

PPLA Programmer's Manual

Mercury Series

Revision V1.00

Date:9/9/2016

CONTENTS

A1. COORDINATES SYSTEM	6
A2. COMMAND CATEGORIES	7
A3. INTERACTION COMMANDS.....	8
<SOH>* Restarts the printer	8
<SOH>A Sends a readable status string	9
<SOH>B Toggles pause condition.....	10
<SOH>E Sends preset label to be printed	10
<SOH>F Sends one byte printer status	11
A4. SYSTEM LEVEL COMMANDS.....	12
<STX>A Sets date and time	12
<STX>a Enables label echo character	13
<STX>B Gets date and time.....	14
<STX>c Sets continuous label length	15
<STX>E Sets copy count for stored label	16
<STX>e Selects edge sensor for gap	16
<STX>F Feeds a label.....	17
<STX>f Sets stop position and automatic back-feed for the label stock.....	17
<STX>G Prints stored label format	18
<STX>I Downloads the graphics file	19
<STX>i Downloads the scalable font.....	21
<STX>KcCC Sets control code set	22
<STX>KcCF Sets horizontal shift.....	23
<STX>KcCL Sets continuous label length	23
<STX>KcSPA Sets serial port.....	24
<STX>KW Set label width.....	25
<STX>k Test RS232 port	25
<STX>L Enters label formatting state	26
<STX>M Sets maximum label length	27
<STX>MT Sets transfer type	27
<STX>m Sets measurement to metric	28
<STX>n Sets measurement to inches.....	28

<STX>O	Sets print start position	29
<STX>P	Enters data dump mode.....	30
<STX>p	Enters pause mode.....	30
<STX>Q	Clears memory	31
<STX>q	Clears memory module	31
<STX>r	Selects reflective sensor for gap	32
<STX>S	Retrieves the global register contents	33
<STX>T	Prints test pattern	33
<STX>U	Replaces the data of the specified data.....	34
<STX>V	Sets cutter and dispenser configuration	35
<STX>v	Inquires the printer version	35
<STX>W	Inquires the graphics/fonts and memory status	36
<STX>X	Sets default module	37
<STX>x	Releases file from memory	38
<STX>y	Print ASD smooth font (Symbol set)	39
<STX>Z	Print Configuration	40
A5. LABEL FORMATTING COMMANDS		41
: Sets cut by amount		42
A Sets logic image printing mode		43
C Sets left margin		44
c Sets cut by amount		45
D Sets width and height pixel size.....		46
E Ends the job and exit from label formatting mode		47
G Stores previous data to global register.....		48
H Sets heat value		49
M Toggles the mirror mode		50
m Sets measurement in metric.....		50
n Sets measurement in inch		51
P Sets print speed		51
Q Sets the quantity of labels to print		52
R Sets vertical offset.....		53
r Retrieves label data to printer buffer		53
s Stores label data to printer buffer.....		54
T Sets end-of-line code		55
X Exits from Label Formatting mode		56

z	Changes slash zero to normal 0.....	57
+	Makes auto increment for numeric.....	58
>	Makes auto increment for alphanumeric.....	58
-	Makes auto decrement for numeric.....	59
<	Makes auto decrement for alphanumeric.....	59
^	Sets count by amount.....	60
<STX>T	Prints date and time.....	61
A6.	IMAGE EDITING COMMANDS	62
Text		64
Bar Codes		67
Line		69
Box		71
Graphic		73
A7.	FONT DOWNLOADING COMMANDS.....	75
A8.	PROGRAMMING EXAMPLES FOR TEXTS	76
A9.	PROGRAMMING EXAMPLES FOR BAR CODES.....	82
Bar code A:	Code 3 of 9.....	82
Bar code B:	UPC-A	83
Bar code C:	UPC-E	84
Bar code D:	Interleaved 2 of 5 (I25)	85
Bar code E:	Code 128 including subset A, B and C	86
Bar code F:	EAN-13	87
Bar code G:	EAN-8.....	88
Bar code H:	HBIC	89
Bar code I:	Coda bar	90
Bar code J:	Interleaved 2 of 5 with a modulo 10 checksum	91
Bar code K:	Plessey.....	92
Bar code L:	Interleaved 2 of 5 with a modulo 10 checksum and shipping bearer bars	93
Bar code M:	UPC2	94
Bar code N:	UPC5	95
Bar code O:	Code 93.....	96
Bar code P:	Postnet.....	97
Bar code Q:	UCC/EAN Code 128	98

Bar code R: UCC/EAN Code 128 K-MART 99

Bar code S: UCC/EAN Code 128 Random weight..... 100

Bar code T: Telepen..... 101

Bar code V: FIM (Facing Identification Mark) 102

Bar code U: UPS MaxiCode 103

Bar code W: DataMatrix..... 104

Bar code W1d: QR Code (Auto format) 106

Bar code W1D: QR Code (Manual formatting) 108

Bar code W1f: Aztec Bar Code (Variable Length) 111

Bar code W1F: Aztec Bar Code (Specified Length/Byte count) 113

Bar code W1k: RSS (GS1 DataBar)..... 115

Bar code Z: PDF-417 120

Bar code W1z: MiocroPDF417 121

Bar code W1Z: MiocroPDF417 (with Specified Byte Count)..... 123

APPENDIX AA: HEX GRAPHIC FORMAT 124

APPENDIX AB: FONTS 125

APPENDIX AC: Bar Codes 148

APPENDIX AD: DENSITY AND SPEED..... 153

APPENDIX AE: COMMAND REFERENCE CHART 154

A1. COORDINATES SYSTEM

The PPLA coordinates system is depicted in Figure A1-1.

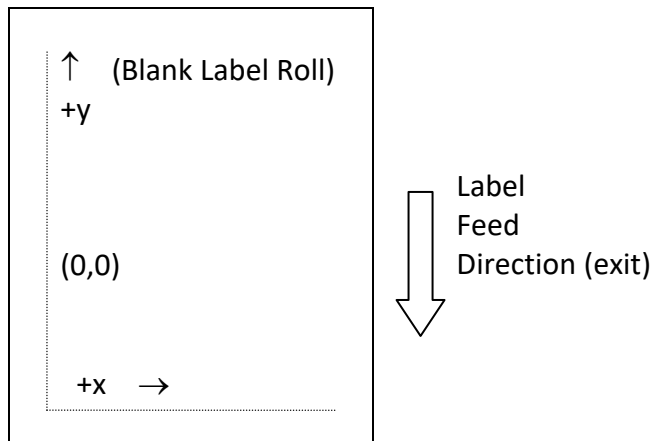


Fig. A1-1 PPLA coordinates system

The point of origin (0,0) of this coordinates system is at the left bottom corner. The origin point remains unchanged, while the texts, bar codes or other objects are being rotated. Negative coordinate value is not allowed. The ranges of X and Y coordinates are shown in the table below:

	Minimum	Maximum
X coordinate	0	It depends on printer models
Y coordinate	0	It depends on printer models

The measurements of the X- and Y-axis of the coordinates system can be in inches or in millimeters.

A2. COMMAND CATEGORIES

According to functions, the PPLA programming commands in this manual are divided into the following five categories:

- ◆ Interaction commands
- ◆ System setting commands
- ◆ System level commands
- ◆ Label formatting commands
- ◆ Font downloading commands

Only the label formatting commands must be grouped to send, other commands can be sent separately. For example:

<u>Command/data from host</u>	<u>Data from printer (RS232C)</u>
Interaction command 0	Printer status
System level command 0	
Interaction command 1	Printer status
System level command 1	
Font downloading 0	
...	
Font downloading N	
System level command 2	
<STX>L (system command)	
Label formatting command 0	
...	
Label formatting command N	
E (Label formatting command)	

A3. INTERACTION COMMANDS

The interaction commands demand the printer's immediate interaction. But the printer only communicates a detailed status to the host via the printer's serial (RS232C) port, since only serial port supports bi-direction communication.

Upon receiving this kind of command, the printer will either perform the command or send back the corresponding data to the host so that the user and programmer can determine what to do with the next step.

<SOH>* Restarts the printer

Syntax <SOH>*

Response <XOFF><XON>R

Description

- 1.This command restarts the printer. Printer has “power off” and “power on” steps.
2. “Power off” responses <XOFF>; “power on” responses <XON>R.
3. This command will clear DRAM memory.

<SOH>A Sends a readable status string

Syntax <SOH>A

Response XXXXXXXX<CR>

Description This command drives the printer to retrieve an 8-byte data string followed by a <CR> to the host. Each 'X' will be 'Y' or 'N' reporting the printer status.

Byte 1	Y - Firmware parser is busy N - Firmware parser is idling
Byte 2	Y - Paper out N - Paper installed
Byte 3	Y - Ribbon out N - Ribbon installed
Byte 4	Y - Printing batch file N - Others
Byte 5	Y – At printing state N - Not at printing state
Byte 6	Y - Printer is paused N - Printer is not paused
Byte 7	Y - Label is present N – Label is not present
Byte 8	N – Always 'N'

<SOH>B **Toggles pause condition**

Syntax <SOH>B

Response None

Description This command toggles the pause state on or off. Byte 6 generated from <SOH>A will reflect the change of the status.

<SOH>E **Sends preset label to be printed**

Syntax <SOH>E

Response XXXX<CR>

Description This command drives the printer to report the numbers of labels queued to print.

This 'XXXX' is a 4-digit decimal number, e.g.

0020<CR>

(There are still 20 labels left in printer buffer waiting to be printed.)

<SOH>F Sends one byte printer status

Syntax <SOH>F

Response X<CR>

Description This command instructs the printer to send a single byte where each bit (1 or 0) represents one of the printer's status flags, followed by a <CR>.

Bit 1	1 - firmware parser busy 0 - firmware parser idling
Bit 2	1 - paper out 0 - paper installed
Bit 3	1 - ribbon out 0 - ribbon installed
Bit 4	1 - printing batch file 0 - others
Bit 5	1 - at printing state 0 - not at printing state
Bit 6	1 - printer is paused 0 - printer is not paused
Bit 7	1 - label present 0 - label not present
Bit 8	0 - always '0'

A4. SYSTEM LEVEL COMMANDS

This group of commands is used to set the printing related parameters or environment for the current or subsequent labels. They will be reset after restarting the printer or by other related system level commands. Unlike system setting commands, these commands will not be saved into E²PROM.

<STX>A Sets date and time

Syntax <STX>AwmmddyyyyhhMMjjj

Description This command sets the date and time. It takes effect only when the RTC (real time clock) function is enable.

Location	Digit no.	Description
w	1	Day of week. 1 for Monday
mm	2	Month. 01 for January
dd	2	Day.
yyyy	4	Year.
hh	2	Hour in 24-hour format.
MM	2	Minutes.
jjj	3	Julian data.

Example <STX>A5100720001230287
This command sets Friday Oct. 7th, 2000, 12:30, the 287 of the year.

<STX>a Enables label echo character

Syntax <STX>a

Description This command enables the printer to feedback the control code of RS (1EH) and US (1FH) the host through the serial port after each label is being printed.

Example <STX>a
 1EH ; 1st label printed
 1EH ; 2nd label printed
 1EH ; 3rd label printed
 1EH ; 4th label printed
 1FH ; 5th label printed

<STX>B Gets date and time

Syntax <STX>B

Response wmmddyyyyhhMMjjj

Description This command gets the date and time from printer through serial (RS232) port. It takes effect only when the RTC (real time clock) board is installed.

Location	Digit no.	Description
w	1	Day of week. 1 for Monday
mm	2	Month. 01 for January
dd	2	Day.
yyyy	4	Year.
hh	2	Hour in 24-hour format.
MM	2	Minutes.
jjj	3	Julian data.

Example <STX>B

<STX>c **Sets continuous label length**

Syntax <STX>cxxxx

Description This command disables the edge sensor sensing the gap or mark between the labels and sets a page length for the printer to feed. (Otherwise the printer will feed the label for 12 inches long.)

This command will overwrite <STX>e and <STX>r commands, which are in conflict with it. The label length means the distance between the start printing position to the stop printing position.

Example <STX>c0200
Set two-inch continuous label.

<STX>c0000
Disable continuous label.

<STX>E **Sets copy count for stored label**

Syntax <STX>Exxxxx

Description This command should be sent in conjunction with <STX>G, as both commands are related to label storage. The stored label format is the last printed format, kept in the print buffer.

Example <STX>E00003<CR>
 <STX>G<CR>

Result Print out 3 pieces of the last printed label format.

<STX>e **Selects edge sensor for gap**

Syntax <STX>e

Description

1. This command is used for see-through media. It sets the printer to sense the gap between labels.
2. Refer to <STX>r for reflective media.
3. Once received this command, the printer will ignore the previous command for continuous label(<STX>cxxxx).

<STX>F Feeds a label

Syntax <STX>F

Description

1. This command will feed a label till where the gap is being detected, in case that the paper type is set to non-continuous.
2. The media paper will be fed for certain length, when the paper type is set to continuous.
3. Basically this function is the same as pressing the “Feed” button.

<STX>f Sets stop position and automatic back-feed for the label stock

Syntax <STX>fxxx

Description

1. This command causes the label stock to stop at a position convenient for the user to tear off. When the next label format is sent to the printer, it will automatically back-feed to the start of print position.
2. Back-feed will not be activated if xxx is less than 220. (Default is <STX>f220.)
3. Under multi-copy or continuous printing, this command is valid only for the first label and last labels.

Example <STX>f320<CR>

Result Label stock feeds backward for one inch before the next label format is printed.

<STX>G Prints stored label format

Syntax <STX>G

Description This command should be used in conjunction with <STX>E.
The stored label format is the last printed label format
kept in the buffer.

Example <STX>L<CR>
121100000200100This is a label<CR>
E<CR>
<STX>E00002<CR>

Result Print the label 3 times (1+2 copies).

<STX>I Downloads the graphics file

Syntax <STX>Imbfnn...n

Parameter *m*: Memory module.
‘A’ for RAM module, ‘B’ for flash memory module,
‘C’ for default module, ‘H’ for USB host.
b: ‘A’ - 7-bit data image file.
f: Image file format. The printer supports 4 image file
formats, PCX, BMP, IMG and HEX formats.
This parameter specifies graphic format type and
direction.

f parameter	Image File Format	Direction
‘B’	8-bit BMP file format	Flipped
‘b’	8-bit BMP file format	
‘I’	8-bit IMG file format	Flipped
‘I’	8-bit IMG file format	
‘P’	8-bit PCX file format	Flipped
‘p’	8-bit PCX file format	
‘F’	7-bit HEX file format	

nn...n: file name, maximum 16 characters.

- The file name can be accessed with label formatting commands (Y).
- The file name must be same as defined.
- Refer to the appendix AA for details of the HEX format.

Example The example of downloading and retrieving HEX graphics is as below:

Part 1

<STX>IAFhexfile<CR>

Part 2

... (HEX file)

Part 3

<STX>L<CR>

...

1Y1100001000100hexfile<CR>

1Y1100002000100hexfile<CR>

E<CR>

Result

The HEX graphics image file “hexfile” will print on the same label.

<STX>i

Downloads the scalable font

Syntax

<STX>imtnn<Name><CR>xxx...x<data>

Parameter

m: Memory module.

‘A’ for RAM module, ‘B’ for flash memory module,

‘C’ for default module, ‘H’ for USB host.

t: Download scalable font type. ‘T’ - 7-bit data image file.

nn: 2-digit font ID. ID range is 50-99, 9A-9Z and 9a-9z.

Name: Up to 15 characters for font name.

<CR>: 0x0d can end the Name.

xxx...x: 8-digit size of font data, bytes, hexadecimal; should be padded with “0.”

data: Scalable font data.

Example

```
<STX>iBT52Water<CR>00008E74
```

```
... (Binary file)
```

```
<STX>L<CR>
```

```
1911S5101500010P022P022FISH<CR>
```

```
E<CR>
```

<STX>KcCC Sets control code set

Syntax <STX>KcCCn

Description Refer to the list below for standard and alternative control codes. To exit from the alternative mode, just reset the menu on the printer panel or send the command of “!KcCC1” in alternative mode 1 or send the command of “~KcCC2” in alternative mode 2.

Parameter ‘S’: Standard control codes.
‘1’: Alternative control codes.
‘2’: Alternative-2 control codes.

Standard (default)		Alternative 1		Alternative 2	
Control codes	Hexadecimal value	Control codes	Hexadecimal value	Control codes	Hexadecimal value
SOH	0x01	^	0x5E	^	0x5E
STX	0x02	~	0x7E	~	0x7E
CR	0x0D	CR	0x0D	CR	0x7C
ESC	0x1B	ESC	0x1B	ESC	0x1B
“Count By” ^	0x5E	@	0x40	@	0x40

<STX>KcCF Sets horizontal shift

Syntax	<ESC>KcCF: <u><i>n</i></u>
Description	This command shifts the image print position in the X coordinate.
Parameter	<i>n</i> is a binary signed byte and in terms of pixels. 00H ~ 7FH are positive and 80H ~ FFH are negative.

<STX>KcCL Sets continuous label length

Syntax	<STX>KcCL <u>xxxx</u>
Description	This command disables the edge sensor sensing the gap or mark between the labels and sets a page length for the printer to feed. (Otherwise the printer will feed the label for 12 inches long.) This command will overwrite <STX>e and <STX>r commands, which are in conflict with it. The label length means the distance between the start printing position to the stop printing position.
Example	<STX>KcCL0200 Set two-inch continuous label. <STX>KcCL0000 Disable continuous label.

<STX>KcSPA Sets serial port

Syntax KcSPAhpdtb

Parameter *h*: Hardware.
 'PH' - hardware, 'PS' – software, p: Parity,
 'pE' - even parity, 'pN' - none parity',
 'pO' - odd parity
d: Data length.
 'D7' - 7- bit data, 'D8' - 8-bit data
t: Stop bit.
 'S1' - 1 stop bit, 'S2' - 2 stop bits
b: Baud rate.
 'B24' - 2400, 'B48' – 4800, 'B96' - 9600, 'B19' - 19200
 'B28' - 28800, 'B38' – 38400, 'B57' - 57600,
 'B15' – 115200

Example <STX>KcSPAPH; SPApN; SPAD8; SPAS1; SPAB96<CR>

The above example will set baud rate to 9600, bit data to 8, parity to odd, stop bit to 1 and hardware.

<STX>KW **Set label width**

Syntax <STX>KWxxxx

Description This command sets the label width. The range is from 0000 to TPH width.

Example <STX>KW0300<CR>
Set label width to 3 inches.

<STX>k **Test RS232 port**

Syntax <STX>k

Description This command drives the printer to send the character Y via its RS232 serial port. Failure to receive the character Y at the host may indicate an interfacing problem.

Example <STX>k

Response Y

<STX>L **Enters label formatting state**

Syntax <STX>L

Description The above command switches the printer into the label-formatting mode. The printer will process the label formatting commands until it receives the command to exit from this mode.

In the process of the label formatting, the system level commands will be ignored.

Example <STX>L<CR>
121100001000050THIS LABEL IS MADE BY
JIMMY<CR>
E<CR>

Output THIS LABEL IS MADE BY JIMMY

Fig. A4-1

<STX>M Sets maximum label length

Syntax <STX>Mxxxx

Description Above command sets the maximum label length and the printer will search for gap or mark within the specified length accordingly. The default length is 12 inches.

Parameter A 4-digit decimal. (1 to 12 inches)

Example <STX>M0300<CR>

Result Set maximum label length to 3 inches.

<STX>MT Sets transfer type

Syntax <STX>MTn

Description This command can set transfer type.

Parameter ‘D’ for direct thermal type
 ‘T’ for thermal transfer type

<STX>m **Sets measurement to metric**

Syntax <STX>m

Description There are two measurements for the printer, in metric and in inch.

Example <STX>m
 <STX>M0600

Result 60 mm for maximum label length

<STX>n **Sets measurement to inches**

Syntax <STX>n

Description There are two measurements for the printer, in metric and in inch. Default unit is inch.

Example <STX>n
 <STX>M0600

Result 6 inches for maximum label length.

<STX>O Sets print start position

Syntax <STX>Oxxxx

Description This command sets the offset value for start print position. The default 0220 sets the start print position exactly below the TPH (print head) line. You may change it to meet the specific label format requirements.

This parameter will be ignored if continuous label command is sent. (<STX>cxxxx).

<STX>P **Enters data dump mode**

Syntax <STX>P

Description This command drives the printer to dump the HEX value of the data that is transmitted to the printer afterwards. The printer will not return to normal function, unless restarted.

Example <STX>P

<STX>p **Enters pause mode**

Syntax <STX>p

Description This command drives the printer to enter pause mode, often applied between batches of paper.

Example <STX>p

<STX>Q Clears memory

Syntax <STX>Q

Description This command instructs the printer to clear all memory modules. Normally, this command is sent at the end of each job to avoid that the graphics and fonts become accumulated up and overflow the memory.

In case of the memory full, the printer will erase the first-in graphics or fonts. To avoid this situation and to save the data re-processing time, you are advised to send this command at the end of a job.

Example <STX>IAFhexfile<CR>
... (HEX file)
<STX>L<CR>

1Y1100001000100hexfile<CR>
E<CR>
<STX>Q<CR>

<STX>q Clears memory module

Syntax <STX>qn

Description This command clears the selected memory module.

Parameter 'A' for RAM module,
 'B' for flash memory module,
 'C' for default module, 'H' for USB host.

<STX>r **Selects reflective sensor for gap**

Syntax <STX>r

Description This command selects the reflective sensor for label detection. It is used for “Black stripe” media sensing. If the label stock is non-continuous type, refer to <STX>e for see-through media. Once this command is received, the previous continuous paper command (<STX>cxxxx) will be ignored.

<STX>S Retrieves the global register contents

Syntax <STX>Sn

Description Refer to “G” command.

Parameters *n* is name of the register ranges from “A” TO “Z”.

<STX>T Prints test pattern

Syntax <STX>T

Description This command is used for testing the printout quality or checking the print head for debugging or maintenance purpose. Normally users do not use this command.

Example <STX>T

Output

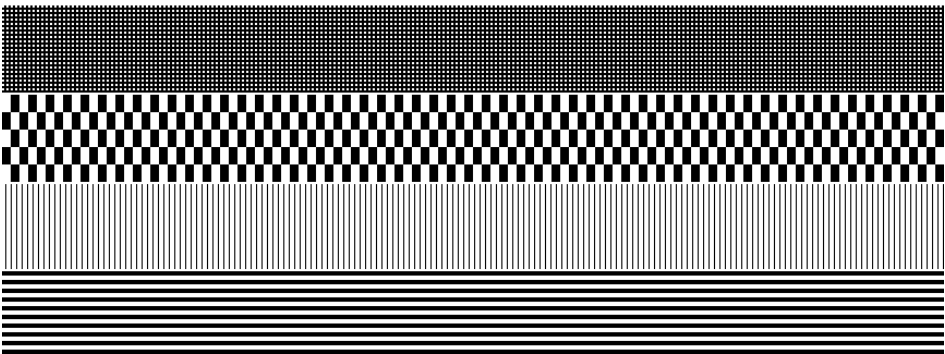


Fig. A4-2

<STX>U **Replaces the data of the specified data**

Syntax <STX>Unncccc

Description This command is used to replace the field data in a form.
Other data are kept the same as the previous ones.

Parameter *nn* field number
 cccc data to be replaced

Example 1 <STX>L
 D11
 121100000100020filed 1 data
 121100000400020filed 2 data
 Q0001
 E

Output 1

filed 2 data

filed 1 data

Fig. A4-3

Example 2 <STX>U01NEW DATA 1
 <STX>E00001
 <stx>G

Output 2

filed 2 data

NEW DATA 1

Fig. A4-4

<STX>V **Sets cutter and dispenser configuration**

Syntax <STX>Vn

Parameter '0': no cutter and peeler function
 '1': Enables cutter function

Example <STX>V0 ; no cutter and peeler function
 <STX>V1 ; Enables cut operation

<STX>v **Inquires the printer version**

Syntax <STX>v

Description This command is used for maintenance purpose. It inquires the printer of the firmware version. The printer responses with version and date code through RS232C.

Example <STX>v

<STX>W **Inquires the graphics/fonts and memory status**

Syntax <STX>Wn

Description The command inquires the printer of the contents of graphics/fonts/labels as well as the available RAM size. The printer will check its memory and response through RS232C.

Parameter F: Show fonts and memory status
G: Show graphics and memory status
L: Show stored labels and memory status

Example <STX>WF

Response MODULE: A<CR>
A100<CR>
A101<CR>
AVAILABLE BYTES: 3743744<CR>
MODULE: B<CR>
AVAILABLE BYTES: 2574336<CR>
MODULE: H<CR>
AVAILABLE BYTES: 3107545<CR>

The “A100” and “A101” are fonts that were downloaded before.

<STX>X Sets default module

Syntax <STX>Xn

Description This command sets the default memory module when loading PCL bitmap fonts (Refer to the section A8 Font Downloading Commands.).

Parameter ‘A’ for RAM module
 ‘B’ for flash memory module
 ‘C’ for default module, ‘H’ for USB host

Note: If font is defined to download to ‘USB disk’ the USB disk is not installed, the printer will automatically download the soft font into RAM.

<STX>x Releases file from memory

Syntax <STX>xmtn...n

Description This command deletes the specific file from the specified memory module, evacuating memory space to load other data, further to avoid memory overflow.

Parameter *m*: The memory module identification character.
Please refer to <STX>I command.
t: The file type identification code.
‘G’ for graphics.
‘F’ for fonts.
n...n: file name (maximum 16 characters).

Example <STX>IAFstar<CR>
... (HEX file)
<STX>L<CR>
...
1Y1100001000100star<CR>
E<CR>
<STX>xAGstar<CR>

Results Download the graphics with file name “star”.
Print the graphic image.
Delete the “star” image file.

<STX>y Print ASD smooth font (Symbol set)

Syntax <STX> ytxx

Parameter t: Set Byte-size symbol sets.
 'S' is for Single byte symbol sets.
 xx: Select symbol sets.
 'PM'- PC850
 'W1'- Windows 3.1 Latin 1
 'WE'- Windows 3.1 Latin 2
 'WG'- Windows Latin, Greek
 'WR'- Windows Latin, Cyrillic
 'WT'- Windows 3.1 Latin 5

Example <STX>ySW1 ; Select 'Windows 3.1 Latin' 1 set.

<STX>Z **Print Configuration**

Syntax <STX>Z

Description This command drives the printer to print the printer configuration including settings, firmware setting, firmware version, etc...

Example <STX>Z

A5. LABEL FORMATTING COMMANDS

The label formatting commands will:

- Set the print environment, margins, print modes, multi-copies, etc.
-
- Set cursor position and print graphics, texts, bar codes, lines and boxes.
-
- Control the heat of the printing, label print position and user interface.
-
- All commands after < STX>L are interpreted as Label Formatting Commands.

Different from other group commands, the label formatting commands have no leading control code, e.g. SOH, STX or ESC.

: **Sets cut by amount**

Syntax **:xxxx**

Description This command is valid only when the cutter is installed. It allows a predetermined number of labels to be printed before a cut is made. Please refer to [cxx].

Example <STX>V1<CR>
 <STX>L<CR>
 131100002000050CHCK THE CUT FUNCTION<CR>
 Q0010<CR>
 :0003<CR>
 E<CR>

Result Enable the cutter to cut after 3 labels have been printed.

A Sets logic image printing mode

Syntax	$A\underline{n}$
Description	This command puts the printer on logical OR operation or XOR operation, which makes the printout more attractive.
Parameter	‘1’ for logical XOR (exclusive OR); the default mode is ‘1’. ‘2’ for logical OR.

Example 1 <STX>L<CR>
 A1<CR>
 151100002000050ABC<CR>
 151100002000050---<CR>
 E<CR>

Output 1



Fig. A5-1

Example 2 <STX>L<CR>
 A2<CR>
 151100002000050ABC<CR>
 151100002000050---<CR>
 E<CR>

Output 2



Fig. A5-2

C Sets left margin

Syntax Cxxxx

Description This command allows horizontal adjustment of the point where printing begins. Different margin value makes image shift to the left or right. Default value is C0000.

Example C0100

Result Set left margin to one inch

c **Sets cut by amount**

Syntax **cxx**

Description This command is valid only when the cutter is installed. It instructs the printer to cut the label media after the specified numbers of labels have been printed.

Its function is same as the command “:xxx”, except only a 2 digit value can be entered. The default value is c01.

Example <STX>V1<CR>
 <STX>L<CR>
 131100002000050CHCK THE CUT FUNCTION<CR>
 Q0010<CR>
 c03<CR>
 E<CR>

Result Enable the cutter to cut the label media after 3 labels have been printed.

D Sets width and height pixel size

Syntax Dwh

Description Default value of 203 dpi is D22; Default value of 303 dpi is D11. Though the maximum resolution is up to the printer model, besides the smallest one, the other pixel sizes can be set by this command. However, reducing the resolution may cause the image pixel to be amplified and the printout get zigzagged. The minimum pixel size set by D11 is

203dpi Printer:	0.0049 inch (0.125 mm)
300dpi Printer:	0.0033 inch (0.084 mm)

Parameter *w* – is pixel width.
 h – is pixel height.

Example <STX>L<CR>
 D23<CR>
 120000002000050PIXEL SIZE FOR D23<CR>
 E<CR>

Output

PIXEL SIZE FOR D23

Fig. A5-3

E **Ends the job and exit from label formatting mode**

Syntax E

Description When the printer is in label formatting mode and receives an “E” command, it will immediately exit from the mode and will print a label based on the data that has already been received. Even if no printable data has been received, the printer will generate and feed a label.

G Stores previous data to global register

Syntax G

Description The command saves the previous data to global register and retrieves it to print only when the restore command <STX>Sn is sent. This command may be used more than one time and the global registers are named in the order created, beginning with register 'A' and ending at register 'Z'.

Example

```
<STX>L<CR>
D11<CR>
140000000700150DATA A<CR>
G<CR>
140000000700050DATA B<CR>
G<CR>
140000001000050<STX>SA<CR>
140000001300050<STX>SB<CR>
140000001600050<STX>SA<CR>
E<CR>
```

Output

```
DATA A
DATA B
DATA A
DATA B    DATA A
```

Fig. A5-4

H

Sets heat value

Syntax

Hxx

Description

The heat value affects the darkness of the image. To get a better quality printout, some of the factors like paper media, ribbon types (wax, semi-resin and resin) and image pattern itself etc. should also be taken into consideration. The default value is defined by model.

Parameter

xx – is 00~20.

Note: Refer to [APPENCIX AD: DENSITY AND SPEED](#).

M **Toggles the mirror mode**

Syntax M

Description This command toggles the mirror mode. At mirror state the printer mirrors the following field data.

Example <STX>L<CR>
M<CR>
150000001800000MIRROR SAMPLE<CR>
E<CR>

Output



Fig. A5-5

m **Sets measurement in metric**

Syntax m

Description There are two measurements in the printer - metric and inch.

n Sets measurement in inch

Syntax n

Description Default measurement is inch.

P Sets print speed

Syntax Pn

Description This command controls the print speed. Default speed is defined by printer. Unit is ips (Inches per second).

Parameter

<u>n</u> value	Speed	<u>n</u> value	Speed	<u>n</u> value	Speed
A	1.0 ips	H	4.5 ips	O	8.0 ips
B	1.5 ips	I	5.0 ips	P	8.5 ips
C	2.0 ips	J	5.5 ips	Q	9.0 ips
D	2.5 ips	K	6.0 ips	R	9.5 ips
E	3.0 ips	L	6.5 ips	S	10.0 ips
F	3.5 ips	M	7.0 ips		
G	4.0 ips	N	7.5 ips		

Note: Refer to [APPENCIX AD: DENSITY AND SPEED](#).

Q Sets the quantity of labels to print

Syntax Qxxxx

Description This command is used to set the number of the labels to be printed. If the printout contents are same or just different in certain auto increment/decrement fields, sending this command can save the communication and processing time.

Parameter xxxx is a 4-digit decimal. The default value is 0001.

Example <STX>c0060<CR>
 <STX>L<CR>
 D11<CR>
 1300000002000002 COPIES<CR>
 Q0002<CR>
 E<CR>

Output 2 COPIES

 2 COPIES
 Fig. A5-6

R **Sets vertical offset**

Syntax	<u>Rxxxx</u>
Description	The command sets the vertical start point to be printed. By this command the print image can be shifted vertically.
Parameter	xxxx is a 4-digit decimal. The default value is 0000.
Example	R0100
Result	Set 1 inch vertical offset.

r **Retrieves label data to printer buffer**

Syntax	<u>r</u><i><u>n...n</u></i>
Description	This command can recall the data of the label format sorted in the printer.
Parameter	<i>n ... n</i> : File name with maximum 16 characters.

s **Stores label data to printer buffer**

Syntax *smn...n*

Description The data of the label format can be stored in the printer memory. With this command, the printer will exit from label formatting mode.

Parameter *m*: Name of the memory module.
 ‘A’ for RAM module
 ‘B’ for flash memory module
 ‘C’ for default module
 ‘H’ for USB host.
 (Normally, the default module is RAM module.)
n ... n: File name with maximum 16 characters.

Example <STX>L<CR>
 D11<CR>
 130000000200100STORED LABEL<CR>
 sASLAB<CR>
 <STX>L<CR>
 rSLAB<CR>
 130000000500100TEXT 1<CR>
 E<CR>

Output TEXT 1
 STORED LABEL

Fig. A5-7

T

Sets end-of-line code

Syntax

$T\underline{nn}$

Description

This command is used to set end-of-line code.
The default value is $T<0x0D>$.

Parameter

nn is represented by HEX value.

Example

```
<STX>L<CR>
D11<CR>
T40<CR>
130000000200100ABC@
E<CR>
```

Output

ABC

Fig. A5-8

The above example changes the end-of-line code from
<CR> to <0x40>(ASCII character: '@').

X Exits from Label Formatting mode

Syntax X

Description The command will exit from Label Formatting mode and switch to the System Level command mode. If you have data in Label Formatting mode, it will not be printed out. Unless sending <STX>G command.

Example <STX>L<CR>
130000000200100TEST<CR>
X<CR>

z **Changes slash zero to normal 0**

Syntax z

Description The alphanumeric fonts support both normal and slash zeros (ø & 0).
If this kind of fonts has been selected, the default slash zero (ø) is being used. This command puts the normal zero 0 in use.

Example <STX>L<CR>
D11<CR>
130000000200100NO. 0228<CR>
z<CR>
130000000500100NO. 0228<CR>
E<CR>

Output NO. 0228

NO. 0228

Fig. A5-9

+ **Makes auto increment for numeric**

> **Makes auto increment for alphanumeric**

Syntax ***+mnn; >mnn***

Description This command can increment field on each label printed to save the time used in communication and data processing between the host and the printer.

Parameter *m*: A character is set in the left-hand of data string.
 nn: The amount which is increment the data string.

Example <STX>c0050<CR>
 <STX>L<CR>
 D11<CR>
 130000000200100100<CR>
 +010<CR>
 Q0003<CR>
 E<CR>

Output

120

110

100

Fig. A5-10

- **Makes auto decrement for numeric**
- < **Makes auto decrement for alphanumeric**

Syntax *-mnn; <mnn*

Description This command can decrement the field on each label printed to save the time use in communication and data processing between the host and the printer.

Parameters *m*: A character is set in the left-hand of data string.
nn: The amount which is decrement the data string.

Example <STX>c0050<CR>
 <STX>L<CR>
 D11<CR>
 130000000200100111<CR>
 - 15<CR>
 Q0003<CR>
 E<CR>

Output

81

96

111

Fig. A5-11

^ **Sets count by amount**

Syntax ^xx

Description An application using incrementing or decrementing fields will occasionally require that more than one label be printed with the same values before the field data is updated. This command can be applied in this situation, but it can only be sent once per label format.

Parameter xx is a 2-digit value used to specify the number of labels to be generated before incrementing or decrementing fields on the label.

Example <STX>c0050<CR>
 <STX>L<CR>
 D11<CR>
 120000000200020COUNT:<CR>
 120000000200100 123<CR>
 -01<CR>
 ^02<CR>
 Q0003<CR>
 E<CR>

Output

COUNT : 122

COUNT : 123

COUNT : 123

Fig. A5-12

<STX>T Prints date and time

Syntax <STX>T<string>

Description This command takes effect only when the RTC battery is installed. It prints current date and time. The <string> is any set of characters A ~ Z or a~f.

Parameter

Characters	Description	Characters	Description
A	Day of week	VW	Hour, 24 format.
BCD	Day of week name	XY	Hour, 12 format.
EF	Month number	Za	Minutes.
GHI	Month name	gh	Seconds
PQ	Day	bc	AM or PM
RSTU	Year	def	Julian data

Example <STX>L<CR>
121100000800010<STX>TBCD GHI PQ, TU<CR>
121100000100010<STX>TA EF PQ, RSTU
VW:Za:gh<CR>
E<CR>

Output

TUE DEC 02, 14

2 12 02, 2014 22:24:58

Fig. A5-13

A6. IMAGE EDITING COMMANDS

The following group of commands is the subset of label formatting commands. They control the position and the scale of the image and put the image directly into the frame buffer of the printer memory. All of them are led by '1', '2', '3' '4' respectively. These numbers represent the orientation or rotation direction. The image types include:

- ◆ Texts - internal hard fonts and downloadable soft fonts.
- ◆ Bar Codes - both one and two dimension (2D) bar codes.
- ◆ Graphics - PCX, BMP, IMG and HEX format files.
- ◆ Lines - solid lines.
- ◆ Boxes - variable sizes, length and thickness.

Rotation

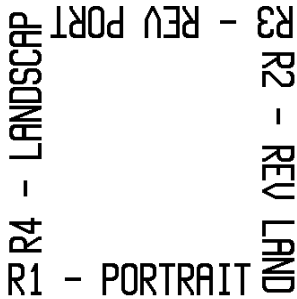


Fig. A6-1

There are 4 print directions shown as figure A6-1. The leading character controls the direction or rotation.

1. Portrait.
2. Reverse landscape.
3. Reverse portrait.
4. Landscape.

Example

```
<STX>L<CR>  
D11<CR>  
141100000800060R1 - PORTRAIT<CR>  
241100002150200R2 - REV LAND<CR>  
341100002400217R3 - REV PORT<CR>  
441100001030079R4 - LANDSCAP<CR>  
E<CR>
```

Output

Please see Fig. A6-1.

Text

Syntax Rthvoooyyyxxxx[data string]

Parameter

R: Print direction. '1', '2', '3' or '4'.

t: Font type. Please refer to the font tables in User's Manual.

t_character	ooo sub font type	Font type
'0', '1', '2', '3', '4', '5', '6', '7', '8'	'000'	font 0 ~ font 8 respectively.
'9'	'A04' '000' ~ '010' 'A72'	ASD smooth fonts. 'A04': 4 points, '000': 5 points, '001': 6 points, '002': 8 points, '003': 10 points, '004': 12 points, '005': 14 points, '006': 18 points, '007': 24 points, '008': 30 points, '009': 36 points, '010': 48 points, 'A72': 72 points. <i>Note: 'A04' and 'A72' are only supported on 300dpi printer.</i> <i>(Refer to <STX>yS command)</i>
'9'	' <u>xxx</u> '	For PCL soft font selection. <u>xxx</u> : A 3-digit decimal represents the soft font ID. Refer to section A8.
'.'	'000' ~ '007'	Courier fonts, (ooo represents symbol set) 000 - Roman-8, 001 - ECMA-94,

		002 - PC set, 003 - PC set A, 004 - PC set B, 005 - Legal, 006 – Greek and 007 - Russian.
--	--	---

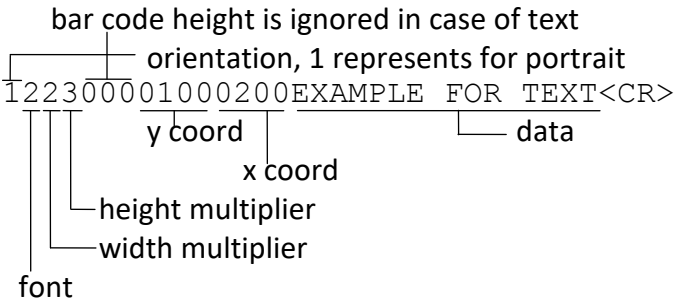
h: Horizontal scale. ‘0’ through ‘9’ and ‘A’ through ‘O’ represent scale factors. (‘A’=10, ‘B’=11, .. and ‘O’=24).

v: Vertical scale. ‘0’ through ‘9’ and ‘A’ through ‘O’ represent scale factors. (‘A’=10, ‘B’=11, .. and ‘O’=24).

yyyy: A 4-digit value for Y coordinate. The lower left corner is the origin point of the XY coordinate system and the Y value is the vertical offset from the origin point.

xxxx: A 4-digital value for X coordinate. The lower left corner is the origin point of the XY coordinate system. The X value is the horizontal offset from the origin point.

data string: A string of printable data with maximum 255 characters in length. The data string ends with a <CR> control code or pre-defined code by Txx command.



Example <STX>L<CR>

```
D11<CR>
1211000010000000FONT2, H=1, V=1<CR>
1221000012000000FONT2, H=2, V=1<CR>
1212000014000000FONT2, H=1, V=2<CR>
1911002017000000SMOOTH, 8 POINTS<CR>
1911003020000000SMOOTH, 10 POINTS<CR>
E<CR>
```

Output

SMOOTH, 10 POINTS

SMOOTH, 8 POINTS

FONT2, H=1, V=2

FONT2, H=2, V=1

FONT2, H=1, V=1

Fig. A6-2

Bar Codes

Syntax RthvoooYYYYxxx[data string]

Parameter

R: Print direction. '1', '2', '3' or '4'.

t: Bar code type. The range can be 'A' through 'T' and 'a' through 'z', each character represents a bar code type and rule. Refer to section A10 for more details on bar codes.

h: '0' through '9' and 'A' through 'O' represent the width of wide bar. ('A'=10, 'B'=11, .. and 'O'=24).

v: '0' through '9' and 'A' through 'O' represent the width of narrow bar. ('A'=10, 'B'=11, .. and 'O'=24).

ooo: A 3-digit value that represents the bar code height.

yyyy: A 4 digit value for Y coordinate. The lower left corner is the origin of the XY coordinate system. The Y value is the vertical offset from origin point.

xxxx: A 4-digit value for X coordinate. The lower left corner is the origin point of the XY coordinate system. The X value is the horizontal offset from origin point.

Data string: A string of data with maximum 255 characters in length, ended by <CR> or pre-defined EOL (end of line) code. The length of the string may be varied from the type of the bar code.

bar code height, 000 stands for default height
 orientation, 1 represents for portrait
 1A5200001000200CODE 3 OF 9<CR>
 y coord x coord data
 narrow bar width
 wide bar width
 bar code type

Example

```

<STX>L<CR>
D11<CR>
1A0000000200000BC 1<CR>
1A0005000200120BC 2<CR>
1A6300000200240BC 3<CR>
E<CR>
  
```

Output



Fig. A6-3

Line

Syntax *RX11000yyyyxxxLaaabbb*
 or
 RX11000yyyyxxxlaaaabbbb

Parameter

R: Print direction. '1', '2', '3' or '4'.

yyyy: A 4-digit value for Y coordinate. The lower left corner is the origin point of the XY coordinate system. The Y value is the vertical offset from origin point.

xxxx: A 4-digit value for X coordinate. The lower left corner is the origin of the XY coordinate system. The X value is the horizontal offset from origin point.

aaa or aaaa: A 3 or 4-digit value that specifies the width of line.

bbb or bbbb: A 3 or 4-digit value that specifies the height of line.

1X1100001000200L200030<CR>
└── ignored ─┘ └── line with 3-digit width and height
└── ignored ─┘ └── line height
└── ignored ─┘ └── line width
└── line or box

Example

<STX>L<CR>
D11<CR>
1X1100000200000L100020<CR>
1X1100000800000100100100<CR>
E<CR>

Output



Fig. A6-4

Box

Syntax *RX11000yyyyxxxBaaabbbtttsss*
 or
 RX11000yyyyxxxbaaaabbbtttssss

Parameter

R: Print direction. '1', '2', '3' or '4'.

yyyy: Y coordinate. A 4-digital decimal. The lower left corner is the origin of the XY coordinate system. The Y value is the vertical offset from origin.

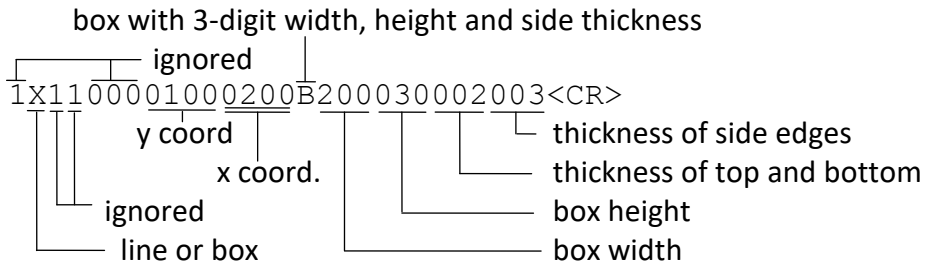
xxxx: X coordinate. A 4-digital decimal. The lower left corner is the origin of the XY coordinate system. The X value is the horizontal offset from origin.

aaa or *aaaa*: A 3 or 4-digit value that specifies the width of box.

bbb or *bbbb*: A 3 or 4-digit value that specifies the height of box.

ttt or *tttt*: A 3 or 4-digit value that specifies the thickness of top and bottom box edges.

sss or *ssss*: A 3 or 4-digit value that specifies the thickness of side edges.



Example

```

<STX>L<CR>
D11<CR>
1X1100000200100B100040002005<CR>
1X1100000800100b0250010000030004<CR>
E<CR>

```

Output



Fig. A6-5

Graphic

Before sending the graphic selection command, make sure that the graphic was already loaded into the flash memory; otherwise send the command led by <STX>I to download the graphic first.

Syntax 1Y11000yyyyxxxxn..n

Parameter

yyyy: A 4-digit value for Y coordinate. The lower left corner is the origin point of the XY coordinate system. The Y value is the vertical offset from origin point.

xxxx: A 4-digit value for X coordinate. The lower left corner is the origin point of the XY coordinate system. The X value is the horizontal offset from origin point.

n..n: Graphic file name that was downloaded (maximum 16 characters).

Example

[illegible]

8020FF00FF00FF00FF00FF00FF00FF00FF00FF00FF00FF00FF00F
F00FF00FF00FF00FF00<CR>
8020FF00FF00FF00FF00FF00FF00FF00FF00FF00FF00FF00FF00F
F00FF00FF00FF00FF00<CR>
0000FF10<CR>
8020FFFFFF000
0000000000000000FFFFFF<CR>
FFFF<CR>

<STX>L<CR>
D11<CR>
1Y1100001000050HEXFILE<CR>
E<CR>

Output



Fig. A6-6

A7. FONT DOWNLOADING COMMANDS

The following commands are used for downloading the soft fonts with the PCL bitmap format. Refer to the PCL technical manual for their descriptions. (PCL4 or PCL5).

Command	Description
<ESC>*c###D	Assigns the soft fonts ID number (### : 0 ~ 999)
<ESC>)s###W	Downloads font descriptor (### : length of font descriptor)
<ESC>*c###E	Sets character code (### : 1 ~ 255)
<ESC>(s###W	Downloads character descriptor and image (### : length of character descriptor and image)

For more information on the soft font format, please refer to the related PCL technical manual.

Example

```
<ESC>*c100D
<ESC>) s26W ...
<ESC>*c33E
<ESC>(s32W ...
. . .
. . .
<STX>XB

<STX>L
190010002000200THIS IS A TEST FOR PCL SOFT FONT.
E
```

Result This example downloads a PCL soft font with ID 100 into flash memory, and then selects and prints it.

A8. PROGRAMMING EXAMPLES FOR TEXTS

This section explains how to select the internal fonts to format the desired printout and generate the font styles as well as font characteristics. Please refer to the User’s Manual for the symbol table of each font.

Font 0 This is a USASCII set from code <0x21> to <0x7F>.
Example

```
1000000000200000Font 0 : ASCII Character Set<CR>
Font 0 : ASCII Character Set
```

Fig. A8-1

Font 1 This is a USASCII and extension set.
Example

```
1100000000400000Font 1 : ASCII+Extension Character Set<CR>
Font 1 : ASCII+Extension Character Set
```

Fig. A8-2

Font 2 This is a USASCII and extension set.
Example

```
1200000000600000Font 2 : ASCII+Extension Character Set<CR>
Font 2 : ASCII+Extension Character Set
```

Fig. A8-3

Font 3 This font includes numeric and uppercase letters.
Example

1300000000900000Font 3 : Alphanumeric Uppercase font<CR>

FONT 3 : ALPHANUMERIC UPPERCASE FONT

Fig. A8-4

Font 4 This font includes numeric and uppercase letters.
Example

1400000001200000Font 4 : Alphanumeric Uppercase font<CR>

FONT 4 : ALPHANUMERIC UPPERCASE FONT

Fig. A8-5

Font 5 This font includes numeric and uppercase letters.
Example

1500000001600000Font 5 : Alphanumeric Uppercase font<CR>

FONT 5 : ALPHANUMERIC UPPERCASE FONT

Fig. A8-6

Font 6 This font includes numeric and uppercase letters.
Example

1600000001000000Font 6 : Alphanumeric<CR>
1600000000500000 Uppercase font<CR>

FONT 6 : ALPHANUMERIC UPPERCASE FONT

Fig. A8-7

Font 7 This font includes OCR-A ASCII characters.

Example

1700000001500000Font 7 : OCR-A font<CR>

Font 7 : OCR-A font

Fig. A8-8

Font 8 This font includes numeric and some special characters only. It is an OCR-B set.

Example

1700000002000000Font 8 : OCR-B font<CR>
18000000020002000123456789<><CR>

Font 8 : OCR-B font 0123456789<>

Fig. A8-9

Font 9 Font 9 (ASD smooth font sets) include 6 symbol sets. There are PC850/ Windows 3.1 Latin 1/ Windows 3.1 Latin 2/Windows Latin, Greek/ Windows Latin, Cyrillic/ Windows 3.1 Latin 5. The sizes are 4, 5, 6, 8, 10, 12, 14, 18, 24, 30, 36, 48 and 72 points. (4, 5 and 72 points are for 300dpi printer only.)

Symbol sets	Point Size	203DPI	300DPI
PC-850 (Default symbol set) Windows 3.1 Latin 1 Windows 3.1 Latin 2 Windows 3.1 Latin 5. Windows Latin/Cyrillic Windows Latin/Greek (Refer to <STX>yS command)	4		○
	5		○
	6	○	○
	8	○	○
	10	○	○
	12	○	○
	14	○	○
	18	○	○
	24	○	○
	30	○	○
	36	○	○
	48	○	○
	72		○

Example

```

1900001004500000ASD : 6 Points. ABCabc<CR>
1900002006000000ASD : 8 Points. ABCabc<CR>
1900003008000000ASD : 10 Points. ABCabc<CR>
1900004010000000ASD : 12 Points. ABCabc<CR>
1900005012000000ASD : 14 Points. ABCabc<CR>
1900006014000000ASD : 18 Points. ABCabc<CR>
1900007016000000ASD : 24 Points. ABCabc<CR>

```

ASD : 24 Points. ABCabc
 ASD : 18 Points. ABCabc
 ASD : 14 Points. ABCabc
 ASD : 12 Points. ABCabc
 ASD : 10 Points. ABCabc
 ASD : 8 Points. ABCabc
 ASD : 6 Points. ABCabc

Fig. A8-10

Courier Fonts The Courier font includes 8 symbol sets. The font type is ‘:’ and the height field represents the symbol set.

Height	Symbol set
000	Roman 8
001	ECMA 94
002	PC
003	PC-A
004	PC-B
005	Legal
006	PC437 (Greek)
007	Russian

Example

```
1:00000000200000Courier : Roman 8 : [\]^{|}~<CR>
1:00001005000000Courier : ECMA-94 : [\]^{|}~<CR>
1:00002008000000Courier : PC : <03H><04H><05H>[\]^<CR>
1:00005011000000Courier : Legal : [\]^{|}~<CR>
1:00006014000000Courier : PC 437 (GREEK)<CR>
```

Courier : PC 437 (GREEK)
 Courier : Legal : [®]©\$¶†™
 Courier : PC : ♥♦♣[\]^
 Courier : ECMA-94 : [\]^{|}~
 Courier : Roman 8 : [\]^{|}~

Fig. A8-11

A9. PROGRAMMING EXAMPLES FOR BAR CODES

The functions of parameters are varied from the specific bar codes.
The bar code cannot be printed out, if the input code is invalid or its length is not up to the specification.

Bar code A: Code 3 of 9

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
Variable	No	'A'	'a'	0 ~ 9, A ~ Z, \$%*+-. / and space	2 : 1 ~ 3 : 1

Example 130000001320010BAR CODE A : 3 OF 9<CR>
1A300000080011019450228<CR>

BAR CODE A : 3 OF 9



Fig. A9-1

Bar code B: UPC-A

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
12 digits (11+1)	Yes	'B'	'b'	0 ~ 9	2 : 3 : 4

Example 1300000020000000BAR CODE B : UPC-A<CR>
1B000000180015502281234567<CR>

BAR CODE B : UPC-A



Fig. A9-2

Bar code C: UPC-E

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
7 digits (6+1)	Yes	'C'	'c'	0 ~ 9	2 : 3 : 4

Example 1300000010000000BAR CODE C : UPC-E<CR>
1C0005000800160654321<CR>

BAR CODE C : UPC-E



Fig. A9-3

Bar code D: Interleaved 2 of 5 (I25)

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
Variable	No	'D'	'd'	0 ~ 9	2 : 1 ~ 3 : 1

The digit count should be an even number otherwise a '0' will be automatically entered at the first position.

Example 1300000022000000BAR CODE D :<CR>
 1300000020000000INTERLEAVED 2 of 5<CR>
 1D5308001800170135792468<CR>

BAR CODE D :
INTERLEAVED 2 OF 5

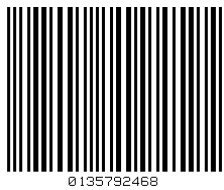


Fig. A9-4

Bar code E: Code 128 including subset A, B and C

The default code subset is B.
Subset A: Place an ASCII 'A' (DEC 65 or HEX 41) before the data to be encoded.
Subset B: place an ASCII 'B' (DEC 66 or HEX 42) before the data to be encoded.
Subset C: place an ASCII 'C' (DEC67, HEX43) before the data to be encoded.
Subset C can only encode numeric data with even byte count (DEC 30~DEC 39).
If user not place an ASCII 'A',' B' or 'C' before the data to be encoded, the barcode will be judged into Subset B.

Example 1: Encoded data: "BBar code", Output data: "Bar code"
Example 2: Encoded data: "12345", Output data: "12345"

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
Variable	Yes	'E'	'e'	from code 0 to 127 (128 codes)	2 : 3 : 4

Example 130000001100000BAR CODE E :<CR>
 130004000900000CODE 128<CR>
 1E0004000800140TO JIMMY<CR>

BAR CODE E :
CODE 128



Fig. A9-5

The following example will print "24681357" by Code 128 subset C.

Example 1E0004000800160C24681357<CR>

Bar code F: EAN-13

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
13 digits (12+1)	Yes	'F'	'f'	0 ~ 9	2 : 3 : 4

Example 130000002060000BAR CODE F : EAN-13<CR>
1F0005001800160135792468228<CR>

BAR CODE F : EAN-13



Fig. A9-6

Bar code G: EAN-8

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
8 digits (7+1)	Yes	'G'	'g'	0 ~ 9	2 : 3 : 4

Example 130000001100000BAR CODE G : EAN-8<CR>
1G30000008001600228001<CR>

BAR CODE G : EAN-8



Fig. A9-7

Bar code H: HBIC

Health Industry Bar Code (HBIC) is same as bar code A (code 3 of 9), except that it includes an additional modulo 43 checksum.

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
Variable	Yes	'H'	'h'	0 ~ 9, A ~ Z, \$%*+-. / and space	2 : 1 ~ 3 : 1

Example 130000002400000BAR CODE H : HBIC<CR>
1H0000001800120HEALTH<CR>

BAR CODE H : HBIC



Fig. A9-8

Bar code I: Coda bar

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
At least 3 characters	No	'I'	'i'	0 ~ 9, A ~ D, \$+-. : And /	2 : 1 ~ 3 : 1

Example 1300000009000000BAR CODE I : <CR>
 1300000011000000CODA BAR<CR>
 1I0000000800100ABCD0123456789<CR>

CODA BAR



Fig. A9-9

Bar code J: Interleaved 2 of 5 with a modulo 10 checksum

Same as bar code D (Interleaved 2 of 5), except that it includes an additional modulo 10 checksum.

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
Variable	Yes	'J'	'j'	0 ~ 9	2 : 1 ~ 3 : 1

Example 130000002100000BAR CODE J : I25<CR>
130000001900000WITH CHECKSUM<CR>
1J000000180016019970701<CR>

BAR CODE J : I25
WITH CHECKSUM



Fig. A9-10

Bar code K: Plessey

An additional checksum will be added to the bar code string where ‘+’ character is inserted.

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
1 ~ 14 digits	Yes	‘K’	‘k’	0 ~ 9	2 : 1 ~ 3 : 1

Example 130000001100000BAR CODE K :<CR>
130000000900000PLESSEY<CR>
1K000000080012050381978<CR>



Fig. A9-11

Bar code L: Interleaved 2 of 5 with a modulo 10 checksum and shipping bearer bars

Same as bar code D (Interleaved 2 of 5) except that it includes a modulo 10 checksum and the horizontal shipping bearer bars.

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
Variable	Yes	'L'	'I'	0 ~ 9	2 : 1 ~ 3 : 1

The horizontal bearer bars exist only when the input digit count is 13.

Example 1300000023000000BAR CODE L : I25<CR>
1300000021000000WITH CHECKSUM &<CR>
1300000019000000BEARER<CR>
1L00060018001401997070187391<CR>

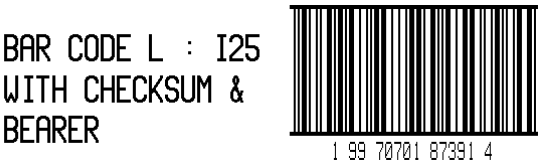


Fig. A9-12

Bar code M: UPC2

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
2 digits	No	'M'	'm'	0 ~ 9	2 : 3 : 4

Example 1300000009000000BAR CODE M : UPC2<CR>
1M000500060016038<CR>

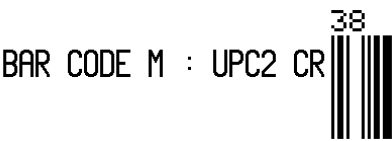


Fig. A9-13

Bar code N: UPC5

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
5 digits	No	'N'	'n'	0 ~ 9	2 : 3 : 4

Example 1300000020000000BAR CODE N : UPC5<CR>
1N000500180016002280<CR>

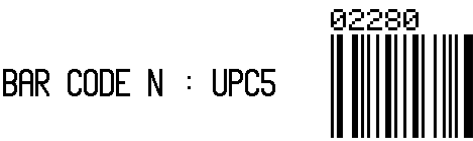


Fig. A9-14

Bar code O: Code 93

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
Variable	Yes	'O'	'o'	0 ~ 9, A ~ Z, \$%+-. / and space	2 : 3 : 4

Example 130000001100000BAR CODE O :<CR>
 130000000900000CODE 93<CR>
 100000000800120CODE 93 OK<CR>



Fig. A9-15

Bar code P: Postnet

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
Variable	Yes	Not defined	'p'	0 ~ 9	

Example 130000002100000BAR CODE P :<CR>
 130000001900000POSTNET<CR>
 1p0006001800120199707<CR>

BAR CODE P :
POSTNET 

Fig. A9-16

Bar code Q: UCC/EAN Code 128

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
20 digits (19+1)	Yes	'Q'	'q'	0 ~ 9	2 : 3 : 4

Example 130000001300000BAR CODE Q<CR>
 130000001100000: UCC/EAN<CR>
 130000000900000128<CR>
 1Q00070010000857812989089990899998<CR>



Fig. A9-17

Bar code R: UCC/EAN Code 128 K-MART

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
18 digits	Yes	'R'	'r'	0 ~ 9	2 : 3 : 4

Example 1300000023000000BAR CODE R<CR>
 130000002100000: UCC/EAN<CR>
 130000001900000128 K<CR>
 1R0006002000083199707011945022800<CR>



Fig. A9-18

Bar code S: UCC/EAN Code 128 Random weight

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
34 digits	Yes	'S'	's'	0 ~ 9	2 : 3 : 4

Example 1300000013000000BAR CODE S : UCC/EAN<CR>
1300000011000000CODE 128 RANDOM WEIGHT<CR>
1S0005000400000123456789012345678901234
5678901234<CR>



Fig. A9-19

Bar code T: Telepen

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
Variable	Yes	'T'	't'	0 ~ 127	2 : 3 : 4

Example 1300000000900000BAR CODE T :<CR>
 1300000000700000TELEPEN<CR>
 1T0005000600120ABC!-=.<CR>

BAR CODE T :
TELEPEN



Fig. A9-20

Bar code V: FIM (Facing Identification Mark)

Length	Check sum	Type for readable string	Type for non-readable string	Valid codes	Bar ratio
1 character	No	Not defined	'v'	A, B, C and D	

Example 130000000900000BAR CODE V :<CR>
 130000000700000FIM<CR>
 1v0000000600160B<CR>

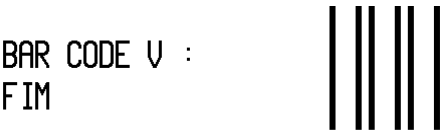


Fig. A9-21

Bar code U: UPS MaxiCode

This is a two dimensional bar code defined by UPS and AIM International. It applies the Reed-Solomon encoding rule. The bar code's data stream consists of 5 different sections:

- a 5-digit primary zip code
- a 4-digit secondary zip code
- a 3-digit country code
- a 3-digit class of service code
- a data string that cannot exceed 84 characters

Example 1300000021000000BAR CODE U :<CR>
 1300000019000000MAXICODE<CR>
 1u0000001500160329874444840555TO
 JIMMY<CR>

BAR CODE U :
MAXICODE

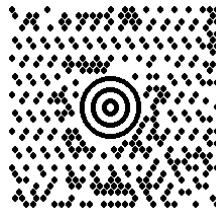


Fig. A9-22

Bar code W: DataMatrix

The Datamatrix is a two dimensional bar code (Only for ECC200 of Datamatrix symbol)

Syntax *aW1c**bd**eeeffffg**ggg**200**jjkkkk**ddddd...**dd***

Parameter

- a*: Print direction: '1', '2', '3' or '4'.
- W1c*: Fixed value, for extended bar code set.
- c*: DataMatrix bar code.
- b*: Horizontal multiplier for module size.
- d*: Vertical multiplier for module size.
- eee*: Always is 000.
- ffff*: Y coordinate.
- gggg*: X coordinate.
- 200*: Fixed value for ECC200 of Datamatrix symbol.
- 0*: Fixed value.
- jj*: A 3 digit even number (or 000) of rows requested.
 000 causes rows to be automatically determined.
- kkk*: A 3 digit even number (or 000) of columns requested.
 000 causes columns to be automatically determined.
- dddd...dd*: data to be encoded and printed.

Example 130000000700050BAR CODE W :<CR>
 130000000500050DATAMATRIX<CR>
 1W1c55000005001702000000000DATA
 MATRIX<CR>

Encode the data "DATA MATRIX"
Horizontal multiplier : 4 Vertical multiplier : 4
Y coordinate : 50 X coordinate : 31

BAR CODE W :
DATAMATRIX



Fig. A9-23

Bar code W1d: QR Code (Auto format)

QR Code has four type valid characters (Numeric data, alphanumeric data, 8-bit byte data, and Kanji). QR Code also has variable length.

Syntax *aW1dcdeeeffffgggghh...h*

Parameter

- a*: Print direction: '1', '2', '3' or '4'.
- W1d*: Fixed value, for extended bar code set.
Automatic format. The data string (hh...h) to be data only.
- c*: Horizontal multiplier for module size. The "c" and "d"
must be equal, because each cell in the bar code is square.
The cell unit depends on the the conversion mode
(*<STX>n* or *<STX>m*). Value: 1~9, A~Z and a~z.
- d*: Vertical multiplier for module size. Value: 1~9, A~Z and
a~z.
- eee*: Value:000~999.(Ineffective; must be numeric)
- ffff*: Y coordinate.
- gggg*: X coordinate.
- hh...h*: Data string.

Example

```
<STX>L<CR>  
1W1d44000010001001233214567<CR>  
121100000400050QR Code-auto  
format (W1d) <CR>  
E<CR>
```



QR Code-auto format(W1d)

Fig. A9-24

Bar code W1D: QR Code (Manual formatting)

QR Code has four type valid characters (Numeric data, alphanumeric data, 8-bit byte data, and Kanji). QR Code also has variable length.

Syntax *a*W1D*cdeeeffffgggg*[*q*,][*e*[*m*]*i*,]*cdata cdata cdata...cdata*

Parameter

- a*: Print direction: '1', '2', '3' or '4'.
- W1D*: Fixed value, for extended bar code set.
Manual format. The data string (hh...h) to be entered with a comma (,) as a separated field, the separated fields are optional in each QR code specifications, and the first field indicates Model1 or Model 2 QR Code. Default model: model 2.
- c*: Horizontal multiplier for module size. The "c" and "d" must be equal, because each cell in the bar code is square. The cell unit depends on the the conversion mode (<STX>*n* or <STX>*m*). Value: 1~9, A~Z and a~z.
- d*: Vertical multiplier for module size. Value: 1~9, A~Z and a~z.
- eee*: Value:000~999.(Ineffective; must be numeric)
- ffff*: Y coordinate.
- gggg*: X coordinate.
- q*: QR Code model number; optional.
Values: 1, 2
Default value: Model 2.
- e*: Error correction level.
Values: H= Ultra reliability level (30%)
Q= High reliability level (25%)
M= Standard reliability level (15%)

L= High density level (7%)
m: Mask number; optional.
 Values: None= Automatic selection
 0~7= Mask 0~Mask 7
 8= No mask
i: Data input mode.
 Values: A= Automatic setting, ASCII
 a=Automatic, hex-ASCII
 M=Manual setting, ASCII
 m=Manual, hex-ASCII
cdata: Character mode.
 Values: N=Numeric, N data
 A=Alphanumeric, A data
 Bxxxx= For 8-bit byte mode; xxxx is indicate number of characters. (A Simple or Traditional Chinese word is 2 characters)
 K=Kanji, K data
 (The four values are immediately followed by data.)

Example

```

<STX>L<CR>
1W1D55000008001002,HA,Athis QR is OK<CR>
1W1D55000008002002,HM,N7533967<CR>
1W1D55000018001002,Hm,Athis QR is OK<CR>
121100000400020QR Code-Manual format(W1D)<CR>
E<CR>
  
```



QR Code-Manual format(W1D)

Fig. A9-25

Bar code W1f: Aztec Bar Code (Variable Length)

Syntax *aW1fcdeeeffffgggqijjkk...k*

Parameter

<i>a</i> :	Print direction: '1', '2', '3' or '4'.
<i>W1f</i> :	Fixed value, for extended bar code set. This parameter sets Aztec bar code length as variable.
<i>c</i> :	Horizontal multiplier for module size. Default: 0; Value: 1~9, A~Z and a~z.
<i>d</i> :	Vertical multiplier for module size. Default: 0; Value: 1~9, A~Z and a~z.
<i>eee</i> :	Ineffective; must be 000.
<i>ffff</i> :	Y coordinate.
<i>gggg</i> :	X coordinate.
<i>i</i> :	ECI (Extended Channel Interpretation) mode. Disable ECI mode: 0, Enable ECI mode: 1
<i>jjj</i> :	Error correction. 000: Default value, approximately 23%. 001~099: Error correction percentage. 101~104: 1 to 4 layers in compacted symbol 201~232: 1 to 32 layers in full symbol. 300: Rune symbol
<i>kk...k</i> :	Data string.

Example

```
<STX>L<CR>
1W1f88000017001001000AZTEC
barcode(W1f)<CR>
122200002700100Aztec Barcode(W1f) <CR>
E<CR>
```

Aztec Barcode(W1f)



Fig. A9-26

Bar code W1F: Aztec Bar Code (Specified Length/Byte count)

Syntax *a*W1F*cdeeeffffgggghhhhiijjkk...k*

Parameter

<i>a</i> :	Print direction: '1', '2', '3' or '4'.
W1F:	Fixed value, for extended bar code set. This parameter sets Aztec bar code length as specified with a Byte Count Specifier.
<i>c</i> :	Horizontal multiplier for module size. Default: 0; Value: 1~9, A~Z and a~z.
<i>d</i> :	Vertical multiplier for module size. Default: 0; Value: 1~9, A~Z and a~z.
<i>eee</i> :	Ineffective; must be 000.
<i>ffff</i> :	Y coordinate.
<i>gggg</i> :	X coordinate.
<i>hhhh</i> :	Length Specifies. The field termination is set by Byte Count. This value includes all of the data following this byte count field, but not include itself.
<i>i</i> :	ECI (Extended Channel Interpretation) mode. Disable ECI mode: 0, Enable ECI mode: 1
<i>jjj</i> :	Error correction. 000: Default value, approximately 23%. 001~099: Error correction percentage. 101~104: 1 to 4 layers in compacted symbol 201~232: 1 to 32 layers in full symbol. 300: Rune symbol
<i>kk...k</i> :	Data string.

Example

```
<STX>L<CR>  
1W1F880000025010000220000AZTEC barcode (W1F) <CR>  
122200001250100Aztec Barcode (W1F) <CR>  
E<CR>
```

Aztec Barcode (W1F)



Fig. A9-27

Bar code W1k: RSS (GS1 DataBar)

This command can print all RSS-14 bar code. (RSS-14, RSS Limited, RSS-14 Stacked, RSS-14 Truncated, RSS Expanded and RSS Stacked Omnidirectional)

Syntax aW1kcdeeeffffqqqghijmn...n / p...p (parameter1)
 aW1kcdeeeffffqqqghijmnn p...p / q...q (parameter2)

Parameter 1

- a*: Print direction: '1', '2', '3' or '4'.
W1: Fixed value, for extended bar code set.
k: RSS bar code.
c: Bar width ratio. No reference. (*Value= bar narrow ratio.*)
d: Bar narrow ratio. Default value=2. (Valid value=1~9)
eee: Barcode height. Default value: refer to the following table.
 (If barcode height value=0, the actual barcode height
 ='default'*'d'*'i', the actual barcode height='eee'.
 The 'eee' unit is set by the "<STX>n" or "STX>m"
 command.)

Barcode	Description
R	RSS-14 (GS1 DataBar Omnidirectional) Width multiplier: 96 pixels; Min. Height Multiplier: 33 pixels (default)
L	RSS Limited (GS1 DataBar Limited) Width multiplier: 74pixels; Min. Height Multiplier: 10 pixels (default)
S	RSS Stacked (GS1 DataBar Stacked) Width multiplier: 50 pixels; Min. Height Multiplier: 13 pixels (default)
T	RSS Truncated

	(GS1 DataBar Truncated) Width multiplier: 96pixels; Min. Height Multiplier: 13 pixels (default)
D	RSS Stacked Omnidirectional (GS1 DataBar Stacked Omnidirectional) Width multiplier: 50 pixels; Min. Height Multiplier: 69 pixels (default)

ffff: Y coordinate. Value: 0000~9999
gggg: X coordinate. Value: 0000~9999
h: RSS type
R = RSS-14 (GS1 DataBar Omnidirectional)
T = RSS-14 Truncated (GS1 DataBar Truncated)
S = RSS-14 Stacked (GS1 DataBar Stacked)
D = RSS Stacked Omnidirectional
(GS1 DataBar Stacked Omnidirectional)
L = RSS Limited (GS1 DataBar Limited)
i: Pixel multiplier. Default: 1; Value: 1~9
j: No reference. Value= always 0.
m: No reference. Value= always 0.
n...n: Value: 0~9, numeric ; numeric linear data, length 13.
/: Optional. Vertical bar separates primary data from secondary 2D data.
p...p: 2D data.

Parameter 2

a: Print direction: '1', '2', '3' or '4'.
W1: Fixed value, for extended bar code set.
k: RSS bar code.
c: Bar width ratio. No reference. (Value= bar narrow ratio.)
d: Bar narrow ratio. Default value=2. (Valid value=1~9)
eee: Barcode height. Default value: refer to the following table.

(If barcode height value=0, the actual barcode height='default'*'d'*'i', the actual barcode height='eee'. The 'eee' unit is set by the "<STX>n" or "STX>m" command.)

Barcode	Description
E	RSS Expanded (GS1 DataBar Expanded) Width multiplier: 102~534 pixels; Min. Height Multiplier: 34pixels (default) Row:1~11

ffff: Y coordinate. Value: 0000~9999
gggg: X coordinate. Value: 0000~9999
h: RSS type
 E = RSS Expanded (GS1 DataBar Expanded)
i: Pixel multiplier. Default: 1; Value: 1~9
j: No reference. Value= always 0.
m: No reference. Value= always 0.
nn: Segments per row. Only work in RSS Expanded
 Default value: 22; Accepted Values: 2-22(even only)
p...p: Alphanumeric. *Value: 0~9*
/: Optional. Vertical bar separates primary data from
 secondary 2D data.
q...q: 2D data.

Example

```
<STX>L<CR>
D11<CR>
1W1k0000000350050R10089121121|RSS-14<CR>
1W1k0000001150050L10089121121|RSS Limited<CR>
1W1k0000001750050S10089121121|RSS Stacked<CR>
1W1k0000002450050T10089121121|RSS Truncated<CR>
1W1k0000003150050D10089121121|RSS Stacked
Omnidirectional<CR>
```

1W1k0000004550050E100220188689121121|RSS
Expanded<CR>
1W1k0000005450050E100040188689121121|RSS Expanded
Stacked<CR>
121100000200050RSS-14<CR>
121100001000050RSS Limited<CR>
121100001600050RSS Stacked<CR>
121100003000050RSS Truncated<CR>
121100002300050RSS Stacked Omnidirectional<CR>
121100004400050RSS Expanded<CR>
121100005300050RSS Expanded StackedQ0001<CR>
E<CR>



RSS Expanded



RSS Truncated



RSS Stacked Omnidirectional



RSS Stacked



RSS Limited



RSS-14

Fig. A9-28

Bar code Z: PDF-417

This is a two dimensional bar code defined by AIM International. It applies the Reed-Solomon encoding rule and includes all ASCII characters. It provides the function of multiple-level error detection and correction. The bar code's data stream consists of 6 different sections:

Length	Description
1	F : normal, T : truncated
1	0 ~ 8 : security level
2	00 ~ 99 : aspect ratio, 00 stands for 1:2
2	03 ~ 90 : row number, 00 for best fit
2	01 ~ 30 : column number, 00 for best fit
Variable	Data string

Example 130000002100000BAR CODE Z :<CR>
 130000001900000PDF-417<CR>
 1z4900001800140F0001002ABCDEFGHI<CR>



Fig. A9-29

Bar code W1z: MiocroPDF417

Syntax gW1zcdeeeffffgggghijkom...m

Parameter

- a: Print direction: '1', '2', '3' or '4'.
- W1z: Fixed value, for extended bar code set.
This parameter sets MicroPDF417.
- c: Horizontal multiplier for module size.
Default: 0; Value: 1~9, A~Z and a~z.
- d: Vertical multiplier for module size.
Default: 0; Value: 1~9, A~Z and a~z.
- eee: Ineffective; must be 000.
- ffff: Y coordinate.
- gggg: X coordinate.
- h: Column number; Value: 1~4.
- i: Row number; Value: 0~9 and A.
- j: Byte compaction mode for binary data.
Auto mode: 0; Byte compaction mode: 1.
- k: Substitute Macro character function.
Disable substitution mode: 0; Enable substitution mode: 1
- o: Fixed value.
- m...m: Data string.

h	i	Columns	Rows	Max Binary Data (Bytes)	Max Alphabetic Characters	Max Numeric Characters
1	0	1	11	3	6	8
1	1	1	14	7	12	17
1	2	1	17	10	18	26
1	3	1	20	13	22	32
1	4	1	24	18	30	44
1	5	1	28	22	38	55
2	0	2	8	8	14	20
2	1	2	11	14	24	35

2	2	2	14	21	36	52
2	3	2	17	27	46	67
2	4	2	20	33	56	82
2	5	2	23	38	67	93
2	6	2	26	43	72	105
3	0	3	6	6	10	14
3	1	3	8	10	18	26
3	2	3	10	15	26	38
3	3	3	12	20	34	49
3	4	3	15	27	46	67
3	5	3	20	39	66	96
3	6	3	26	54	90	132
3	7	3	32	68	114	167
3	8	3	38	82	138	202
3	9	3	44	97	162	237
4	0	4	4	8	14	20
4	1	4	6	13	22	32
4	2	4	8	20	34	49
4	3	4	10	27	46	67
4	4	4	12	34	58	85
4	5	4	15	45	76	111
4	6	4	20	63	106	155
4	7	4	26	85	142	208
4	8	4	32	106	178	261
4	9	4	38	128	214	313
4	A	4	44	150	250	366

Example

```
<STX>L<CR>
1W1z460000130002044000MICROPDF417<CR>
121100001800030Micro PDF417(W1z)<CR>
E<CR>
```

Micro PDF417(W1z)



Fig. A9-30

Bar code W1Z: MiicroPDF417 (with Specified Byte Count)

Syntax *aW1Zcdeeeffffgggghhhhm...m*

Parameter

- a*: Print direction: '1', '2', '3' or '4'.
- W1Z*: Fixed value, for extended bar code set. This parameter sets MicroPDF417 (with Specified Byte Count).
- c*: Horizontal multiplier for module size.
 Default: 0; Value: 1~9, A~Z and a~z.
- d*: Vertical multiplier for module size.
 Default: 0; Value: 1~9, A~Z and a~z.
- eee*: Ineffective; must be 000.
- ffff*: Y coordinate.
- gggg*: X coordinate.
- hhhh*: Specify Byte count.
- m...m*: Data string; data string must refer to *hhhh*.

Example <STX>L<CR>
 1W1Z4600001300020001220000MICRO<0X0D> Z<
 CR>
 121100001800030Micro PDF417 (W1Z) <CR>
 E<CR>

Micro PDF417(W1z)



Fig. A9-31

APPENDIX AA: HEX GRAPHIC FORMAT

Unlike the PCX, BMP and IMG formats, the HEX format is a proprietary one. It consists of 3 types of records.

Record type	Format	Description
Data	80 <u>xx</u> [...]	<u>xx</u> : hex value, stands for byte count. [...] : image data, 2 hex digits represent one byte raster image.
Repeat	0000FF <u>xx</u>	<u>xx</u> : repeat count. The repeated data will appear at the subsequent data record. The maximum value is 255 (FFH). If the actual repeat count is more than 255, split it to fit the range.
End	FFFF	End the HEX file.

APPENDIX AB: FONTS

Internal Fonts Fonts 0~8 have single symbol set.

Font 0

20H~2FH	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/	
30H~3FH	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40H~4FH	a	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50H~5FH	P	Q	R	S	T	U	U	W	X	Y	Z	[	\	]	^	_
60H~6FH	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70H~7FH	p	q	r	s	t	u	v	w	x	y	z	{		}	~	8
F0H~FFH																e

Fig. AB-1

Font 1

20H~2FH	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/	
30H~3FH	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40H~4FH	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50H~5FH	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
60H~6FH	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70H~7FH	p	q	r	s	t	u	v	w	x	y	z	{		}	~	■
80H~AFH	á	í	ó	ú	ñ	Ñ	æ	ø	¿			½	¼			
E0H~EFH	ß															
F0H~FFH																€

Fig. AB-2

Font 2

20H~2FH	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/	
30H~3FH	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40H~4FH	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50H~5FH	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
60H~6FH	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70H~7FH	p	q	r	s	t	u	v	w	x	y	z	{		}	~	⌘
80H~AFH	á	í	ó	ú	ñ	Ñ	à	ä	ç						½	¼
E0H~EFH	ß															
F0H~FFH																€

Fig. AB-3

Font 3

20H~2FH	# \$ % & () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 :
40H~4FH	A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z
60H~6FH	A B C D E F G H I J K L M N O
70H~7FH	P Q R S T U V W X Y Z
80H~8FH	Ñ ¿
E0H~EFH	ß
F0H~FFH	€

Fig. AB-4

Font 4

20H~2FH	# \$ % & () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 :
40H~4FH	A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z
60H~6FH	A B C D E F G H I J K L M N O
70H~7FH	P Q R S T U V W X Y Z
80H~AFH	Ñ ÿ
E0H~EFH	ß
F0H~FFH	€

Fig. AB-5

Font 5

20H~2FH	# \$ % & () * + , - . /															
30H~3FH	0	1	2	3	4	5	6	7	8	9	:					
40H~4FH	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
50H~5FH	P	Q	R	S	T	U	V	W	X	Y	Z					
60H~6FH	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	
70H~7FH	P	Q	R	S	T	U	V	W	X	Y	Z					
80H~AFH						Ñ				¿						
E0H~EFH	ß															
F0H~FFH																€

Fig. AB-6

Font 6

20H~2FH	# \$ % & () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 :
40H~4FH	A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z
60H~6FH	A B C D E F G H I J K L M N O
70H~7FH	P Q R S T U V W X Y Z
80H~8FH	Ñ ¿
90H~9FH	ß
E0H~EFH	
F0H~FFH	€

Fig. AB-7

Font 7

20H~2FH	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/	
30H~3FH	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40H~4FH	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50H~5FH	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
60H~6FH	h	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70H~7FH	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

Fig. AB-8

Font 8

20H~2FH											+		
30H~3FH	0	1	2	3	4	5	6	7	8	9		<	>
40H~4FH				C		E							N
50H~5FH				S	T	U			X	Z			
60H~6FH				C		E							N
70H~7FH				S	T				X	Z	I		

Fig. AB-9

Font 9

Font 9 (ASD smooth font sets) include 6 symbol sets. There are PC850/Windows 3.1 Latin 1/ Windows 3.1 Latin 2/Windows Latin, Greek/Windows Latin, Cyrillic/ Windows 3.1 Latin 5.

PC850

20H~2FH ! " # \$ % & ' () * +, - . /
 30H~3FH 0 1 2 3 4 5 6 7 8 9 : ; < = > ?
 40H~4FH @ A B C D E F G H I J K L M N O
 50H~5FH P Q R S T U V W X Y Z [\] ^ _
 60H~6FH ` a b c d e f g h i j k l m n o
 70H~7FH p q r s t u v w x y z { | } ~ □
 80H~AFH á í ó ú ñ Ñª ° ¿ ® ¬ ½ ¼ ; « »
 B0H~BFH ■■■■ | | Á Â À © ¶ || ¶ ¢ ¥ ¶
 C0H~CFH ¯ ⊥ ⊥ ⊥ ⊥ ⊥ ã Ã ℒ ℞ ℞ ℞ ℞ ℞
 D0H~DFH ð ð Ê Ë È Í Î Ï ¯ ■ ■ ■
 E0H~EFH Ó Ô Õ Ö Õ μ ρ ρ Ú Û Ü Ý Ý ¯
 F0H~FFH - ± ¾ ¶ § ÷ ° ° ° ° ° ° ° °

Fig. AB-10

Windows 3.1 Latin 1

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ 
80H~8FH	¡ ¢ £ ¤ ¥ ¦ § ¨ © ª « ¬ ® ¯
90H~9FH	° ± ² ³ ´ µ ¶ · ¸ ¹ º » ¼ ½ ¾ ¿
A0H~AFH	À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï
B0H~BFH	Ð Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ ß
C0H~CFH	à á â ã ä å æ ç è é ê ë ì í î ï
D0H~DFH	ð ñ ò ó ô õ ö ÷ ø ù ú û ü ý þ €
E0H~EFH	
F0H~FFH	

Fig. AB-11

Windows 3.1 Latin 2

! " # \$ % & ' () * +, - . /
 0 1 2 3 4 5 6 7 8 9 : ; < = > ?
 @ A B C D E F G H I J K L M N O
 P Q R S T U V W X Y Z [\] ^ _
 ` a b c d e f g h i j k l m n o
 p q r s t u v w x y z { | } ~ 
 ˇ ˘ Ł Œ Å | § ¨ © Š « ¬ ® Ž
 ° ± ı ´ μ ¶ · ¸ ą ș » Ľ " ĺ ž
 Ř Á Â Ã Ä Å Ĺ Ć Ĉ Ċ É Ê Ë Ě Í Î Ď
 Đ Ň Ń Ó Ô Õ Ö × Ŕ Ů Ú Ů Ü Ý Ţ ß
 ı á â ã ä å í ċ ĉ ċ é ě ě ě í î ð
 đ ń ň ó ô õ ö ÷ ř ů ú ů ü ý ț €

Fig. AB-12

Windows Latin, Greek

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ ▒
80H~8FH	“ ’ ¤ £ ¤ ¥ § ¨ © ª « ¬ ® —
90H~9FH	° ± ² ³ ´ µ ¶ · ¸ ¹ º » ¼ ½ ¾
A0H~AFH	ı̇ Α Β Γ Δ Ε Ζ Η Θ Ι Κ Λ Μ Ν Ξ Ο
B0H~BFH	Π Ρ Σ Τ Υ Φ Χ Ψ Ω Ĭ Ÿ á é ħ í
C0H~CFH	û α β γ δ ε ζ η θ ι κ λ μ ν ξ ο
D0H~DFH	π ρ ς σ τ υ φ χ ψ ω ĩ ü ó ú €
E0H~EFH	
F0H~FFH	

Fig. AB-13

Windows Latin, Cyrillic

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ 
80H~8FH	Ў ў Ј ѡ Г § Ё © € « ¬ ® Ĭ
90H~9FH	° ± І і г μ ¶ · ё № » ј S s ĩ
A0H~AFH	А Б В Г Д Е Ж З И Й К Л М Н О П
B0H~BFH	Р С Т У Ф Х Ц Ч Ш Щ Ъ Ы Ь Э Ю Я
C0H~CFH	а б в г д е ж з и й к л м н о п
D0H~DFH	р с т у ф х ц ч ш щ ъ ы ь э ю я
E0H~EFH	
F0H~FFH	

Fig. AB-14

Windows 3.1 Latin 5

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ 
80H~AFH	ı ¢ £ ¤ ¥ ¦ § ¨ © ª « ¬ ® ¯
B0H~BFH	° ± ² ³ ´ µ ¶ · ¸ ¹ º » ¼ ½ ¾ ¿
C0H~CFH	À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï
D0H~DFH	Ğ Ń Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ
E0H~EFH	à á â ã ä å æ ç è é ê ë ì í î ï
F0H~FFH	ğ ñ ò ó ô õ ö ÷ ø ù ú û ü ý ÿ

Fig. AB-15

Courier Font set

The Courier font set is only for 200dpi printers. It includes Roman-8, PC, PC-A, PC-B, EAMA-94, Legal, Greek and Russian symbol sets.

Roman-8

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ ▒
80H~AFH	À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ ß à á â ã
B0H~BFH	ä å æ ç è é ê ë ì í î ï ð ñ ò ó ô õ ö ù
C0H~CFH	â ê ô û á é ó ú à è ò ù ä ë ö ü
D0H~DFH	Å î ø Æ å í ø æ Ä ì Ö Ü É ï ß Ô
E0H~EFH	Á Ã ã Ð ð Í Î Ó Ò Õ õ Š š Ú Ÿ Ÿ
F0H~FFH	Ɔ Ɔ · μ ¶ ¾ − ¼ ½ ¾ ¾ « ■ » ±

Fig. AB-16

ECMA-94

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ ☐
80H~AFH	ı ċ £ ¤ ¥ ¦ § ¨ © ª « ¬ ® ¯
B0H~BFH	° ± ² ³ ´ µ ¶ · ¸ ¹ º » ¼ ½ ¾ ¿
C0H~CFH	À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï
D0H~DFH	Ð Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ ß
E0H~EFH	à á â ã ä å æ ç è é ê ë ì í î ï
F0H~FFH	ð ñ ò ó ô õ ö ÷ ø ù ú û ü ý þ ÿ

Fig. AB-17

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ △
80H~AFH	á í ó ú ñ Ñ ã ö ç ÿ ½ ¼ ; « »
B0H~BFH	⋮ ⋱ ⋲ † ‡ † † † † † † † † † † † † † †
C0H~CFH	⌞ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥
D0H~DFH	⌞ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥
E0H~EFH	α β γ π Σ σ μ τ Φ Θ Ω δ ∞ ϕ ε ∩
F0H~FFH	≡ ± ≥ ≤ ∫ ∫ ÷ ≈ ° • • √ n 2 ■

Fig. AB-18

PC-A

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ △
80H~AFH	á í ó ú ñ Ñ ã Ã ¨ ñ ; ³ ¤
B0H~BFH	⋮ ⋯ ⋰ † ‡ † ‡ † ‡ † ‡ † ‡ † ‡ † ‡ † ‡
C0H~CFH	⌞ ⊥ ⊥ † † † † † † † † † † † † † † † †
D0H~DFH	⌞ ⊥ ⊥ † † † † † † † † † † † † † † † †
E0H~EFH	α β γ π Σ σ μ τ Φ Θ Ω δ ∞ ϕ ∈ ∩
F0H~FFH	≡ ± ≥ ≤ ∫ ∫ ÷ ≈ ° • · √ n ² ■

Fig. AB-19

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ △
80H~AFH	á í ó ú ñ Ñ ã ö ç ® ¬ ½ ¼ ¡ « »
B0H~BFH	⋮ ⋘ ⋙ † Á Â Ã © ¶ ¶ ¶ ¤ ¥ 7
C0H~CFH	ℒ ⊥ ⊥ † − † ã Ã ℒ ¶ ⊥ ¶ ¶ = ¶ α
D0H~DFH	ö ð ê ë è ı í î ï 7 ¶ ■ ■ î ■
E0H~EFH	ó ß ô ò õ ã μ þ ð ú û û ý ý - ´
F0H~FFH	− ± = ¾ ¶ § ÷ ˘ ° ° . 1 3 2 ■

Fig. AB-20

Legal

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; _ = ¢ ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [®] © _
60H~6FH	° a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z \$ ¤ † ™ ▒
A0H~AFH	
B0H~BFH	
C0H~CFH	
D0H~DFH	
E0H~EFH	
F0H~FFH	

Fig. AB-21

Greek

20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ Δ
80H~AFH	ι η λ μ ν ξ ο π ρ σ ς τ υ φ χ ψ
B0H~BFH	▒ ▓ ▔ ⊥ ⊢ ⊣ ⊤ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥
C0H~CFH	⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥
D0H~DFH	⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥ ⊥
E0H~EFH	ω α ε η ÿ í ó ú ü Å Æ Ħ ı Ÿ
F0H~FFH	Ω ± ≥ ≤ ∫ ∫ ÷ ≈ ° £ ¥ √ n 2 ■

Fig. AB-22

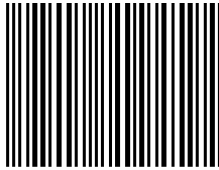
Russian

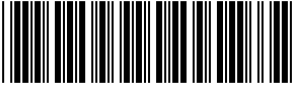
20H~2FH	! " # \$ % & ' () * + , - . /
30H~3FH	0 1 2 3 4 5 6 7 8 9 : ; < = > ?
40H~4FH	@ A B C D E F G H I J K L M N O
50H~5FH	P Q R S T U V W X Y Z [\] ^ _
60H~6FH	` a b c d e f g h i j k l m n o
70H~7FH	p q r s t u v w x y z { } ~ Δ
A0H~AFH	а б в г д е ж з и й к л м н о п
B0H~BFH	⋮ ▨ ▩ † ‡ ╥ ╦ ╧ ╨ ╩ ╪ ╫ ╬ ╭ ╮ ╯ ╰
C0H~CFH	L ⊥ ⊥ † − ‡ ‣ ․ ‥ ‧     ‪ ‫ ‬ ‭ ‮   ‰
D0H~DFH	‖ ⁀ ⁁ ⁂ ⁃ ⁄ ⁅ ⁆ ⁇ ⁈ ⁉ ⁊ ⁋ ⁌ ⁍ ⁎ ⁏
E0H~EFH	р с т у ф х ц ч ш щ ъ ы ь э ю я
F0H~FFH	Ë Ò ≥ ≤ ∫ ∫ ÷ ≈ ° • · √ n ² ■

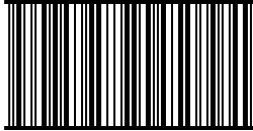
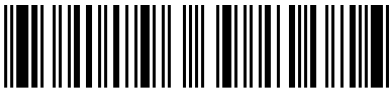
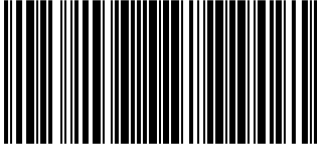
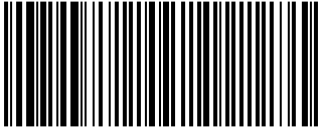
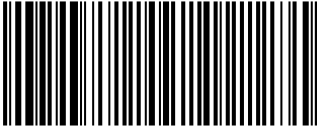
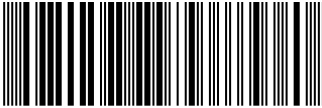
Fig. AB-23

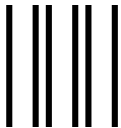
APPENDIX AC: Bar Codes

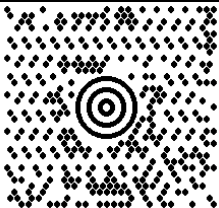
The PPLA supports the following internal bar codes.

1D Bar Code Type	
Code 3 of 9	 19450228
UPC-A	 0 22812 34567 4
UPC-E	 0 654321 7
Interleaved 2 of 5 (I25)	 0135792468
Code 128	 TO JIMMY

EAN-13	 1 357924 682287
EAN-8	 0228 0011
HBIC	 HEALTHM
Coda bar	 RBCD0123456789
Interleaved 2 of 5 with a modulo 10 checksum	 0199707018
Plessey	 50381978

Interleaved 2 of 5 with a modulo 10 checksum and shipping bearer bars	 1 99 70701 87391 4
UPC2	
UPC5	
Code 93	 CODE 93 OK
Postnet	
UCC/EAN Code 128	<78> 1 2989089 990899998 1 
UCC/EAN Code 128 K-MART	70 701194 502 2800 
UCC/EAN Code 128 Random weight	70 701194 502 2800 
Telepen	 ABC1--

FIM (Facing Identification Mark)	
RSS-14 (GS1 DataBar Omnidirectional)	
RSS Limited (GS1 DataBar Limited)	
RSS Stacked (GS1 DataBar Stacked)	
RSS Truncated (GS1 DataBar Truncated)	
RSS Stacked Omnidirectional (GS1 DataBar Stacked Omnidirectional)	
RSS Expanded (GS1 DataBar Expanded)	

2D Bar Code Type	
UPS MaxiCode	

DataMatrix	
QR Code (Auto format)	
QR Code (Manual formatting)	
Aztec Bar Code (Variable Length)	
Aztec Bar Code (Specified Length/Byte count)	
PDF-417	
MiicroPDF417	
MiicroPDF417 (with Specified Byte Count)	

APPENDIX AD: DENSITY AND SPEED

PPLA	Command Syntax	WLP-660 WLP-666	WP-660 WP-666	WP-880 WP-888	WL-880 WL-888	I4-240	I4-250	I4-350
Density range	Hxx	00~20	00~20	00~20	00~20	00~20	00~20	00~20
Default density		5	10	10	10	10	10	10
Speed range	Pn (Unit: ips)	2~6	2~7	2~6	2~6	2~6	2~8	2~6
Default speed		5	5	3	3	5	6	5

APPENDIX AE: COMMAND REFERENCE CHART

INTERACTION COMMANDS	DESCRIPTION	WLP series	WL series WP series	I4 series	
<SOH>*	Restarts the printer		*	*	
<SOH>A	Sends a readable status string		*	*	
<SOH>B	Toggles pause condition		*	*	
<SOH>E	Sends preset label to be printed		*	*	
<SOH>F	Sends one byte printer status		*	*	
SYSTEM LEVEL COMMANDS	DESCRIPTION	WLP series	WL series WP series	I4 series	
<STX>A	Sets date and time	*	*	*	
<STX>a	Enables label echo character		*	*	
<STX>B	Gets date and time		*	*	
<STX> c	Sets continuous label length	*	*	*	
<STX>E	Sets copy count for stored label	*	*	*	
<STX>e	Selects edge sensor for gap		*	*	
<STX>F	Feeds a label	*	*	*	
<STX>f	Sets stop position and automatic back-feed for the label stock	*	*	*	
<STX>G	Prints stored label format	*	*	*	
<STX>I	Downloads the graphics file	*	*	*	
<STX>i	Downloads the scalable font	*	*	*	
<STX>KcCC	Sets control code set	*	*	*	
<STX>KcCF	Sets horizontal shift	*	*	*	
<STX>KcCL	Sets continuous label length	*	*	*	
<STX>KcSPA	Sets serial port		*	*	
<STX>KW	Set label width	*	*	*	
<STX>k	Test RS232 port		*	*	
<STX>L	Enters label formatting state	*	*	*	
<STX>M	Sets maximum label length	*	*	*	
<STX>MT	Sets transfer type				
<STX>m	Sets measurement to metric	*	*	*	
<STX>n	Sets measurement to inches	*	*	*	
<STX>O	Sets print start position	*	*	*	
<STX>P	Enters data dump mode	*	*	*	
<STX>p	Enters pause mode	*	*	*	
<STX>Q	Clears memory	*	*	*	
<STX>q	Clears memory module	*	*	*	
<STX>r	Selects reflective sensor for gap	*	*	*	
<STX>S	Retrieves the global register contents	*	*	*	
<STX>T	Prints test pattern	*	*	*	
<STX>U	Replaces the data of the specified data	*	*	*	
<STX>V	Sets cutter and dispenser configuration	*	*	*	
<STX>v	Inquires the printer version		*	*	
<STX>W	Inquires the graphics/fonts and memory status		*	*	
<STX>X	Sets default module	*	*	*	
<STX>x	Releases file from memory	*	*	*	
<STX>y	Print ASD smooth font (Symbol set)	*	*	*	
<STX>Z	Print Configuration	*	*	*	
LABEL FORMATING COMMANDS	DESCRIPTION	WLP series	WL series WP series	I4 series	
:	Sets cut by amount	*	*	*	
A	Sets logic image printing mode	*	*	*	
C	Sets left margin	*	*	*	
c	Sets cut by amount	*	*	*	

D	Sets width and height pixel size	*	*	*	
E	Ends the job and exit from label formatting mode	*	*	*	
G	Stores previous data to global register	*	*	*	
H	Sets heat value	*	*	*	
M	Toggles the mirror mode	*	*	*	
m	Sets measurement in metric	*	*	*	
n	Sets measurement in inch	*	*	*	
P	Sets print speed	*	*	*	
Q	Sets the quantity of labels to print	*	*	*	
R	Sets vertical offset	*	*	*	
r	Retrieves label data to printer buffer	*	*	*	
s	Stores label data to printer buffer	*	*	*	
T	Sets end-of-line code	*	*	*	
X	Exits from Label Formatting mode	*	*	*	
z	Changes slash zero to normal 0	*	*	*	
+	Makes auto increment for numeric	*	*	*	
>	Makes auto increment for alphanumeric	*	*	*	
-	Makes auto decrement for numeric	*	*	*	
<	Makes auto decrement for alphanumeric	*	*	*	
^	Sets count by amount	*	*	*	
<STX>T	Prints date and time	*	*	*	
IMAGE EDITING COMMANDS	DESCRIPTION	WLP series	WL series WP series	I4 series	
<u>Rthvoooyyyxxxx[d ata string]</u>	Text	*	*	*	
<u>Rthvoooyyyxxxx[d ata string]</u>	Bar Codes	*	*	*	
<u>RX11000yyyxxL aaabbb</u>	Line	*	*	*	
<u>RX11000yyyxxla aaabbbb</u>		*	*	*	
<u>RX11000yyyxxxB aaabbbtttsss</u>	Box	*	*	*	
<u>RX11000yyyxxxb aaaabbbtttsss</u>		*	*	*	
<u>1Y11000yyyxxn. .n</u>	Graphic	*	*	*	