[Default]

[Reference] $10 $04, $1B $76

[Example]

$1D $76 $30 m xL xH yL yH d1...dk

[Name] **Print raster bit image.**

[Format]

ASCII	GS	v	0	m	xL xH yL yH d1...dk
Hex	1D	76	30	m	xL xH yL yH d1...dk
Decimal	29	118	48	m	xL xH yL yH d1...dk

[Range] 0 ≤ m ≤ 3, 48 ≤ m ≤ 51
0 ≤ xL ≤ 255
0 ≤ xH ≤ 255 (1 ≤ xL + xH x 256 ≤ 65535)
0 ≤ yL ≤ 255
0 ≤ yH ≤ 8 (1 ≤ yL + yH x 256 ≤ 2047)
0 ≤ d ≤ 255
k = (xL + xH x 256) + (yL + yH x 256)
(except for k = 0)

[Description] Selects raster bit image mode. The value of m selects the mode as follows:

m	Mode
0,48	Normal
1, 49	Double-width
2, 50	Double-height
3, 51	Quadruple

• xL, xH selects the number of data bits (xL + xH * 256) in the horizontal direction for the bit image.
• yL, yH selects the number of data bits (yL + yH * 256) in the vertical direction for the bit image.
• k shows the number of data of the image. It's an explanation parameter so it isn't necessary to transmit it.
• d shows the data of the image.

[Notes] • In standard mode for receipt paper, this command is effective only when there is no data in the print buffer.
• The data (d) identify as 1 a printed bit and as 0 a non printed bit.
• If a raster bit image is longer than one line, the surplus data aren't printed.
• This command has no effect in all print modes (character size, emphasized,double-strike, upside-down, underline, hite/black reverse printing, etc.) for raster bit image, except the reverse mode (90° anticlockwise rotation).
• This command feed the paper as much as is necessary to print the raster bit image, though the spacing set by $1B $32 or $1B $33.
• Don't use this command during a macro execution because it can't be included in a macro.
• After the printing, the printing position moves to the beginning of the line.
• The following table shows the report between the image data and the printing result:

d1	**d2**	**...**	**dx**
dX+1	**dX+2**	**...**	**dX x 2**
:	**:**	**...**	**:**
...	**dk-2**	**dk-1**	**dk**

[Default]

[Reference]

[Example]

$1D $77 n

[Name] **Set bar code width**

[Format]

ASCII	GS	w	n
Hex	1D	77	n
Decimal	29	119	n

[Range] 1 ≤ n ≤ 6

[Description] Sets the horizontal size of the bar code. n specifies the bar code width as follows:

n	MODULE WIDTH (mm)
1	0.125
2	0.25
3	0.375
4	0.5
5	0.625
6	0.75

[Notes]

[Default] n = 3

[Reference] $1D $6B

[Example]

$1D $7C n

[Name] **Set printing density**

[Format]

ASCII	GS	|	n
Hex	1D	7C	n
Decimal	29	124	n

[Range] 0 ≤ n ≤ 8, 48 ≤ n ≤ 56

[Description] Sets printing density. n specifies printing density as follows:

n	PRINTING DENSITY
0, 48	- 50%
1, 49	- 37.5%
2, 50	- 25%
3, 51	- 12.5%
4, 52	0%
5, 53	+ 12.5%
6, 54	+ 25%
7, 55	+ 37.5%
8,56	+ 50%

[Notes] • Printing density reverts to the default value when the printer is reset or turned off.

[Default] n = 4

[Reference]

[Example]

$1D $E0 n

[Name] Enable/disable automatic FULL STATUS back.

[Format]

ASCII	GS	{ }	n
Hex	1D	E0	n
Decimal	29	224	n

[Range] 0 ≤ n ≤ 255

[Description] Enable / disable automatic FULL STATUS back.

n speci es the composition of FULL STATUS as follows:

Bit	Off/On	Hex	Decimal	FUNCTION
0	Off	00	0	Disable Paper status
	On	01	1	Enable Paper status
1	Off	00	0	Disable User status
	On	02	2	Enable User status
2	Off	00	0	Disable Recoverable Error Status
	On	04	4	Enable Recoverable Error Status
3	Off	00	0	Disable Unrecoverable Error Status
	On	08	8	Enable Unrecoverable Error Status
4	-	-	-	Non definito
5	-	-	-	Non definito
6	-	-	-	Non definito
7	-	-	-	Non definito

[Notes] • Once enable at least one byte of the FULL STATUS, for each change of at least one of the bits which compose the required status, the status sent in automatic from the printer will be so composed as follows:
1° byte = $10 (DLE)
2° byte = n
Next byte (depends how many bits are active in n)

[Default]

[Reference] $10 $04

[Example]

$1D $E1

[Name] **Reading of length paper (cm) available before virtual paper-end**

[Format]

ASCII	GS	{ }
Hex	1D	E1
Decimal	29	225

[Description] Reading of length (cm) paper available before virtual paper-end.
The command return a string pointing out how much paper is available, for example if there are 5.1 m before the paper end, it will be: ‘510cm’

[Notes] • The lenght of residual paper reported is just as an indication because tolerances and other factors are not taken into consideration (paper thickness, roll core diameter, roll core thickness). The virtual paper-end limit is set by the command $1D $E6.
• To set virtual paper-end limit, measure the length of the paper from near paper end to the end of the roll, using several of them.

[Default]

[Reference] $1D $E6

[Example]

$1D $E2

[Name] **Reading number of cuts performed from the printer**

[Format]

ASCII	GS	{ }
Hex	1D	E2
Decimal	29	226

[Description] Reading the number of cuts performed from the printer.
The command return a string that points out how many cuts are performed by the printer, for example if there are performed 2376 cuts, it will be: ‘2376 cuts’

[Notes]

[Default]

[Reference]

[Example]

$1D $E3

[Name] **Reading of length (cm) of printed paper**

[Format]

ASCII	GS	{ }
Hex	1D	E3
Decimal	29	227

[Description] Reading of length (cm) of printed paper.
The command return a string pointing out how much paper is printed, for example if the printer has print about 2515,5 m, it will be: '251550cm'

[Notes]

[Default]

[Reference]

[Example]

$1D $E5

[Name] **Reading number of power up**

[Format]

ASCII	GS	{ }
Hex	1D	E5
Decimal	29	229

[Description] Reading number of power up of the printer.

[Notes] • The command return a string pointing out the number of turning on of the printer, for example if the printer is turned on 512 times, it will be: '512on'

[Default]

[Reference]

[Example]

$1D $E6 nH nL

[Name] **Virtual paper-end limit**

[Format]

ASCII	GS	{ }	nH	nL
Hex	1D	E6	nH	nL
Decimal	29	230	nH	nL

[Range] 0 ≤ nH, nL ≤ 255

[Description] This command sets the limit after which is pointed out the virtual paper-end.

[Notes] • The calculation limit of the near paper-end is in centimetres.
• This value is expressed as [(nH x 256)+nL]

[Default] nH = 0x00
nL = 0xF0

[Reference]

[Example] To set the virtual paper-end limit so that is pointed out after 15 meters from the first detetction of near paper end, it's necessary convert 15 meters in 1500 cm and then, calculate the nH and nL value in the following mode :
nH = 1500 / 256 = 5
nL = 1500 - (nH x 256) = 1500 - (5 x 256) = 220
and then send the following command :

Hex:	$1D	$E6	$05	$DC
Decimal:	29	230	5	220

$1D $E7 nH nL

[Name] **Set notch distance**

[Format]

ASCII	GS	{ }	nH	nL
Hex	1D	E7	nH	nL
Decimal	29	231	nH	nL

[Range] 0 ≤ nH ≤ 255
0 ≤ nL ≤ 255

[Description] Sets notch distance in mm from the beginning of the document (see Appendix B).

[Notes] • This value is expressed as [(nH * 256)+nL].

• The maximum value is 99,9 mm.

[Default] nH = $00
nL = $00

[Reference]

[Example]

$1D $F0 n

[Name] **Set printing speed**

[Format]

ASCII	GS	{ }	n
Hex	1D	F0	n
Decimal	29	240	n

[Range] 0 ≤ n ≤ 2

[Description] Sets printing speed; n specifies the printing speed as follows:

n	PRINTING SPEED
0	Low
1	Normal
2	High

[Notes] • Printing speed reverts to the default value when the printer is reset or turned off.

[Default] n = 1

[Reference]

[Example]

$1D $F6

[Name] **Align the print head with the notch**

[Format]

ASCII	GS	{ }
Hex	1D	F6
Decimal	29	246

[Description] Set the print head notch alignment.

[Notes] • Range from 0 to 32 mm for of programmable distances.

[Reference] **$1D $E7, $1D $F8**

[Example]

$1D $F8

[Name] **Align the autocutter with the notch**

[Format]

ASCII	GS	{ }
Hex	1D	F8
Decimal	29	248

[Description] Set the autocutter notch alignment.

[Notes]

[Reference] $1D $F6

[Example]

1.2.2 Svelta Emulation

The following table lists all the commands for function management. The commands must be transmitted to the printer ascommand string enclosed between '<' character and '>' character.

COMMAND DESCRIPTION TABLE (Tab.1.2)

ASCII Command	Description
<CB>	Clear data in the print buffer
<NR>	Restore the text horizontal
<RR>	Rotate text 90° clockwise
<RL>	Rotate text 90° counter-clockwise
<RU>	Rotate text 180°
<p>	Printing command (cut and buffer cleaning) in reverse
<q>	Printing command (only buffer cleaning) in reverse
<P>	Printing command (cut and buffer cleaning) in normal
<Q>	Printing command (only buffer cleaning) in normal
<BF x1, y1, x2, y2>	Command to create filled BOX
<BV x1, y1, x2, y2>	Command to create empty BOX
<BX x1, y1, x2, y2, s, t>	Command to create parametric BOX
<RC row, column>	Position the cursor
<HW height, width>	Set height and width of the current font
<F n>	Select font
<BS height, width>	Define area for the BOX mode
<X n, M>	Define the barcode lines dimension
<NFL s>Data	Print horizontal ITF barcode
<NFP s>Data	Print vertical ITF barcode
<NL s>Data	Print an horizontal code 39 barcode
<NP s>Data	Print a vertical code 39 barcode
<BA n>	Change the ticket print intensity
<LHT length, width, notch, dimnotch>	Set the ticket dimension to print
<T>	Get the ticket dimension to print
<S n>	Status request
<BC n>	Read a BarCode
<PN>	Get number of stored logo
<PL>	Get pictures header list
<PI n>	Get pictures header info
<PR n, x, y, sp>	Print rotated image
<PP n, x, y, sp>	Print image in graphic page
<PC HexNumLogo HexXDim HexYDim HexTBD Id HexData>	Save the image in flash
<PE n>	Delete image
<SP n>	Change speed
<TIME>	Print time
<DATE>	Print date

<DT m>	Read date/time through serial port	
<SDT m Data>	Set date/time through serial port	
<TDF m Data>	Set User-Defined Date/Time Formats	
<bXnn>	Sets the scan timeout of the barcode reader	Only in the version with barcode scanner
<B>	Return the scan timeout value of the barcode reader	
<EPOS>	Change printer emulation to ESC/ POS	
<SVEL>	Change printer emulation to SVELTA	
<COM2>	Select the communication toward RFID module	Only in the version with RFID (mifare/ icode)
<COM1>	Terminate the communication toward RFID module	

Given below are more detailed descriptions of each command.

<CB>

[Name] **Clear data in the print buffer**
[Format] ASCII <CB>
[Description] Clear data in the print buffer, move the cursor to column 0, row 0, resets the text rotation, set the deault font as current and disables the Box Size function during the character writing.
[Notes]
[Default]
[Reference]
[Example]

<NR>

[Name] **Restore the text in horizontal**
[Format] ASCII <NR>
[Description] Restore the text in horizontal, without rotation.
[Notes]
[Default]
[Reference]
[Example]

<RR>

[Name] **Rotate text 90° clockwise**
[Format] ASCII <RR>
[Description] Rotate text 90° clockwise, (to the right).
[Notes]
[Default]
[Reference]
[Example]

<RL>

[Name] **Rotate text 90° counter-clockwise**
[Format] ASCII <RL>
[Description] Rotate text 90° counter-clockwise, (to the left).
[Notes]
[Default]
[Reference]
[Example]

<RU>

[Name] **Rotate text 180°**
[Format] ASCII <RU>
[Description] Rotate text 180°.
[Notes]
[Default]
[Reference]
[Example]

<p>

[Name] **Printing command (cut and buffer cleaning) in reverse**
[Format] ASCII <p>
[Description] This command executes the following operations :
- align the ticket to notch;
- barcode reader turn ON;
- prints ticket;
- clear the data in the print buffer;
- align the ticket to cut;
- executes a ticket cut.

[Notes] • Print ticket in reverse
• After printing, the data of the barcode read and the reading result, are stored in a circular buffer.
• To read the barcode acquired during printing, use the <BC1> or <BCA> commands.
[Default]
[Reference] <CB>
[Example]

<q>

[Name] **Printing command (only buffer cleaning) in reverse**
[Format] ASCII <q>
[Description] This command executes the following operations :
- align the ticket to notch;
- barcode reader turn ON;
- prints ticket;
- clear the data in the print buffer;

[Notes] • Print ticket in reverse
• After printing, the data of the barcode read and the reading result, are stored in a circular buffer.
• To read the barcode acquired during printing, use the '<BC1>' or '<BCA>' commands.
[Default]
[Reference] <CB>
[Example]

<P>

[[Name] **Printing command (cut and buffer cleaning) in normal**
[Format] ASCII <P>
[Description] This command executes the following operations :
- align the ticket to notch;
- barcode reader turn ON;
- prints ticket;
- clear the data in the print buffer;
- align the ticket to cut;
- executes a ticket cut.

[Notes] • Print ticket in normal

• After printing, the data of the barcode read and the reading result, are stored in a circular buffer.
• To read the barcode acquired during printing, use the '<BC1>' or '<BCA>' commands.

[Default]
[Reference] <CB>
[Example]

<Q>

[Name] **Printing command (only buffer cleaning) in normal**
[Format] ASCII <Q>
[Description] This command executes the following operations :
- align the ticket to notch;
- barcode reader turn ON;
- prints ticket;
- clear the data in the print buffer;

[Notes] • Print ticket in normal
• After printing, the data of the barcode read and the reading result, are stored in a circular buffer.
• To read the barcode acquired during printing, use the '<BC1>' or '<BCA>' commands.

[Default]
[Reference] <CB>
[Example]

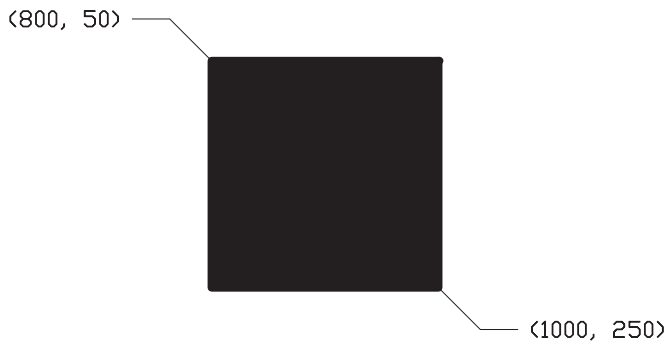
<BF x1, y1, x2, y2>

[Name] **Command to create filled Box**
[Format] ASCII <BF x1,y1,x2,y2>
[Description] Create a filled box on the basis of x1, y1, x2, y2 coordinates where :
x1 -> minimum horizontal coordinate
y1 -> minimum vertical coordinate
x2 -> maximum horizontal coordinate
y2 -> maximum vertical coordinate

[Notes] • If the coordinates are reversed, the printer automatically turns the points to create in any case the box.
• If the x2 is greater than the maximum horizontal width of graphic page, the box is drawn using the maximum width as last point.
• If the y2 is greater than the maximum length of graphic page defined by <LHT...> command, the box is drawn using the maximum length (defined by this command) as last point.

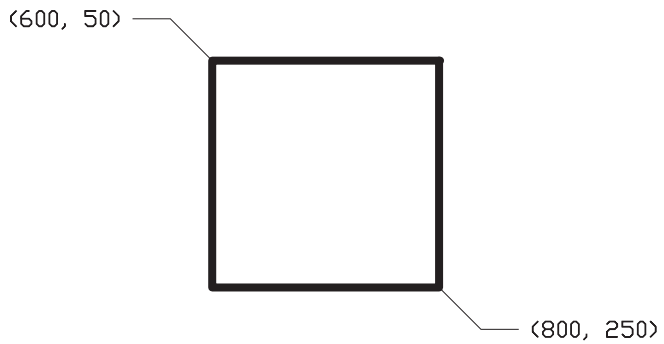
[Default]
[Reference]
[Example] Ticket example that use a filled box

```
<CB><BA8>
<BF800,50,1000,250>
<q>
```



<BV x1, y1, x2, y2>

[Name] **Command to create empty Box**
[Format] ASCII <BF x1,y1,x2,y2>
[Description] Create an empty box on the basis of x1, y1, x2, y2 coordinates where :
x1 -> minimum horizontal coordinate
y1 -> minimum vertical coordinate
x2 -> maximum horizontal coordinate
y2 -> maximum vertical coordinate
[Notes] • The box border is fixed to 1mm (8 dots)
• If the coordinates are reversed, the printer automatically turns the points to create in any case the box.
• If the x2 is greater than the maximum horizontal width of graphic page, the box is drawn using the maximum width as last point.
• If the y2 is greater than the maximum length of graphic page defined by <LHT...> command, the box is drawn using the maximum length (defined by this command) as last point.
[Default]
[Reference]
[Example] Ticket example that use an empty box
<CB><BA8>
<BV600,50,800,250>



<BX x1, y1, x2, y2, s, t>

[Name] **Command to create parametric Box**
[Format] ASCII <BX x1,y1,x2,y2, s, t >
[Description] Create a box defined by the following parameters where :
x1 -> minimum horizontal coordinate
y1 -> minimum vertical coordinate
x2 -> maximum horizontal coordinate
y2 -> maximum vertical coordinate
s -> border thickness in dot (8 dot = 1mm) s ≤ 255
t -> Fill mode 0 ≤ t ≤ 9

t	Fill mode
0	Deletes area
1	Fills area
2..8	Fills area with specific pattern
9	the area leaves unchanged (only for rectangle border)

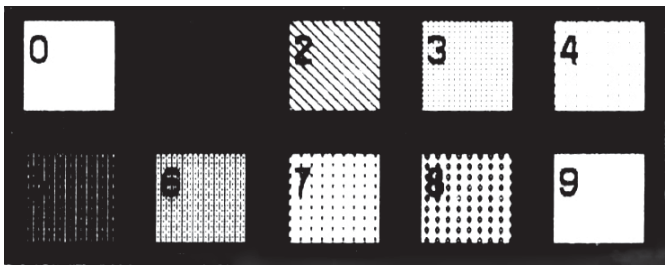
[Notes] • If t > 9 the fill mode is set to 9
• If the coordinates are reversed, the printer automatically turns the points to create in any case the box.
• If the x2 is greater than the maximum horizontal width of graphic page, the box is drawn using the maximum width as last point.

• If the y2 is greater than the maximum length of graphic page defined by <LHT...> command, the box is drawn using the maximum length (defined by this command) as last point.
• If the defined thickness is greater than the half of box width, then the thickness is set to the half of box width to print (filled box).

[Default]
[Reference]
[Example] Command sequence to generate a demo ticket with differents kinds of box

```
<CB><BA8><BS0,0>
<NR>
<BX200,100,300,200,16,0><RC120,220><F3><HW1,1>0
<BX300,100,400,200,16,1><RC120,320><F3><HW1,1>1
<BX400,100,500,200,16,2><RC120,420><F3><HW1,1>2
<BX500,100,600,200,16,3><RC120,520><F3><HW1,1>3
<BX600,100,700,200,16,4><RC120,620><F3><HW1,1>4
<BX200,200,300,300,16,5><RC220,220><F3><HW1,1>5
<BX300,200,400,300,16,6><RC220,320><F3><HW1,1>6
<BX400,200,500,300,16,7><RC220,420><F3><HW1,1>7
<BX500,200,600,300,16,8><RC220,520><F3><HW1,1>8
<BX600,200,700,300,16,9><RC220,620><F3><HW1,1>9
<q>
```

Example of what will be printed on ticket



<RC row, column>

[Name] **Position the cursor**
[Format] ASCII <RC row, column>
[Description] Moves the cursor at the position specified by row and column parameters.
[Notes] • The row and column values must be a number with four digit at most, otherwise the command will be ignored.
[Default]
[Reference]
[Example] To move the cursor at row (dot) 10, column (dot) 30 the command sequence is :

```
<RC 10,30>
```

<HW height, width>

[Name] Set height and width of the current font
[Format] ASCII <HW height, widht>
[Description] Modifies the height and width of the current font where height and width are the multiplier coefficients of heigth and width of how enlarge the font.Both values can be:
1: Font dimension x1
2: Font dimension x2
4:Font dimension x4
[Notes] • The command is ignored if height or width has different value from that reported above.
[Default]
[Reference]
[Example]

<F n>

[Name] **Select the font**
[Format] ASCII <F n>
[Description] Selects the current font where n indicates the font to use.
[Notes]
[Default]
[Reference]
[Example]

<BS height, width>

[Name] Define area for the box mode
[Format] ASCII <BS height, width>
[Description] Defines the area where position a character. If the box dimensions are bigger than the font, then the empty spaces are filled with white spaces, whereas if the box dimensions are smaller than the font, then the font is cutted.
[Notes] • To disable the Box Size set height and width parameters to 0 (<BS0,0>).
[Default]
[Reference]
[Example]

<X n, M>

[Name] **Define the barcode lines dimension**
[Format] ASCII <X n, M>
[Description] n defines the thins lines dimension (in dot) of barcode. The M parameter defines the barcode printing speed if it must be printed rotated.
[Notes] • if the M parameter = 'H' as ASCII value, the barcodes will be printed in high speed. Otherwise if if the M parameter = 'L' as ASCII value the barcodes will be printed at reduced speed (only if n is less than 4).
[Default]
[Reference]
[Example]

<NFL s> Data

[Name] **Print horizontal ITF BarCode**
[Format] ASCII <NFL s>Data
[Description] Print an ITF barcode type in horizontal. The s parameter indicates the barcode height in millimetres. The Data parameter contains the data to convert, with start and stop characters of barcode.
[Notes]
[Default]
[Reference]
[Example]

<NFP s> Data

[Name] **Print vertical ITF BarCode**
[Format] ASCII <NFP s>Data
[Description] Print an ITF barcode type in vertical. The s parameter indicates the barcode height in millimetres. The Data parameter contains the data to convert, with start and stop characters of barcode.
[Notes]
[Default]
[Reference]
[Example]

<NL s> Data

[Name] **Print an horizontal code 39 barcode**
[Format] ASCII <NL s>Data
[Description] Print a code 39 barcode type in horizontal. The s parameter indicates the barcode height in millimetres. The Data parameter contains the data to convert, with start and stop characters of barcode.
[Notes]
[Default]
[Reference]
[Example]

<NP s> Data

[Name] **Print a vertical code 39 barcode**
[Format] ASCII <NP s>Data
[Description] Print a code 39 barcode type in vertical. The s parameter indicates the barcode height in millimetres. The Data parameter contains the data to convert, with start and stop characters of barcode.
[Notes]
[Default]
[Reference]
[Example]

<BA n>

[Name] **Change the ticket print intensity**
[Format] ASCII <BA n>
[Description] Changes the ticket print intensitywhere n indicates the print mode. The possible values of n are as follows :

n	Print mode
0	Black/whyte printing at 100% of maximum intensity
8	Black/whyte printing at 50% of maximum intensity
16	Black/whyte printing at 25% of maximum intensity
24	Black/whyte printing at 12% of maximum intensity
32	Black/whyte printing at 7% of maximum intensity
40	Black/whyte printing at 5% of maximum intensity

[Note]
[Default]
[Reference]
[Example]

<LHT length, height, notch, dimnotch>

[Name] **Set ticket dimension to print**
[Format] ASCII <LHT length, height, notch, dimnotch>
[Description] Sets the ticket dimension to print in the following mode:
lenght is the ticket length (in dot);
height is the ticket height (in dot);
notch is the distance (in dot) between the ticket upper edge and strobe backside preprinted black mark;
dimnotch is the notch dimension (in dot).
[Notes] • 1mm = 8dot.
• If using the point (.) chararcter as decimal separator instead of commas then the passed value are stored in EEProm.

• IIt's recommended to not use this command for each printed ticket beacuse the total rewriting number of EEProm is limited (max 10000).

[Default]
[Reference]
[Example]

<T>

[Name] **Get the ticket dimension to print**
[Format] ASCII <T>
[Description] Get the ticket dimensions to print, in the Ticket Size format.
[Note]
[Default]
[Reference]
[Example]

<Sn>

[Name] **Status request**
[Format] ASCII <Sn>
[Description] The host can ask to the printer many differents status infos; the n parameter indicates which type of request :

If n = 1 the printer return a byte that represent the status:
$10: Paper end
$11: Correct functioning
$18: Paper jam during printing (or the print head is open)
$19: Last received command is not correct.
If n=3 the printer return ACK ($06) if printing is properly finished, otherwise return NACK ($15). If the request will be transmitted during printing phase, it waits the end of the process and then is sent the answer.

[Note]
[Default]
[Reference]
[Example]

<BC n>

[Name] **Read a BarCode**
[Format] ASCII <BC n>
[Range] n = 0, 1, A
[Description] • With **n = 0** the scan command is sent and the returned string is:
<BC0 ↵ x barcode ↵>
where
- ↵ corresponds to CR character ($0D).
- x indicate the reading result ; the x value can be :
'!' : the barcode is read
'#': the barcode is not correctly read
- barcode is the barcode's characters read

• With n = 1 the returned string is :
<BC1 ↵ x barcode ↵>
where barcode is the last barcode read through the printing commands <p>, <P>, <q>, <Q>.

• With n = A returns the last barcodes read up to ten as maximum;the returned string is:
<BCA ↵ x barcode1 ↵

x barcode2 ↵
...
x barcode n ↵
>
where
- ↵ corresponds to CR character ($0D).
- x indicate the reading result ; the x value can be :
'!' : the barcode is read
'#': the barcode is not correctly read
- barcode is the barcode's characters read

[Notes] • The barcode read through the printing commands '<p>', '<P>', '<q>', '<Q>'.
[Default]
[Reference]
[Example]

<PN>

[Name] **Get number of stored logo**
[Format] ASCII <PN>
[Description] This command sends to the printer the request of number of stored logo; the printer returns a bytes sequence as follows :
<PNn>
where n (in ASCII format) indicates the number of stored images.
[Notes]
[Default]
[Reference]
[Example] If in the flash memory are stored 10 logos send this command :

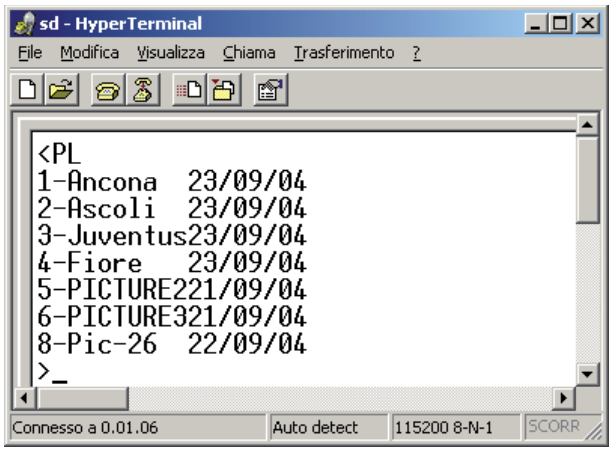
Hex	$1C	$90
ASCII	FS	{}

The printer's answer will be :

Hex	$3C	$50	$4E	$31	$30	$3E
ASCII	<	P	N	1	0	>

<PL>

[Name] **Get pictures header list**
[Format] ASCII <PL>
[Description] This command requests to the printer the list of stored logo. The printer returns a bytes sequence as follows :
<PL CrLf [N-ID CrLf]> where
• CrLf indicates the two characters $0D (Carriage return) and $0A (Line Feed);
• N is the number of stored logo;
• [ID] indicates the logo identifier, a sequence of 16 bytes that was defined when the logo is stored.This field is optional because it's returned only if the logo has been found.
[Notes] • The fields enclosed in square bracket are repeated for all number of stored images.
[Default]
[Reference]
[Example] The figure here reported showes the printer's answer after sending this command



<PIn>

[Name] **Get pictures header info**

[Format] ASCII <PIn>

[Description] Gets the logo header info stored specified by n (express in ASCII). The printer returns a bytes sequence as follows :
<PIe[ID]> where
• e indicates the search result
e = 0 picture not found
e = 1 picture found
• [ID] indicates the logo identifier, a sequence of 16 bytes that was defined when the logo is stored. This fiedl is optional because it's returned only if the logo has been found.

[Notes]

[Default]

[Reference]

[Example]

<PR n, x ,y sp>

[Name] **Print rotated image**

[Format] ASCII <PR n, x, y, sp>

[Description] Prints rotated image in graphic page where
• n is the number of image to print;
• x indicates the horizontal position inside the graphic page
• y indicates the vertical position inside the graphic page
• sp indicates the thickness value of the image border (express in dot).

[Notes] • if n is a negative number the image is printed as a background image, without deleting the area below.

[Default]

[Reference]

[Example] Several printing commands in graphic page; in the first printing command the image no. 2 is printed with border, instead the other images are printed without border :

```
<CB><n><BA8><HW1,1><BS0,0>
<PR2,10,10,8>                (image printed with border)
<PR1,10,200,0>               (image printed without border)
<PR3,210,200,0               (image printed without border)
<PR4,620,200,0               (image printed without border)
<q>
```

<PP n, x ,y sp>

[Name] **Print image in graphic page**

[Format] ASCII <PP n, x, y, sp>

[Description] Prints image in graphic page where
- n is the number of image to print;
- x indicates the horizontal position inside the graphic page
- y indicates the vertical position inside the graphic page
- sp indicates the thickness value of the image border (express in dot).

[Notes] • if n is a negative number the image is printed as a background image, without deleting the area below.

[Default]

[Reference]

[Example] Several printing commands in graphic page; in the first printing command the image no. 2 is printed with border, instead the other images are printed without border :

```
<CB><n><BA8><HW1,1><BS0,0>
<PP2,10,10,8>          (image printed with border)
<PP1,10,200,0>         (image printed without border)
<PP3,210,200,0>        (image printed without border)
<PP4,620,200,0>        (image printed without border)
<q>
```

<PCHexNumLogo HexXDim HexYDim HexTBD Id Hexdata>

[Name] **Save the image received from serial port into flash**

[Format] ASCII <PCHexNumLogo HexXDim HexYDim HexTBD Id Hexdata>

[Description] Save the image received from serial port into printer flash; if the number used to store logo is not already present inside the printer, the new logo is appended to stored logos, otherwise the image is overwritten and moved in the last position of flash.

- HexNumLogo — indicates the number of logo, 2 bytes expressed in hexadecimal notation;
- HexXDim — indicates the logo horizontal dimension in pixe, 2 bytes expressed in hexadecimal notation; the value must be multiple of 16;
- HexYDim — indicates the logo vertical dimension in pixel, 2 bytes expressed in hexadecimal notation;
- HexTBD — 2 bytes fixed to $00 (RESERVED);
- Id — indicates the logo Id, a sequence of 16 bytes that identify univocally the logo;
- Hexdata — are the image data.

The printer returns a sequence of bytes as follows :

<PC0>	if the saving include an incorrect syntax or the available memory in flash for logos is finished (128Kbyte);
<PC1n>	if the syntax command is correct and there's enough memory in flash for saving logos; n returns the status of the flash programming :
$88 ->	Sector not erased
$77 ->	Error during programming
$AA ->	Programming done.

[Note]

[Default]

[Reference]

[Example] The following example shows the bytes sequence received from serial port to store a logo into the printer flash :

```
Offset      Hexadecimal                                      ASCII

00000000:  3C 50 43 00-08 01 C0 02-49 00 00 50-69 63 2D 32  <PC □□+□I  Pic-2
00000010:  36 20 20 32-32 2F 30 39-2F 30 34 00-00 00 00 00  6  22/09/04
00000020:  00 00 00 00-00 00 00 00-00 00 00 00-00 00 00 00
....
....                                                        Dati dell'immagine
....

00008000:  00 00 00 00-00 00 00 00-00 00 00 00-00 00 00 00
00008010:  00 00 00 3E
>
```

If the programming is successful, the printer's answer will be :
$3C $50 $43 $31 $AA $3E

<PE n>

[Name] **Delete image**
[Format] ASCII <PE n>
[Description] Deletes image defined by n.
The printer returns a sequence of bytes as follows :

<PE0>	Image n not found;
<PE1n>	Image found; n ritorna lo stato di programmazione della flash
$88 ->	Sector not erased
$77 ->	Error during erasing operation
$AA ->	Erasing done.

[Note]
[Default]
[Reference]
[Example]

<SP n>

[Name] **Change speed**
[Format] ASCII <SP n>
[Description] Sets prinitng speed using n as follows :

n	Printing speed
0	High quality
1	Normal
2	High speed

[Note]
[Default]
[Reference]
[Example]

<TIME>

[Name] **Print Time**
[Format] ASCII <TIME>
[Description] Prints time with the format specified by the command '<TDF>'.
[Note]
[Default] "hh:nn:ss"
[Reference] <DATE>
[Example]

<DATE>

[Name] **Print date**
[Format] ASCII <DATE>
[Description] Prints date in the format specified by the command '<TDF>'.
[Note]
[Default] "dd/mm/yy"
[Reference] <TIME>
[Example]

<DT m>

Name] **Read date/time through serial port**
[Format] ASCII <DT m>
[Description] Read date/time of the real time clock and send it through serial port, in the format specified by m values as follows :

m	FORMAT
0	DD/MM/YY hh:mm:ss
1	DDMMYYhhmmss
2	YYMMDDhhmmss
3	YYMMDDhhmmss

where :
DD = represents the dayof the date
MM = represents the month of the date
YY = represents year of the date
hh = represents the hour of the time
mm = represents the minutes of the time
ss = represents the seconds of the time
d = indicates the day of the week
The printer's answer will be : <DT ↵ x data ↵>

where
- ↵ corresponds to CR character ($0D).
- x indicate the reading result ; the x value can be :
'!' : the command is executed successfully
'#': the command is not executed successfully
- data are the ASCII characters that represent the date/time.

[Note]
[Default]
[Reference]
[Example] To read date/time in the "DDMMYYhhmmss" format, transmit :
<DT 1>

For example if the current date/time are “15 September 2006 at 10:56:20 (AM)” the printer’s answer is as follows :
<DT ↵ ! 151006105620 ↵ > if the transmission is succesfully, otherwise
<DT ↵ # ↵> if the transmission is not succesfully

<SDT m data>

[Name] **Set date/time of the real time clock through serial port**
[Format] ASCII <SDT m data>
[Description] Set the date/time of the real time clock through serial port, in the format specified by m values as follows :

m	FORMAT
0	DD/MM/YY hh:mm:ss
1	DDMMYYhhmmss
2	YYMMDDhhmmss
3	YYMMDDhhmmss

where:
DD = represents the dayof the date
MM = represents the month of the date
YY = represents year of the date
hh = represents the hour of the time
mm = represents the minutes of the time
ss = represents the seconds of the time
d = indicates the day of the week
• data are the ASCII characters relative to the date and time to set.

If the transmission has been received correctly and the command is valid, the printer returns the following string :
<SDT ↵ x ↵>
where
- ↵ corresponds to CR character ($0D).
- x indicate the reading result ; the x value can be :
‘!’ : the command is executed successfully
‘#’: the command is not executed successfully

[Note] • the day of the week is calculated automatically from the printer and then it’s possible that the returned value is different from the one transmitted.

[Default]
[Reference]
[Example] For example to set the date and time to “29 September 2006 at 13:51:00 (PM)” in the “YYMMDDhhmmss” format transmit:
<SDT 2 061029135100>
The printer’s answer will be :
<SDT ↵ ! ↵> if the transmission is succesfully, otherwise
<SDT ↵ # ↵> if the transmission is not succesfully

<TDF m data>

[Name] **Set User-Defined Date/Time Formats**
[Format] ASCII <TDF m data>
[Description] Sets the format string for date and time used to printing;
• m specifies which user-defined string format is set
D for date, T for time
• data are the ASCII characters relative to user-defined date/time formats.
• the maximum length of fthe user-defined date/time format string is 64 chars.

The following table shows characters used to create user-defined date/time formats :

Character	Description
I	Selects Italian language
E	Selects English language (is the default language)
c	Selects default date/time
d	Displays the day as a number without a leading zero (1-31).
dd	Displays the day as a number with a leading zero (01-31).
ddd	Displays the day as an abbreviation (for example, Sun).
dddd	Displays the day as a full name (for example, Sunday).
ddddd	Displays the date as a complete date in the short format where date values are formatted with day, month and year (the short date format is dd/mm/yy).
dddddd	Displays the date as a complete date in the extended format where date values are formatted with day, month and year (the extended date format is dd mmmm, yyyy).
m	Displays the month as a number without a leading zero (1-12). If the character m is immediately after the character h or hh , displays the minutes instead of month (see also the n character formatting).
mm	Displays the month as a number with leading zeros (01-12). If the character m is immediately after the character h or hh , displays the minutes instead of month (see also the nn character formatting).
mmm	Displays the month as an abbreviation (for example, Jan).
mmmm	Displays the month as a full month name (for example, January).
yy	Displays the year in two-digit numeric format with a leading zero.
yyyy	Displays the year in four digit numeric format.

Character	Description
h	Displays the hour as a number without leading zeros (0-23)
hh	Displays the hour as a number with leading zeros (00-23)
n	Displays the minutes as a number without leading zeros (0-59)
nn	Displays the minutes as a number with leading zeros (00-59)
s	Displays the seconds as a number without leading zeros (0-59)
ss	Displays the seconds as a number with leading zeros (00-59)
ttttt	Displays the time in the extended format where time values are formatted with hour, minutes and seconds (the extended time format is h:mm:ss).
AM/PM	Using the 12-hour clock and displays the AM prefix in uppercase next to the hours that preceding midday and the PM prefix in uppercase next to the hours between midday and midnight.
am/pm	Using the 12-hour clock and displays the am prefix in lowercase next to the hours that preceding midday and the pm prefix in lowercase next to the hours between midday and midnight.
A/P	Using the 12-hour clock and displays the A prefix in uppercase next to the hours that preceding midday and the a prefix in uppercase next to the hours between midday and midnight.
a/p	Using the 12-hour clock and displays the a prefix in lowercase next to the hours that preceding midday and the a prefix in lowercase next to the hours between midday and midnight.

[Note]
[Default]
[Reference]
[Example]

For example to print the current time with the string format ‘yy/mm/dd hh:mm:ss’ follow these steps :
1. Send the following command to define the user-defined Time string format:
<TDF T yy/mm/dd hh:mm:ss>
2. Send the following command to print the time :
<TIME>
If the date and time is 22 October 2006 at 17:35:27 (PM) the output string printed will be:
06/10/22 17:35:27

<bXnn>

[Name] **Sets the scan timeout of the barcode reader**
[Format] ASCII <bXnn>
[Description] Sets the scan timeout of the barcode reader, using nn parameter value, expressed in tenth of second (10-1 second).
If the X parameter value is equal to ASCII character 'e' ($65) the nn value (the scan timeout) is stored in EEProm. Otherwise its value is loaded into RAM so that it's possible to make different tests before save the correct value in EEProm.
[Notes]
[Default] X = 3
[Reference]
[Example]

<B>

[Name] **Return the scan timeout value of the barocde reader**
[Format] ASCII <B>
[Description] Returns the scan timeout value of the barcode reader.
[Notes]
[Default]
[Reference]
[Example]

<EPOS>

Name] **Change printer emulation to ESC/ POS**
[Format] ASCII <EPOS>
[Description] Set the ESC/ POS emulation.
[Notes]
[Default]
[Reference]
[Example]

<SVEL>

Name] **Change printer emulation to SVELTA**
[Format] ASCII <SVEL>
[Description] Set the SVELTA emulation.
[Notes]
[Default]
[Reference]
[Example]

<COM2>

Name] **Select the communication toward RFID module**
[Format] ASCII <COM2>
[Description] Set the communication toward RFID module.
[Notes]
[Default]
[Reference]
[Example]

<COM1>

Name] **Terminate the communication toward RFID module**
[Format] ASCII <COM1>
[Description] Terminates the communication toward RFID module.
[Notes]
[Default]
[Reference]
[Example]

Blank page

Part Number : DOMC-TK300II-E Rev. 1.00

CUSTOM

UNI EN ISO 9001
CERTIFIED QUALITY
SYSTEM

TÜV SÜD ISO 9001

M.U.R.S.T.
Ministry University
Research Scientific
Technology
Authorized laboratory
no.50846ZYZ

CUSTOM ENGINEERING SPA
World Headquarters
Via Berettine, 2 - 43100 Fontevivo
Tel. +39 0521 680111 - Fax +39 0521 610701
info@custom.biz - www.custom.biz

www.custom.biz

Always On!