

Thermal SMICE label Printer with integrated power supply 112/80 mm

User Manual



All rights reserved. The reproduction of this manual, either totally or partially, in any form whatsoever, whether it be on paper or through computer processes, is strictly prohibited. CUSTOM ENGINEERING SPA and the resource teams employed in the making of this manual will not be held responsible for any problems arising as a result of improper use of this manual, as they guarantee that the information contained in it has been subjected to careful inspection. Any suggestions regarding errors in its contents or possible improvements will be greatly appreciated. The products are continuously checked and improved. For this reason CUSTOM ENGINEERING SPA reserves the right to modify the information contained in this manual without prior notice.

COD. DOME – SMICE-LP4

VERS. PRELIMINARY

Copyright © 2002 CUSTOM ENGINEERING SPA – Italy

Custom Engineering

Str. Berettine 2 - 43010 Fontevivo (PARMA) - Italy

Tel. : +39 0521-680111 - Fax : +39 0521-610701

http: www.custom.it Email : support@custom.it

To contact our Technical Support :

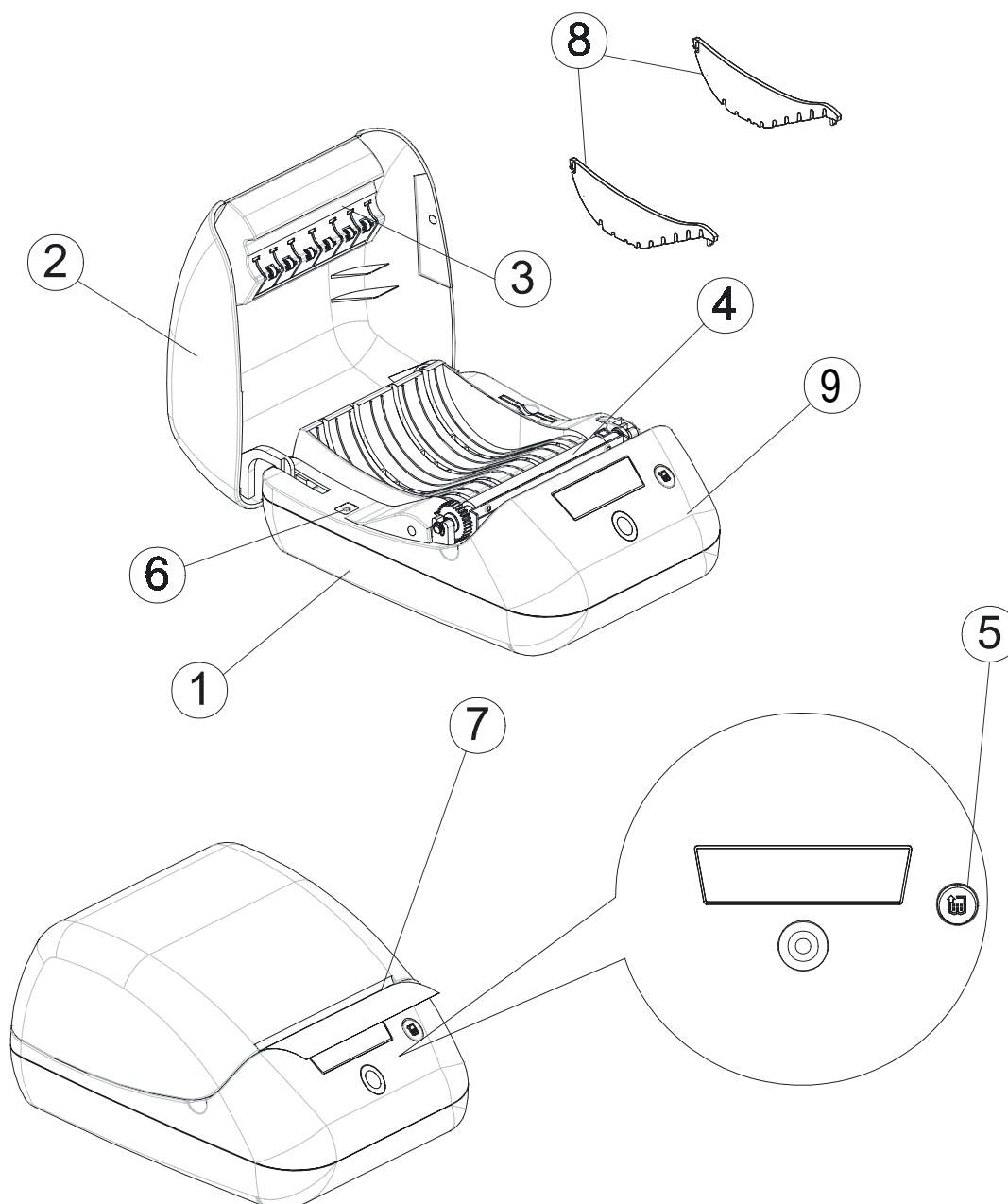
Phone. : +39 0521-680163 - Fax : +39 0521-680146

Email : support@custom.it

PRINTER COMPONENTS

A. SMICE-LP4 - front exterior view

- 1- Printer base
- 2- Cover
- 3- Paper guide
- 4- Print mechanism + Cutter
- 5- Backlighting FEED key
- 6- Key "KEY2"
- 7- Paper opening
- 8- Paper control edges for 80mm roll
- 9- Plate



B. SMICE-LP4 - rear view

- 1- Interface connector
- 2- Drawer connector
- 3- Power supply connector
- 4- ON/OFF switch

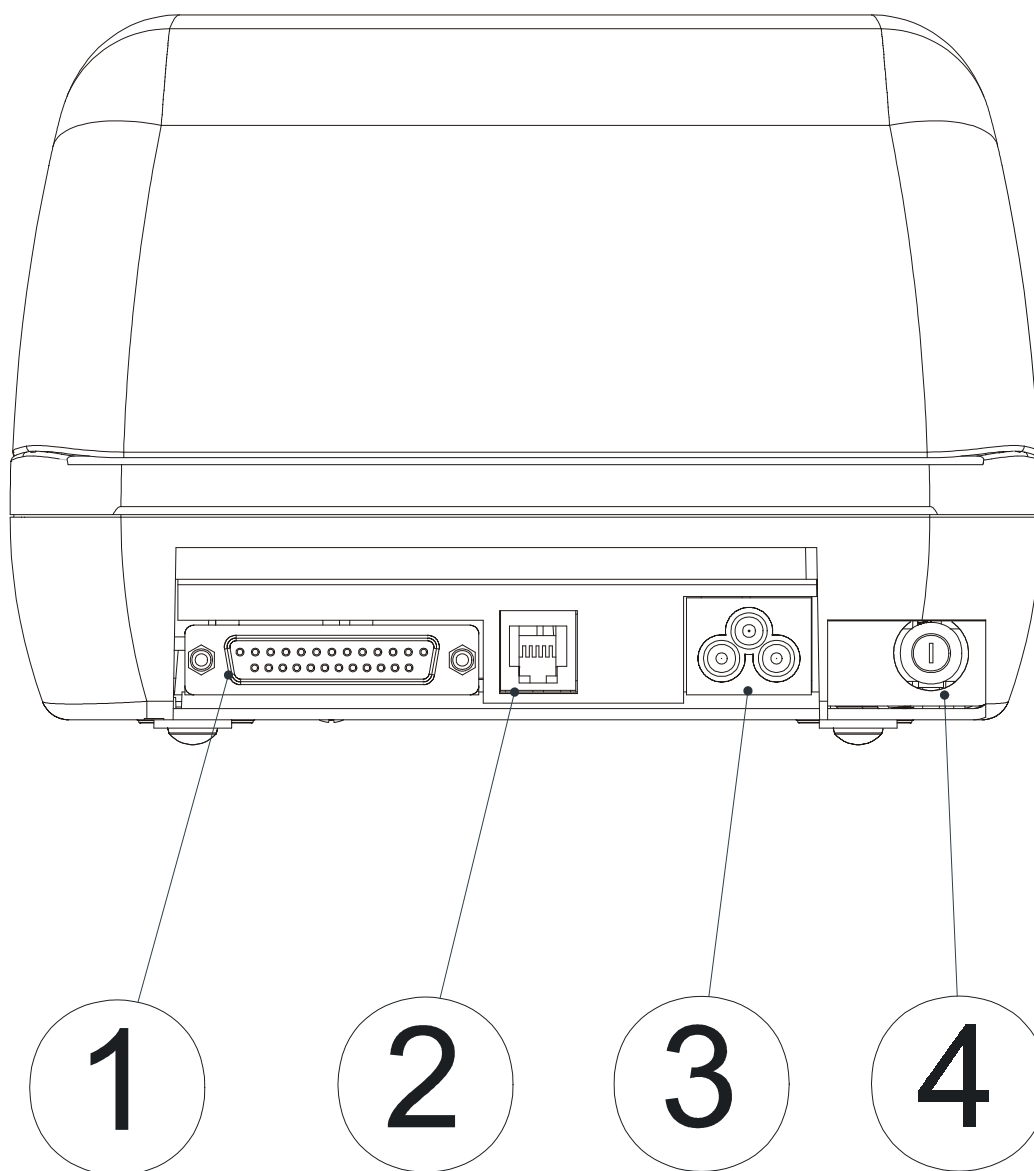


TABLE OF CONTENTS

INTRODUCTION

UNPACKING THE PRINTER
GENERAL FEATURES
DESCRIPTION OF THE PRINTER

1. COMMANDS DESCRIPTION

- 1.1 Labels commands
- 1.2 Layout handling commands
- 1.3 Fields handling commands
- 1.4 Label handling commands
- 1.5 Previous commands that may be useful
- 1.6 Example of a label handling mode application

2. TECHNICAL SPECIFICATIONS

- 2.1 Technical specifications
- 2.2 Dimensions

INTRODUCTION

UNPACKING THE PRINTER

Remove the printer from the carton, taking care not to damage the packing materials which should be retained for future shipping/moving.

Make sure all components listed below are present and not damaged. If any part is missing and/or damaged, contact customer service.

1. Manual (or CD-rom)
2. Printer

GENERAL FEATURES

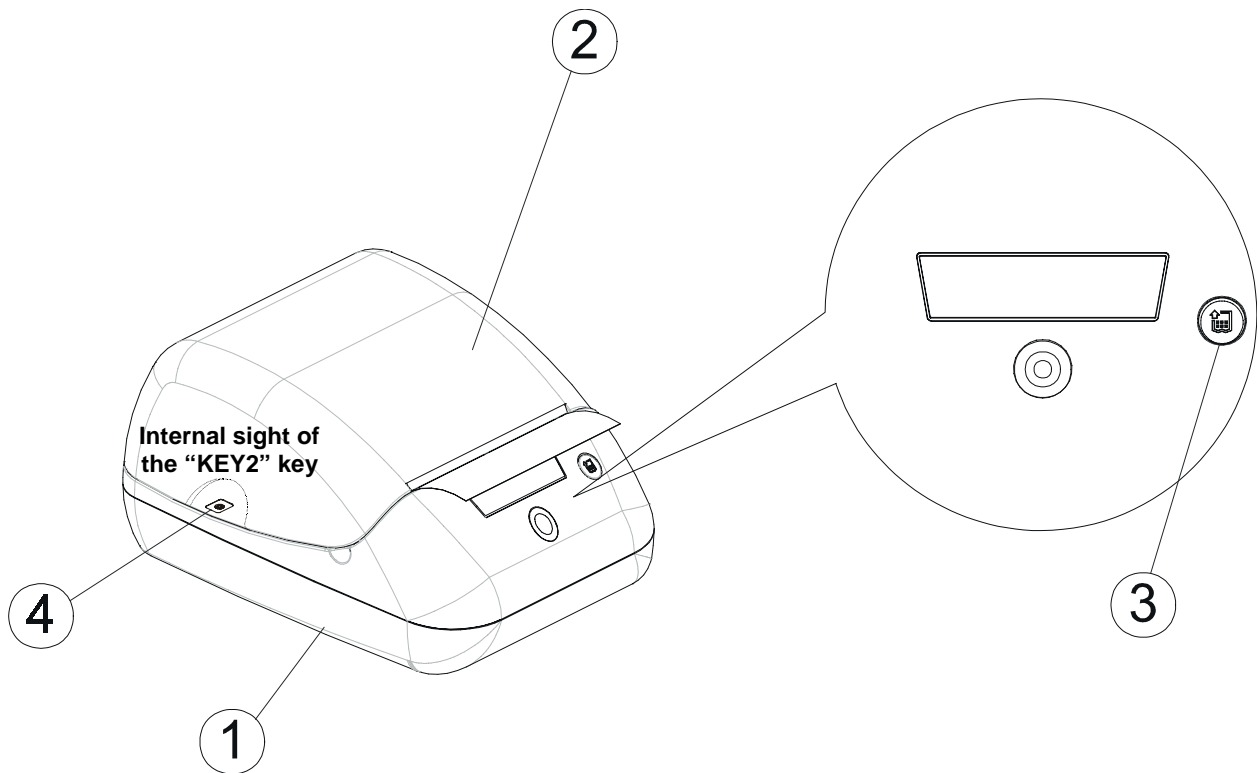
The SMICE-LP4 is suitable for industrial applications, for point of sale where space requirements is priority.

- Label width 112mm / 80mm, with different height.
- 200dpi printing resolution.
- 110 mm/sec printing speed.
- RS232 serial and Centronics parallel interface.
- Power supply 100-240 Vac.
- Sensors: label end, cover open, head temperature.
- Supplied with "Easy Label" software to manage the labels for PC (Windows 98/2000).
- Different coloured covers are available.

DESCRIPTION OF THE PRINTER

The printer (fig.2) consists of a shell in ABS-V0 (1) equipped with a cover (2) that provides access to the label roll and printing mechanism.

On the front are the backlighting FEED key (3); the "KEY2" key is located inside of the printer by side of the roll holder opening (4).



- **FEED key.** When the FEED key is pressed, the printer advances the paper. During machine power-up, pressing the FEED key, the printer performs the GRAPHIC TEST and the FONT TEST.
The GRAPHIC TEST allows to verify the print of logos stored with the **FS q** command (see paragraph 3.1), or without these, the print of logo located in the flash memory.
The FONT TEST allows to verify the print of characters font stored.
- **"KEY2" key.** During machine power-up, pressing the "KEY2" key, the printer goes in the SETUP procedure that allows to print and modify the mains setup parameters (see paragraph 1.2) besides to perform the Hexadecimal Dump function (see paragraph 1.3).

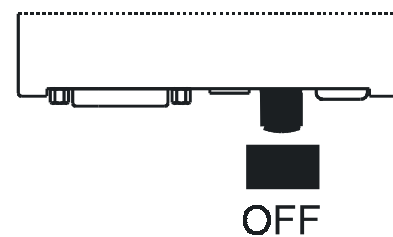
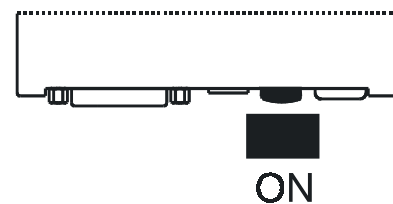
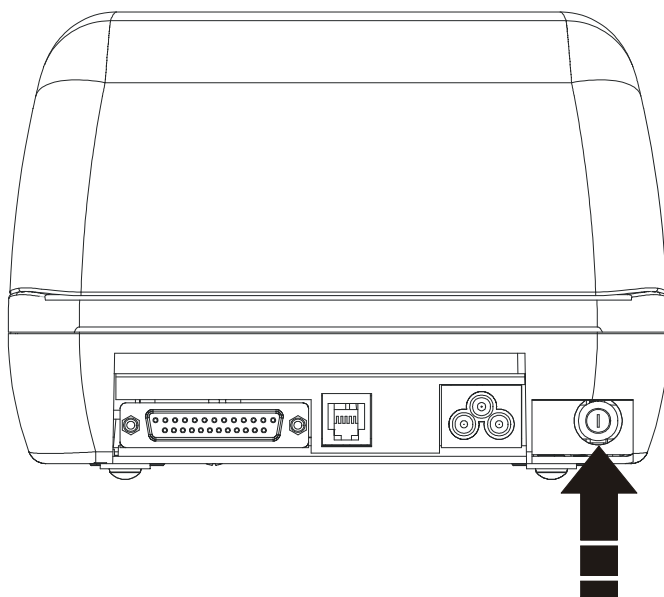
INTRODUCTION

- The backlighting of the key displays the printer functioning status as follows:

(Tab.1)

LED status	Description
None	Printer OFF
Green	Printer ON : correct functioning (stand-by)
	Flashing : Data processing phase
Yellow	Printer in warning status : paper out, cover opened
Red	Printer in error status : power supply voltage incorrect, heading over temperature.

- ON/OFF key. When pressed, it turns on the printer (fig.3); when released, the printer shuts off.



1. COMMANDS DESCRIPTION

1.1 Labels commands

Printer serial setup

A command has been created that permits modification of the configuration parameter value. Given below is the description of this command.

GS	n
[Name]	Printer serial set-up
[Format]	ASCII GS LF 2 n m ck Hex 1D FF 0A 32 n m ck Decimal 29 255 10 50 n m ck
[Range]	$0 \leq n \leq 255$ $0 \leq m \leq 255$ $0 \leq ck \leq 255$
[Description]	Printer serial set-up <ul style="list-style-type: none"> • <i>n</i> specifies the parameter to be modified • <i>m</i> specifies the value of the parameter to be modified • <i>ck</i> is the command's 8-bit check sum • All settings made using this command remain in effect even after the printer has been turned off because they are saved in EEPROM • Following the execution of this command, the printer resets for some parameters • The table below summarizes <i>m</i> and <i>n</i> values in relation to the parameter to be modified

n	Parameter	M	Description
1	Paper width	0	112 mm
		1	80 mm
2	Busy Condition	0	RxFull
		1	Offline / RxFull
3	Receive Error	0	Ignored
		1	Print '?'
4	Rx Buffer Dimension	0	16 Bytes
		1	64 Bytes
		2	1K Bytes
		3	4K Bytes
		4	8K Bytes
5	Autofeed	0	CR disabled
		1	CR enabled
6	Panel Key	0	Disabled
		1	Enabled
7	Print Mode	0	Normal
		1	Reverse
8	Char / Inch	0	A=11 B=15 cpi
		1	A=15 B=20 cpi
9	Notch alignment (Note1)	0	Disabled
		1	Enabled

1. COMMANDS DESCRIPTION

10	Notch threshold	0	0.5 V
		1	1.0 V
		2	1.5 V
		3	2.0 V
		4	2.5 V
		5	3.0 V
		6	3.5 V
		7	4.0 V
		8	4.5 V
11	Alignment correction	0	-2.00 mm
		1	-1.75 mm
		2	-1.50 mm
		3	-1.25 mm
		4	-1.00 mm
		5	-0.75 mm
		6	-0.50 mm
		7	-0.25 mm
		8	0.00 mm
		9	+0.25 mm
		10	+0.50 mm
		11	+0.75 mm
		12	+1.00 mm
		13	+1.25 mm
		14	+1.50 mm
		15	+1.75 mm
		16	+2.00 mm
12	Speed / Quality	0	High Quality
		1	Normal
		2	High Speed
13	Current consumption	0	Low
		1	Normal
		2	High
14	Red Printing	0	Disabled
		1	Enabled

1. COMMANDS DESCRIPTION

15	Print Density	0	- 50 %
		1	-37 %
		2	-25 %
		3	-12 %
		4	Normal
		5	+12 %
		6	+25 %
		7	+37 %
		8	+50 %
		9	+62 %
		10	+75 %
		11	+87 %
		12	+100 %
		13	+112 %
		14	+125 %
		15	+137 %
		16	+150 %
16	Paper Type (Note 2)	0	Single form
		1	Label roll
17	Notch distance from edge (Note 3)	n	Given in mm
18	Notch width (Note 4)	n	Given in mm

(Note 1): If the alignment sensor is enabled, sensor calibration is performed automatically.

(Note 2): If the alignment sensor is enabled, the paper support utilized is indicated by the Paper Type (Form/Label) parameter; if it is not enabled, the default setting is for a continuous paper roll.

(Note 3): This parameter may be modified in key set-up, but only if label alignment has been enabled and the single form option is in use. A value table has not been included for this parameter, but there is a routine that permits the complete value (between 0 and 255) to be entered and modified directly from the keys.

(Note 4): This parameter may be modified in key set-up, but only if label alignment has been enabled. A value table has not been included for this parameter, but there is a routine that permits the complete value (between 0 and 255) to be entered and modified directly from the keys. This parameter indicates notch width when pre-printed forms are utilized, and indicates the GAP between successive labels with the “**Paper = LABEL**” setting.

[Example] To set a print density of +125%, send the following commands:

0x1D 0xFF 0x0A 0x32 n m ck

where the value of n is 15 (Print Density given in the table) and the value of m is 14 (+125% in the table). The command thus becomes:

0x1D 0xFF 0x0A 0x32 0x0F 0x0E ck

1. COMMANDS DESCRIPTION

The ck is calculated as the sum of the preceding command

$$Ck = 0x1D + 0xFF + 0x0A + 0x32 + 0x0F + 0x0E = 0x0175$$

The final command is as follows:

0x1D 0xFF 0x0A 0x32 0x0F 0x0E 0x75

Print buffer cancellation

A command has been created that permits the cancellation of the print buffer. This command is performed as soon as it is received and has priority over other commands already stored in the buffer. Buffer cancellation is performed automatically during paper loading.

A description of the command is given below:

DLE DC4 ACK ENQ EOT						
[Name]	Print buffer cancellation					
[Format]	ASCII	DLE	DC4	ACK	ENQ	EOT
	Hex	10	14	06	05	04
	Decimal	16	20	6	5	4
[Description]	Cancels the receive buffer of the communication port and the line buffer. The command is comprised of five characters in order to severely limit the possibility that a random string of bytes could be misconstrued as the buffer reset command.					

Printer functioning with the cover open

When the cover is open, no roller and cutter movement from commands received from the communication port is permitted on the printer. However, roller movement is permitted if activated by pressing the FEED key.

From tests performed regarding possible conflict between “cover open” and “printer buffer full” conditions, it was found that if handshaking is enabled and it is of the same type on both the host and printer, there will not be any data loss. If handshaking is not enabled, when the cover is open the printer buffer could fill up and it is possible that the first data received will be overwritten by the last to be received since circular buffering is used.

UCC/EAN128 Application Identifier

The command for barcode printing has been modified so that the UCC/EAN128 may be printed. The command format is as follows:

GS 'k' m , in which the value of m must be 9 or 74. If m = 9 is transmitted, it must be followed by the data to be codified followed by the stop character 0x00. If, on the other hand, m = 74 is transmitted, it must be followed by the total number of bytes to be codified (n) and then the characters to be codified.

1. COMMANDS DESCRIPTION

For example, to codify the string “CIAO”, either one of the two command strings may be sent:

```
0x1D 0x6B 0x09 0x43 0x49 0x41 0x4F 0x00
```

or

```
0x1D 0x6B 0x4A 0x04 0x43 0x49 0x41 0x4F
```

Please note: The second coding type ($m = 74$) is preferable because the stop character utilized in the first case ($m = 9$) is a character that may be codified in UCC/EAN128 and therefore could give rise to ambiguous situations.

Syntax of the string to be decoded:

This command makes it possible to decode formatted strings to ensure compatibility with the “UCC/EAN128 Application Identifier Standard”.

In this standard, barcodes contain an “Application Identifier” (shown in parentheses within the HRI) followed by a “Data Field” of either fixed or variable length. Therefore, to decode an AI with a fixed-length data field, all that is required is to place it within parentheses (), whereas an AI with a variable-length data field must be enclosed in brackets []. The printer will automatically run the decoder algorithm making provision for the start characters, code changes for decoding, shifts, number compression, stop characters for AIs with variable-length data fields and print the HRI according to UCC/EAN128 standard (for AIs within parentheses and without control characters).

PLEASE NOTE: To shorten barcode length, it is recommended that AIs with variable-length data field be codified in the final part of the barcode. Placing variable data field AIs in other positions does not affect barcode decoding, but it does make it longer.

Example:

If we want to codify the production date (AI = 11) with a fixed data field of 6 characters, the packaging date (AI = 13) also with a fixed field of 6 characters, and we must indicate an Electronic serial number (AI = 8002) with a variable length data field of up to a maximum of 20 characters, we must format the string so that AI = 8002 is in the last position in order to maximize length. The resulting string to be sent will be:

```
(11)YYMMDD(13)YYMMDD[8002]12345
```

If the characters ‘(’, ‘)’, ‘[’ and ‘]’ must also be codified, they must be transmitted twice within the string to be decoded so that they are not confused with the AI identification.

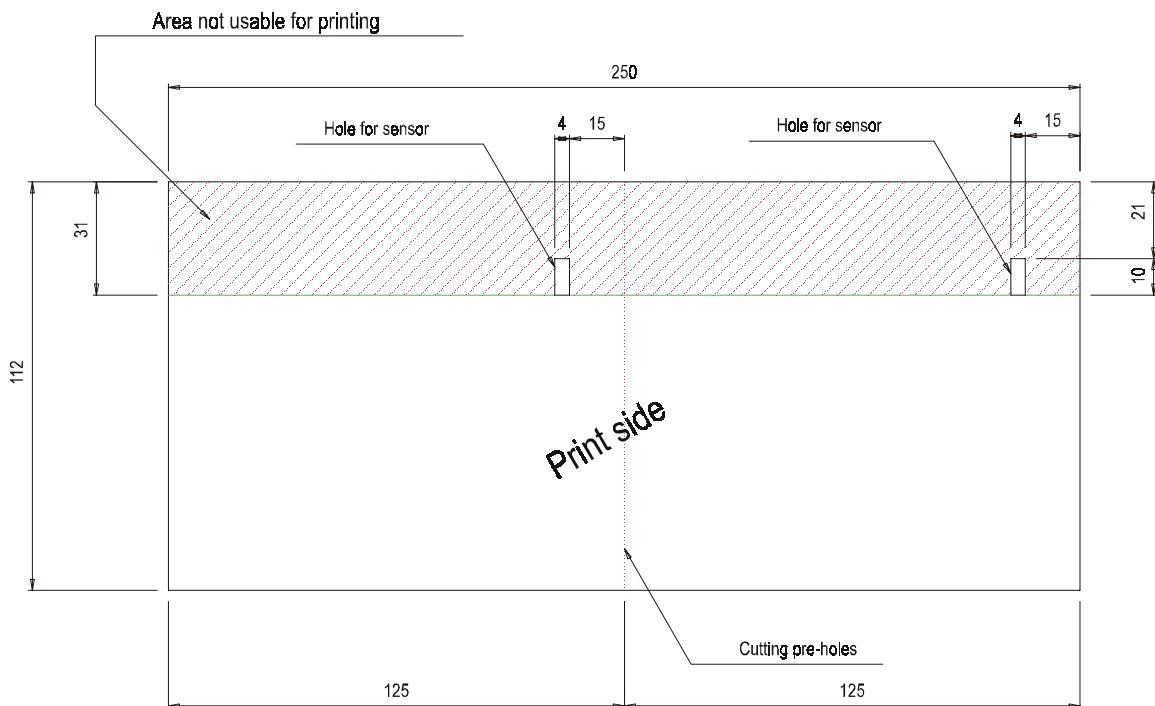
1. COMMANDS DESCRIPTION

For example, if we want to codify the string "(Ciao)" as a fixed length data field in a hypothetical AI codified by the number 45, the command must be formatted as follows:

```
(45)((Ciao))
```

Single form handling

A series of commands for comprehensive handling of single forms has been created. To select the single form option, printer alignment must be enabled ("Notch Align" parameter in set-up) and, using the "Paper Type" parameter, the single form option selected (see, Printer serial set-up). The dimensions of the form used are shown in the figure below.



Label set-up parameters:

For correct label handling, the parameters for notch width and distance from the edge must be entered into the printer. The distance of the notch from the label border (15mm) is used by the printer to determine the correct number of motor steps required to align the label, while the notch width (4mm) makes it possible to detect the paper end from the notch because the same sensor is used for both paper end and alignment.

An alignment correction parameter has also been created ("Alignment Correction" in set-up), that makes it possible to fine-tune alignment ($\pm 2\text{mm}$) and also compensate for slight variations in printer assembly.

These parameters may be changed using the serial set-up command and remain active even after the printer has been turned off.

1. COMMANDS DESCRIPTION

Calibration of the alignment sensor.

Calibration of the alignment sensor is performed automatically when alignment is enabled. If alignment is enabled from key set-up, a receipt containing data regarding calibration is printed out, while if it is enabled from serial set-up, no message is printed but either the character ACK (0x06) is transmitted over the serial line if calibration is performed successfully, or the character NACK (0x15) in the event of calibration error. The values relative to sensor calibration are stored in EEPROM, therefore it is not necessary for the calibration procedure to be run each time the printer is powered-up.

Alignment commands:

There are two label alignment commands: GS 0xF6 and GS 0xF8. The former aligns the label at the first printing line, while the latter aligns the label at the cut. To better illustrate how these two commands function, here is an example: if a string print command is sent following the GS 0xF6 command and if all transmitted parameters are correct, the string will be printed starting from the upper edge of the label, while if a complete cut command (ESC 'i') is sent following the GS 0xF8 command, the label will be cut along the dotted line.

Another command which has been created is that for expelling the form (GS 0xF9): the paper feeds until "paper end" is detected, indicating that the form has been completely ejected .

Label handling

A series of commands have been created to permit the STB112 to handle labels. To select label mode, printer alignment must be enabled ("Notch Align" parameter in set-up) and, using the "Paper Type" parameter, label support selected (see Set-up). The labels that may be used must have a minimum length of 45mm (due to printer mechanics) and minimum width of 80mm. Maximum label width is 112mm and there is no maximum length, although the label search goes into timeout after 160mm of white paper.

From tests performed, the GAP between successive labels may be utilized for alignment, but the black recognition threshold must be lowered to 1.0V (see Set-up). This is because the signal furnished by the sensor peaks at between 1.5V and 2.0V when the sensor is over the GAP, while it is approx. 0.5 volt when covered by the paper.

In this mode, reverse movement of the motor is not recommended as reverse movement distances the label from the sensor, jeopardizing correct functioning during alignment.

1. COMMANDS DESCRIPTION

Label alignment:

For the commands relative to head and cutting alignment, please refer to those described in previous chapters.

Label mode printing:

Once the label has been aligned, it may be printed in one of three modes. The first involves sending the strings of which the label is comprised to the printer in text mode; the second makes use of the graphics page (Page Mode or GAM); the third is a special mode that handles the label to be printed “by field”, offering the possibility of modifying only those printed areas that vary from label to label. This third mode also makes it possible to save and re-call various label styles and has been designed to make printing as fast as possible and reduce the amount of information from host to printer to a minimum.

Label layout handling:

The layout is the label “style” that includes a number of basic elements, called fields. Each field has its own, unique, identification code that defines it within the layout and which is sent when the fields are declared. This ID code is always comprised of a letter indicating the type of field and by a number that must fall between ‘0’ (0x30) and the maximum number of elements of that type. Each layout has four different types of fields which are: text, barcode, box and image.

Type of field	Sample ID		Max. no. fields
TEXT	‘t’	0x74	8
BARCODE	‘b’	0x62	2
BOX	‘x’	0x78	6
IMAGE	‘i’	0x69	1

Defining a label layout:

Each field within a layout must first be defined in order to be used later within a label. For example, if I need to insert a barcode on a label, I must first define a barcode-type field. In addition to the field ID, this definition includes all data that will always remain unchanged from one label to the next. In the specific case of a barcode, the x and y positions for barcode insertion (expressed in dots), as well as the barcode size must be sent. When the time comes to print the barcode within an individual label, all that is required is to use the field selection command, followed by the barcode print command (GS ‘k’ – see SmicePS manual).

1. COMMANDS DESCRIPTION

Saving and re-calling a label layout:

Once a layout has been completed, using the define command, the layout may be saved using the proper command and then re-called when required in order to avoid having to transmit each time the data that does not change between successive labels. Up to five different layouts may be stored in EEPROM.

Table summarizing label handling commands:

LAYOUT HANDLING

GS B0h <num>	Save layout
GS B1h <num>	Re-call layout

FIELD HANDLING

GS B8h <type> ...	Define field
GS B9h <type> ...	Select field
GS BAh <type> ...	Write field
GS BBh <type> ...	Cancel field

LABEL HANDLING

GS BDh	Print label
GS BEh	Cancel label

PREVIOUS USEFUL OR MODIFIED COMMANDS

ESC & I [len] P	Set page length on label
ESC * r B	Print graphic page

IMPORTANT

The parameters of some commands marked <value> utilize the following syntax:

- the value is expressed by a single character (byte) ;
- valid characters are all ASCII from 0 to 255;
- the value coincides with the character ASCII value;
- if the value exceeds the limits, it is ignored.

Example:

a value of 65 is obtained from the character

ASCII	A
Hex	41
Decimal	65

1. COMMANDS DESCRIPTION

The parameters of some commands indicated by [value] utilize the following syntax:

- the value is expressed in decimal format;
- valid characters are digits from '0' to '9' (ASCII from 48 to 57) with the comma as the stop character ',' (ASCII 44);
- following the character sequence that makes up the value, the comma stop character ',' must be inserted;
- if the value exceeds the limits, it is ignored.

Example:

the value 345 is obtained from the following character sequence

ASCII		3	4	5	,
Hex	33	34	35	2C	
Decimal	51	52	53	44	

1. COMMANDS DESCRIPTION

1.2 Layout handling commands

GS B0h <num>					
[Name]	Save layout				
[Format]	ASCII	GS	B0h	<num>	
	Hex	1D	B0	<num>	
	Decimal	29	176	<num>	
[Range]	num = 0..4				
[Description]	Saves current layout in EEPROM as layout num.				
[Notes]	If not saved, the definitions are lost when the printer is turned off.				
[Default]					
[Reference]					
[Example]	To save the current layout as number 1				
	ASCII	GS	B0h	01h	
	Hex	1D	B0	01	
	Decimal	29	176	01	
	Layout number 1 may be re-called using the command <GS B2h>				

GS B1h <num>					
[Name]	Re-call layout				
[Format]	ASCII	GS	B1h	<num>	
	Hex	1D	B1	<num>	
	Decimal	29	177	<num>	
[Range]	num = 0..4				
[Description]	Re-calls layout num from EEPROM as current layout				
[Notes]					
[Default]					
[Reference]					
[Example]	To re-call layout no. 1 as current layout				
	ASCII	GS	B1h	01h	
	Hex	1D	B1	01	
	Decimal	29	177	01	
	The layout maybe modified and stored using command <GS B1h>				

1.3 Field handling commands

GS B8h <type> ...									
[Name]	Define field								
[Format]	ASCII	GS	B8h	<type>	...				
	Hex	1D	B8	<type>	...				
	Decimal	29	184	<type>	...				
	Text field:								
	ASCII	GS	B8h	t	[numT]	[posX]	[posY]	[modeT]	
	Hex	1D	B8	74	[numT]	[posX]	[posY]	[modeT]	
	Decimal	29	184	116	[numT]	[posX]	[posY]	[modeT]	
	Barcode field:								
	ASCII	GS	B8h	b	[numB]	[posX]	[posY]	[heightB]	
	Hex	1D	B8	62	[numB]	[posX]	[posY]	[heightB]	
	Decimal	29	184	98	[numB]	[posX]	[posY]	[heightB]	

1. COMMANDS DESCRIPTION

	Box field: ASCII GS B8h x [numX] [posX][posY][dimX][dimY] [modeX] Hex 1D B8 78 [numX] [posX][posY][dimX][dimY] [modeX] Decimal 29 184 120 [numX] [posX][posY][dimX][dimY] [modeX]
	Image field: ASCII GS B8h i [numI] [posX][posY][dimX][dimY][logo] [OLX] [OLY] Hex 1D B8 69 [numI] [posX][posY][dimX][dimY][logo] [OLX] [OLY] Decimal 29 184 105 [numI] [posX][posY][dimX][dimY][logo] [OLX] [OLY]
[Ranges]	numT = 0..7; numB = 0..1; numX = 0..5; numI = 0 posX = 0..1280; posY = 0..1500; heightB = 1..1500; dimX = 0..1280; dimY = 0..1500; modeT = 0..255; modeX = 0..29; logo = 0..0 OLX = 0.. 1280; OLY = 0.. 1500
[Description]	Defines a field within the current layout. Field definition varies on the basis of the type of field. numT -> text field number numB -> barcode field number numX -> box field number posX -> field horizontal position posY -> field vertical position dimX -> field horizontal dimension dimY -> field vertical dimension modeT -> character print mode (analogous to command ESC/POS <ESC ! n>) modeX -> type of box to be printed: - decimal unit value u: 0 -> cancels box area 1 -> fills box area 1 < u < 9 -> fills box area with pre-set pattern 9 -> leaves box area unchanged (used for borders and rectangles) - decimal tens value d: 0 < d < 10 -> box outline with solid border to thickness d logo -> number of stored logo to be printed OLX -> Horizontal offset position of logo read, expressed in dots. Identifies the horizontal starting point within the logo from which to copy the image inside the label graphic page OLY -> Vertical offset position of logo read, expressed in dots. Identifies the vertical starting point within the logo from which to copy the image inside the label graphic page

1. COMMANDS DESCRIPTION

[Notes]	Text and barcode fields, once defined using this command, are re-called using the <GS B9h> command; following this latter command, the text or barcode may be entered using the applicable ESC/POS command. Box and image fields are inserted directly using the <GS BAh> command. Barcode height can only be fixed in the case of single-dimension barcodes. Image dimensions define the portion of the stored logo to be printed.
[Default]	
[Reference]	
[Examples]	<p>To define a text field</p> <pre> ASCII GS B8h t 1 , 3 2 0 , 1 - > Hex 1D B8 74 31 2C 33 32 30 2C 31 - > Decimal 29 184 116 49 44 51 50 48 44 49 - > ASCII 6 0 , 6 4 , Hex 36 30 2C 36 34 2C Decimal 54 48 44 54 52 44 Text field number 1 in position x = 320 y = 160 and font A italic </pre> <p>To define a barcode field</p> <pre> ASCII GS B8h b 0 , 2 2 0 , 1 - > Hex 1D B8 62 30 2C 32 32 30 2C 31 - > Decimal 29 184 98 40 44 50 50 48 44 49 - > ASCII 0 0 , 1 6 0 , Hex 30 30 2C 31 36 30 2C Decimal 48 48 44 49 54 48 44 Barcode field number 0 in position x = 220 y = 100 with barcode height = 160 (single-dimension barcodes only) </pre> <p>To define a box field</p> <pre> ASCII GS B8h x 1 , 1 0 , 2 0 - > Hex 1D B8 78 31 2C 31 30 2C 32 30 - > Decimal 29 184 120 49 44 49 48 44 50 48 - > ASCII 0 , 8 0 , 5 0 , 3 0 , Hex 30 2C 38 30 2C 35 30 2C 33 30 2C Decimal 48 44 56 48 44 53 48 44 51 48 44 Box field number 1 in position x = 10 y = 200, width = 80 and height = 50; cancel enclosed area and draw border thickness 3 </pre>

1. COMMANDS DESCRIPTION

To define an image field											
ASCII	GS	B8h	i	0	,	1	0	,	1	0	->
Hex	1D	B8	69	30	2C	31	30	2C	31	30	->
Decimal	29	184	105	48	44	49	48	44	49	48	->
ASCII	0	,	8	0	,	5	0	,	0	,	->
Hex	30	2C	38	30	2C	35	30	2C	30	2C	->
Decimal	48	44	56	48	44	53	48	44	48	44	->
ASCII	0	,	8	0	,	5	0	,	0	,	->
Hex	30	2C	38	30	2C	35	30	2C	30	2C	->
Decimal	48	44	56	48	44	53	48	44	48	44	->
ASCII	1	0,	0	,	6	0	,				
Hex	30	2C	38	2C	35	30	2C				
Decimal	48	44	56	44	53	48	44				
Image field number 0 in position x = 10 y = 100, width = 80 and height = 50; write logo number 0, starting from the position x = 100, y = 60 from within the logo. This type of image handling makes it possible to create a logo as a grouping of a number of smaller images stored in different positions that may then be re-called within the label as if they were a number of separate images.											

GS B9h <type> <num>					
[Name]	Select field				
[Format]	ASCII	GS	B9h	<type>	<num>
	Hex	1D	B9	<type>	<num>
	Decimal	29	185	<type>	<num>
	Text field:				
	ASCII	GS	B9h	t	<numT>
	Hex	1D	B9	74	<numT>
	Decimal	29	185	116	<numT>
	Barcode field:				
	ASCII	GS	B9h	b	<numB>
	Hex	1D	B9	62	<numB>
	Decimal	29	185	98	<numB>
[Ranges]	numT = 0..7; numB = 0..1				
[Description]	Selects a field within the current layout. The next text or barcode will utilize the position and characteristics set during field definition using command <GS B8h> numT -> text field number numB -> barcode field number				
[Notes]	If the field was not defined, current settings will not be changed.				
[Default]					
[Reference]					
[Examples]	To write a text field				
	ASCII	GS	B9h	t	1
	Hex	1D	B9	74	31
	Decimal	29	185	116	49
	Writes the string 'Prova' in the position and with the characteristics set in the definition of text field number 1 of the current layout				

1. COMMANDS DESCRIPTION

	To write a barcode field											
ASCII	GS	B9h	b	1	,	GS	k	06h	1	2	-	
>												
Hex	1D	B9	62	31	2C	1D	6B	06	31	32	-	
>												
Decimal	29	185	98	49	44	29	107	06	49	50	-	
>												
ASCII	3	4	5	0h								
Hex	33	34	35	0								
Decimal	51	52	53	0								
Writes the barcode (type code32 value 12345) in the position and with the characteristics set in the definition of barcode field number 1 of the current layout												

GS BAh <type> <num>					
[Name]	Write field				
[Format]	Box field:				
	ASCII	GS	BAh	x	<numX>
	Hex	1D	BA	78	<numX>
	Decimal	29	186	120	<numX>
	Image field:				
	ASCII	GS	BAh	i	<numI>
	Hex	1D	BA	69	<numI>
Decimal	29	186	105	<numI>	
[Range]	numX = 0..3 numI = 0				
[Description]	Writes a field within the current layout. The box or image will utilize the position and characteristics set during field definition using command <GS B8h> numX -> box field number numI -> image field number				
[Notes]	If the field was not defined, nothing will be written.				
[Default]					
[Reference]					
[Example]	To write a box field				
	ASCII	GS	BAh	x	2 ,
	Hex	1D	BA	78	32 2C
	Decimal	29	186	120	50 44
	Writes the box in the position and with the characteristics entered in the definition of box field number 2 of the current layout.				
	To write an image field				
	ASCII	GS	BAh	i	0 ,
	Hex	1D	BA	69	30 2C
	Decimal	29	186	105	48 44
	Writes the image in the position and with the characteristics entered in the definition of image field number 0 of the current layout.				

1. COMMANDS DESCRIPTION

GS BBh <type> <num>						
[Name]	Cancel field					
[Format]	ASCII	GS	BBh	<type >	<num>	
	Hex	1D	BB	<type >	<num>	
	Decimal	29	187	<type >	<num>	
	Text field:					
	ASCII	GS	BBh	t	<numT>	
	Hex	1D	BB	74	<numT>	
	Decimal	29	187	116	<numT>	
	Barcode field:					
	ASCII	GS	BBh	b	<numB>	
	Hex	1D	BB	62	<numB>	
	Decimal	29	187	98	<numB>	
	Box field:					
	ASCII	GS	BBh	x	<numX>	
	Hex	1D	BB	78	<numX>	
	Decimal	29	187	120	<numX>	
	Image field:					
	ASCII	GS	BBh	i	<numI>	
	Hex	1D	BB	69	<numI>	
	Decimal	29	187	105	<numI>	
[Ranges]	numT = 0..7; numB = 0..1; numX = 0..5					
[Description]	Eliminates a field within the current layout. numT -> text field number numB -> barcode field number numX -> box field number numI -> image field number					
[Notes]						
[Default]						
[Reference]						
[Examples]	To cancel a previously-defined text field ASCII GS BBh t 1 , Hex 1D BB 74 31 2C Decimal 29 187 116 49 44 Cancels text field number 1 of the current layout To cancel a previously-defined barcode field ASCII GS BBh b 0 , Hex 1D BB 62 30 2C Decimal 29 187 98 48 44 Cancels barcode field number 0 of the current layout To cancel a previously-defined box field ASCII GS BBh x 1 , Hex 1D BB 78 31 2C Decimal 29 187 120 49 44 Cancels box field number 1 of the current layout					

1. COMMANDS DESCRIPTION

	To cancel a previously-defined image field					
ASCII	GS	BBh	i	0	,	
Hex	1D	BB	69	30	2C	
Decimal	29	187	105	48	44	
	Cancels image field number 0 of the current layout					

1.4 Label handling commands

GS BDh	
[Name]	Print label
[Format]	ASCII GS BDh Hex 1D BD Decimal 29 189
[Range]	
[Description]	Prints the label and does not cancel the graphic page
[Notes]	The ESC *rB command prints the label and cancels it from the memory; the command described here retains the just-printed label and allows changes to be made in it. This procedure can be useful when there are only slight variations from label to label. The GS BEh command cancels the label.
[Default]	
[Reference]	ESC *rB, GS BEh
[Example]	To print the label in memory ASCII GS F9h Hex 1D F9 Decimal 29 249 Once printing has finished, the label in memory may be modified, for example a box may be cancelled and text inserted.

GS BEh	
[Name]	Cancel label
[Format]	ASCII GS BEh Hex 1D BE Decimal 29 190
[Range]	
[Description]	Cancels the label currently stored in memory. May be useful when the next label is very different from the current one and it is not worthwhile making changes to it.
[Notes]	The previous use of the command is retained
[Default]	
[Reference]	GS BDh

1. COMMANDS DESCRIPTION

[Example]	Further use of the current label is not required
	ASCII GS BEh Hex 1D BE Decimal 29 190
	The print buffer is cleared and a new label may be entered

GS F6h	
[Name]	Print alignment
[Format]	ASCII GS FAh Hex 1D FA Decimal 29 250
[Range]	
[Description]	Aligns the next label at the first line of printing
[Notes]	
[Default]	
[Reference]	
[Example]	To request alignment ASCII GS F6h Hex 1D F6 Decimal 29 246 The printer is positioned on the first line of printing of the next label

GS F8h	
[Name]	Cutting and print alignment
[Format]	ASCII GS F8h Hex 1D F8 Decimal 29 248
[Range]	
[Description]	Cuts the current label at the bottom and aligns the next label at the first line of printing
[Notes]	
[Default]	
[Reference]	
[Example]	When printing has finished, to request cutting and successive alignment ASCII GS F8h Hex 1D F8 Decimal 29 248 The printer places the label in cutting position, makes the cut and then positions itself on the first printing line of the next label

1. COMMANDS DESCRIPTION

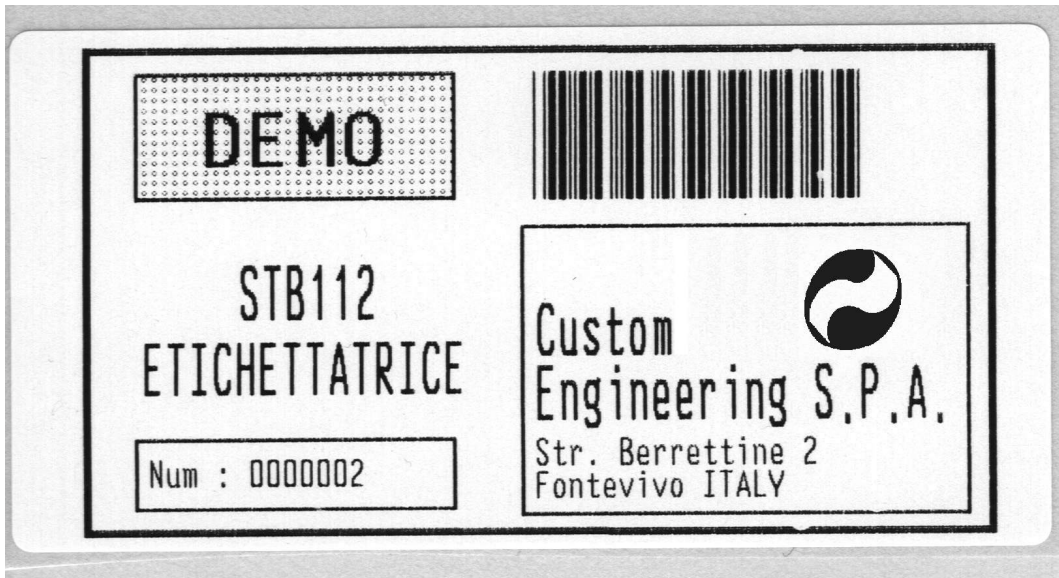
1.5 Previous commands that may be useful

ESC & I [len] P								
[Name]	Sets page length on label							
[Format]	ASCII	ESC	&	I	[len]	P		
	Hex	1B	26	6C	[len]	50		
	Decimal	27	38	108	[len]	80		
[Range]	20 < len < 2320							
[Description]	Sets the number of printable graphic lines on a label. If the current label length is shorter, only the number of graphic lines actually available on the label are printed.							
[Notes]	This command is useful when a print length shorter than the current label length is desired							
[Default]								
[Reference]	ESC & I [len] R							
[Example]	To limit the length of the printable page on the label to 50 mm (400 graphic lines)							
	ASCII	ESC	&	I	4	0	0	P
	Hex	1B	26	6C	34	30	30	50
	Decimal	27	38	108	52	48	48	80
	The print buffer is cleared and a new label may be entered							

ESC * r B					
[Name]	Print graphic page				
[Format]	ASCII	ESC	*	r	B
	Hex	1B	2A	72	42
	Decimal	27	42	114	66
[Range]					
[Description]	Prints the label and then cancels it				
[Notes]	The GS F9 command prints the label and does not cancel it from memory; this command cancels the just-printed label. This may be useful when there are significant variations from one label to the next. Page length may not be zero				
[Default]					
[Reference]					
[Example]	To print the label stored in memory				
	ASCII	ESC	*	r	B
	Hex	1B	2A	72	42
	Decimal	27	42	114	66
Following printing, the label is cancelled from memory and must be completely re-entered.					

1. COMMANDS DESCRIPTION

1.6 Example of a Label handling mode application



Let's assume that we want to create a label like the one shown here. It is comprised of 4 box fields, 8 text fields, 1 barcode field and 1 image field. The first step is to define the various fields that make up the label using the definition command—given below is the list of the commands to be sent to define the label. **Note:** The characters enclosed between single quote marks (') are expressed in ASCII, while the numbers separated by commas are in hexadecimal format. Everything that appears following the semicolon (;) is considered a comment.

; Definition of BOX fields

; See Note 1.

```
1D,B8,'x0,40,16,710,380,69;',           ; box 0 : external border
1D,B8,'x1,80,36,250,100,38;',           ; box 1 : upper left
1D,B8,'x2,380,156,350,225,39;',         ; box 2 : address box
1D,B8,'x3,80,320,250,56,30;',           ; box 3 : serial number box
```

; Definition of image field

; See Note 2.

```
1D,B8,'i0,530,160,100,100,00,723,523;', ; Image 0
```

; Definition of text fields

; See Note 3.

```
1D,B8,'t0,130,66,56;',                 ; text 0
1D,B8,'t1,160,186,24;',                 ; text 1
1D,B8,'t2,85,246,24;',                 ; text 2
1D,B8,'t3,90,336,01;',                 ; text 3
1D,B8,'t4,390,216,24;',                 ; text 3
1D,B8,'t5,390,266,24;',                 ; text 4
1D,B8,'t6,390,321,01;',                 ; text 5
1D,B8,'t7,390,346,01;',                 ; text 6
```

1. COMMANDS DESCRIPTION

; Definition of Barcode field

; See Note 4.

1D,B8,'b0,390,36,100;',

; Defines Barcode field 0

; Save current Layout

; See Note 5.

1D,B0,01,

; saves current layout as number 1

Notes:

1. The type of fill used for BOX fields can also be specified. In this example, in box 1 we have chosen a pre-set fill pattern (see command description), while for boxes 0 and 2, the inside of the box has been left blank once drawn. For box 3, we have established that the area inside the box must be cancelled before it is drawn. During label handling, this will allow us to redraw just the contents of this box while leaving the rest of the label unchanged.
2. The image field allows us to insert within the label a portion of the logo stored in flash memory. In this way, the logo may be stored as part of a group of smaller graphic images and then, using this command, be inserted within the graphic page. In this example, it has been established that the image must be inserted within the label at position 580,171, that it must have a horizontal dimension equal to 100 dots and a vertical dimension equal to 100 dots, that the logo must be number zero (this printer has just one logo) and that it must start to copy the logo image starting with point 370,170.
3. The text fields require the sending of the position inside the layout and the character print mode; for more information on print mode, refer to command **ESC `!`**)
4. Barcode fields require the sending of the start position within the label and the dimension of the barcode in dots; in this example it is 12.5mm high.
5. Once the layout has been defined it may be saved and then re-called for use without all this data being resent. It is not mandatory that layout data be saved once it has been defined, but if it is not, the label data sent is lost when the printer is turned off.

At this point the various fields must be filled with specific values to be printed on each label and the boxes and images previously defined be drawn. Given below is an example of how to print a label.

1B,'&l400P',

; Page length = 50mm [400 dots]

1D,B1,01,

; load the layout into the current layout

; stored in address 1

1D,BE,

; cancel previous label

; insert box

1D,BA,'x0;',

; Draw box 0

1D,BA,'x1;',

; Draw box 1

1D,BA,'x2;',

; Draw box 2

1D,BA,'x3;',

; Draw box 3

1. COMMANDS DESCRIPTION

```
; insert image
1D,BA,'i0;',           ; Draw image 0

; insert text
1D,B9,'t0;DEMO',       ; text 0
1D,B9,'t1;SMICE-LP4',   ; text 1
1D,B9,'t2;ETICHETTATRICE', ; text 2
1D,B9,'t3;Num : 0000001', ; text 3
1D,B9,'t4;Custom',      ; text 4
1D,B9,'t5;Engineering S.P.A.', ; text 5
1D,B9,'t6;Str. Berrettine 2', ; text 6
1D,B9,'t7;Fontevivo ITALY', ; text 7

; insert barcode
1D,B9,'b0;',           ; Select barcode 0
1D,'w',02,             ; select barcode width
1D,'k',04,'STB112',00, ; select barcode CODE39

1D,F6,                 ; align label
1D,BD,                 ; print label (does not cancel)

1D,BA,'x3;',           ; Draw box 3
1D,B9,'t3;Num : 0000002', ; Rewrite only text 3

1D,F6,                 ; align label
1D,BD,                 ; print label (does not cancel)
```

Notes:

1. As can be seen, the first instruction selects the graphic page length that must correspond to the label length. This serves to provide the print command with the length of the label to be printed.
2. The commands for inserting the boxes and image are very simple in that they require only the box or image ID number. All other data relating to the box and image to be drawn are taken from the box and image definition.
3. For text, in addition to the text itself to be inserted, ESC/POS commands designating print mode but not included in the label definition may also be inserted (see, barcode width selection in the barcode print example).
4. When entering the barcode, the complete ESC/POS command must be entered in order to print the barcode, as described in the manual and in previous pages (e.g., UCC/EAN128).
5. In this example, the label is first aligned and then printed using the non-cancelling command; following this just one box and one text element are redrawn. This makes it possible to change only those elements that vary from label to label. However, if it had been necessary to cancel the printed label, the **1B, '*rB'** command would have had to have been used instead of **1D,BD**.

2. TECHNICAL SPECIFICATIONS

2.1 Technical specifications

In the following table are listed the main technical features of the printer.

(Tab. 2.1)

Printing Method	Thermal line printing
Resolution	203 dpi
Printing Speed	70 mm/s High Quality 90 mm/s Normal >110 mm/s High Speed
Character Set	ASCII standard, International
Type Styles	Normal, 1x a 8x height and width, reverse, underscored, italic, bold
Printing Direction	Straight, 90°, 180°, 270°
Label Width	112 mm 80 mm
Roll Dimension	80 mm max. \pm 5 mm
Emulation	ESC/POS™
Interface	RS 232, Centronics
Baud Rate	From 1.200 BPS To 115.200 BPS
Data Buffer	8 Kb
Flash Memory	256 Kb
Graphic Memory	1 logo da 832 x 630 DOTS
Drivers	Windows™ 95, 98, me, NT4, 2K, Linux
Software Tools	"Easy Label" for Win98/2k
Power Consumption	1,5 A
Power	100÷240 Vac \pm 10% 50 ÷60 Hz
DKD Function	2 drivers
Reliability MTBF	50 Km
Safety	EN60950 +A1 +A2 +A3 +A4
Dimension	242 x 186 x h 132 mm
Options	RS485, RS422

2. TECHNICAL SPECIFICATIONS

2.2 Dimensions

The dimensions of the SMICE-LP4 printer are shown in the figure below.

(Dimensions in mm)

(Fig. 2.1)

