

# **EPM203-HRS**

## **Technical reference**





## **EPM203-HRS Printer Technical reference**

### **A.P.S. ADVANCED PRINTING SYSTEMS**

A.P.S. reserves the right to make changes to the product, without notice, 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.

The warranty terms of the product are described in a separate document, ask A.P.S. to obtain this document.

Any part of the present document may not be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of A.P.S.

The information contained in the present document must be used for the purpose of operating or servicing the product to which the document relates, unless you obtain a prior written permission from A.P.S. for different use.

The contents of the present document are subject to change without notice. Please contact A.P.S. offices for the latest updating.

While every precaution has been taken in the preparation of this document, A.P.S. assumes no responsibility for errors or omissions.

## Preface

- This manual provides a complete technical information about **EPM203-HRS** thermal printer mechanism.
- For customized printers, **A.P.S.** supplies documentation in addition to the present specification.
- The present specification is valid also for customized types, where the different condition has not effects for common data (eg. different length of elec. Cables, colour case, etc.).

## 1. REVISION HISTORY

REV.	DATE	PAGE	REVISION ITEM	TDP	AUTHOR
A	11/feb/2009	-	Document issue	2009-029	GP-AF

## 2. TABLE OF CONTENTS

Preface.....	3
1. Revision history.....	3
2. Table of contents.....	4
3. General features.....	5
4. General specifications.....	6
5. Printer connectors.....	7
5.1 Connectors summary.....	7
5.2 Power supply connector.....	8
5.3 RS232 communication connector.....	8
5.4 USB communication connectors (J5 and J6).....	9
5.5 RS232 / USB mode selection.....	10
5.6 Switches and LED.....	10
6. Printer device operations.....	11
6.1 Self-test mode.....	11
6.2 Text printing fonts.....	12
6.3 Control codes.....	14
6.3.1 Control codes cross reference.....	14
6.3.2 Setup and hardware control codes.....	17
6.3.3 Text and general control codes.....	24
6.3.4 Graphic control codes.....	29
6.3.5 Bar code control codes.....	32
6.4 Real time control codes.....	36
6.4.1 RS232.....	36
6.4.2 USB.....	36
7. Mechanical and housing.....	39
7.1 Overall dimensions, outward appearance and fixing points.....	39
7.2 Mounting precautions.....	40
8. HANDLING THE EPM203-HRS.....	43
8.1 How to open the cover group.....	43
8.2 How to load paper rolls.....	43
8.3 How to close the Cover Group correctly.....	44
8.4 How to cut the paper correctly.....	46
9. Environmental Protection.....	46
10. Ordering code.....	46

## 3. GENERAL FEATURES

The EPM203HRS is a 2 inch, from 4V to 8.5V, Easy Loading Printing Module with integrated paper roll housing and integrated controller board using RS232 and USB communication interfaces.

- **High grade of integration and Easy Door Opening System**
- **Ultra light (120g)**
- **Easy loading printers**
- **Fully hot pluggable printers**
- **Sleep mode**
  - Current consumption <math><300\mu\text{A}</math>
  - Wake-up on serial/USB port or external switch
- **User extension:**
  - I<sup>2</sup>C, 2 GPIO (A/D), 3.3v, VBAT, GND
- **Software programmable consumption (dynamic division)**
- **Full control over printing quality/speed (up to 90mm/s)**
  - Speed clamping via control code.
- **2 communication ports**
  - RS232 (speed up to 115200 bauds),
    - RS232 C:  $\pm 12\text{V}$
    - RS232 TTL: 3.3V and 5V level compliant
  - USB 2.0 (full-speed).
- **Integrated keyboard with Paper Feed and ON-OFF Line push buttons**
- **Three internal default 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.
- **Hole / black mark detection**
- **Setup parameters saved in flash are recovered at next power-up**
  - One simple command saves all important parameters.
- **Easy firmware update**
- **Windows® and linux drivers available**

## 4. GENERAL SPECIFICATIONS

ITEM	SPECIFICATION
Printing method	Thermal dot-line printing
Dimensions WxDxH (mm)	Approx. 76.8 x 77.4 x 55
Weight (g)	Approx. 120 (without paper roll)
Paper roll size	Max. Ø32 mm (outside diameter)
Total number of dots	384
Dot density (dots/mm)	8
Paper width (mm)	58 +0/-1
Printing width (mm)	48 (centred on paper)
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
Recommended paper	JUJO-AF50KS-E (standard grade) JUJO-AF50KS-E3 (high sensitivity) Equivalent types can be used
Voltage range (Volts)	From 4 to 8.5V
Current consumption (Amps) <i>(can be limited with dynamic division)</i>	From 1.5A to 5A at 5V (peak for 3ms) <300µA (in sleep mode)
Operating temperature (°C)	From 0 to +50
Operating humidity (RH%)	From 20 to 85 (no condensation)
Storage temperature (°C)	From -25 to +70
Storage humidity (RH%)	From 10 to 90 (no condensation)
EMC standard	Designed to comply with FCC/CE Class B
Mechanism life	<b>Thermal printed head : 100 million of cycles</b> <b>Abrasion/wear resistance : 50 km of paper</b> <b>Cover Group Opening/closing : 2000 operations or more</b>

## 5. PRINTER CONNECTORS

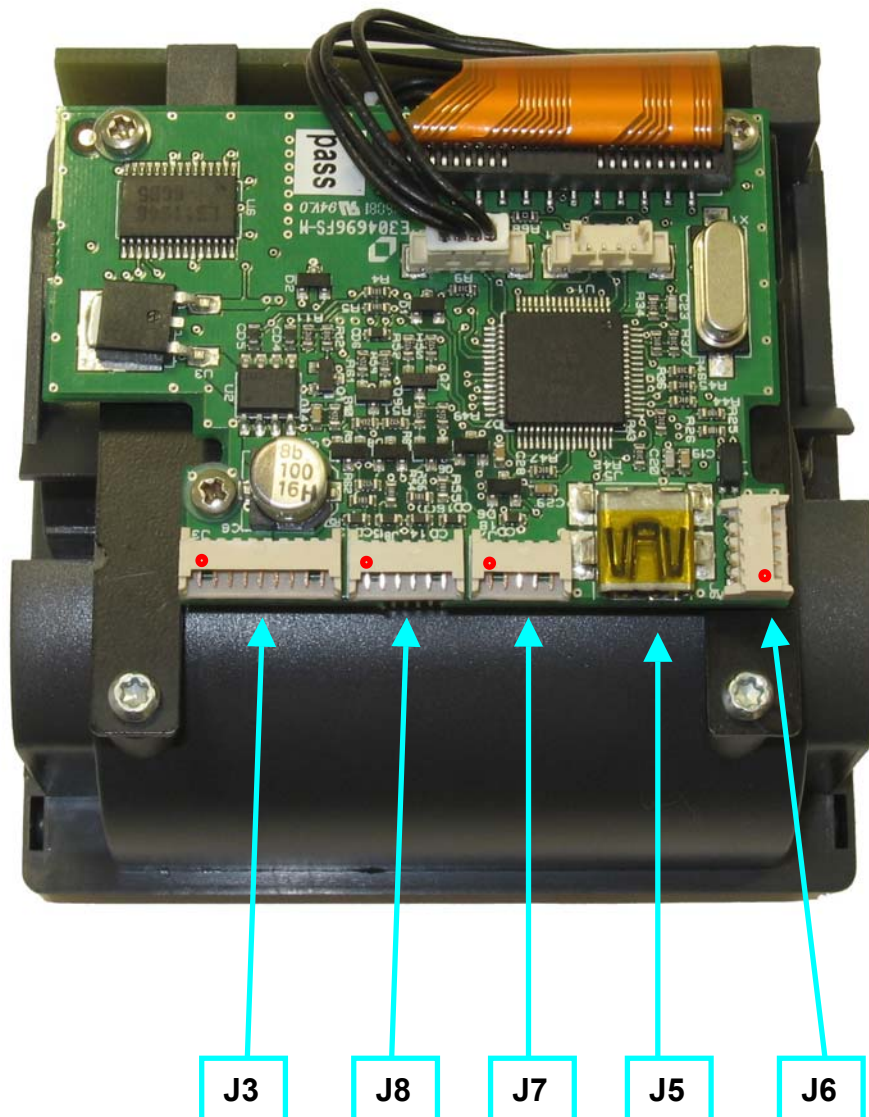
### 5.1 Connectors summary

Reference	Description
J3	Power supply
J5	Standard mini-B USB interface
J6	Low-cost USB interface
J7	RS232 serial interface
J8	RS232 TTL serial interface

Please refer to the figure below for the numbering and pin n.1 position of each connector available on the EPM203-HRS board.

Pin n. 1 position is indicated with a red dot directly on the connector.

This printer is fully hot pluggable: any connector shown here can be connected or disconnected without damaging printer.



## 5.2 Power supply connector

**Connector J3:** 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 4V to 8.5V DC.

## 5.3 RS232 communication connector

**Connector J7:** RS232C ( $\pm 12V$ ) it is a 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, not managed)
5	RTS or DTR (printer handshaking output)

**Connector J8:** RS232TTL (3.3V and 5V tolerant), it is aMOLEX 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, not managed)
5	RTS or DTR (printer handshaking output)
6	Not connected

**Note:** Printer TTL output (pin 2 and 5) are open drain type, so you need to add one pull-up (1K2 ohms recommended) between each output pin and host logic power.

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. Anyway, 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).

## 5.4 USB communication connectors (J5 and J6)

Two USB interface connectors are available:

- **Connector J5:** MOLEX 54819-0578, **standard** mini-B receptacle (compatible with MOLEX mini-B plug),
- **Connector J6:** 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 behavior may be affected. Please contact APS for further information.

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

Connectors **J5** and **J6** 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, 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.

## 5.5 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.

## 5.6 Switches and LED

Switches and LED functions are defined in the following table:

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:

- As soon as 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.



## 6.2 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. An APS tool enables you to perform this customization (please contact APS).

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 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 EPM203HRS (384 pixels width), number of characters per line can be up to 42 in standard text, 21 in double width, and 7 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	{		}	~	ó
8	€	ü	é	á	ä	à	ç	è	ë	ì	í	î	ï	ñ	ä	å
9	é	æ	ŕ	ø	ö	ù	û	ü	ö	ü	o	ø	ø	ŕ	ŕ	ŕ
A	á	í	ó	ú	ñ	ñ	ñ	ñ	ñ	ñ	ñ	ñ	ñ	ñ	ñ	ñ
B	☼	☼	☼		†	†	†	†	†	†	†	†	†	†	†	†
C	L	L	T	†	†	†	†	†	†	†	†	†	†	†	†	†
D	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø
E	ó	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø	ø
F	-	±	∏	¼	¶	§	÷	,	°	°	°	°	°	°	°	°

- 12x20 characters set:** minimum character area is actually *13 pixels* (12 “active dots” plus 1 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 EPM203HRS (384 pixels width), number of characters per line can be up to 29 in standard text, 14 in double width, and 7 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	{		}	~	
8	€	ü	é	â	ä	à	ç	ê	ë	è	ï	î	ï	Ä	Å	
9	É	æ	Æ	ô	ö	ò	ó	û	ü	ÿ	ö	Ü	ß	£	¢	ƒ
A	ã	í	õ	ú	ñ	ñ										
B	█	█	█		+	À	À	À	©	¶	¶	¶	¶	¶	¶	¶
C	L	⊥	τ	†	-	+	ä	Ä	ℓ	ℓ	π	π	π	π	=	¶
D	ð	þ	ê	ë	è	ï	î	ÿ								
E	ð	þ	ø	ö	ö	ö	μ	ℓ								
F	-	±	□	¼	¾	§	÷	·	°	..	.	!	3	z	■	€

- 7x16 characters set:** minimum character area is actually *8 pixels* (7 “active dots” plus 1 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 EPM203HRS (384 pixels width), number of characters per line can be up to 48 in standard text, 24 in double width, and 12 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	
9	É	æ	Æ	ô	ö	ò	ó	û	ü	ÿ	ö	Ü	£	¢	ƒ	
A	◦	「	」	、	・	ヲ	イ	ウ	エ	オ	ト	チ	リ	ヨ	ツ	
B	-	ア	イ	ウ	エ	オ	カ	キ	ク	ケ	コ	サ	シ	ス	セ	ソ
C	タ	チ	ツ	テ	ト	ナ	ニ	ノ	ネ	ヘ	ホ	フ	フ	フ	フ	フ
D	ミ	メ	ム	モ	ム	ム	ム	ム	ム	ム	ム	ム	ム	ム	ム	ム
E	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
F	▲	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

## 6.3 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.

### 6.3.1 Control codes cross reference

#### *Setup and hardware control codes*

CONTROL CODE	DESCRIPTION
<b>GS / n</b>	Set maximum peak current via dynamic division
<b>GS s n1 n2</b>	Set maximum printing speed
<b>GS D n</b>	Set printing intensity
<b>ESC @</b>	Reset printer
<b>ESC v</b>	Send printer status
<b>ESC I</b>	Send printer identity
<b>GS B n</b>	Set serial communication settings
<b>ESC o n</b>	Set “End of Paper” optosensor type
<b>GS O n1 n2</b>	Do an “End of Paper” optosensor calibration
<b>ESC O</b>	Send “End of Paper” optosensor parameters
<b>GS o</b>	Send “End of Paper” optosensor current level
<b>ESC s</b>	Save in flash setup parameters
<b>ESC d</b>	Recover factory setup parameters
<b>GS c n</b>	Enable/disable historic heat
<b>ESC S</b>	Puts the printer in sleep mode

## *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

## 6.3.2 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 = 11

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

Printer peak consumption @5V =

(0.30A (stepper motor) + 0.2A (logic power supply) + 8V\*48/176) = 2.68A

*176 Ohms is the dot resistance.*

**At the moment, minimum available regarding n is 10 for EPM203HRS.**

### 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 90mm 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 = 1389 $\mu$ s (90mm/s).

Example : (n1 = 4, n2= 226) => T = 2083  $\mu$ s

=> Maximum printing speed = (1 / (8 \* 2083e-6)) = 60 mm/s

*8 dot lines/mm is the line pitch.*

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

Vmin = 10mm/s and Vmax = 90mm/s.

b) Maximum speed is also limited in function of the power supply voltage.

### 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:

EPM203HRS                      X.XX

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 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).

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,
- 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.

#### “Calibration” parameters

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

## ESC S

---

Description: Puts the printer in sleep mode

Format: <1Bh> <53h>

Comments: This command puts the printer in sleep mode giving the major benefit of very low power consumption ( $<300\mu\text{A}$ ).

There are 3 ways of waking the printer up:

- Through the USB port by sending event (data, status etc...)
- Through the serial port by sending the character "00 hex" (wake-up character)
- Press the paper feed button

Note:

1. Sleep mode consumption is increased to  $700\mu\text{A}$  if one theses conditions are not true:  
On RS232C: CTS voltage pin must remaine lower than 0.2V.  
On RS323TTL: CTS voltage pin must remaine upper or equal than 3V2.
2. Sleep mode consumption is increased to 17mA if host usb is not switched in stanby mode.
3. Wait 500 ms before sending the next character for the printer to execute the power-up sequence.

When waking-up through the serial port, the wake-up character will be ignored

### 6.3.3 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, please contact A.P.S.

#### ESC R n

Description: Select international characters set.

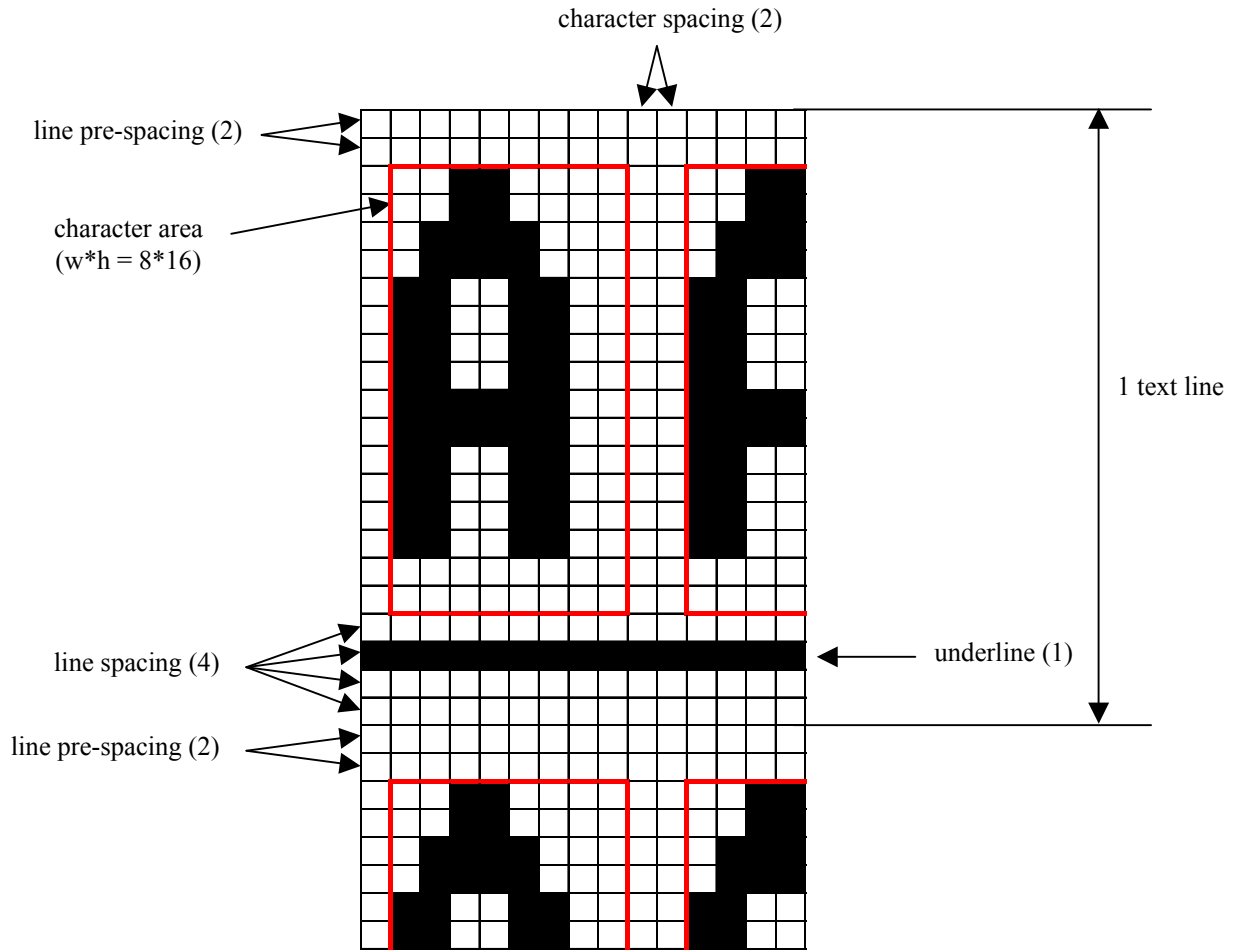
Format: <1Bh> <52h> <n>

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	#	¤	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	#	¤	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 3 to 15 dot lines. Line spacing pitch is 1/8mm.

Default: 3.

Note: underline is included in this set of dot lines, it is the 2<sup>nd</sup> dot line.

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 1 to 16 dots. Character spacing pitch is 1/8mm.

Default: 2.

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).

## 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).

## 6.3.4 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).

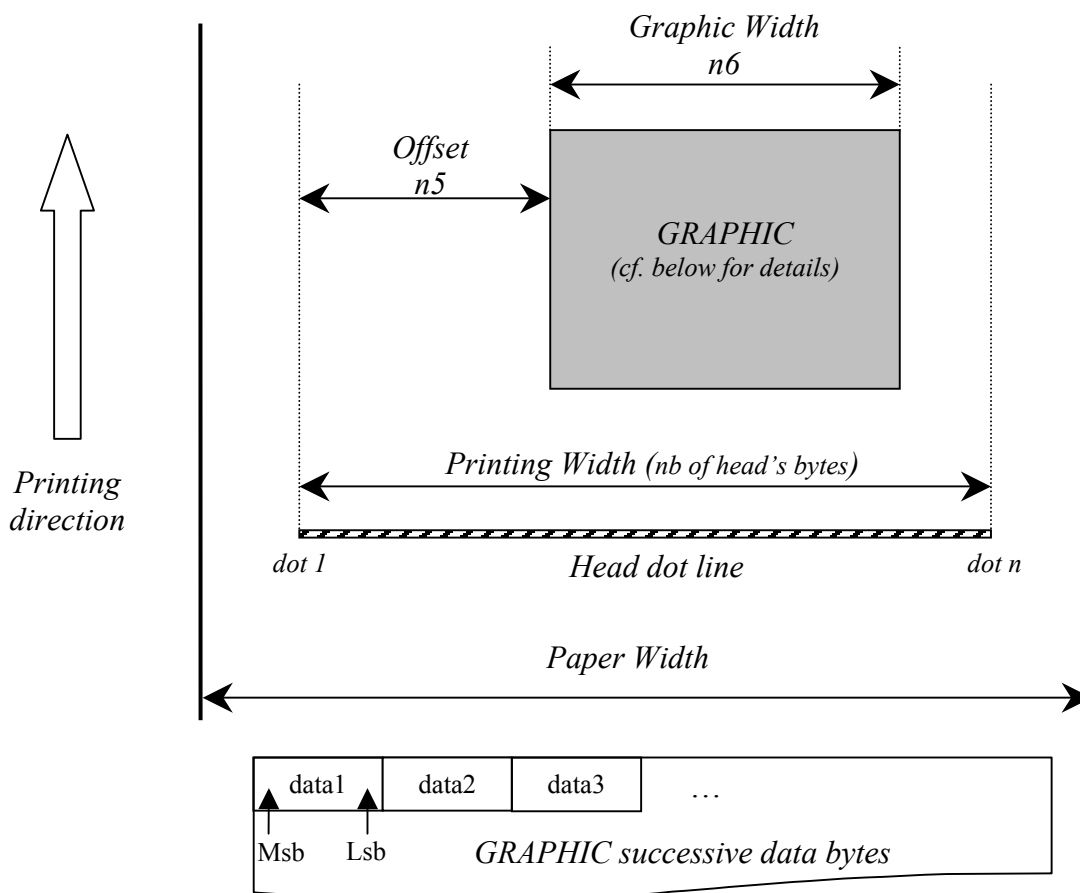
Byte n5 sets the number of dot bytes to be skipped (offset) before printing out the first graphic bit:

- 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 EPM203HRS is  $384 \text{ dots} / 8 = 48 \text{ bytes}$ .

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 EPM203HRS (384 pixels),

Image size =  $368 * 242 / 8 = 11\,132 \text{ bytes}$

$n1 = 124d, n2 = 43d, n3 = 0d, n4 = 0d, n5 = 4d, n6 = 46d$  or  
 $n1 = 7Ch, n2 = 2Bh, n3 = 00h, n4 = 00h, n5 = 04h, n6 = 2Eh$ .

---

## ESC \$ n1 n2

Description: Set graphic offset in line mode.

Format:  $\langle 1Bh \rangle \langle 24h \rangle \langle n1 \rangle \langle n2 \rangle$

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 ( $< \text{total number of head's bytes}$ ).  $N = (256 * n2) + n1$ . 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 EPM203HRS is  $384 \text{ dots} / 8 = 48 \text{ bytes}$ .*

---

## ESC V n1 n2 n3 <data>

Description: Print graphic in line mode.

Format:  $\langle 1Bh \rangle \langle 56h \rangle \langle n1 \rangle \langle n2 \rangle \langle n3 \rangle \langle \text{data} \rangle$

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 ( $\leq \text{total number of head's bytes}$ ):

$N = (256 * n3) + n2$ . 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 EPM203HRS is  $384 \text{ dots} / 8 = 48 \text{ bytes}$ .*

## 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.

## 6.3.5 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
	136d	Code 128 (start with subset B)	00h
	137d	Code 128 (start with subset C)	00h
	138d	Code 128 (A,B,C: automatic subset selection)	8Bh
8	No	PDF417	No

*MRS compatibility not yet implemented*  
*MRS compatibility not yet implemented*  
*MRS compatibility not yet implemented*

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.
- 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.

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

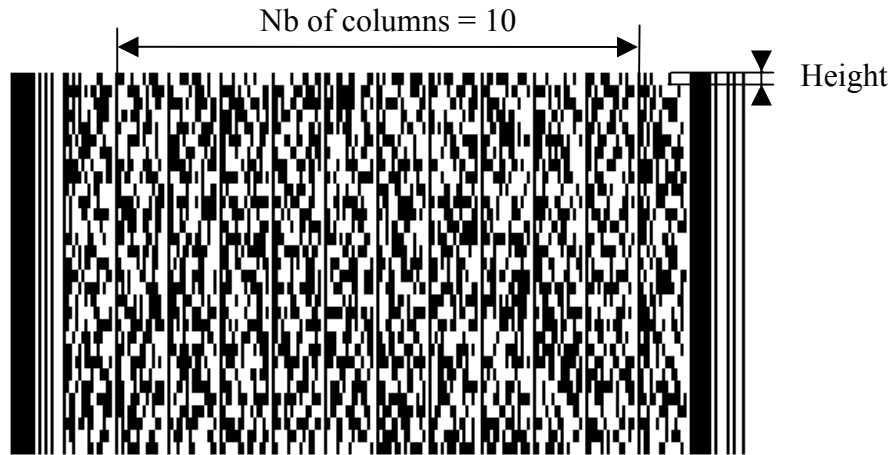
<n1> <n2> <n3> <n4> <n5> <data1> <data2>

- 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; please contact APS for removing/downloading operations (APS tool). 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 !

## 6.4 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.

### 6.4.1 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 !).

### 6.4.2 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 EPM203HRS 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.

**7. MECHANICAL AND HOUSING****7.1 Overall dimensions, outward appearance and fixing points**Overall dimensions:

See attached drawings for overall dimensions and recommended screws.

3D-IGES file, for mechanical details, is available upon request, ask APS for more information.

Standard outward appearance:

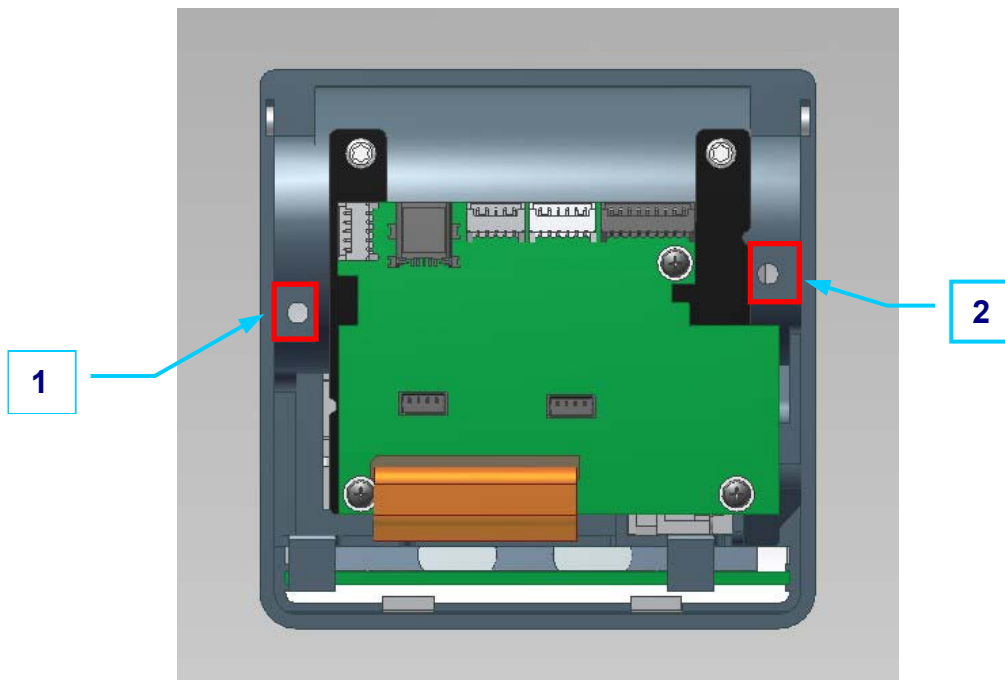
Housing colour is grey and the cover lever is transparent dark grey. For details please refer to photo on front page of the present document.

Fixing points:

The mechanism has to be fixed using the fixing points 1 and 2 provided for this purpose.

To avoid any kind of deformation or distortion, a flat surface for contact areas is required, if not, the print quality and printer's life will be drastically reduced.

The image below, shows the matching areas to be used for fixing, they are highlighted in red colour.

**Bottom view of EPM203-HRS**

## 7.2 Mounting precautions

Orientation according to figure A-B is to be preferred (i.e. parallel to the ground plane); reliability and life tests have been based only according to this orientation.

Alternatively, it is possible to choose different orientation angles as shown in figures C-D-E.

Fig. A

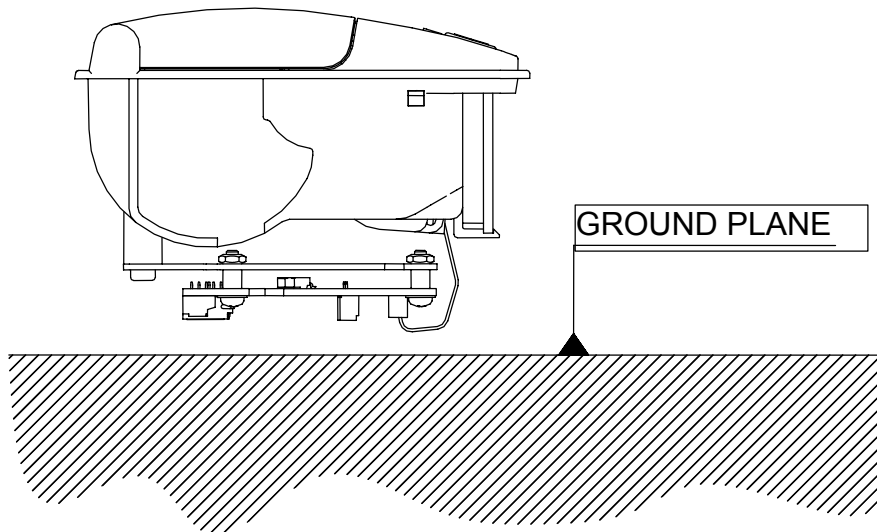
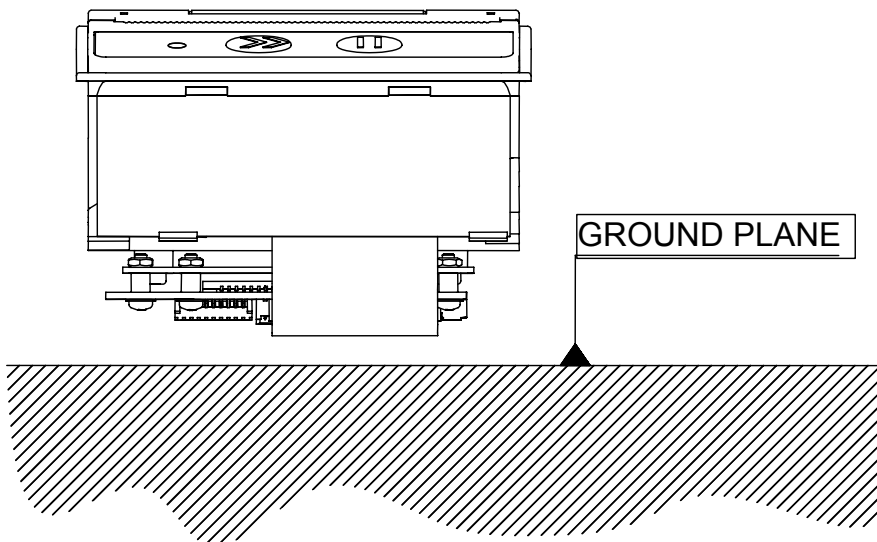


Fig. B



## Mounting precautions (continued)

Fig. C

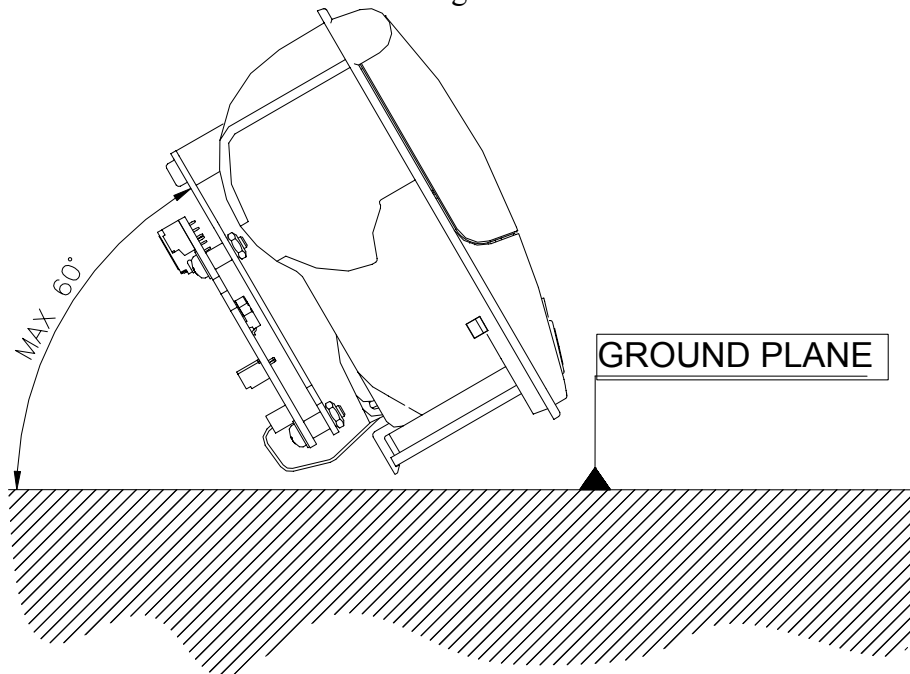
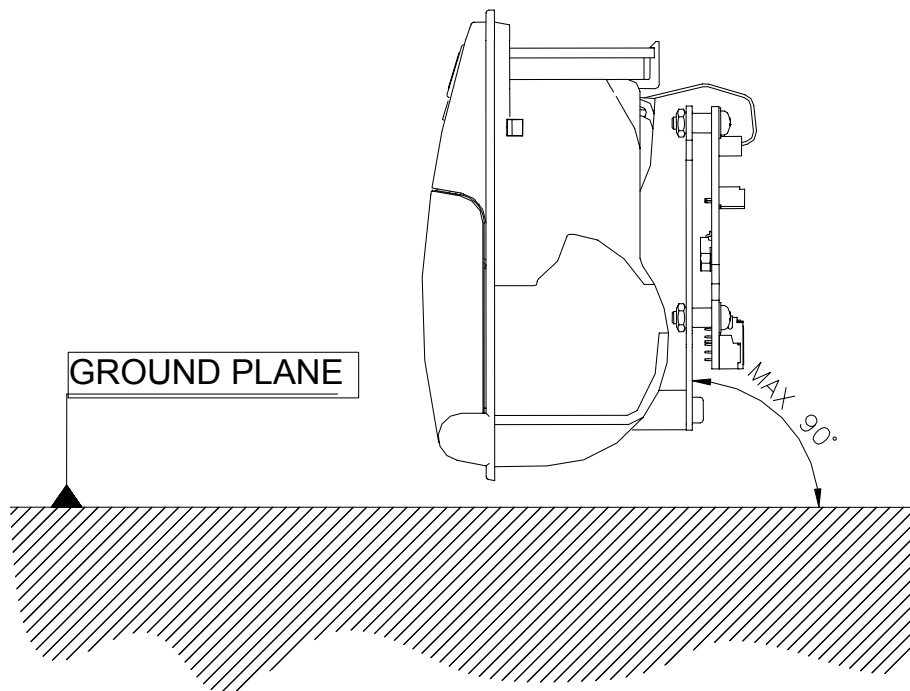
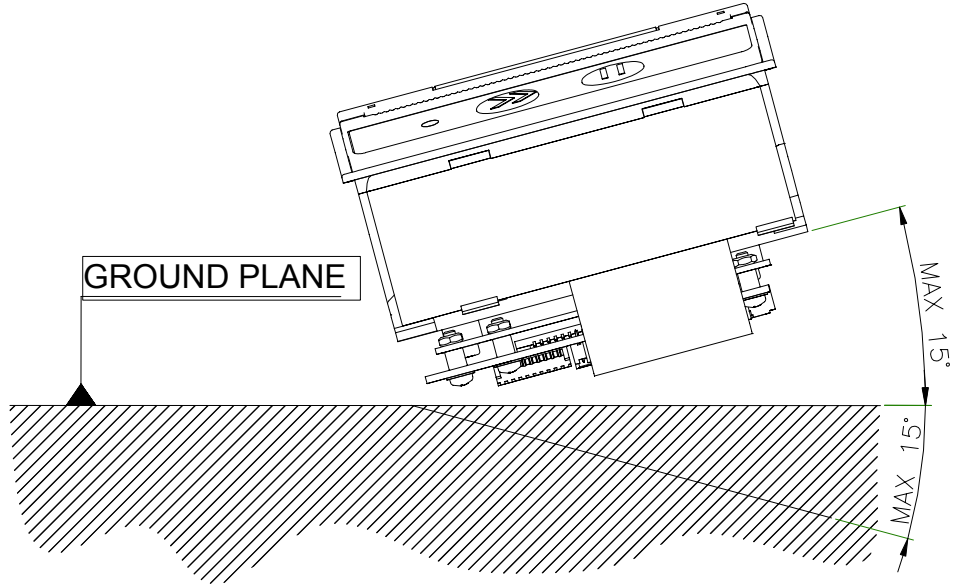


Fig. D



Mounting precautions (continued)



## 8. HANDLING THE EPM203-HRS

### 8.1 How to open the cover group

Pull the lever until the Cover Group is released from its locking position.  
To avoid damages to the lever do not use excessive force.



### 8.2 How to load paper rolls

#### STEP 1



**STEP 2****8.3 How to close the Cover Group correctly**

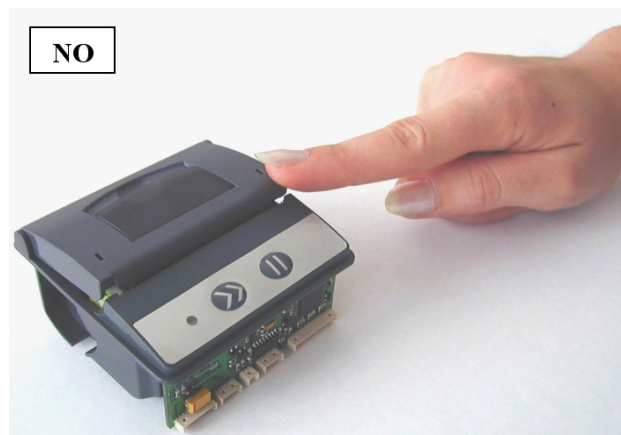
Press on both sides of the Cover Group simultaneously.



Alternatively:  
Press on the middle area of Cover Group, near the paper exit.



**Do not close the Cover Group pressing only on one side.**



## 8.4 How to cut the paper correctly

Pull the paper towards the Tear Bar from one side to the other.



## 9. ENVIRONMENTAL PROTECTION

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

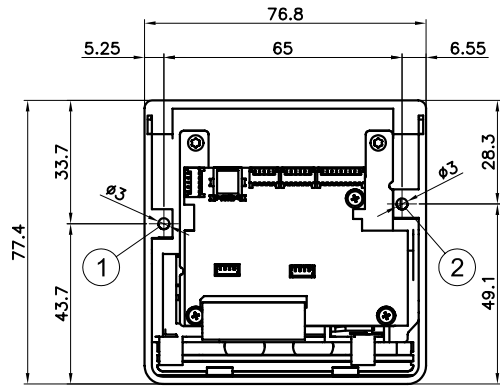
Declaration of RoHS compliance is available upon request.

## 10. ORDERING CODE

MODEL	ORDERING CODE	RoHS COMPLIANT
<b>EPM203-HRS</b> (Grey colour case)	<b>002 90 770 003</b>	<b>Yes</b>

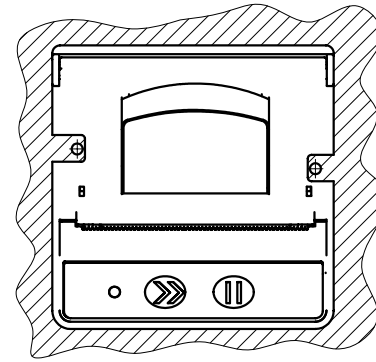
REVISIONI (Revisions) : [A] [TDP-2009-029] ISSUED. (11-FEB-09-FP)

① ② **FIXING POINTS**

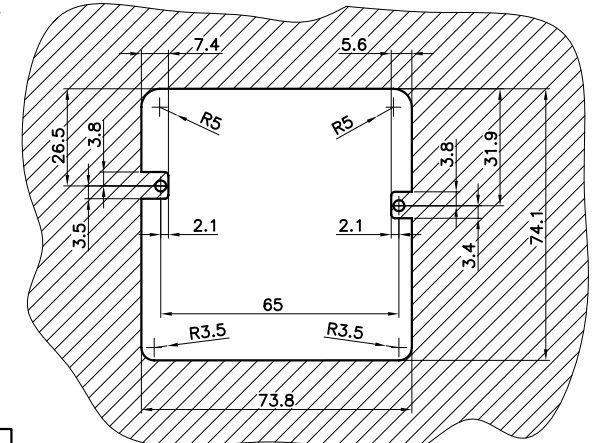


**BOTTOM VIEW**

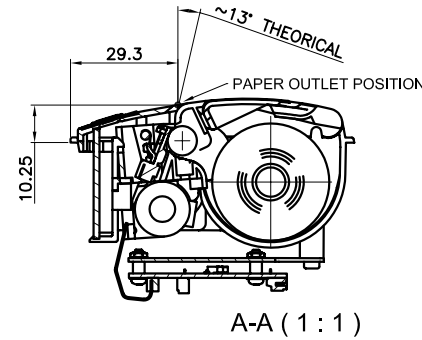
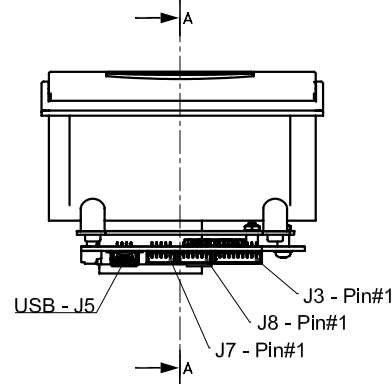
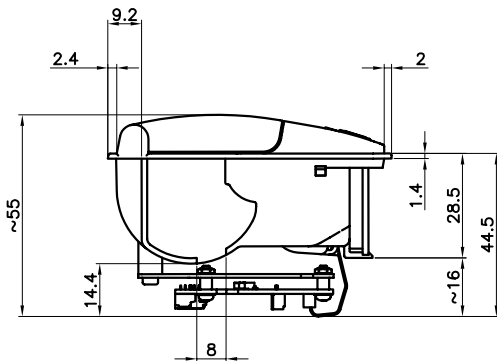
**MOUNTING PANEL**



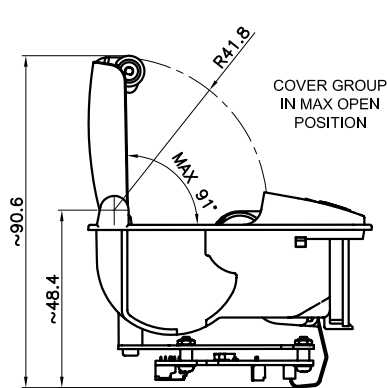
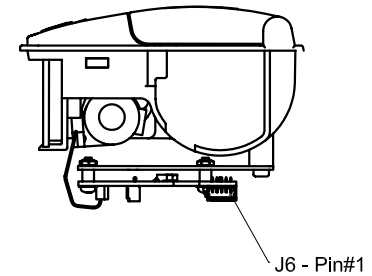
EXAMPLE OF PANEL MOUNTING



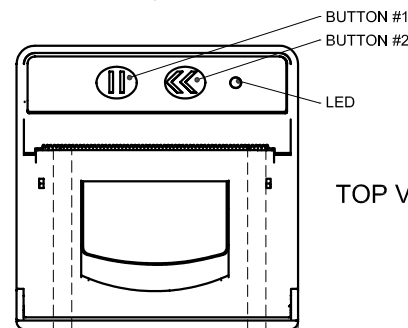
SUGGESTED PANEL DIMENSIONS



A-A (1 : 1)

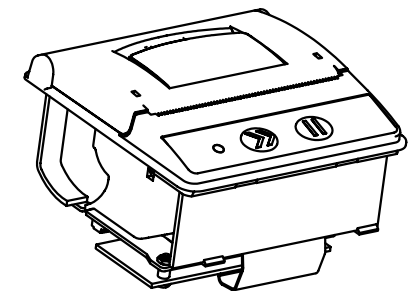


COVER GROUP  
IN MAX OPEN  
POSITION



**TOP VIEW**

CONNECTOR	FUNCTION
J3	Power Supply
J5	Standard Mini-B USB Interface
J6	Low-cost USB Interface
J7	RS232 Serial Interface
J8	RS232 TTL Serial Interface



DENOMINAZIONE Denomination	I	STAMPANTE	MASSA-Moss	UNITA'-Unit	
	GB	PRINTER	-	mm	
	F	IMPRIMANTE	DISIGNATO DA drawn by	CONTROLLATO DA Checked by	
	D	DRUCKER	F.P.	A.F.	
<b>DISEGNO D'INGOMBRO</b> <b>OVERALL DIMENSIONS DRAWING</b>			DATA-Date 09-FEB-09	SCALA-Scale 1:1	
CATALOGO/Catalog			FOGLIO-Sheet 1/1	FORMATO-Size A2	REV.
EPM203-HRS			90 770 003		A