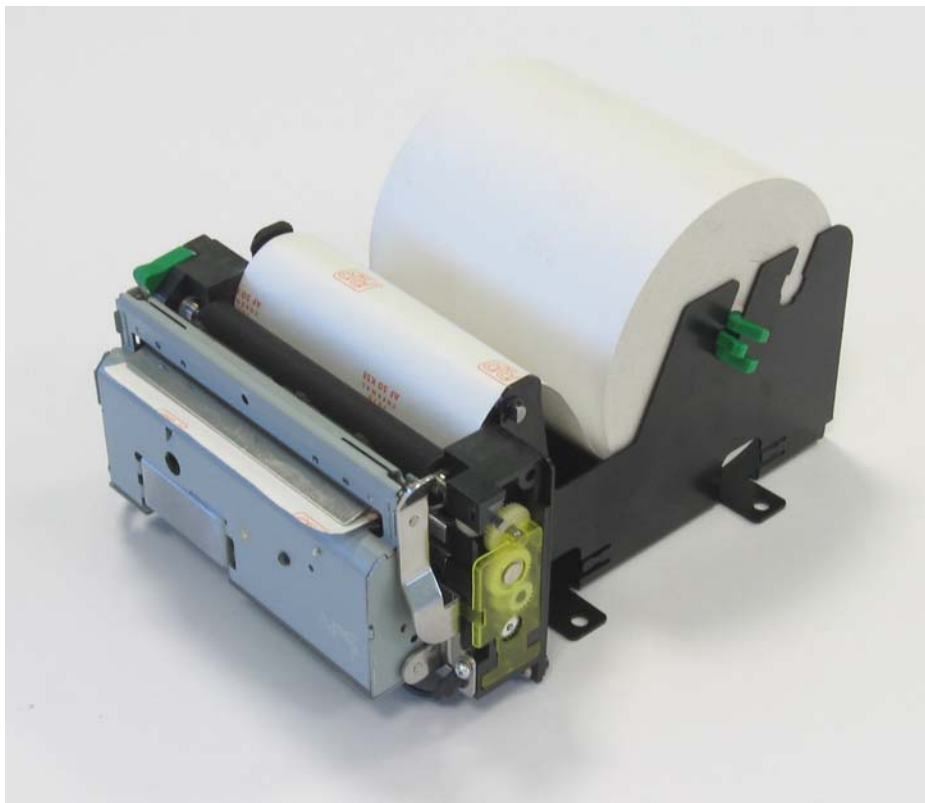


# KM324-HRS-E

Technical reference



KM324-HRS-E-FL

## 1. TABLE OF CONTENTS

Front page.....	1
1. TABLE OF CONTENTS .....	2
2. REVISION HYSTORY .....	3
3. GENERAL FEATURES .....	4
4. GENERAL SPECIFICATION .....	5
5. PRINTER DEVICE INTERCONNECTIONS .....	6
5.1 Printer connectors table .....	6
5.2 Connectors location .....	7
5.3 Connectors location (continued).....	8
5.4 Connectors location (continued).....	9
5.5 Power supply connector .....	10
5.6 RS232 communication connector.....	10
5.7 USB communication connectors .....	11
5.8 RS232 / USB mode selection .....	11
5.9 Near end of paper sensor connector (NEOP) .....	12
5.10 Switches/led connector .....	13
6. SWITCHES AND LED FUNCTIONS .....	14
7. KEYBOARD KIT (Option) .....	15
7.1 Keyboard structure .....	15
7.2 Keyboard specification.....	16
8. PRINTER DEVICE OPERATIONS .....	17
8.1 Self-test mode.....	17
8.2 Paper loading.....	18
8.3 Text printing fonts .....	19
8.4 Control codes.....	21
8.4.1 Control codes cross reference.....	21
8.4.2 Setup and hardware control codes.....	24
8.4.3 Text and general control codes.....	34
8.4.4 Graphic control codes.....	39
8.4.5 Cutter control codes.....	42
8.4.6 Bar code control codes .....	43
8.4.7 Hole / black mark detection control codes .....	47
8.5 Real time control codes .....	51
8.5.1 RS232 .....	51
8.5.2 USB .....	51
9. ADDITIONAL INFORMATION .....	54
9.1 USB .....	54
9.2 Firmware updating .....	54
9.3 Font management .....	55
10. MECHANICAL & HOUSING.....	56
10.1 Mounting holes.....	56
10.2 Mounting position .....	57
10.3 Paper roll position .....	58
10.4 Paper roll position (continued) .....	59
10.5 How to remove the roll holder correctly .....	60
11. ENVIRONMENTAL PROTECTION.....	61
12. ORDERING CODES .....	61



<http://www.aps-printers.com/>

This manual provides complete information about A.P.S KM324-HRS-E printer module.

A.P.S reserves the right to make changes without notice to the product to improve reliability, function or design.  
A.P.S does not assume any liability arising out of the application or use of the product or circuit described herein.

## 2. REVISION HYSTORY

REV.	DATE	PAGE	REVISION ITEM	AUTHOR
Preliminary	27-JUL-2007	-	Firmware revision: <b>CP324-HRS 0.13</b>	AF
A	18-JULY-2008	-	Document issue.	AF

## 3. GENERAL FEATURES

The KM324-HRS-E is a 3 inch, 24Volts Kiosk module with an integrated controller board using RS232 and USB communications, it consists of a CP324-HRS printer mounted on a sheet metal paper holder, it is also provided with a NEOP (near end of paper sensor) and a guillotine cutter.

- **Two versions available, Front paper loading (FL) and Bottom paper loading (BL)**
- **Fully hot pluggable printer**
- **Single power supply voltage ranges from 18V to 27V**
- **Software programmable consumption (dynamic division)**
- **Full control over printing quality/speed (up to 120mm/s)**
- **Speed clamping via control code.**
- **2 communication ports**
- **RS232 (speed up to 115200 bauds),**
- **USB 2.0 (full-speed).**
- **Near End of Paper opto-sensor**
- **Cutter (Guillotine type).**
- **Three internal fonts**
- **Easy font update.**
- **Powerful text printing modes**
- **Horizontal (normal or rotated 180°),**
- **Double and quadruple width and height,**
- **Underlining,**
- **Inverse video,**
- **3 justifications (centred, right, left).**
- **Powerful graphic modes**
- **Variable width and offset,**
- **Double width and height.**
- **11 Barcodes**
- **Normal and rotated 90° for 1D bar codes, PDF417 2D bar code.**
- **Automatic paper loading**
- **Pause between detection of paper and printer beginning to load.**
- **Hole / black mark detection**
- **Setup parameters saved in flash are recovered at next power-up**
- **One simple command saves all important parameters.**
- **Easy firmware upgrades**
- **Windows and Linux drivers available**
- **Option External pluggable switches and LED (Keyboard KIT)**

## 4. GENERAL SPECIFICATION

ITEM		SPECIFICATION	
Printing method		Thermal dot-line printing	
Dimensions WxDxH (mm)		See Overall dimensions drawing attached	
Total number of dots		576	
Average dot resistance (ohm)		1500	
Dot density (dots/mm)		8	
Paper width (mm)		82.5 +0/-1	
Printing width (mm) (centred on paper)		72	
Heat element pitch (mm)		0.125	
Paper feed pitch (mm)		0.125	
Paper feed tension (g)		50 or more	
Paper hold tension (g)		80 or more	
Paper roll diameter (mm)		Max. 100	
Paper roll core diameter (mm)		13 ±0.5 - Core material must be plastic only - Carton or other materials are not suitable	
Recommended paper	KM324-HRS-E-FL	JUJO-AF 50 KSE or equivalent JUJO-AF 50 KSE3 or equivalent KANZAN-KPW460 or equivalent	Max. paper thickness allowed= 82 µm
	KM324-HRS-E-BL	JUJO-AF 50 KSE or equivalent JUJO-AF 50 KSE3 or equivalent KANZAN-KPW460 or equivalent KOEHLER-KT100P or equivalent	Max. paper thickness allowed= 110 µm
Voltage range (Volt)		From 18 to 27	
Current consumption (Amps) (Max instantaneous value @ 24V) <i>Starting value is when only paper feeding</i>		from 1 to 10.5 <i>(Can be limited with dynamic division : 3.3A is the default value.</i>	
Operating temperature (°C)		From -20 to +70	
Operating humidity (RH%)		From 20 to 85 (no condensation)	
Storage temperature (°C)		From -40 to +85	
Storage humidity (RH%)		From 10 to 90 (no condensation)	
EMC standard		Designed to comply with FCC/CE Class B	
<b>Printer life</b>			
	<b>Durability</b>	<b>Basic conditions</b>	<b>Maximum variations</b>
Thermal head pulse resistance	100 million pulses	<ul style="list-style-type: none"> <li>Room temp.: 20 ÷ 25 °C</li> <li>Head temp.: 65 °C max.</li> <li>Rated energy</li> </ul>	Max. 15% in resistance value (Ω) of any dot, from its initial value
Abrasion/wear resistance (TPH)	50 km of paper		
Cutter life (cuts)	≥ 500.000 (max. 100g/m <sup>2</sup> paper weight)		

## 5. PRINTER DEVICE INTERCONNECTIONS

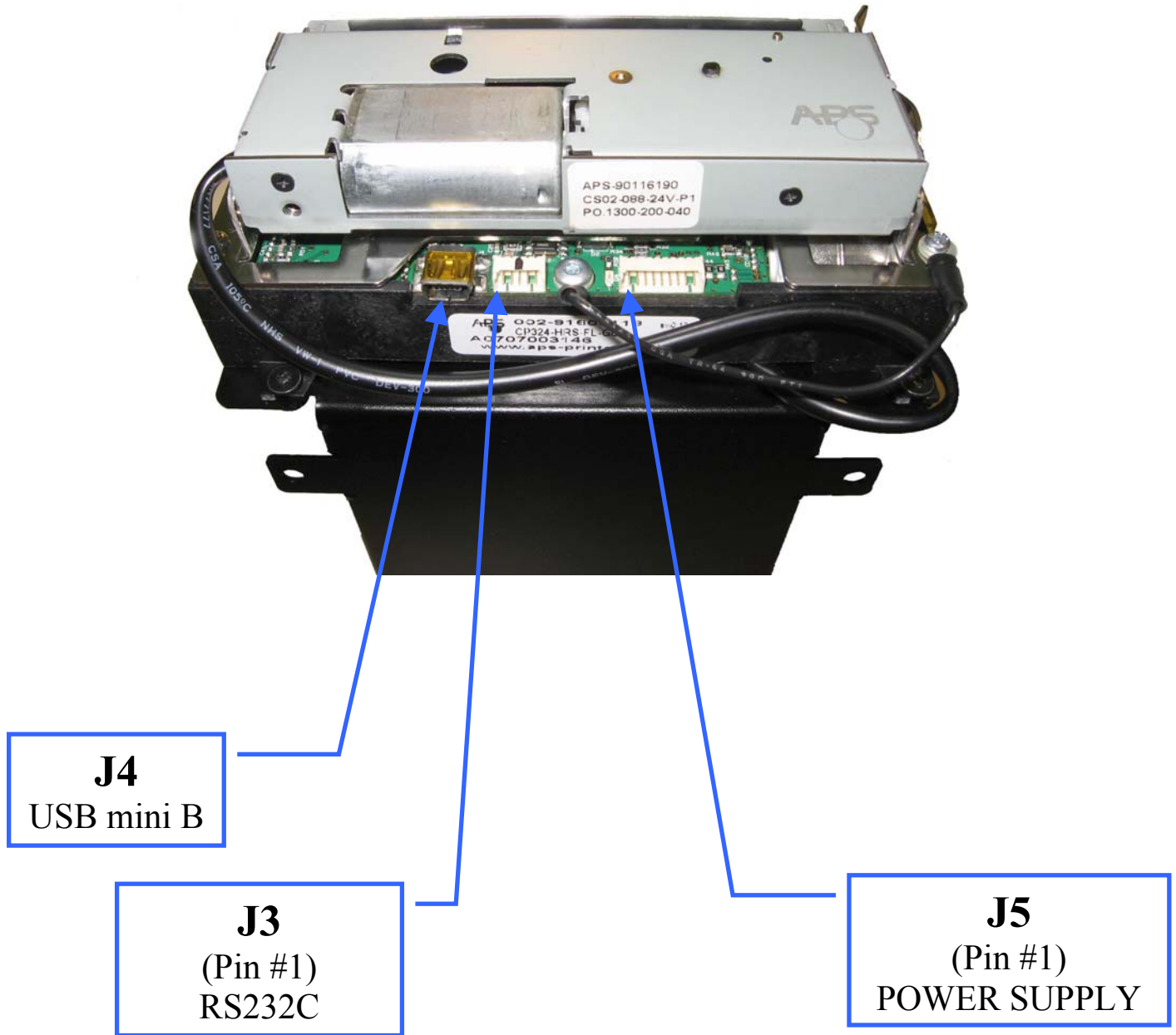
The printer is fully hot pluggable : any connector hereafter can be connected or disconnected without damaging the printer.

### 5.1 Printer connectors table

PRINTER DEVICE INTERCONNECTIONS	
CONNECTOR	FUNCTION
J3	RS232C COMMUNICATION
J4	USB COMMUNICATION (MINI B)
J5	POWER SUPPLY
J8	SWITCHES/LED (Keyboard)
J12	NEAR END OF PAPER SENSOR (already connected)
J13	USB COMMUNICATION (MOLEX)
J2	(for future use)
J14	

## 5.2 Connectors location

The following image shows the Connectors location and their identification.



### 5.3 Connectors location (continued)

To have access to the others connectors that are located in the bottom side of the printer, it is necessary to dismount the printer from the metal housing.

To dismount the printer, unscrew the 4 screws that are used to fix the printer, connect the necessary cables, mount again the printer by using the same screws.

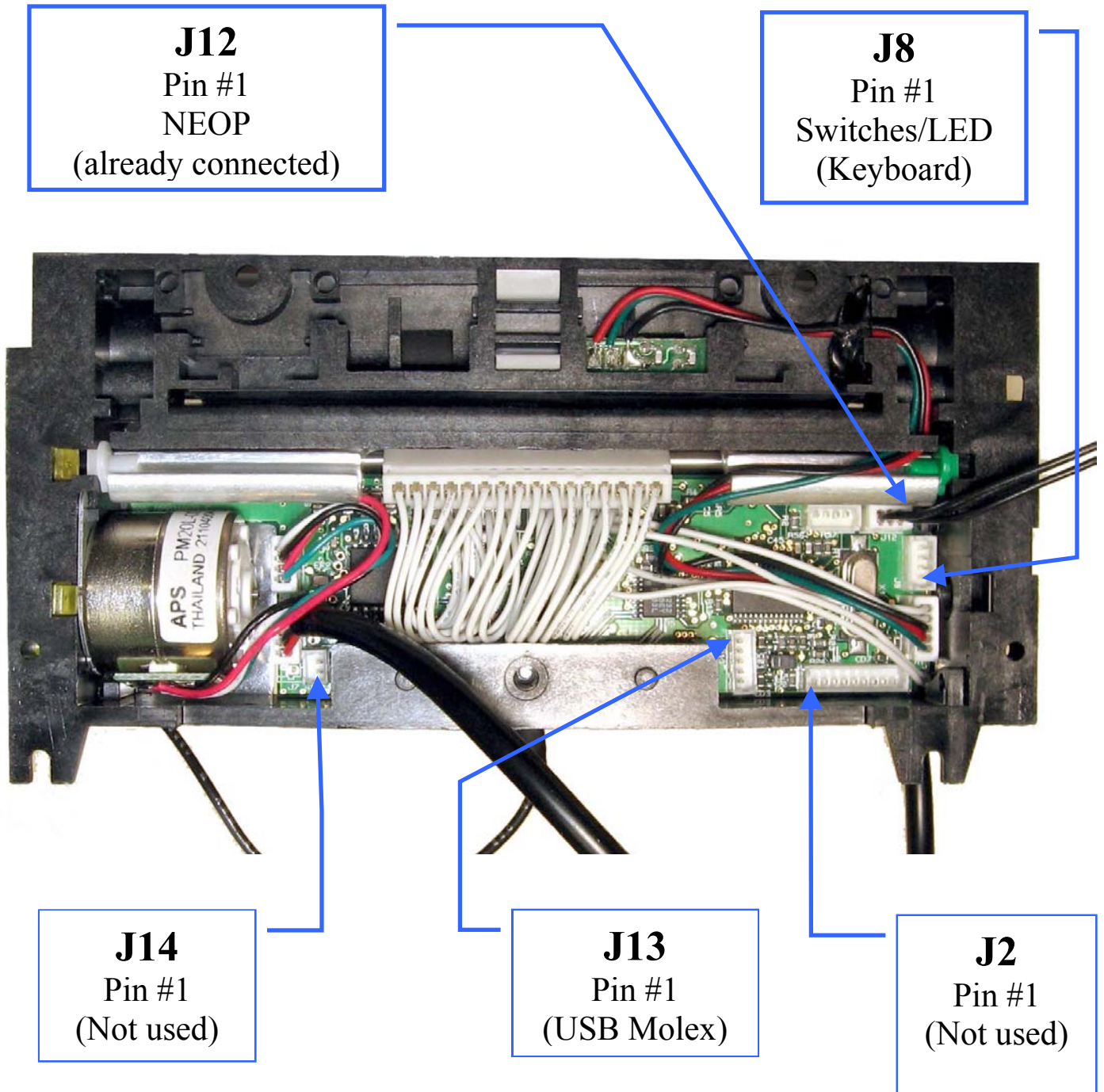
It is necessary to screw the 4 screws with a torque-meter screwdriver, tightening torque for screws is 30 Ncm  $\pm 5$ , the seat of screws is PH1.





## 5.4 Connectors location (continued)

The image shows the bottom side of the printer, once it has been dismantled from the metal housing.



## 5.5 Power supply connector

**Connector J5:** MOLEX 53048 series, 9 contacts (compatible with MOLEX female 51021 series).

PIN NUMBER	SIGNAL NAME
1	GND
2	GND
3	GND
4	GND
5	GND
6	VH
7	VH
8	VH
9	Not connected

**IMPORTANT NOTES:** - wires AWG28 must be used in order to avoid current losses,  
 - power supply (VH) should range from 18V to 27V DC.

## 5.6 RS232 communication connector

**Connector J3:** MOLEX 53048 series, 5 contacts (compatible with MOLEX female 51021 series).

PIN NUMBER	SIGNAL NAME
1	GND
2	Transmit data (TxD, printer output)
3	Receive data (RxD, printer input)
4	CTS or DSR (printer handshaking input, only managed during firmware updating)
5	RTS or DTR (printer handshaking output)

Couple of signals from pins 4 and 5 should be considered as “CTS and RTS” or “DSR and DTR” in function of host handshaking implementation. They are basically managed only in hardware handshaking mode. When printer FIFO is full, RTS (or DTR) signal is set to logical level “1” to point out to host that sending should be stopped otherwise data will be lost (handshaking is OFF). As soon as this signal returns to logical level “0” (free space in printer FIFO), host sending can resume (handshaking is ON).

**IMPORTANT NOTE:** during firmware updating, handshaking is automatically set in hardware mode and so all signals should be connected.

## 5.7 USB communication connectors

There are 2 possibilities:

- **Connector J4:** MOLEX 54819-0578, **standard** mini-B receptacle (compatible with MOLEX mini-B plug),
- **Connector J13:** MOLEX 53047 series, 5 contacts, **customized** connector useful when spacing constraint (compatible with MOLEX female 51021 series). However, this connector has not been designed with taking care EMC issue. So, EMC behaviour may be affected when USB communication uses this way. Please contact APS for further information.

PIN NUMBER	SIGNAL NAME
1	Vbus
2	D-
3	D+
4	GND
5	Shield

Connectors J4 and J13 are wired in parallel. Both connectors are bound to the same USB interface; consequently, both connectors cannot be used at the same time.

The USB LED indicates the state of the USB communication interface. After power-on/plug-in, during enumeration step, this LED should blink and then remains on. Once initialized, this LED then indicates the activity on the USB communication link by blinking at each data transmission.

## 5.8 RS232 / USB mode selection

Only one communication mode (RS232 or USB) can be active at a time. It is set automatically by printer as soon as first character is received. At power-up, both RS232 and USB communication lines are ready to work. The first used communication line becomes active until next power-down. Therefore, if first character is received on RS232 port, then RS232 becomes the active communication line and then all characters sent via USB will be lost. In addition, this first character will be interpreted as any other incoming byte into the printer.

## 5.9 Near end of paper sensor connector (NEOP)

**Connector J12:** MOLEX 53047 series, 3 contacts (compatible with MOLEX female 51021 series). The KM324-HRS-E is provided of NEOP, the **J12** connector is already occupied by the NEOP connection .

PIN NUMBER	SIGNAL NAME
1	LED
2	OPTO
3	GND

### IMPORTANT NOTE

Due to variations in optosensor specifications, type of paper, and different mounting positions, it will generally be necessary to perform a calibration procedure. This is very straightforward, as the firmware has a command that will automatically perform a reflection measurement, determine an appropriate threshold, and permanently store it in the flash memory of the microcontroller. This is normally done only once in production. The application software can query for the “Near End of Paper” status, and the firmware will respond with a single yes/no answer.

Please refer to the “Near End of Paper” control codes (“ESC n ...”) in section 6.4.2 for more information.

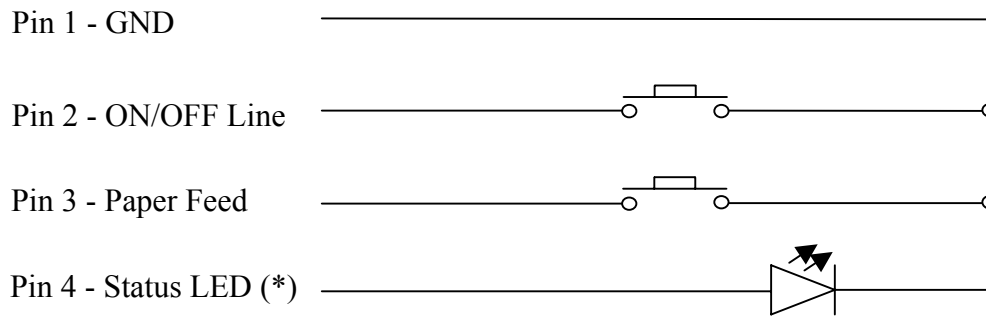
## 5.10 Switches/led connector

**Connector J8:** MOLEX 53047 series, 4 contacts (compatible with MOLEX female 51021 series).

PIN NUMBER	SIGNAL NAME
1	GND
2	ON/OFF Line switch
3	Paper Feed switch
4	Status LED (anode)

This connector allows you to design a remote interface “Paper Feed” button, “ON/OFF Line” button and “Status” LED or connect the Optional APS Keyboard.

External circuitry is as following:



(\*) A serial resistor (270 ohms) is on the printer, setting the LED current at about 5mA. In addition, an on-board status led is also present near power supply connector of the printer.

## 6. SWITCHES AND LED FUNCTIONS

The following table describes the switches and LED functions

Printer status	OFF	OFF Line	ON Line	Head-up or Cutter error	End of paper or Mark error	Over/under voltage or temperature
ON/OFF Line switch	-	ON Line	OFF Line	-		
		cf. remarks (self-test)				
Paper Feed switch	-	feeds paper	feeds paper if not already printing	-		
Status LED (cf. remarks)	OFF	1 flash "ON"	always "ON"	2 flashes "ON"	3 flashes "ON"	4 flashes "ON"

### Remarks :

- When the printer is powered on, a self-test can be run at any time by making following switches sequence (in less of 5s):

*Push on "ON/OFF Line" button and stay pressed on it*

*then*

*Push twice on "Paper Feed" button*

- Status LED is normally always ON. When an error is detected, it is blinking following a specific pattern which is function of error. Periodicity of pattern is 1,024s. Flashes last 128ms. Time between two successive flashes is also 128ms.

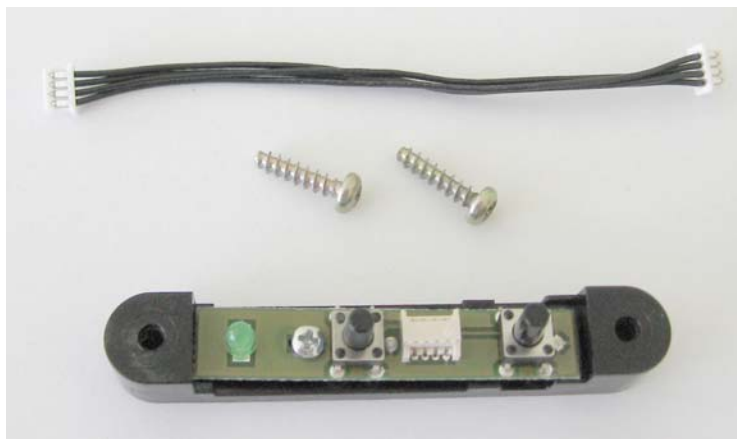
## 7. KEYBOARD KIT (OPTION)

The **KM324-HRS-E** is available with a small expansion Keyboard KIT (optional), supplied separately. It can be connected to the Printer elec. Board by means of a own connecting cable.

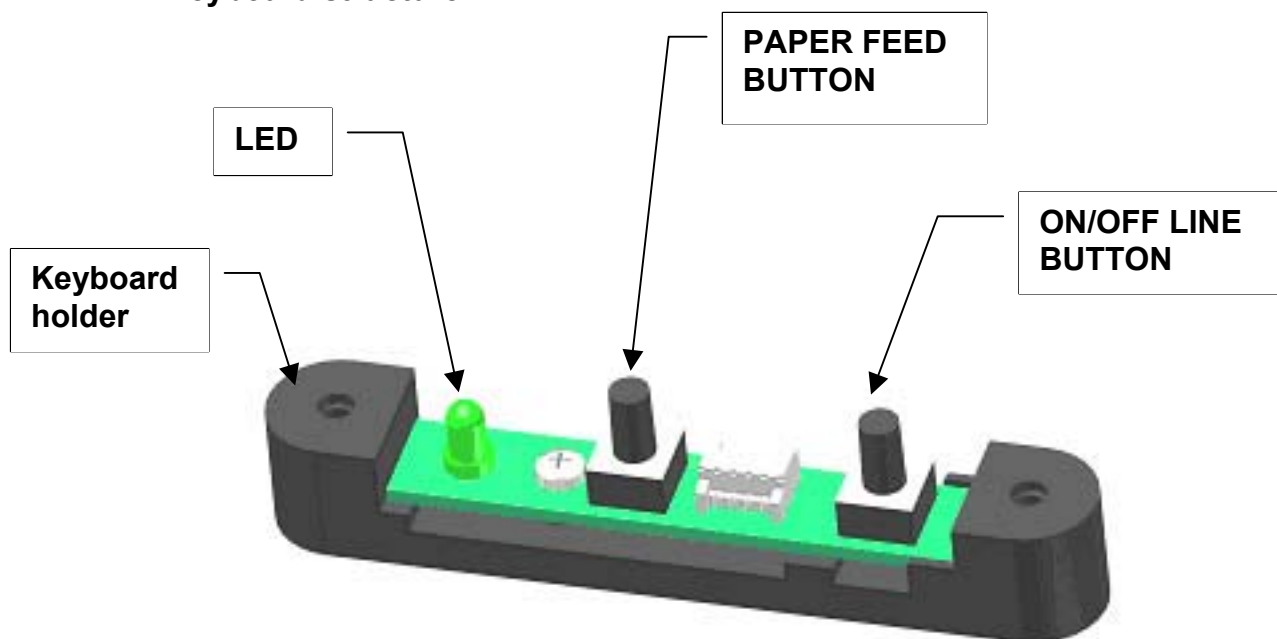
The Kit is composed by the following items:

- Keyboard (with 2 buttons and one led)
- Cable (90mm in length)
- Fixing screws Ø3x12 (2 pcs.)

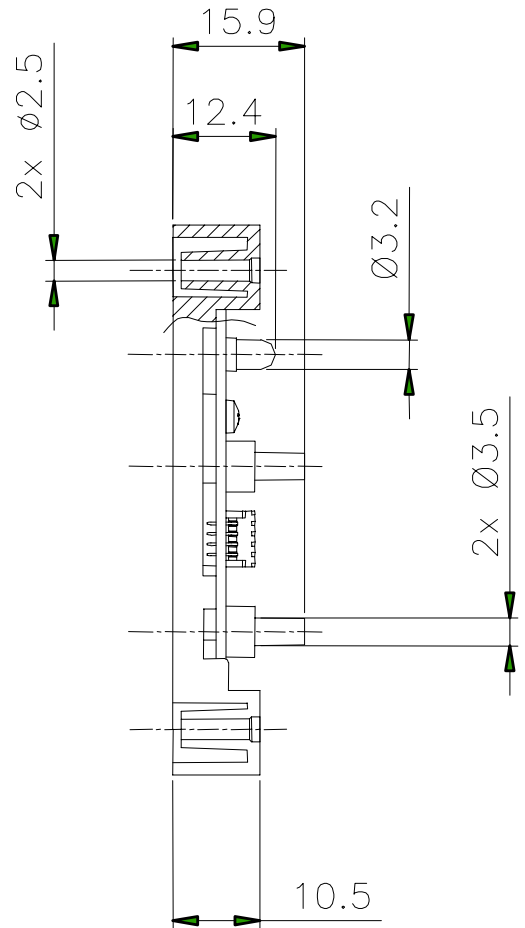
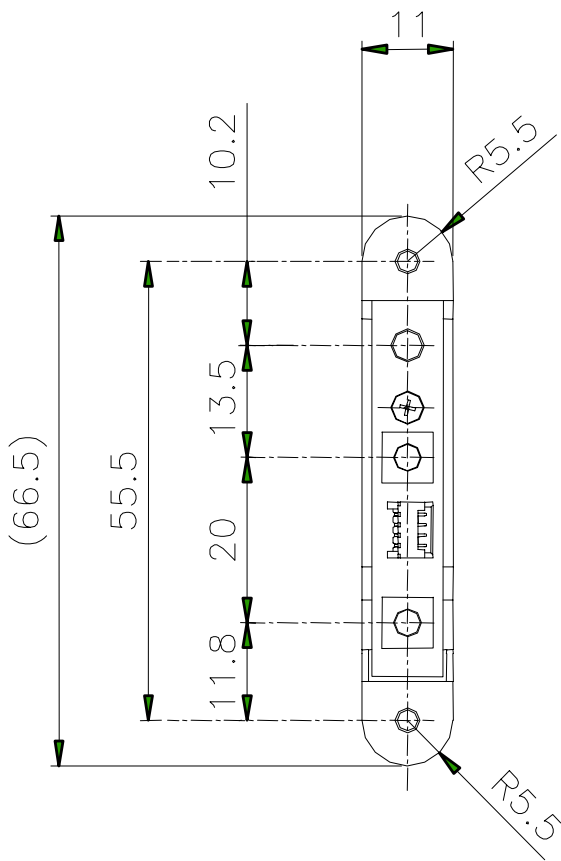
The following image shows the Keyboard KIT composition



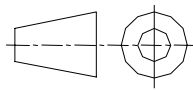
### 7.1 Keyboard structure



7.2 Keyboard specification



UNIT: mm





## 8. PRINTER DEVICE OPERATIONS

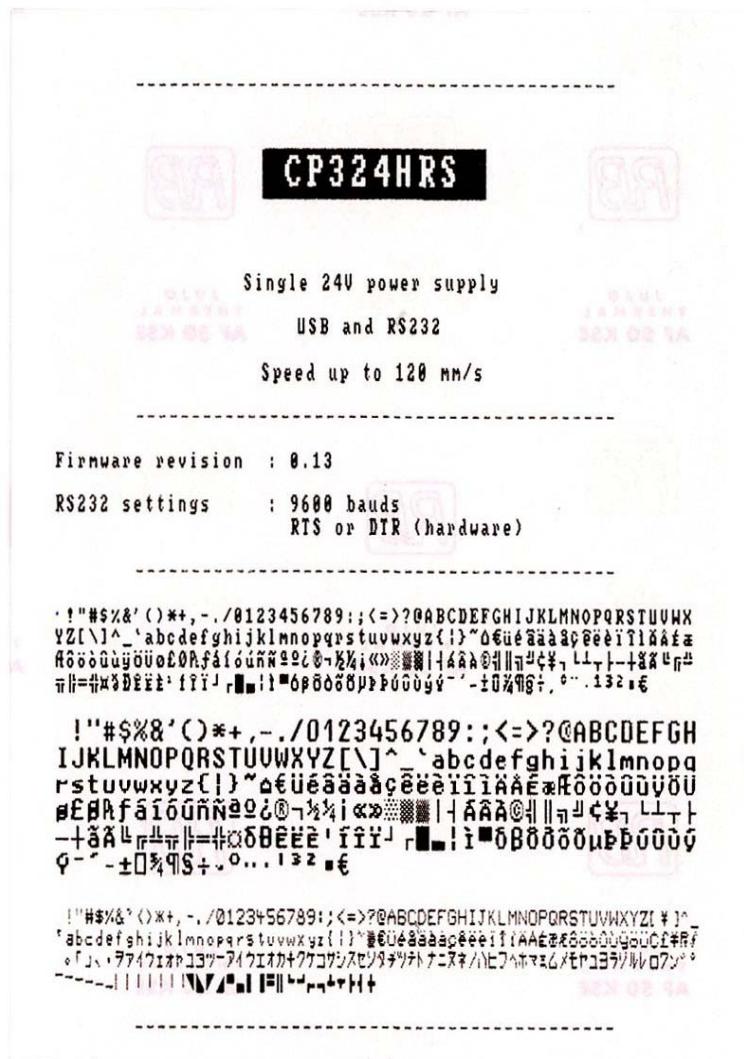
### 8.1 Self-test mode

This mode is run by the combination of the 2 external switches (see Switches and LED section). It prints the printer name and its main features, the firmware revision, the serial port settings and finally all internal character sets (3 by default).

8x16 internal character set

12x20 internal character set

7x16 internal character set



- Notes:
- After self-test running, setup parameters return to their latest saving values.
  - During self-test running, communication is disabled. Basically, received data are not buffered (lost if RS232 or in waiting state if USB). Only real time requests are processed (see section 6.5 for further details).

## 8.2 Paper loading

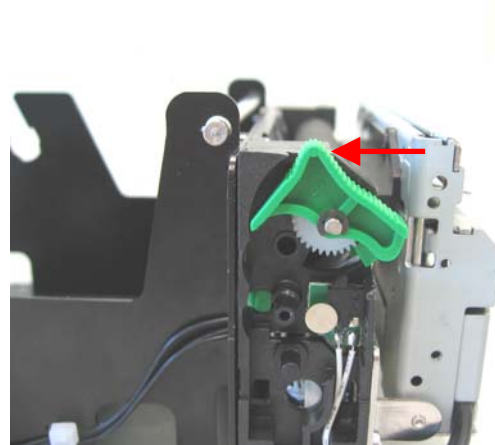
Paper loading can be performed by two different methods:

- Automatic paper loading:** With the green head-up lever in the down position, insert the paper inside the printer, and then the roller will automatically feed the paper for about 40 mm (default value). When continuous paper feed mode is active, a paper cut will be done just after the loading phase (if option enabled). When hole/black mark detection mode is active, paper will be only moved to the TOF position just after the loading phase (no cut). In every case, printer is then ready to print. This function can be performed only if power supply is valid (between 18 and 27 volts).
- Manual paper loading:** With the green head-up lever in the up position, manually feed the paper into the printer until it exits from the printer (cutter side). Then turn the green lever in down position. Now the printer is ready to print.

Lever in down position (Working posit.)



Lever in up position



## 8.3 Text printing fonts

The controller board is provided with **3 resident sets of 224 characters: 8x16, 12x20, and 7x16**. However, it is possible to remove them and add your own fonts. Cf. “9.3 Font management” for further information.

All these resident fonts include the **Euro currency symbol** (*Position 128, 80h*).

12 characters are selectable from the international character set: refer to “ESC R n” control code for more information.

All character bitmaps are presented below with their hexadecimal code (row being the most significant nibble, and column the least significant nibble). Example: ASCII code for ‘A’ is 0x41 (or 65 in decimal).

- 8x16 characters set:** minimum character area is actually *9 pixels* (8 “active dots” plus 1 dot character spacing) *x 19 pixels* (16 “active” dots plus 3 lines spacing including underline), or *1.125mm x 2.375mm*. With double and quadruple height and width, maximum character area can go up to *4.5mm width x 9.5mm height*. Horizontal character spacing and vertical line spacing may be adjusted via software. Therefore, with CP324-HRS (576 pixels width), number of characters per line can be up to 64 in standard text, 32 in double width, and 16 in quadruple width.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/	
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	Δ
8	€	ü	é	á	ä	à	ç	ë	è	ï	í	ì	ñ	â	ã	
9	é	æ	ß	ö	ò	û	ù	ÿ	ö	ü	ø	£	Ø	℞	f	
A	á	í	ó	ú	ñ	ñ	ã	õ	ç	©	¬	½	¼	¿	«	»
B	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
C	L	⊥	τ	†	-	‡	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
D	§	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
E	ó	β	ö	ö	ö	μ	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
F	-	±	∏	¼	¶	§	÷	,	°	∞	.	1	3	2	■	€

- 12x20 characters set:** minimum character area is actually *13 pixels* (12 “active dots” plus 1 dot character spacing) *x 23 pixels* (20 “active” dots plus 3 lines spacing including underline), or *1.625mm x 2.875mm*. With double and quadruple height and width, maximum character area can go up to *6.5mm width x 11.5mm height*. Horizontal character spacing and vertical line spacing may be adjusted via software. Therefore, with CP324-HRS (576 pixels width), number of characters per line can be up to 44 in standard text, 22 in double width, and 11 in quadruple width.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	Δ
8	€	ü	é	â	ä	à	ç	ê	ë	è	ï	î	ï	Ä	Å	ƒ
9	É	æ	Æ	ô	ö	õ	û	ü	ÿ	ö	Ü	ß	£	¢	ƒ	€
A	ã	í	ó	ú	ñ	Ñ	â	â	â	â	â	â	â	â	â	â
B	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘	⌘
C	L	L	T	T	-	+	ä	Ä	ℒ	℞	℥	π	ℙ	=	≠	⊙
D	δ	θ	ê	Ë	È	·	í	î	ÿ	J	r	■	■	■	■	■
E	δ	θ	ø	ø	ø	ø	μ	ℙ	ℙ	Ü	Ü	Ü	Ü	Ü	Ü	Ü
F	-	±	∅	¼	½	¾	÷	·	°	..	.	!	3	z	■	€

- 7x16 characters set:** minimum character area is actually *8 pixels* (7 “active dots” plus 1 dot character spacing) *x 19 pixels* (16 “active” dots plus 3 lines spacing including underline), or *1mm x 2.375mm*. With double and quadruple height and width, maximum character area can go up to *4mm width x 9.5mm height*. Horizontal character spacing and vertical line spacing may be adjusted via software. Therefore, with CP324-HRS (576 pixels width), number of characters per line can be up to 72 in standard text, 36 in double width, and 18 in quadruple width. This font includes the Katakana characters set.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
2		!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	⌘
8	€	ü	é	â	ä	à	ç	ê	ë	è	ï	î	ï	Ä	Å	ƒ
9	É	æ	Æ	ô	ö	õ	û	ü	ÿ	ö	Ü	ß	£	¢	ƒ	€
A	。	「	」	、	・	ヲ	イ	ウ	エ	オ	ト	チ	リ	ヲ	シ	ヲ
B	-	ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ	サ	シ	ス	セ	ソ
C	ツ	テ	ト	チ	リ	ヲ	ニ	ヲ	ヲ	ヲ	ヲ	ヲ	ヲ	ヲ	ヲ	ヲ
D	ミ	メ	モ	ム	メ	ム	ヨ	リ	ル	レ	ロ	フ	ン	°	°	°
E	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
F	▲	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

## 8.4 Control codes

Control codes are non-printable characters or sequences of characters that control the operation of the printer. Within the following description, a control code causes the printer to interpret the following byte(s) as part of a command and not as a printable character.

### 8.4.1 Control codes cross reference

#### *Setup and hardware control codes*

CONTROL CODE	DESCRIPTION
GS / n	Set maximum peak current via dynamic division
GS s n1 n2	Set maximum printing speed
GS a n	Set acceleration smoothing (not implemented in 24V printers)
GS D n	Set printing intensity
ESC @	Reset printer
ESC v	Send printer status
ESC I	Send printer identity
GS B n	Set serial communication settings
ESC o n	Set “End of Paper” optosensor type
GS O n1 n2	Do an “End of Paper” optosensor calibration
ESC O	Send “End of Paper” optosensor parameters
GS o	Send “End of Paper” optosensor current level
ESC s	Save in flash setup parameters
ESC d	Recover factory setup parameters
GS p n	Set paper loading pause
GS P n1 n2	Set paper loading length
GS e n	Eject paper (not implemented)
GS M n1 n2	Set paper loading speed
GS c n	Enable/disable historic heat
GS A msk1 msk2 app1 app2	Set some applicative behaviours
ESC n p	Send “Near End of Paper” extension presence
ESC n c	Do a “Near End of Paper” optosensor calibration
ESC n s	Send “Near End of Paper” status
ESC n l	Send “Near End of Paper” optosensor current level

## *Text and general control codes*

CONTROL CODE	DESCRIPTION
ESC % n	Select internal font
ESC R n	Select international characters set
ESC 2 n	Set line pre-spacing
ESC 3 n	Set line spacing
ESC SP n	Set character spacing
ESC b n	Set normal/inverse video mode
ESC c n	Set maximum number of columns
ESC C n	Set text justification
ESC ! n	Set printing modes
ESC { n	Set normal/rotated text line
LF	Do a text line feed
CR	Do a carriage return
ESC J n	Feed paper forward (n dot lines)
ESC j n	Feed paper backward (n dot lines)
CAN	Cancel text line data buffer

## *Graphic control codes*

CONTROL CODE	DESCRIPTION
ESC * n1 n2 n3 n4 n5 n6 <data>	Print graphic in full mode
ESC \$ n1 n2	Set graphic offset in line mode
ESC V n1 n2 n3 <data>	Print graphic in line mode

## *Cutter control codes*

CONTROL CODE	DESCRIPTION
ESC m	Do a partial cut
ESC i	Do a full cut

### *Bar code control codes*

CONTROL CODE	DESCRIPTION
<b>GS k n</b> [Start] <data> [Stop]	Print bar code
<b>GS h n</b>	Set bar code height
<b>GS w n</b>	Set bar code magnification
<b>GS H n</b>	Set bar code text position
<b>GS R n</b>	Set normal/rotated bar code

### *Hole and black mark detection control codes*

CONTROL CODE	DESCRIPTION
<b>GS L n</b>	Set mark length, switch continuous/mark mode
<b>GS E</b>	Move to “Top Of Form” (TOF) position
<b>GS T n1 n2</b>	Set “Mark” to TOF position length
<b>GS Y n1 n2</b>	Set “Optosensor” to “Head dot line” length
<b>GS X n1 n2</b>	Set “Mark” to “Cut” position length
<b>GS x n1 n2</b>	Set “Head dot line” to “Cut” position length

## 8.4.2 Setup and hardware control codes

### GS / n

---

Description: Set maximum peak current via dynamic division (impact on printing speed).

Format: <1Dh> <2Fh> <n>

Comments: n = 0 : enable maximum peak current on power supply (and so maximum printing speed).  
 n = 1 to 32 : software programmable consumption (dynamic division). The maximum number of black dots which can be simultaneously heated is (n+1) x 8. So maximum printing speed may be never reached because successive heats can take a lot of time.

Default : n = 17

Example : n = 5 => maximum black dots simultaneously heated : (5+1)\*8 = 48.

Printer peak consumption @24V =

(0.72A (stepper motor) + 0.2A (logic power supply) + 24\*48/1500) = 1.69A

1500 Ohms is the dot resistance.

**For the moment, minimum available regarding n is 15, CP324HRS (standard).**

### GS s n1 n2

---

Description: Set maximum printing speed.

Format: <1Dh> <73h> <n1> <n2>

Comments: This control code may be used to reduce printing speed in case of paper roll diameter above 120mm and/or if rewinding mechanism is connected to the printer.

Bytes n1 and n2 set the time T (in  $\mu$ s) between each step:  $T = (256*n1) + n2$ .

Default : T = 1042 $\mu$ s (120mm/s).

Example : (n1 = 4, n2= 226)=> T = 1250  $\mu$ s

=> Maximum printing speed =  $(1 / (8 * 1250e-6)) = 100$  mm/s

8 dot lines/mm is the line pitch.

Notes : If parameter T is zero, speed is reset at its default power-on value (**not yet implemented**).

Vmin = 10mm/s and Vmax = 120mm/s.

### GS D n

---

Description: Set printing intensity.

Format: <1Dh> <44h> <n>

Comments: n=80h (128d) : nominal printing intensity (default value).

n>80h (128d) : printout becomes darker (until n = FFh : +60% intensity).

n<80h (128d) : printout becomes lighter (until n = 00h : -60% intensity).



## ESC @

---

Description: Reset printer.

Format: <1Bh> <40h>

Comments: Reset printer device. This is a real time request. See section 6.5 for further details.

## ESC v

---

Description: Send printer status.

Format: <1Bh> <76h>

Comments: Printer returns a single byte that reflects its status in accordance with the following table:

BIT	FUNCTION	BIT = 0	BIT = 1
0	Head temperature	OK	Too high or too low
1	Head-up	No	Yes
2	End of Paper	No	Yes
3	Power supply	OK	Too high or too low
4	Printer in use	Ready	Action in progress
5	ON/OFF line	OFF	ON
6	Hole/mark detection error	No	Hole/mark not found
7	Cutter error	Yes	No

This is a real time request. See section 6.5 for further details.

## ESC I

---

Description: Send printer identity.

Format: <1Bh> <49h>

Comments: Printer returns an ASCII string ended by zero (00h) that reflects the printer identity.

The string is formed by the concatenation of printer mechanism name and firmware revision, as below:

CP324HRS                      0.13

The string always has a fixed format, that is : the printer mechanism name padded to 16 bytes, a space, then 5 bytes for the firmware revision (the dot being in the middle), ended with zero. So it is a 23 bytes length string.

## GS B n

---

Description: Set serial communication settings.

Format: <1Dh> <42h> <n>

Comments: Sets serial communication speed and handshaking mode.

Bit 7: b7 = 0: “Xon/Xoff” handshaking mode (software control),  
           b7 = 1: “RTS or DTR” handshaking mode (hardware control).

Bits 6, 5, 4, 3: not used.

Bits 2, 1, 0: speed.

b2b1b0	SPEED (BAUDS)
0	1200
1	2400
2	4800
3	9600
4	19200
5	38400
6	57600
7	115200

Default: n = 83h : “RTS or DTR” handshaking mode, 9600 bauds, 8 bits of data, no parity bit, 1 stop bit.

## ESC o n

---

Description: Set “End of Paper” optosensor type.

Format: <1Bh> <6Fh> <n>

Comments: n = 0 : reflective optosensor is mounted.

          n = 1 : transmissive optosensor is mounted.

Default : reflective optosensor.

## GS O n1 n2

---

Description: Do an “End of Paper” optosensor calibration.

Format: <1Dh> <4Fh> <n1> <n2>

Comments: Due to dispersions in optosensor mounting and sensitivity and due to paper features (reflection efficiency, black mark ...), calibration may be required to update thresholds at which an “End of Paper” will be set or a “Black Mark” will be detected. These updated thresholds are also saved in flash memory and will be recovered at next power-on. In addition, “Black”, “Mark” and “Paper” optosensor levels will be saved in flash too for information.

This control code makes printer to perform a calibration procedure which features depends of bytes n1 and n2 :

- n1 : length in cm of paper loading before starting actually calibration,
- n2 : length in cm of paper required for calibrating the optosensor.

At the end of procedure, printer returns a single byte:

- 0x01 if calibration and saving are successful,
- 0x00 if calibration or saving failure.

**CAUTION** : before sending this control code, paper should absolutely be removed from printer, otherwise, calibration will fail (no robustness) and only a hardware RESET could make printer work again. In addition, 1<sup>st</sup> loaded paper length should not contain any black mark which could cause calibration failure too. For information, when reflective optosensor is used, calibration can be performed even with a paper length without any black mark.

Notes :- For the moment, saving of these new thresholds causes saving of all setup parameters too as “ESC s” request (flashing of only few parameters at a time is not available). So it is advised to perform a calibration procedure only in a stable and known context.

- Calibration procedure can take a lot of time in function of paper loading lengths. In addition, during saving, communication is not guarantee (risk of loss of data in RS232) because flashing operation cannot be performed simultaneously with normal operating function. So it is advised for host to wait for printer return status with a sufficient timeout before resuming communication.
- An “End of Paper” optosensor calibration application note and calibration tools are available. Please contact APS for further details.

## ESC O

---

Description: Send “End of Paper” optosensor parameters.

Format: <1Bh> <4Fh>

Comments: Printer returns the 6 next bytes :

- Byte 1 : optosensor type (0 : reflective / 1 : transmissive),
- Byte 2 : black optosensor level (no paper),
- Byte 3 : mark/backing optosensor level,
- Byte 4 : paper optosensor level,
- Byte 5 : paper presence threshold,
- Byte 6 : mark detection threshold.

Levels are basically analogical/digital conversion results and thresholds are also coherent with level type.

All these parameters are determined automatically during “End of Paper” optosensor calibration procedure. This command is intended for test purpose.

Default : reflective / 255 / 255 / 0 / 249 / 249. These values should make most of applications to work properly.

## GS o

---

Description: Send “End of Paper” optosensor current level.

Format: <1Dh> <6Fh>

Comments: Printer returns 1 byte representing the optosensor current level (analogical/digital conversion result). Smaller value means more reflection (paper is closer). This control code is intended mostly for test purpose.

## ESC s

---

Description: Save in flash setup parameters.

Format: <1Bh> <73h>

Comments: Setup parameters are saved in the internal flash memory of controller. Therefore, they are not lost when power is removed or printer is reset, and are recovered at each power-on/reset.

Printer returns 1 byte representing the status of this request :

- 0x01 : saving successful,
- 0x00 : saving failure.

Saving can take more or less time in function of action in progress. During saving, communication is not guaranteed (risk of loss of data in RS232) because flashing operation cannot be performed simultaneously with normal operating function. So it is advised for host to wait for printer return status with a sufficient timeout before resuming communication (100ms minimum, more if printing operation was in progress when request has been sent).

The following parameters are saved by this command. Basically, they represent all setup operations performed with control codes.

### “Custom” parameters

- maximum peak current,
- maximum printing speed,
- intensity,
- RS232 settings,
- paper loading pause,
- paper loading length,
- paper loading speed,
- historic heat management,
- active font,
- active international character set,
- pre-line spacing,
- line spacing,
- character spacing,
- maximum number of columns,
- text rotation,
- inverse video,
- text justification,
- text mode (simple/double/quadruple width/height and underlining),
- bar code height,

- bar code magnification,
- bar code “human readable interpretation” position,
- bar code rotation,
- applicative behaviours.

#### “Calibration” parameters

- mechanism configuration,
- “End of Paper” optosensor type,
- “Black” optosensor level,
- “Mark” optosensor level,
- “Paper” optosensor level,
- “End of Paper” threshold,
- “Mark” threshold,
- “Near End of Paper” additive pullup,
- “Near End of Paper” threshold,
- “Mark” length,
- “Mark” to “Top Of Form” position length,
- “Mark” to “Cut” position length,
- “Optosensor” to “Head dot line” length,
- “Head dot line” to “Cut” position length.

## **ESC d**

---

Description: Recover factory setup parameters.

Format: <1Bh> <64h>

Comments: Revert all setup parameters (cf. above) to their factory default values. This updating is temporary. Indeed, if printer is reset or power is cycled, setup parameters will be initialized with the last set saved in flash (with “ESC s”). If you want to permanently set the parameters to their factory default values, you must send an “ESC d” “ESC s” sequence. Combining the use of this command and the “Reset printer” one enables you to compare the effects of the new saved values with default ones without altering them.

Printer returns 0x01 when recovering is done. It is advised for host to wait for this acknowledgement before resuming printing operations so as to be sure to work with coherent parameters. As above, timeout value is function of action in progress when request has been sent.

Note : be careful regarding possible change of RS232 baudrate. Indeed, after sending this request, host should quickly update its baudrate too in order to be able to receive printer acknowledgement. Basically, it is advised to keep baudrate constant between default and new parameters values.

## GS p n

---

Description: Set paper loading pause.

Format: <1Dh> <70h> <n>

Comments: Set time pause between the moment when printer detects the insertion of paper and the moment when roller starts turning. This allows accurate manual positioning of paper.  
n is in 125 millisecond units (=> 0 to 32s).

Default: n = 0 (instantaneous).

Example: n = 16 => printer will wait for 2 seconds before starting paper loading.

## GS P n1 n2

---

Description: Set paper loading length.

Format: <1Dh> <50h> <n1> <n2>

Comments: Set length of paper to feed during the automatic paper loading.  
Bytes n1 and n2 set the length L (in dot lines) of feeding :  $L = (256 * n1) + n2$

Default : L = 40mm.

Example : (n1 = 3, n2= 32)=> L = 800 dot lines => L = 800 / 8 = 100mm  
*8 dot lines/mm is the line pitch.*

## GS e n Not implemented

---

Description: Eject paper.

Format: <1Dh> <65h> <n>

Comments: Printer will feed paper until an “End of Paper” condition is detected. It will then feed an extra n millimeter (0 to 255), useful for ejecting paper totally.

## GS M n1 n2

---

Description: Set paper loading speed.

Format: <1Dh> <4Dh> <n1> <n2>

Comments: This control code is used to adapt loading speed to various conditions.  
Bytes n1 and n2 set the time T (in  $\mu$ s) between each step:  
 $T = (256 * n1) + n2$  with  $1042 \leq T \leq 12500 \mu$ s ( $10 \leq \text{speed} \leq 120 \text{mm/s}$ ).

Default : T = 12500 $\mu$ s (10mm/s).

Example : (n1 = 24, n2= 106)=> T = 6250  $\mu$ s  
=> paper loading speed =  $(1 / (8 * 6250e-6)) = 20 \text{ mm/s}$   
*8 dot lines/mm is the line pitch.*

## GS c n

Description: Enable/disable historic heat.

Format: <1Dh> <63h> <n>

Comments: When high printing speed is set ( $\geq 60\text{mm/s}$ ), historic heat is required to improve printing quality especially if text is printed. However, this additional heat makes controller to work more and in some case it can cause that maximum speed cannot be reached (too many processing to be done in a few time). So a compromise should be chosen : either speed or quality.

n = 0 : historic heat is disabled,  
other values : historic heat is enabled.

Default : enabled.

Note : When picture graphic is printed, historic should be generally disabled because it makes printout darker. Basically, intensity modification is more suitable to get good printing quality; idem for curve graphic. On the contrary, regarding text graphic, historic is generally required so as to increase edge density. Therefore, user should set suitable historic mode just before transmitting his graphic.

## GS A msk1 msk2 app1 app2

Description: Set some applicative behaviours.

Format: <1Dh> <41h> <msk1> <msk2> <app1> <app2>

Comments: This control code is used to customize a little bit printer behaviour.  
Couple (msk1; msk2) is the activation mask of the next couple (app1; app2) : if  $mskX.n$  is set to 1, then  $appX.n$  is taken into account.  
For the moment, only 2 applicative behaviours can be customized :

BYTE	BIT	APPLICATIVE BEHAVIOUR	BIT = 0	BIT = 1
app1	all	unused	-	-
app2	7 to 2	unused	-	-
	1	full cut after automatic paper loading	No	Yes
	0	full cut after self-test ticket printing	No	Yes

Default : always full cut after automatic paper loading and self-test ticket printing.

Example : msk1=0b00000000 msk2=0b00000010  
app1=0b00000000 app2=0b00000000  
⇒ no full cut after automatic paper loading

## **ESC n p**      **Obsolete (kept for compatibility with old MRS)**

---

Description: Send “Near End of Paper” extension presence.

Format: <1Bh> <6Eh> <70h>

Comments: Printer checks for the presence of a “Near End of Paper” extension and returns a single byte:

- 0x01 if it was found,
- 0x00 if not.

CAUTION : CP324-HRS cannot detect extension presence. Therefore it always returns 0x01 to ensure compatibility with old MRS tools regarding “Near End of Paper” management. In practice, host should know actual configuration (extension present or not) before taking into account “Near End of Paper” status.

## **ESC n c**

---

Description: Do a “Near End of Paper” optosensor calibration.

Format: <1Bh> <6Eh> <63h>

Comments: Due to dispersions in optosensor mounting and sensitivity, calibration may be required to update threshold at which a “Near End of Paper” will be set. This updated threshold is also saved in flash memory and will be recovered at next power-on. This control code makes printer to perform a calibration procedure. At the end of procedure, printer returns a single byte:

- the new threshold if saving is successful,
- 0x00 if not.

Default : 245 (this value provided by firmware is suitable for most of applications; calibration is often optional).

Notes : - For the moment, saving of this new threshold causes saving of all setup parameters too as “ESC s” request (flashing of only one parameter at a time is not available). So it is advised to perform a calibration procedure only in a stable and known context.

- As “ESC s” operation, saving can take more or less time in function of action in progress. During saving, communication is not guarantee (risk of loss of data in RS232) because flashing operation cannot be performed simultaneously with normal operating function. So it is advised for host to wait for printer return status with a sufficient timeout before resuming communication (100ms minimum, more if printing operation was in progress when request has been sent).
- A “Near End of Paper” optosensor calibration application note and calibration tools are available. Please contact APS for further details.



## **ESC n s**

---

Description: Send “Near End of Paper” status.

Format: <1Bh> <6Eh> <73h>

Comments: Printer returns a single byte representing the “Near End of Paper” status, that is, if current optosensor level is under or above calibrated threshold. Value is :

- 0x00 : OK, enough paper still present,
- 0x01 : “Near End of Paper” condition.

## **ESC n l**

---

Description: Send “Near End of Paper” optosensor current level.

Format: <1Bh> <6Eh> <6Ch>

Comments: Printer returns 1 byte representing the optosensor current level (analogical/digital conversion result). Smaller value means more reflection (paper is closer). This control code is intended mostly for test purpose.

## 8.4.3 Text and general control codes

### ESC % n

Description: Select internal font.

Format: <1Bh> <25h> <n>

Comments: n = 0: **8x16** font is selected.  
 n = 1: **12x20** font is selected.  
 n = 2: **7x16** font is selected.

Default: 8x16.

These are the 3 resident fonts provided by default. For customized fonts, cf. “9.3 Font management”.

### ESC R n

Description: Select international characters set.

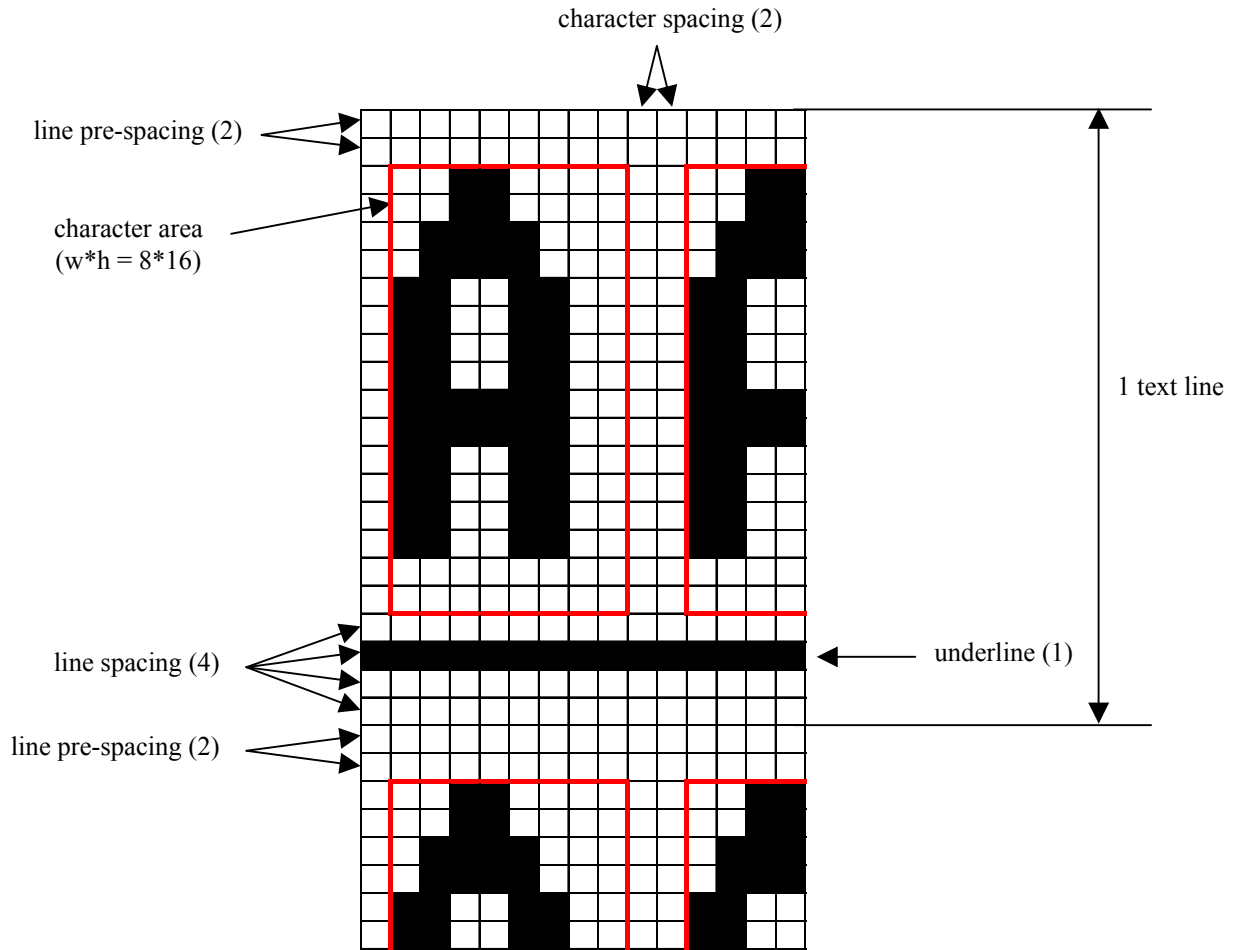
Format: <1Bh> <52h> <n>

Comments: 12 characters are defined as international characters. This control code enables to set which international characters set should be used: n value selects one of the 13 available countries:

n	COUNTRY	23h	24h	40h	5Bh	5Ch	5Dh	5Eh	60h	7Bh	7Ch	7Dh	7Eh
0	USA	#	\$	@	[	\	]	^	‘	{		}	~
1	France	#	\$	à	°	ç	§	^	‘	é	ù	è	“
2	Germany	#	\$	§	Ä	Ö	Ü	^	‘	å	ö	ü	ß
3	UK	£	\$	@	[	\	]	^	‘	{		}	~
4	Denmark 1	#	\$	@	Æ	φ	Å	^	‘	æ	Φ	å	~
5	Sweden	#	α	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
6	Italy	#	\$	@	°	\	é	^	ù	à	ò	è	ì
7	Spain 1	Pt	\$	@	ı	Ñ	¿	^	'	"	ñ	}	~
8	Japan	#	\$	@	[	¥	]	^	‘	{		}	~
9	Norway	#	α	É	Æ	φ	Å	Ü	é	æ	Φ	å	ü
10	Denmark 2	#	\$	É	Æ	φ	Å	Ü	é	æ	Φ	å	ü
11	Spain 2	#	\$	à	ı	Ñ	¿	é	'	ı	ñ	ó	ú
12	Latin Amer.	#	\$	à	ı	Ñ	¿	é	û	ı	ñ	ó	ú

Default: USA.

Example of text settings:



## ESC 2 n

Description: Set line pre-spacing.

Format: <1Bh> <32h> <n>

Comments: Set the number of dot lines to insert before starting actually character dot lines. This is very useful when printing in inverse video mode some characters which pixels start from first dot line. n may vary from 0 to 15 dot lines. Line pre-spacing pitch is 1/8mm.

Default: 0.

CAUTION: actual line pre-spacing is calculated from this user value n and updated proportionally in function of current height mode (simple (n\*1), double (n\*2), quadruple (n\*4)).

## ESC 3 n

---

Description: Set line spacing.

Format: <1Bh> <33h> <n>

Comments: Set the number of dot lines to insert just after last character dot line and before next one. n may vary from 0 to 15 dot lines. Line spacing pitch is 1/8mm.

Default: 3.

Note: underline is managed only from line spacing of 3 and then it will be always the 2<sup>nd</sup> dot line of this set of dot lines.

CAUTION: actual line spacing is calculated from this user value n and updated proportionally in function of current height mode (simple (n\*1), double (n\*2), quadruple (n\*4)).

## ESC SP n

---

Description: Set character spacing.

Format: <1Bh> <20h> <n>

Comments: Set the number of dots to insert just after last character dot and before next one. n may vary from 0 to 16 dots. Character spacing pitch is 1/8mm.

Default: 2.

Note : a last character can be possible provided it holds in the line even if next character spacing overlaps.

CAUTION: actual character spacing is calculated from this user value n and updated proportionally in function of current width mode (simple (n\*1), double (n\*2), quadruple (n\*4)).

## ESC b n

---

Description: Set normal/inverse video mode.

Format: <1Bh> <62h> <n>

Comments: n = 0 : normal video mode,  
n = 1 : inverse video mode.

Default: normal video.

Note: this setting is valid for the whole printing text line. Spaces at the beginning of a text line will be printed as a dark rectangle. In order to shift the black printing from the left margin, one can send TAB (ASCII 0x09) instead SP (ASCII 0x20). This enables an accurate control of white/black edges.

## ESC c n

---

Description: Set maximum number of columns.

Format: <1Bh> <63h> <n>

Comments: Set the maximum number of printable characters that printer accepts in a same text line before automatically going to the next text line.  
n may vary from 3 to 255.

Default: 255.

## ESC C n

---

Description: Set text justification.

Format: <1Bh> <43h> <n>

Comments: Set how text will be justified:  
n = 0: text will be centred,  
n = 1: text will be right justified,  
n = 2: text will be left justified.

Default: left justification.

## ESC ! n

---

Description: Set printing modes.

Format: <1Bh> <21h> <n>

Comments: Select various printing modes as described below:

Bit of n	Function	Bit = 0	Bit = 1
0	Not used	-	-
1	Quadruple Height	Cancelled	Set
2	Quadruple Width	Cancelled	Set
3	Not used	-	-
4	Double Height	Cancelled	Set
5	Double Width	Cancelled	Set
6	Not used	-	-
7	Underlined	Cancelled	Set

Default: simple height and width, no underlining.

Notes:- several printing widths can be mixed on the same text line,

- only one printing height is enabled per text line (change of height requests are ignored and lost),
- underline is actually managed only if enough line spacing ( $\geq 3$ ).

## ESC { n

---

Description: Set normal/rotated text line.

Format: <1Bh> <7Bh> <n>

Comments: n = 0 : printout is normal,  
n = 1 : printout is rotated of 180°.

Default: normal.

## LF

---

Description: Do a text line feed.

Format: <0Ah>

Comments: Move the print position to the beginning of the next text line.

Note: if LF follows a CR, printer will ignore LF. So, CR = LF = CR+LF.

## CR

---

Description: Do a carriage return.

Format: <0Dh>

Comments: Move the print position to the beginning of the next text line.

Note: if CR is followed by LF, printer will ignore LF. So, CR = LF = CR+LF.

## ESC J n

---

Description: Feed paper forward (n dot lines).

Format: <1Bh> <4Ah> <n>

Comments: Paper is fed forward for n dot lines. Print position is reset to the beginning of the next dot line.  
n may vary from 1 to 255. Dot line pitch is 1/8mm.

## ESC j n

---

Description: Feed paper backward (n dot lines).

Format: <1Bh> <6Ah> <n>

Comments: Paper is fed backward for n dot lines. Print position is set to the beginning of the next dot line.  
n may vary from 1 to 255. Dot line pitch is 1/8mm.

## CAN

---

Description: Cancel text line data buffer.

Format: <18h>

Comments: Text line data buffer is cancelled and print position is set to the beginning of the next text line.

Note : cancel operation is not applied on received data buffer (communication buffer) but only on text line data buffer (buffer used to stock all characters of the current text line).

## 8.4.4 Graphic control codes

### ESC \* n1 n2 n3 n4 n5 n6 <data>

Description: Print graphic in full mode.

Format: <1Bh> <2Ah> <n1> <n2> <n3> <n4> <n5> <n6> <data>

Comments: Bytes n1, n2 and n3 set the number of data bytes N to be printed out:

$$N = (65536 * n3) + (256 * n2) + n1$$

Byte n4 sets graphic operator on data bytes:

- n4 = 0 : data bytes are directly printed out (normal size, full printer resolution),
- n4 = 1 : double width (each pixel is repeated horizontally),
- n4 = 2 : double height (each pixel is repeated vertically),
- n4 = 3 : expanded (double width and height).

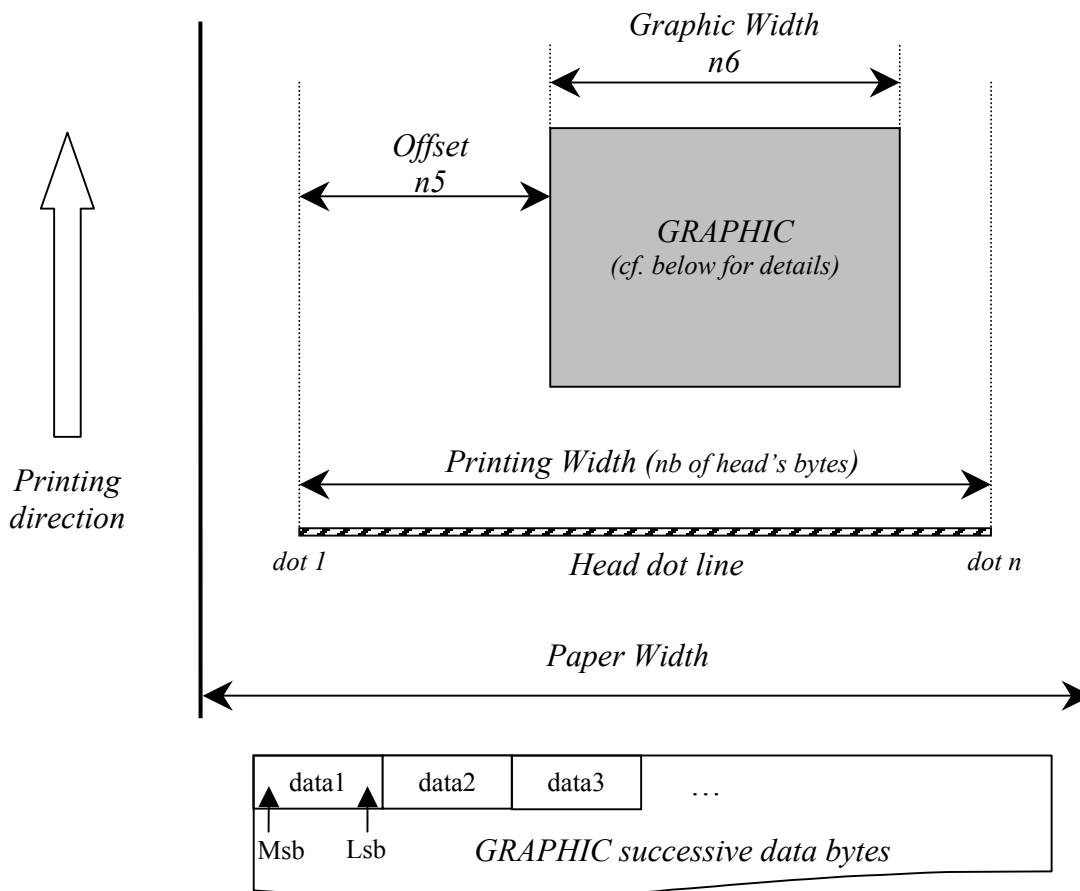
Byte n5 sets the number of dot bytes to be skipped (offset) before printing out the first graphic bit:

- 00h : first graphic bit to be printed out is dot 1 on the head,
- 01h to FFh : 1 to 255 skipped dot bytes (< total number of head's bytes).

Byte n6 sets the width of the graphic to be printed out:

- 01h to FFh : width is 1 to 255 bytes ( $\leq$  total number of head's bytes).

Data bytes : successive graphic dot bytes to be printed out. Host should send them with meeting order requirement drawn below. "1" bit value means black dots while "0" means white dot.



Total number of head's bytes is given by the total number of dots divided by 8. For instance CP324HRS is 576 dots / 8 = 72 bytes.

Example: with the following bitmap:



Black and white, 1 dot per pixel, 368 pixels width and 242 pixels height, printed in full resolution, and centred in a CP324HRS (576 pixels),

Image size =  $368 * 242 / 8 = 11\ 132$  bytes

$n1 = 124d, n2 = 43d, n3 = 0d, n4 = 0d, n5 = 13d, n6 = 46d$  or  
 $n1 = 7Ch, n2 = 2Bh, n3 = 00h, n4 = 00h, n5 = 0Dh, n6 = 2Eh.$

---

## ESC \$ n1 n2

Description: Set graphic offset in line mode.

Format: <1Bh> <24h> <n1> <n2>

Comments: This control code should be used with “ESC V ...” just before sending graphic data in line mode. Set the number N of bytes to be skipped before printing out first graphic bit of all next graphic lines (< total number of head's bytes).  $N = (256 * n2) + n1$ . So n2 will be always 0.  
*Total number of head's bytes is given by the total number of dots divided by 8. For instance CP324-HRS is 576 dots / 8 = 72 bytes.*

---

## ESC V n1 n2 n3 <data>

Description: Print graphic in line mode.

Format: <1Bh> <56h> <n1> <n2> <n3> <data>

Comments: Byte n1 sets graphic operator on data bytes:

- n1 = 0 : data bytes are directly printed (normal size, full printer resolution),
- n1 = 1 : double width (each pixel is repeated horizontally),
- n1 = 2 : double height (each pixel is repeated vertically),
- n1 = 3 : expanded (double width and height).

Bytes n2 and n3 set the number of data bytes N to be printed out ( $\leq$  total number of head's bytes):

$N = (256 * n3) + n2$ . So n3 will be always 0.

Data bytes : successive graphic dot bytes to be printed out. Host should send them with meeting the same order requirement as in full mode (cf. above). “1” bit value means black dots while “0” means white dot.

*Total number of head's bytes is given by the total number of dots divided by 8. For instance CP324-HRS is 576 dots / 8 = 72 bytes.*



## IMPORTANT NOTES FOR GRAPHICS:

- It is advised when graphic printing is required with **RS232** line to set up communication speed at its maximum value (115200 bauds). Indeed, because there is a lot of data to transmit to printer, communication speed is then the main limitation factor on resulted printing speed. In addition, when user maximum printing speed is too high in comparison with RS232 communication speed and graphic features (width, zoom), it may cause some “stop&go” events which affects printing quality. To avoid this bad behaviour, firmware limits automatically and temporary maximum printing speed in function of all these previous parameters (=> lower but constant printing speed without “stop&go” event, better printing quality). However, this automatically setting is performed only in full mode. In line mode, user should set himself the right setting.
- It is advised when graphic printing is required with **USB** line to limit maximum printing speed. Indeed, because printer has a lot of data to process in a low period, when maximum printing speed is too high, it can cause also some “stop&go” which affects printing quality. Unlike RS232/full mode context, firmware does not limit automatically maximum printing speed. User should set himself the right setting.
- Please check that: “**n5 + n6 ≤ total number of head’s bytes**” (that is : *offset + width ≤ printing width*). If it is greater, graphic will be truncated, of course. So printer is robust but speed performance may be altered because useless data should be received and processed while it is not necessary. So it takes useless processing time which makes user to set a lower maximum printing speed to get best printing quality.

For the moment, these previous advices are required to get the best performance as possible. Some enhancements are forecasted to make required settings automatically without user operation.

## 8.4.5 Cutter control codes

The below operations are properly performed only with APS cutters. If any other cutter is used, please contact APS for further information.

### ESC m

---

Description: Do a partial cut.

Format: <1Bh> <6Dh>

Comments: If cutter is present, printer performs a partial cut. Cut position is function of current mode:

- when continuous paper feed mode, partial cut is performed at the current paper position,
- when hole/black mark detection mode, first paper is fed forward until specific cut position (cf. “GS X ...” control code) and then partial cut is performed.

### ESC i

---

Description: Do a full cut.

Format: <1Bh> <69h>

Comments: If cutter is present, printer performs a full cut. Cut position is function of current mode:

- when continuous paper feed mode, full cut is performed at the current paper position,
- when hole/black mark detection mode, first paper is fed forward until specific cut position (cf. “GS X ...” control code) and then full cut is performed.

### IMPORTANT NOTE FOR CUTTER INTEGRATION:

Cutter can be the root cause of printer failures as intempestive reset (mainly just at cut start but also during normal printing). Indeed, in function of paper kind, ticket length and ambient air, cutter can accumulate more or less electrostatic loads due to paper friction and this can cause electrostatic discharges (ESD) and so reset of printer. So it is highly advised to take care of cutter integration.

Therefore, each printer with cutter is delivered with a grounding cable which connects cutter holder to electrical ground (GND) of PCB. It enables to flow electrostatic loads from cutter and prevent any discharge/reset issue. So this cable should not be removed.

To complete/optimize integration, it is advised to ground also cutter to system chassis by means of a large cable (braid + eyelet for example) connected between screw of cutter holder and the closest as possible screw on chassis.

## 8.4.6 Bar code control codes

### GS k n [Start] <data> [Stop]

Description: Print bar code.

Format: <1Dh> <6Bh> <n> [Start] <data> [Stop]

Comments: Byte n specifies bar code standard type as described in the following table.

[Start] is an optional byte only used with Code 128.

[Stop] is an optional byte always used except with PDF417.

n	START BYTE	BAR CODE TYPE	STOP BYTE	
0	No	UPC-A	00h	
1	No	UPC-E	00h	
2	No	EAN13	00h	
3	No	EAN8	00h	
4	No	Code 39	00h	
5	No	Interleaved 2/5 (ITF)	00h	
6	No	Codabar	00h	
7	135d	Code 128 (start with subset A)	00h	<i>Still available for backward compatibility with old host MRS applications</i>
	136d	Code 128 (start with subset B)	00h	
	137d	Code 128 (start with subset C)	00h	
	138d	Code 128 (automatic subset selection)	8Bh	<i>To be used by new applications</i>
8	No	PDF417	No	

Notes (intended for 1D bar codes):

- some checking are performed on <data> in function of bar code type : minimum number of data bytes, correct checksum byte, correct character type (only numerical characters for example), possible UPC-A compression. If data are wrong, bar code will not be printed out and next data bytes following to wrong data may be processed as other data. Therefore, some “unwanted” characters can be printed out instead of barcode if barcode data are not valid.
- when number of data bytes is higher than required number (required ‘Stop’ is not detected), even so bar code data are processed (checking step, ...).
- when checksum byte misses, printer will calculate it and add to <data> (except with Code 39).
- when UPC-E is selected, data to be transmitted can be either initial UPC-A data or directly corresponding compressed UPC-E data (checksum byte is then compulsory) (**not yet implemented**).

- when Codabar is selected, “Start” and “Stop” bytes are compulsory to get a valid encoding (but their presence is not tested).
- when ITF is selected, last byte will be ignored if total number of bytes is odd. Furthermore, if total is null (or = 1), no bar code will be printed out.
- when Code 39 is selected, “Start” and “Stop” bytes are automatically added and so they should not be sent.
- when Code 128 is selected, automatic mode is advised because it enables direct encoding of data. However, to keep backward compatibility with old host MRS applications, “manual” mode is still available. For information, in this last mode, 0x00 cannot be encoded because used as “Stop” byte.

PDF417 : this 2D bar code is more complex and requires further details. <data> field is made up of several sub-fields :

<n1> <n2> <n3> <n4> <n5> <data1> <data2>

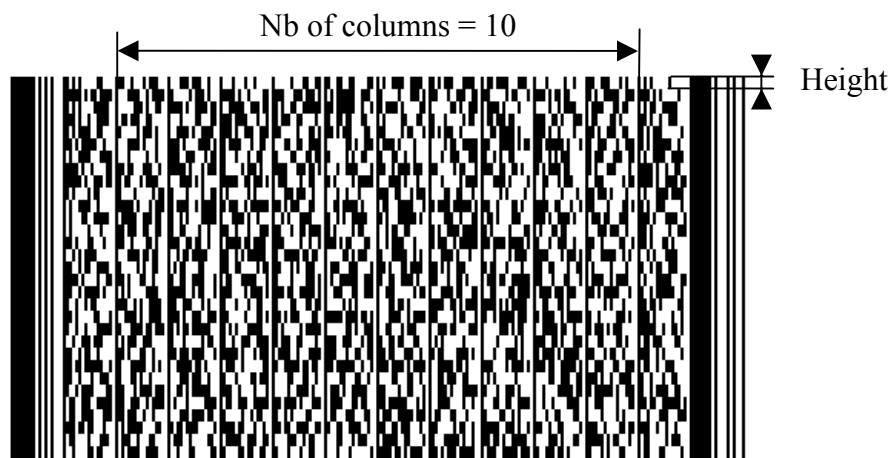
- n1 : compression mode (**for the moment, “Automatic” is set automatically**)
  - o 0 : Text,
  - o 1 : Numeric,
  - o 2 : Byte,
  - o 3 : Automatic.
- n2 : error level (0 to 8) (**maximum of 5 for the moment**),
- n3 : number of columns (1 to 30),
- n4, n5 : number of bytes transmitted in the next data blocks (n4 is MSB, n5 is LSB; 1 to 2862 bytes),
- <data1> : as many bytes of data as indicated by (n4, n5),
- <data2> : repetition of <data1>

Notes :

- <datax> contains the bytes to be encoded. The whole extended ASCII table is allowed. The maximum number of bytes depends basically of their type (“text”, “numeric” or “byte” => compression efficiency) and chain of these types (insertion of specific “switches”). For information, it is possible to encode up to 1850 “text” bytes (TAB, LF, CR and from ASCII code 32d to 126d) or up to 462 “bytes” (others bytes values).
- regarding error level, printer can automatically lower it so as to make printout possible when too much bytes should be encoded. For information, printing execution time is proportional to error level. It should be low when few bytes are encoded and high when a lot of bytes. Generally, it is recommended to set level 5 only from 321 data “CodeWords”.
- regarding number of columns, printer can automatically adjust it so as to meet number of lines requirements (3 to 90).
- in any case, when a feature is not valid and cannot be adjusted automatically, printing is then not performed and data are dumped.
- “Macro PDF417” is not managed.

- PDF417 bar code encoding requires big tables of data and so big space in printer flash. Basically, these tables are not part of main firmware which enables to remove them without updating firmware. Therefore, this feature enables to free flash space for customer who does not use PDF417 and who needs more flash space for bigger customized fonts. By default, these tables are provided with main firmware; cf. “9.2 Firmware updating” for removing/downloading operations. Please note that if tables have not been loaded while a PDF417 bar code is requested, printing will then not be performed and data will be dumped.

Example :



## GS h n

Description: Set bar code height.

Format: <1Dh> <68h> <n>

Comments: Set bar code height as multiple of 1/8 mm. n should range from 1 to 255. When rotated bar code, printed height will be rounded up to mm. When PDF417, it sets basically the height of each line (cf. example above; n=8 (i.e. 1mm) is advised).

Default: 128 (16mm).

## GS w n

Description: Set bar code magnification.

Format: <1Dh> <77h> <n>

Comments: Set bar code module width as multiple of 1/8mm. Module is the elementary bar/space on which bar code symbol is based. n should range from 2 to 6.

Default: 3.

Notes:- **bar code ratio** between thick and thin line is **2:1**.

- bar code printing is always **centred**.

- when magnification is too important, bar code width may exceed printing width. Therefore, bar code will be printed out from left paper side and truncated.

## GS H n

---

Description: Set bar code text position.

Format: <1Dh> <48h> <n>

Comments: Set position where HRI text (Human Readable Interpretation) of next barcodes will be printed out.

n	TEXT POSITION
0	Not printed
1	Above bar code
2	Under bar code
3	Above and under bar code

Default: HRI is not printed.

Notes: - HRI is printed out with the latest text features (font, width, height ...) and always centred.

- when PDF417, HRI text is never printed out. Before PDF417 printing, this feature is set automatically to “Not printed” and it is not restored after printing !

## GS R n

---

Description: Set normal/rotated bar code.

Format: <1Dh> <52h> <n>

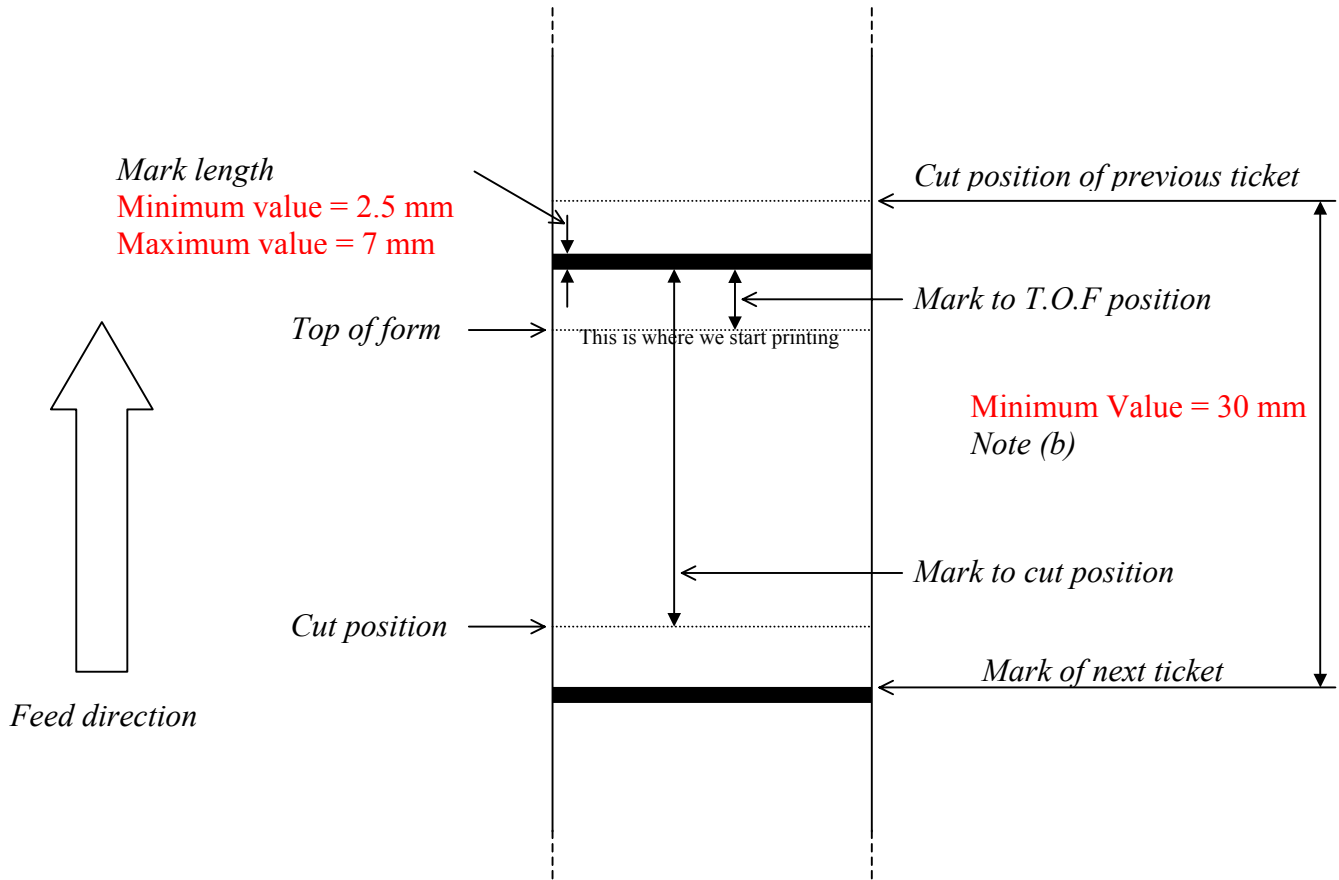
Comments: n = 0: bar code printing is normal (horizontally),

n = 1: bar code printing is rotated of 90° (vertically).

Default: normal.

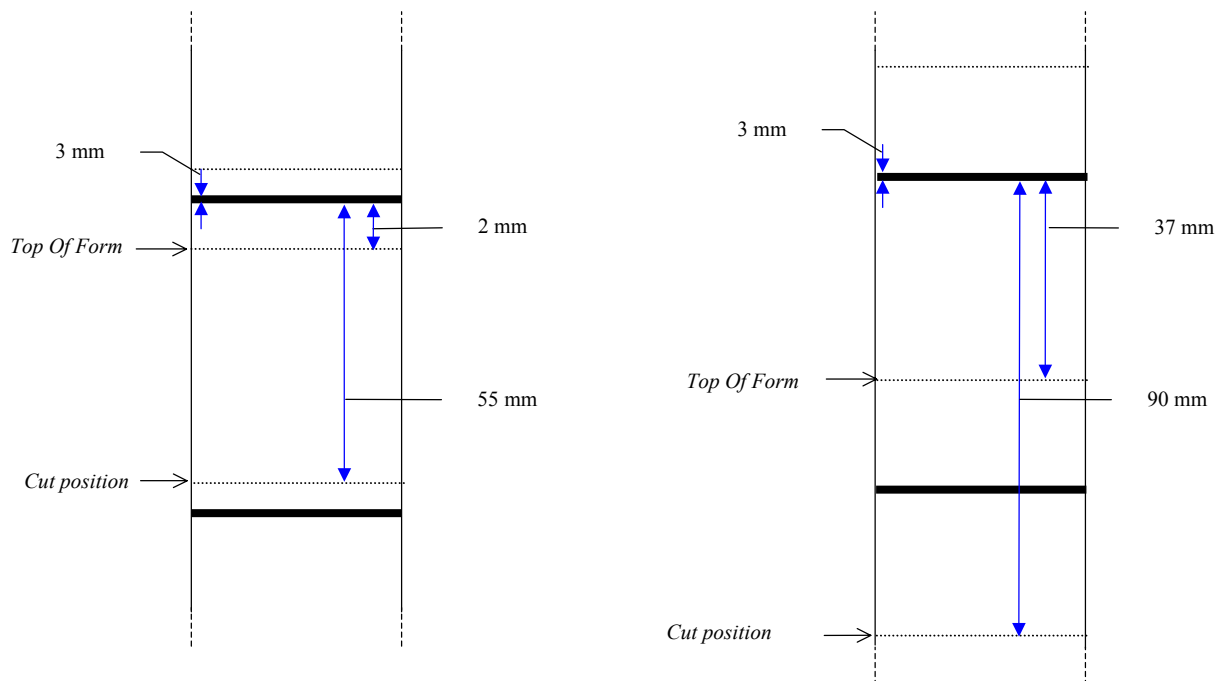
Note: when PDF417, printing is always horizontal. Before PDF417 printing, this feature is set automatically to “Normal” and it is not restored after printing !

## 8.4.7 Hole / black mark detection control codes



Remark : hole/black mark is detected by “End of Paper” optosensor.

### EXAMPLES:



## GS L n

---

Description: Set mark length, switch continuous/mark mode.

Format: <1Dh> <4Ch> <n>

Comments: Set mark length and switch from continuous paper feed to mark detection mode. n specifies mark length as multiple of 1/8 mm. n should range from 20 to 56 (=> mark length should range from 2.5 mm to 7 mm).

Default: 0 (continuous paper feed mode).

Example: mark length = 3mm => n = 24 should be transmitted and printer enters immediately in mark detection mode.

Notes:- n = 0 : printer switches into continuous paper feed mode.

- this command clears also the hole/black mark detection error bit of printer status.
- when switching in mark detection mode, paper error bit of printer status is cleared too.
- mark length is a parameter which can be saved in flash in order to recover directly the right continuous/mark detection mode at each power-on/reset.
- when mark detection mode becomes active at power-on/reset or via this command and no paper has been inserted, "End of Paper" status is not set because printer cannot make difference between "End of Paper" and a "Mark" while enough paper has not been fed. So, at power-on/reset, an automatic forward paper feeding of 2cm is performed so as to be able to detect possible "End of paper". If paper is well detected, an automatic backward paper feeding is then performed so as to return to initial position (TOF for example) and enables immediate start of printing. If no paper is detected, an "End of Paper" status is then set which enables an automatic paper loading. In the other case (mark detection mode activation via this command), "End of Paper" status stays cleared until first "Move to TOF" command which will make paper feeding until actual "End of Paper" detection. At this moment, automatic paper loading becomes available again.
- when green lever is turned to head-down position while mark detection mode is active, the same automatic "forward/backward" paper feeding is performed so as to detect possible "End of paper" and to make automatic paper loading available too.

## GS E

---

Description: Move to "Top Of Form" (TOF) position.

Format: <1Dh> <45h>

Comments: Paper is fed until next TOF position is under head dot line.

Note : hole/black mark detection error bit of printer status is set if no mark has been found during this operation after 50cm of paper feeding. Moving to TOF position is then stopped and printing operations are disabled. To remove this error and enable again printing operations, a "GS L n" command is required.



## GS T n1 n2

---

Description: Set “Mark” to TOF position length.

Format: <1Dh> <54h> <n1> <n2>

Comments: Set length L between end of mark and first printable line (TOF) as multiple of 1/8 mm.  
 $L = (256 * n1) + n2$ .

Default: 0 mm.

Note: negative length is allowed. Negative value is represented with the two’s complement of absolute value. For example, to set a “-5 mm” length :

$$L = -40 = 65536 - 40 = 65496 \Rightarrow n1 = 255, n2 = 216.$$

## GS Y n1 n2

---

Description: Set “Optosensor” to “Head dot line” length.

Format: <1Dh> <59h> <n1> <n2>

Comments: Set paper path length L between optosensor position and head dot line as multiple of 1/8 mm.  
 $L = (256 * n1) + n2$ . This data is required to be able to move with accuracy to TOF and cut positions from end of mark detection.

Default: 104 (13mm, whatever mounting configuration of optosensor (front or bottom)).

Notes: - this control code should be used only if optosensor position is different from default.  
- no negative length is allowed; maximum value is 32767.

## GS X n1 n2

---

Description: Set “Mark” to “Cut” position length.

Format: <1Dh> <58h> <n1> <n2>

Comments: Set length L between end of mark and cut position as multiple of 1/8 mm.  $L = (256 * n1) + n2$ .

Default: 0 mm.

Note: no negative length is allowed; maximum value is 32767.

## GS x n1 n2

---

Description: Set “Head dot line” to “Cut” position length.

Format: <1Dh> <78h> <n1> <n2>

Comments: Set paper path length L between head dot line and cut position as multiple of 1/8 mm.  
 $L = (256 * n1) + n2$ . This data is required to be able to move with accuracy to cut position from end of mark detection.

Default: 88 (11 mm).

Notes: - this control code should be used only if cutter’s blade position is different from default.  
- no negative length is allowed; maximum value is 32767.

### NOTES:

- (a) Make sure that hole/black mark fully covers optosensor window, according to selected paper path (front or bottom).
- (b) Distance between cut position of previous ticket and mark of next ticket should be higher than distance (in terms of paper path) between cutter and optosensor (default : 24.75mm). A minimum distance of 30mm should provide reasonable margin.
- (c) To get optimum performance, paper should be guided, and in particular, distance between paper and optosensor should be kept as constant as possible so as to make hole/black mark detection the most accurate as possible. In addition, it is advised to perform a calibration of “End of Paper” optosensor with paper used (calibration should be performed in continuous paper feed mode).
- (d) During automatic paper loading, if insertion of paper failed just at the end of this automatic step, it might not be possible to make a new attempt because failure might not be detected (if not enough feeding without paper, it might be a mark too) and so, automatic paper loading will stay disabled. The only way to make this step robust is to set good conditions of paper loading (long paper loading length with “GS P n1 n2”, pause before starting paper loading with “GS p n”). Otherwise, if this kind of failure occurs, only a paper feeding operation could make automatic paper loading enabled again (“Move to TOF” of a test ticket for example, ...).

## 8.5 Real time control codes

2 requests are processed in real time: “**Send printer status**” and “**Reset printer**”. However, the suitable way to send these requests depends of communication line and context.

### 8.5.1 RS232

Bytes are received one after one directly by UART of microcontroller. So they can be processed whatever printer state (printer error or full buffer). In the case of full buffer (handshaking was set OFF by printer with “RTS or DTR” or Xoff), host should disable its own handshaking control before sending its real time request. Otherwise this control will prevent request to be sent.

- “**ESC v**” will be used to send a “Send printer status” request. Note that during graphic printing, it is possible that printer interprets graphic data as a “Send printer status” request (edge effect of real time processing implementation). Then printer will return its status. So when host needs to get some data from printer, it is advised to reset its receiving buffer before sending its request (in the case when this noisy answer has been received previously).
- “**ESC @**” will be used to send a “Reset printer” request. Note that this one will be processed in real time only if printer is in error so as to prevent the same edge effect explained above (fatal consequence this time !).

### 8.5.2 USB

Bytes are received by packets not directly by microcontroller but an external USB hardware interface driver. This USB driver handles 3 communication channels: “DATA\_IN”, “DATA\_OUT” and “CONTROL\_IN\_OUT”. Host will use mainly “DATA\_IN” channel to send its printing data and its control requests (printer control codes). However, this channel is not a real time channel unlike “CONTROL\_IN\_OUT”. Therefore, when firmware buffer is full, packets stay in this external component which becomes full too. This component manages itself handshaking with host. So communication is “suspended” until enough space is free to stock new packet. So when printer is in error and buffer is full, it is impossible for host to send its real time request with using this “DATA\_IN” channel. “CONTROL\_IN\_OUT” has to be used in place because this is really a real time channel. However, allowable requests in this channel are regulated by USB standard (cf. “*USB Device Class Definition for Printing Devices*”, version 1.1.). 3 kinds of requests are defined: “Standard Requests”, “Class Specific Requests” and “Vendor Requests”. The 2 first kinds are already defined by USB standard; only last kind is intended to specific user needs. Therefore, available status provided by “GET\_PORT\_STATUS Class Specific Request” is too limited. So, in order to retrieve extended printer status (the one returned with printer control code “ESC v”), a specific “Vendor Request” has been added to the set of standard USB printer requests: APS\_GET\_HRS\_KIOSK\_STATUS. In addition, available reset operation provided by “SOFT\_RESET Class Specific Request” is not suitable because only communication buffer would be reset. So, in order to provide a full reset operation (as a hardware reset during a power cycle), a specific “Vendor Request” has been added too: APS\_HARD\_RESET.



Table below gathers all USB requests (standard and specific) and gives for each one their handling status:

REQUEST TYPE	LABEL	SUPPORTED	DETAILS
Standard	GET_STATUS	Yes	Cf. USB standard
	CLEAR_FEATURE	Yes	Cf. USB standard
	SET_FEATURE	No	
	SET_ADDRESS	Yes	Cf. USB standard
	GET_DESCRIPTOR	Yes	Cf. USB standard
	SET_DESCRIPTOR	No	
	GET_CONFIGURATION	Yes	Cf. USB standard
	SET_CONFIGURATION	Yes	Cf. USB standard
	GET_INTERFACE	No	
	SET_INTERFACE	No	
	SYNCH_FRAME	No	
Class Specific	GET_DEVICE_ID	Yes	Cf. USB standard
	GET_PORT_STATUS	Yes	Cf. USB standard
	SOFT_RESET	No	
Vendor	APS_GET_HSP_STATUS	No	
	APS_GET_HRS_KIOSK_STATUS	Yes	<b>Cf. below</b>
	APS_HARD_RESET	Yes	<b>Cf. below</b>
	APS_ACK_ERROR	No	

Table below specifies the 2 “Vendor Requests” available on our CPxxxHRS printers:

Label	bmRequestType	bRequest	wValue	wIndex	wLength	Data
APS_GET_HRS_KIOSK_STATUS	11000001b (read dir vendor req i/f recipient)	0x01	0x0000	0x0000 (interface)	0x0001	1 byte received (see answer format below)
APS_HARD_RESET	01000001b (write dir vendor req i/f recipient)	0x02	0x0000	0x0000 (interface)	0x0000	[None]

## APS\_GET\_HRS\_KIOSK\_STATUS:

Printer answers with a 1 byte data buffer containing below extended status information:

BIT	FUNCTION	BIT = 0	BIT = 1
0	Head temperature	OK	Too high or too low
1	Head-up	No	Yes
2	End of Paper	No	Yes
3	Power supply	OK	Too high or too low
4	Printer in use	Ready	Action in progress
5	ON/OFF line	OFF	ON
6	Hole/mark detection error	No	Hole/mark not found
7	Cutter error	Yes	No

## APS\_HARD\_RESET:

Printer performs a self-reset (“hardware” reset as a power cycle). Therefore, printer restarts by a whole initialization (all parameters set to their default values, internal buffers cleared, ...) and host will view an USB disconnection/reconnection without prior acknowledgement of its request.

To sum up, when USB is currently used, host should send its real time requests via “CONTROL\_IN\_OUT” channel :

- “APS\_GET\_HRS\_KIOSK\_STATUS” will be used to send a “Send printer status” request.
- “APS\_HARD\_RESET” will be used to send a “Reset printer” request.

Note that these 2 requests are fully real time (no restricted conditions).

However, it is also possible to get status with “ESC v” via “DATA\_IN” channel but real time and well-execution are not guarantee; idem with “ESC @” via “DATA\_IN” when printer reset is required.

## 9. ADDITIONAL INFORMATION

### 9.1 USB

<b>APS VENDOR ID</b>	0x1AD4
----------------------	--------

<b>TYPE OF HRS / MODE</b>	<b>PRODUCT ID</b>
MONITOR MODE	0x0001
CP290HRS	0x0002
CP324HRS “STANDARD”	0x0003
CP324HRS “WIDE”	0x0004
CP424HRS	0x0005

### 9.2 Firmware updating

Software embedded in controller board flash is made up of several items :

<b>ITEM</b>	<b>ALWAYS PRESENT</b>	<b>DOWNLOADABLE</b>	<b>COMMENTS</b>
Monitor	Yes	No	<i>It is always the revision flashed in factory. Useful to download the other items.</i>
Firmware	No/Yes	Yes	<i>If not present, printer stays in monitor mode.</i>
Font	No	Yes	<i>If not present, no text can be printed out.</i>
PDF417 data	No	Yes	<i>If not present, PDF417 barcode cannot be printed out.</i>
Parameters	No	Yes	<i>If no present, default values are used.</i>

Each item can be downloaded alone (except monitor) and this structure enables very powerful and easy way to customize or update your printer. Therefore you can only upgrade the firmware while keeping your parameters settings. You can also remove PDF417 data so as to get enough space for another font if PDF417 is not used ...

A package is a group made up of some of these items above.

An APS PC tool is available to perform flash management : “*Flashing Utility*”.

This tool accepts only “.APS\_Package” working file (APS specific format which contains a group of items to be downloaded).

Unfortunately, flash cannot be fully managed for the moment. Current limitations are :

- “.APS\_Package” file can only be built by APS,
- update requires a “complete” package (with at least a firmware even if you just want to add a font for example) because during updating, all flash is firstly erased except monitor and parameters.

A “full features” revision should be available very soon. Please contact APS for further details.

## IMPORTANT NOTE :

Firmware can be updated either via USB or RS232.

Regarding RS232, flux control of communication is only managed by hardware and is bidirectional. So, all RS232 handshaking signals should be absolutely wired.

### 9.3 Font management

The controller board is provided with 3 resident fonts : 8x16, 12x20 and 7x16.

However, it is possible to remove them and add your own fonts. For the moment, no APS tool enables to perform the full process “creation/conversion/download”. Procedure should be only as following :

- Creation of customized font with the APS PC tool “**Font Editor**” : a bitmap font file is built (“.fna”).
- Conversion of this “.fna” file into an “.APS\_Package” file : only APS can perform this task for the moment.
- Downloading of this “.APS\_Package” file into printer with the APS PC tool “**Flashing Utility**” (cf. 9.2 *Firmware updating*).

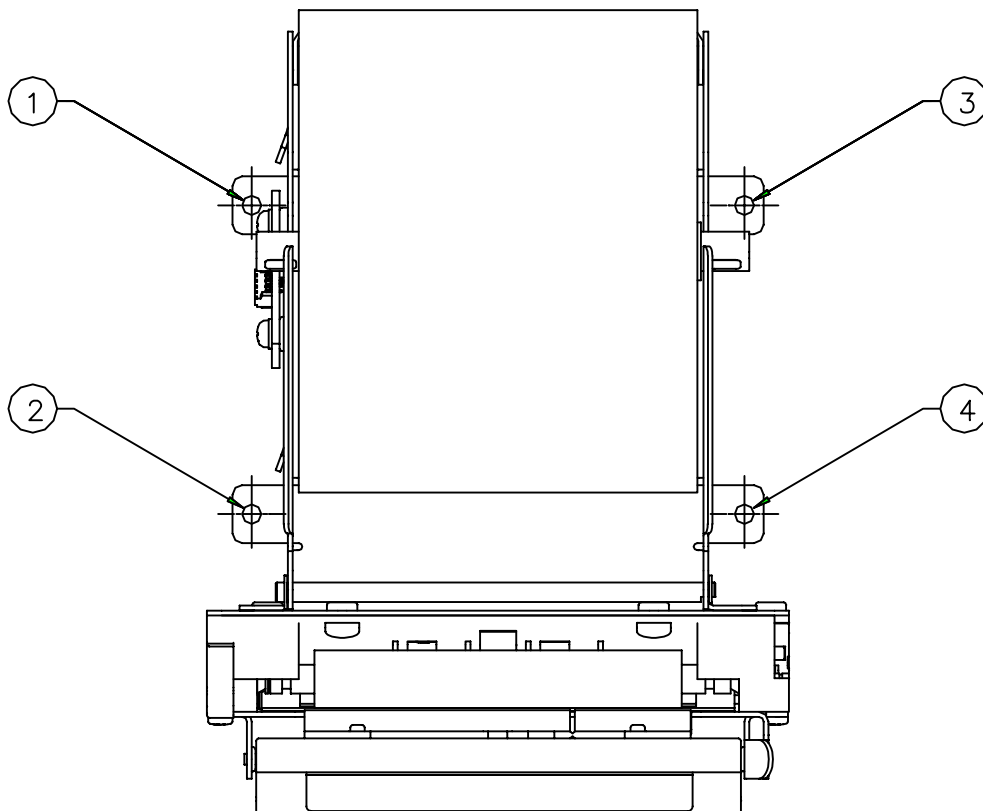
A “3-in-1” tool should be available very soon. Please contact APS for further details.

## 10. MECHANICAL & HOUSING

### 10.1 Mounting holes

The mechanism can be fixed by means of 4 screws Ø3, the figure below indicates the 4 fixing points to be used. See overall dimension drawing on attachment for further indications about fixing holes and others dimensions.

KM324-HRS-E Mechanism

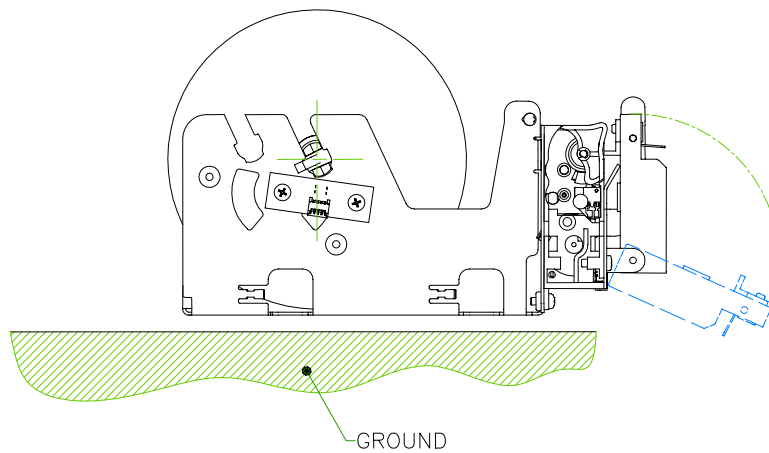




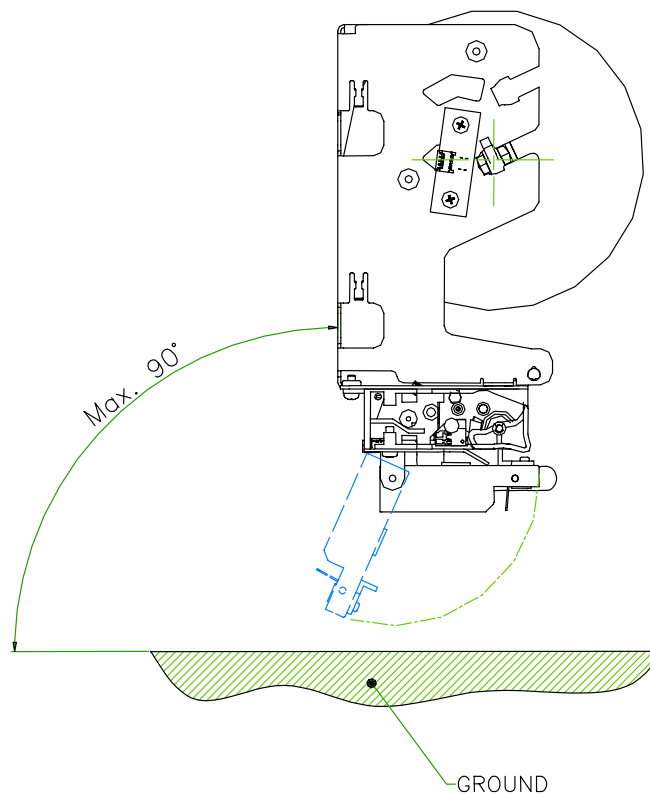
## 10.2 Mounting position

The mechanism can be mounted in horizontal or in vertical position as shown in the figure below. Intermediate angled positions between vertical and horizontal are allowed too, any other position is not allowed.

### Horizontal position



### Vertical position



## 10.3 Paper roll position

### KM324-HRS-E-BL (Bottom loading version)

Place the roll in seat (1), ensure complete insertion of the roll holder (the green plastic axle), see photo-1.  
 Do not use seat (2) as printing position of the roll.  
 The seat (2) can be used only as momentary position to hold the roll , for easy paper loading, once the paper has been loaded move the roll and place in seat (1).

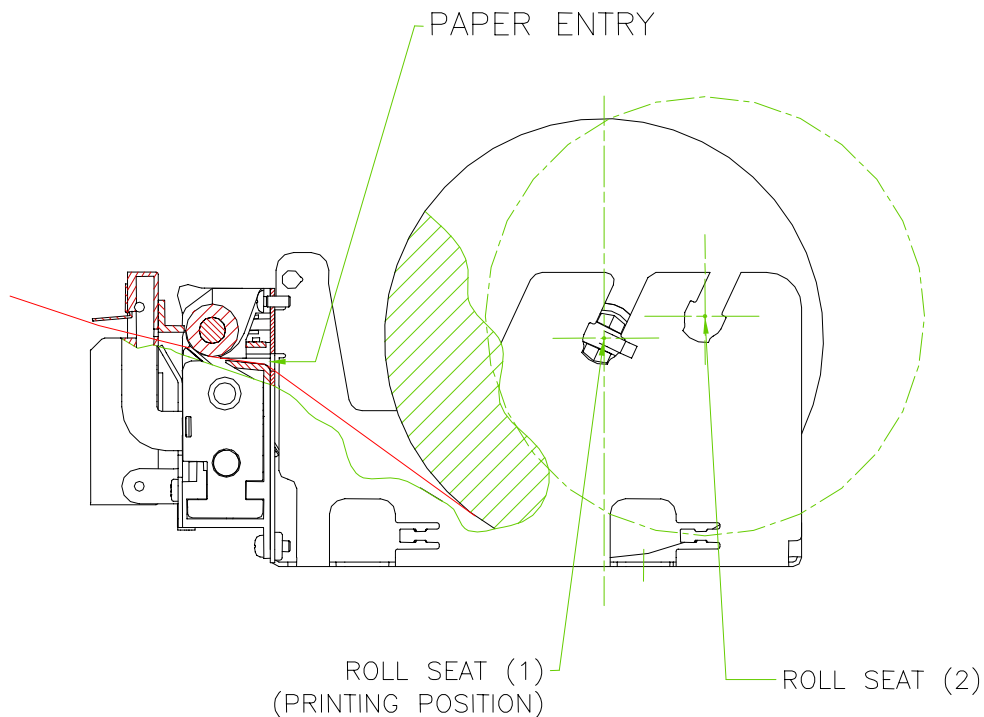


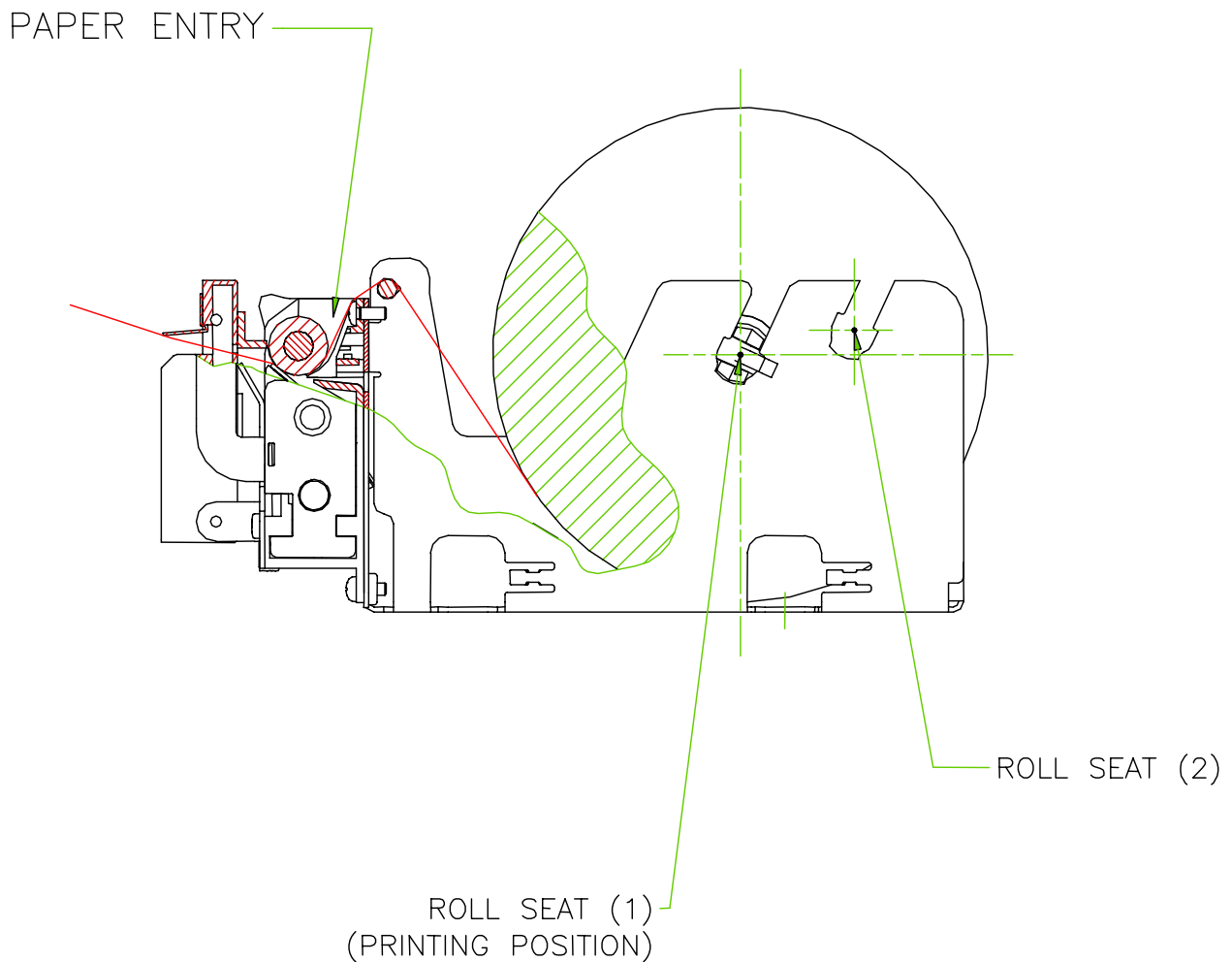
Photo-1



## 10.4 Paper roll position (continued)

### KM324-HRS-E-FL (Front loading version)

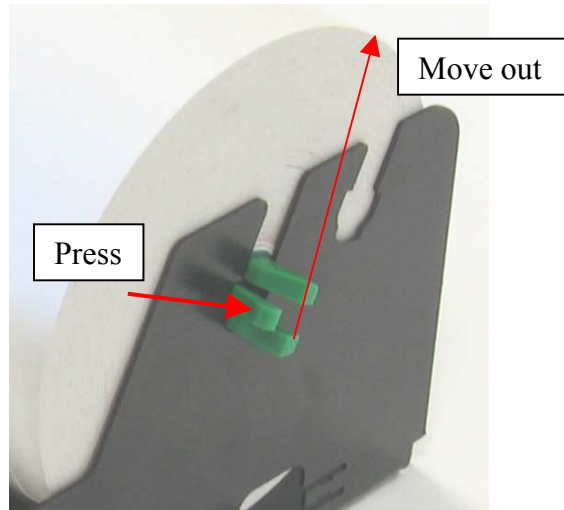
In the KM324-HRS-E-FL the paper loading operation is easier than KM324-E-BL version. Place the roll in seat (1), ensure complete insertion of the roll holder (the green plastic axle). Do not use seat (2) as printing position of the roll. The seat (2) is dedicated to customized printers, other bottom loading versions.



### 10.5 How to remove the roll holder correctly

To remove the roll holder, press the clips on both sides simultaneously then move out of the case (Photo-2)

Photo-2



## 11. ENVIRONMENTAL PROTECTION

The Products described in the present specification are conform to directive 2002/95-EC (RoHS) and following amendments issued by the European Council, the products don't contain the hazardous substances prohibited by the directive.

Declaration of RoHS compliance is available upon request.

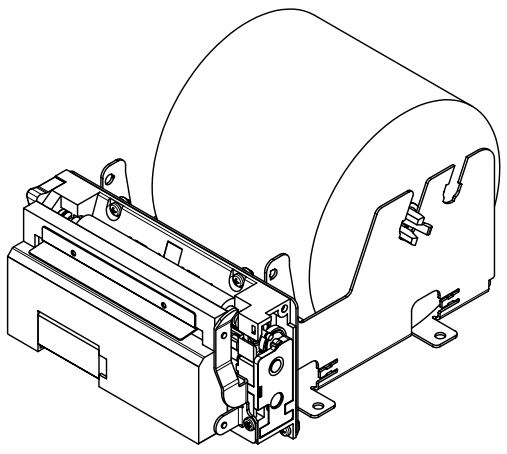
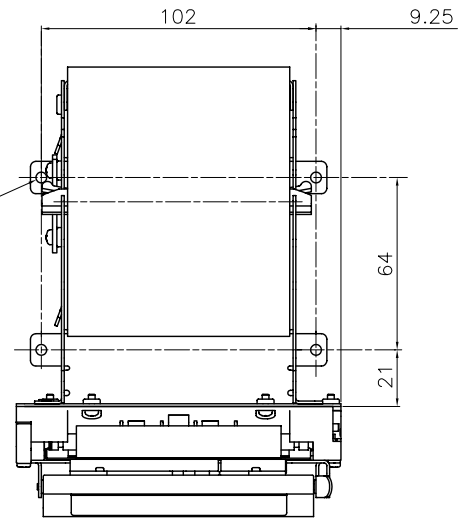
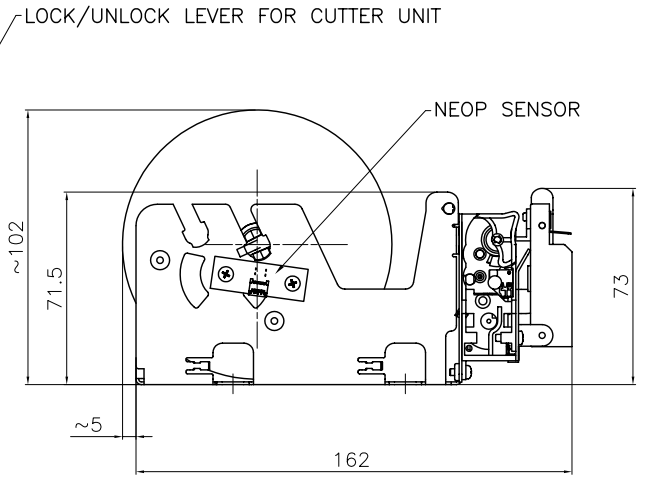
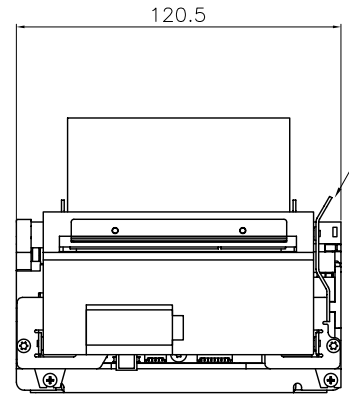
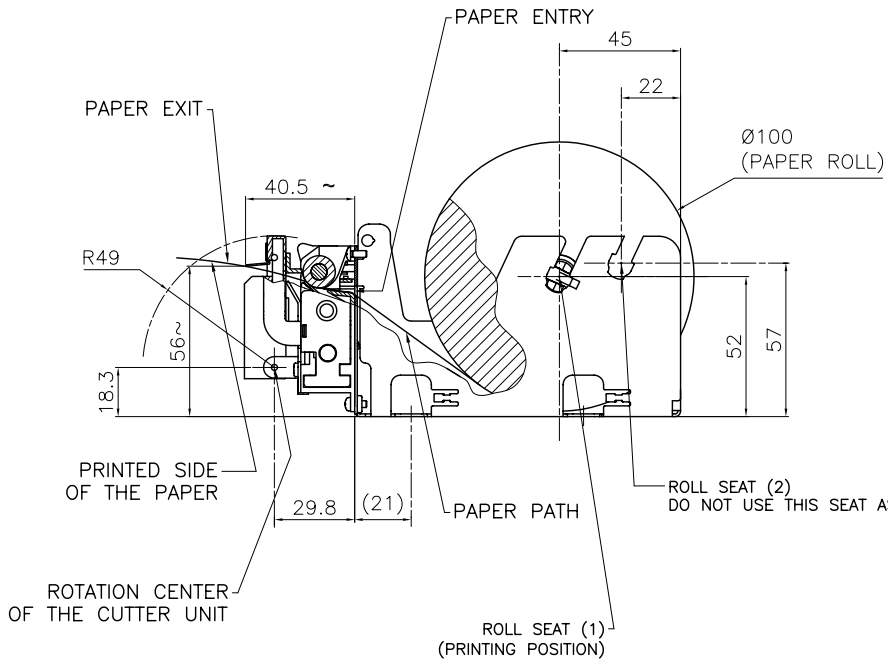
## 12. ORDERING CODES

MODEL	ORDERING CODE
<b>KM324-HRS-E-BL</b> (Bottom loading)	<b>90 600 104</b>
<b>KM324-HRS-E-FL</b> (Front loading)	<b>90 600 105</b>
<b>KEYBOARD KIT</b> (Optional)	<b>90 160 100</b>


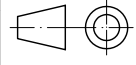
1 2 3 4 5 6 7 8

A  
B  
C  
D  
E  
F

REVISIONI (Revisions) A TDP 2008-111- ISSUED. (18-JULY-2008-FC)



RoHS compliant components  
According to 2002/95-EC and following amendments

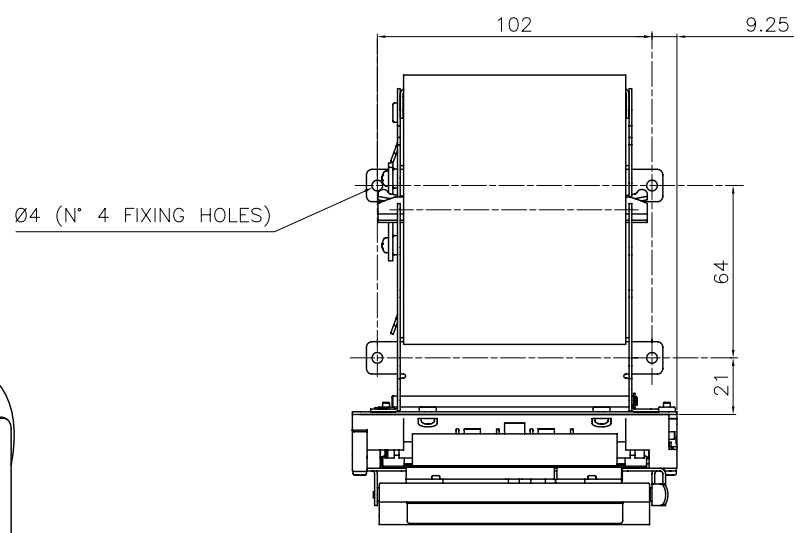
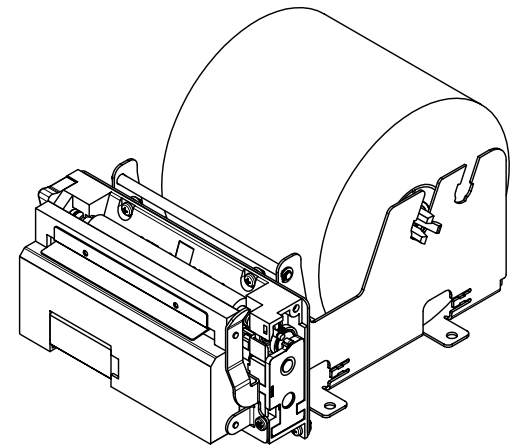
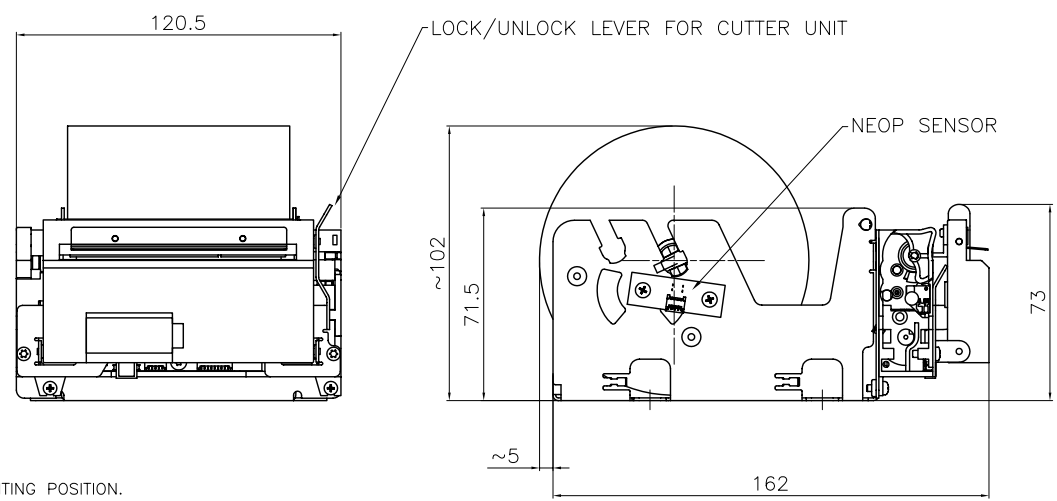
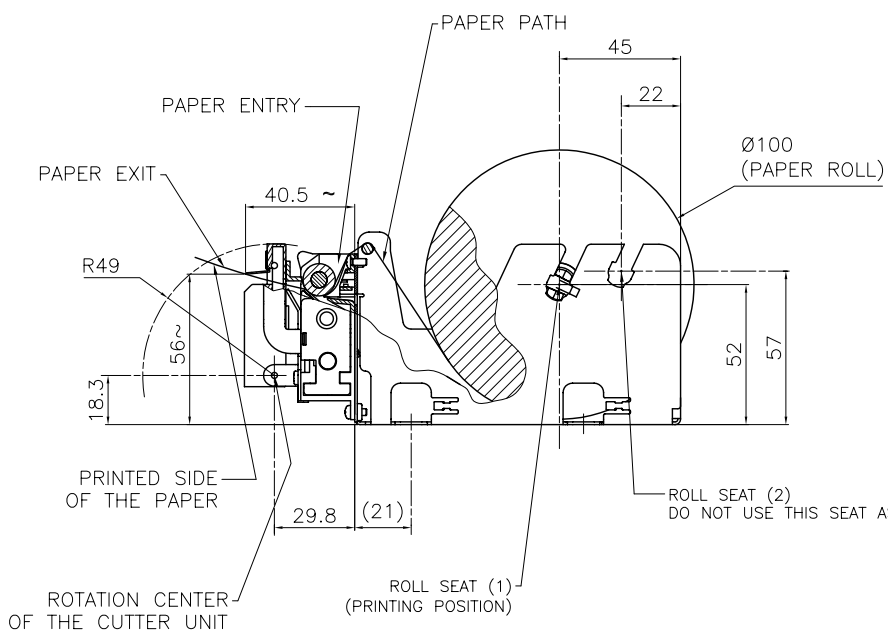
DENOMINAZIONE Denomination	I	MODULO KIOSK	MASSA Mass	UNITA'/Unit	  FORMATO/Size <b>A3</b>
	GB	KIOSK MODUL		mm	
	F	KIOSK MODULE	DISEGNATO DA Drawn by	VISTO DA Checked by	
	D	KIOSK MODUL	FC	AF	
DISEGNO D' INGOMBRO OVERALL DIMENSIONS DRAWING			DATA Date	SCALA Scale	FOGLIO/Sheet 1 / 1
			24-JUNE-2008	1:2	
CATALOGO/Catalog			N° DIS.-Draw. No		REV.
KM324-HRS-E-BL			90 600 104		A

1 2 3 4 5 6 7 8


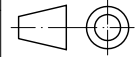
1 2 3 4 5 6 7 8

A  
B  
C  
D  
E  
F

REVISIONI (Revisions) A TDP 2008-111- ISSUED. (18-JULY-2008-FC)



RoHS compliant components  
According to 2002/95-EC and following amendments

DENOMINAZIONE Denomination	I	MODULO KIOSK	MASSA Mass	UNITA'/Unit	  FORMATO/Size <b>A3</b>
	GB	KIOSK MODUL		mm	
	F	KIOSK MODULE	DISEGNATO DA Drawn by	VISTO DA Checked by	
	D	KIOSK MODUL	FC	AF	
DISEGNO D' INGOMBRO OVERALL DIMENSIONS DRAWING			DATA Date	SCALA Scale	N° DIS.-Draw. No <b>90 600 105</b> REV. <b>A</b>
			24-JUNE-2008	1:2	
CATALOGO/Catalog			FOGLIO/Sheet 1 / 1		
<b>KM324-HRS-E-FL</b>					

1 2 3 4 5 6 7 8