

## Table of Contents

<b>Setup and System Commands.....</b>	<b>4</b>
SIZE .....	4
GAP .....	5
BLINE.....	6
OFFSET .....	8
SPEED .....	9
DENSITY .....	9
DIRECTION and Mirror Image.....	10
REFERENCE.....	10
CODEPAGE .....	11
CLS .....	12
FEED .....	12
BACKFEED .....	13
FORMFEED .....	13
HOME.....	14
PRINT .....	14
SOUND.....	15
LIMITFEED .....	16
SELFTEST.....	17
<b>Label Formatting Commands.....</b>	<b>17</b>
BAR .....	17
BARCODE .....	18
BITMAP .....	22
BOX.....	23
ERASE.....	24
PUTBMP.....	25
PUTPCX .....	26
QRCODE.....	27
REVERSE.....	30
TEXT .....	30
<b>Status Polling Commands(RS-232) .....</b>	<b>32</b>
<ESC>!?.....	32
<ESC>!R.....	33

~!@ .....	33
~!A .....	33
~!C .....	34
~!D .....	34
~!F .....	34
~!I .....	35
~!T .....	35
<b>FileManagement Commands .....</b>	<b>35</b>
DOWNLOAD .....	35
EOP .....	38
FILES .....	39
KILL .....	39
MOVE .....	40
RUN .....	40
<b>BASIC Commands and Functions .....</b>	<b>41</b>
ABS( ) .....	41
ASC( ) .....	41
CHR\$( ) .....	42
END .....	42
EOF( ) .....	43
OPEN .....	44
READ .....	45
SEEK .....	46
LOF( ) .....	47
FREAD\$( ) .....	47
FOR...NEXT .....	48
IF...THEN...ELSE...ENDIF .....	49
GOSUB...RETURN .....	52
GOTO .....	53
REM .....	54
INT( ) .....	54
LEFT\$( ) .....	55
LEN( ) .....	55
MID\$( ) .....	56

RIGHT\$( ).....	57
STR\$( ).....	57
VAL( ).....	58
BEEP.....	59
<b>Device Reconfiguration Commands.....</b>	<b>59</b>
SET COUNTER.....	59
SET KEY1, SET KEY2.....	60
SET PEEL.....	61
SET TEAR &SET STRIPPER.....	62
SET HEAD .....	63
SET RIBBON .....	63
SET COM1 .....	64
SETPRINTKEY.....	65
SET REPRINT.....	66
PEEL.....	67
KEY1, KEY2 .....	68
<b>Printer Global Variables .....</b>	<b>68</b>
@LABEL.....	68

# Setup and System Commands

## SIZE

- [Description]

This command defines the label width and length.
- [Syntax]

(1)English system (inch)

SIZE m, n

(2)Metric system (mm)

SIZE m mm, n mm

(3)Dot measurement

This command has been supported since V6.27 EZ and later firmware.

SIZE m dot,n dot

Parameter	Description
m	Label width (inch or mm)
n	Label length (inch or mm)

**Notes:** 203 DPI: 1 mm = 8 dots  
300 DPI : 1mm = 12 dots

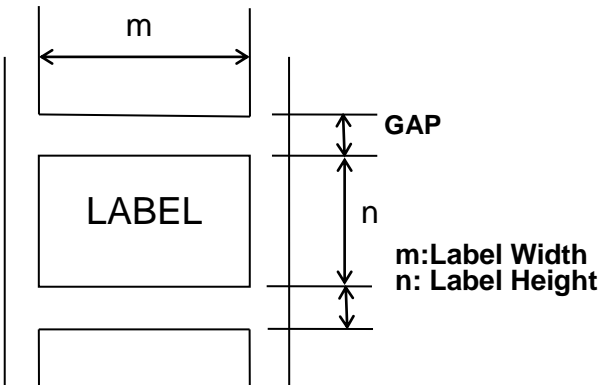
- For metric and dot systems, there must be a space between parameter and “mm” or “dot”.
- [Example]

(1) English system (inch)

SIZE 3.5, 3.00

(2) Metric system (mm)

SIZE 100 mm, 100 mm



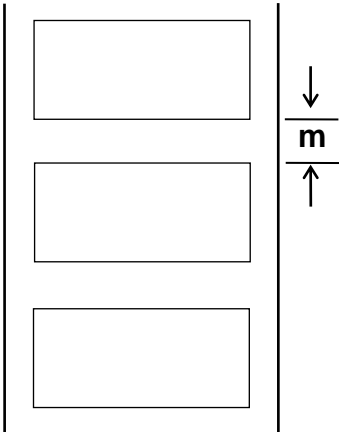
[See Also] GAP,BLINE

# GAP

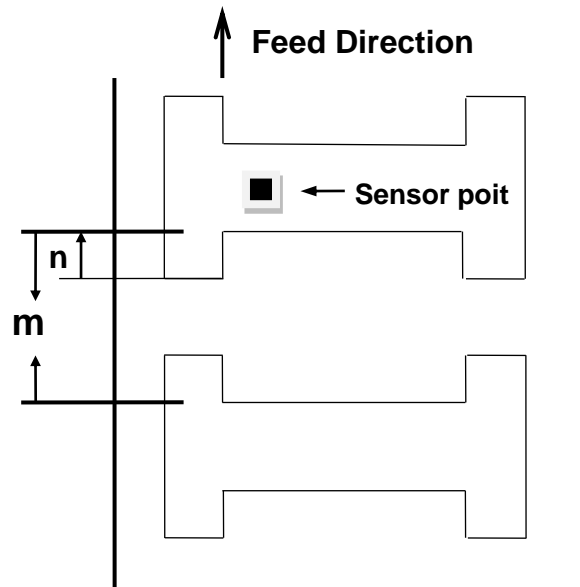
- [Description]

Define the gap distance between two labels.
- [Syntax]

(1)English system (inch)  
GAP m, n  
  
(2)Metric system (mm)  
GAP m mm, n mm  
  
(3)Dot measurement  
This command has been supported since V6.27 EZ and later firmware.  
GAP m dot,n dot



Parameter	Description
m	The gap distance between two labels $0 \leq m \leq 1(\text{inch}), 0 \leq m \leq 25.4 \text{ (mm)}$ $0 \leq m \leq 5 \text{ (inch)}, 0 \leq m \leq 127 \text{ (mm)}$ / <b>since V6.21 EZ and later firmware</b>
n	The offset distance of the gap $n \leq \text{label length (inch or mm)}$
0,0	Continuous label



**Notes: 200 DPI : 1 mm = 8 dots**

**300 DPI : 1mm = 12 dots**

***For metric system, there must be a space between parameter and mm.***

***When the sensor type is changed from “Black Mark” to “GAP”, please send the “GAP” command to the printer first.***

[Example]

#### **Normal gap**

(1).English system (inch)

GAP 0.12,0

(2).Metric system (mm)

GAP 3 mm,0

(3).Continuous label

GAP 0,0

#### **Special gap**

(1).English system (inch)

GAP 0.30,0.10

(2).Metric system (mm)

GAP 7.62 mm, 2.54 mm

[See Also]

SIZE,BLINE

## **BLINE**

[Description]

This command sets the height of the black line and the user-defined extra label feeding length each form feed takes.

[Syntax]

(1)English system (inch)

BLINE m,n

(2)Metric system (mm)

BLINE m mm,n mm

(3) Dot measurement

**This command has been supported since V6.27 EZ and later firmware.**

BLINE m dot,n dot

Parameter	Description
m	The height of black line either in inch or mm $0 \leq m \leq 1$ (inch), $0 \leq m \leq 25.4$ (mm) $0 \leq m \leq 5$ (inch), $0 \leq m \leq 127$ (mm) / <b>since V6.21 EZ and later firmware</b>
n	The extra label feeding length $0 \leq n \leq \text{label length}$
0,0	Continuous label

**Note:** For metric system, there must be a space between parameter and mm.

When the sensor type is changed from “GAP” to “Black Mark”, please send the “BLINE” command to the printer first.

200 DPI : 1 mm = 8 dots

300 DPI : 1mm = 12 dots.

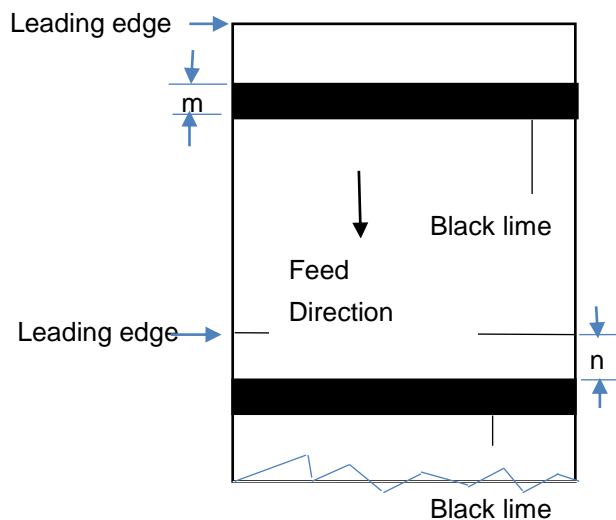
[Example] (1) English system (inch)

BLINE 0.20,0.50

(2)Metric system (mm)

BLINE 5.08 mm,12.7 mm

[See Also] SIZE, GAP



## OFFSET

[Description] This command defines the selective, extra label feeding length each form feed takes, which, especially in peel-off mode, is used to adjust label stop position, so as for label to register at proper places for the intended purposes. The printer back tracks the extra feeding length before the next run of printing.

[Syntax] (1)English system (inch)  
OFFSET m  
(2)Metric system (mm)  
OFFSET m mm  
(3)Dot measurement

**This command has been supported since V6.27 EZ and later firmware.**

OFFSET m dot,n dot

Parameter	Description
m	The offset distance (inch or mm) $-1 \leq m \leq 1(\text{inch})$

**CAUTION:** Improperly offset value may cause paper jam.

For metric system, there must be a space between parameter and mm.

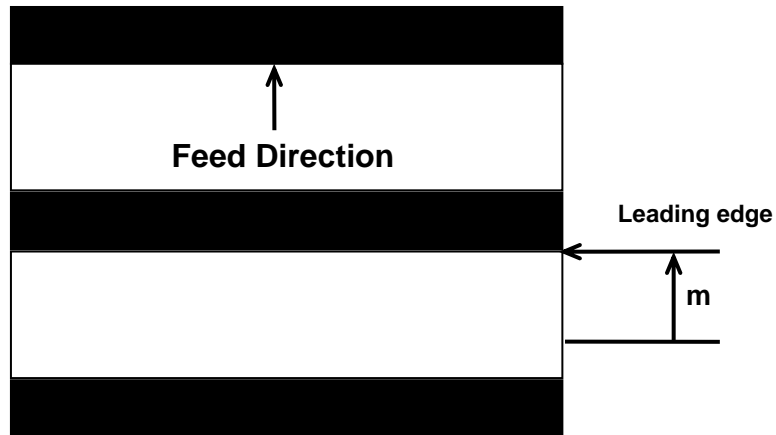
200 DPI : 1 mm = 8 dots

300 DPI : 1mm = 12 dots

[Example] (1) English system (inch)  
OFFSET 0.5  
(2)Metric system (mm)  
FFSET 12.7(mm)

[See Also] SIZE, GAP, SET PEEL, SET CUTTER





## SPEED

[Description] This command defines the print speed.

[Syntax] SPEED n

Parameter	Description
n	printing speed in inch per second

[Example] SPEED 10

[See Also] DENSITY

## DENSITY

[Description] This command sets the printing darkness.

[Syntax] DENSITY n

Parameter	Description
n	0~15 0, specifies the lightest level 15, specifies the darkest level

Note:

Default DENSITY setting is 8.

[Example] DENSITY7

# DIRECTION and Mirror Image

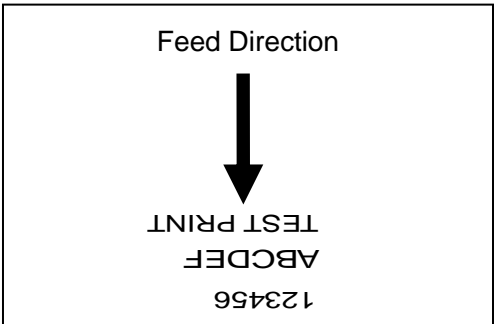
- [Description]

This command defines the printout direction and mirror image. This will be stored in the printer memory.
- [Syntax]

DIRECTION n[,m]

Parameter	Description
n	0 or 1. Please refer to the illustrations below:
m	0: Print normal image. 1: Print mirror image.

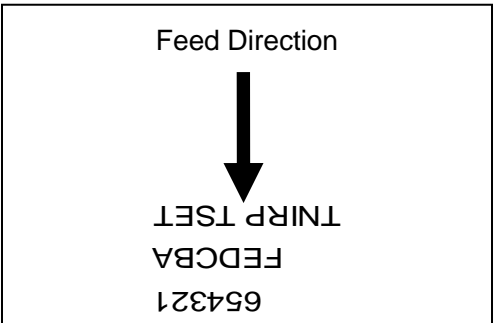
**DIRECTION 0, 0**



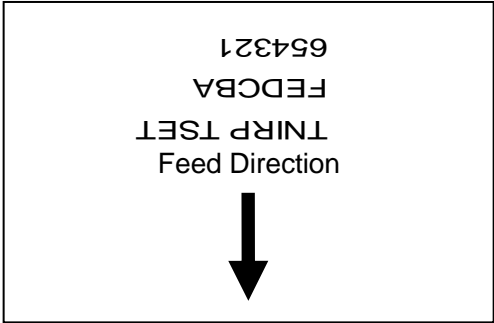
**DIRECTION 1, 0**



**DIRECTION 0, 1**



**DIRECTION 1, 1**



- [Example]

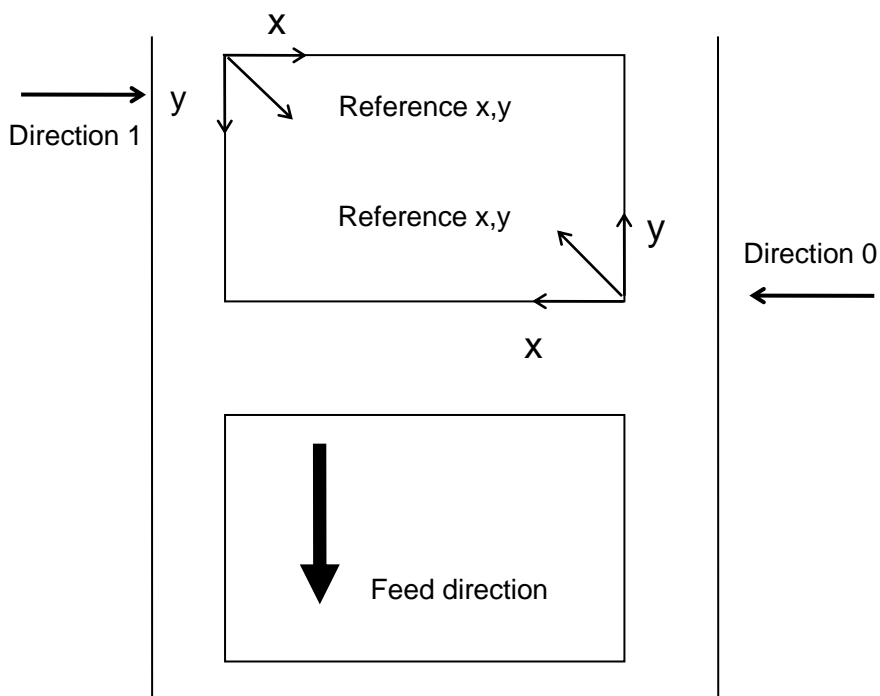
DIRECTION 0  
DIRECTION 0,1
- [See Also]

REFERENCE 0,1

## REFERENCE

- [Description]

This command defines the reference point of the label. The reference (origin) point varies with the print direction, as shown:



[Syntax]      REFERENCE x, y

Parameter	Description
x	Horizontal coordinate (in dots)
y	Vertical coordinate (in dots).

**Note:**    **200 DPI: 1 mm = 8 dots**  
               **300 DPI: 1 mm = 12 dots**

[Example]      REFERENCE 10,10

[See Also]     DIRECTION

## CODEPAGE

[Description]    This command defines the code page of international character set.

[Syntax]        CODEPAGE n

Parameter	Description
n	name or number of code page, which can be divided into 8-bit code page further.

**8-bit code page number**

437: United States
850: Multilingual
852: Slavic
860: Portuguese
863: Canadian/French
865: Nordic
857: Turkish

<b>Windows code page</b>
1250: Central Europe
1252: LatinI
1253: Greek
1254: Turkish

**Note: DATA LENGTH determines 7-bit or 8-bit communications parameter.**

[Example]        CODEPAGE 437  
[See Also]       COUNTRY, ~!l

## CLS

[Description]    This command clears the image buffer.  
[Syntax]         CLS

Parameter	Description
None	N/A

**Note: This command must be placed after SIZE command.**

[Example]        CLS  
[See Also]       SIZE, GAP, BLINE

## FEED

[Description]    This command feeds label with the specified length.  
The length is specified by dot.  
[Syntax]         FEED n

Parameter	Description
-----------	-------------

n	Unit: dot $1 \leq n \leq 9999$
---	-----------------------------------

[Example] FEED 40

**Note:** 200 DPI: 1 mm = 8 dots  
300 DPI: 1 mm = 12 dots

[See Also] BACKFEED, SIZE, GAP, HOME, FORMFEED

## BACKFEED

[Description] To back feed label with the specified length. The length is specified by dot.

[Syntax] BACKFEED n  
TSPL printers only  
BACKFEED n  
TSPL2 printers only

Parameter	Description
n	Unit: dot $1 \leq n \leq 9999$

[Example] TSPL printers  
BACKFEED 40  
TSPL2 printers  
BACKFEED 40

**CAUTION:** Improper back feed value may cause paper jam or wrinkle.

**Note:** 203 DPI: 1 mm = 8 dots  
300 DPI: 1 mm = 12 dots

[See Also] FEED, SIZE, GAP, HOME, FORMFEED

## FORMFEED

[Description] This command feeds label to the beginning of next label.

[Syntax] FORMFEED

Parameter	Description
None	N/A

[Example]        SIZE 4,2.5  
                 GAP 2 mm,0  
                 DIRECTION 1  
                 FORMFEED  
                 CLS  
                 TEXT 25,25,"3",0,1,1,"FORMFEED COMMAND TEST"  
                 PRINT 1,1

[See Also]        FEED, SIZE, GAP, HOME, BACKFEED

## HOME

[Description]    This command will feed label until the internal sensor has determined the origin. Size and gap of the label should be defined before using this command.

[Syntax]        HOME

Parameter	Description
None	N/A

[Example]        Sample code  
                 SIZE 4,2.5  
                 GAP 2 mm,0  
                 SET COUNTER @0+1  
                 @0="000001"  
                 HOME  
                 CLS  
                 BOX 1,1,360,65,12  
                 TEXT 25,25,"3",0,1,1,"HOME COMMAND TEST"  
                 TEXT 25,80,"3",0,1,1,@0  
                 PRINT 3,1

[See Also]        FEED, SIZE, GAP, FORMFEED

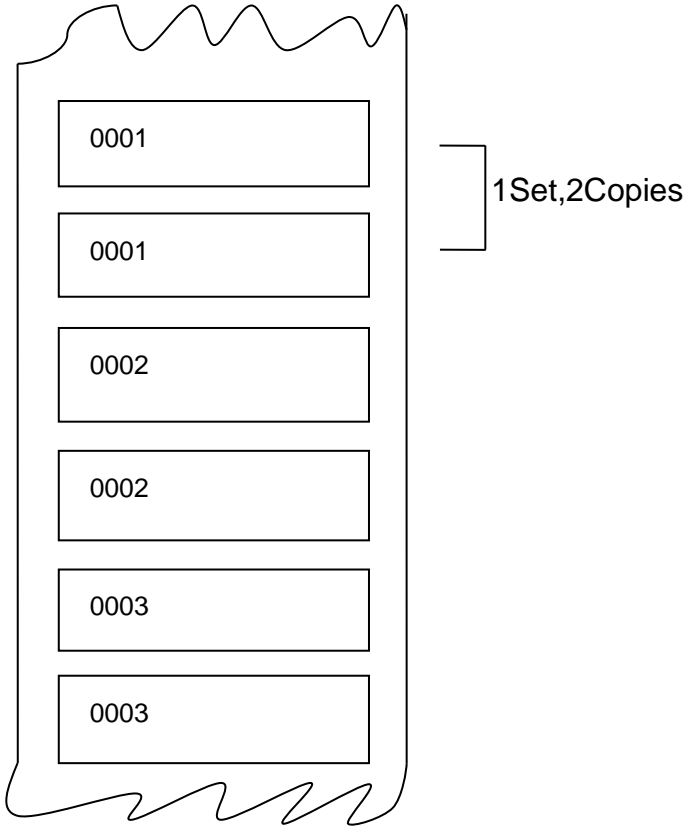
## PRINT

[Description]    This command prints the label format stored in the image buffer.

[Syntax]        PRINT m[,n]

Parameter	Description
-----------	-------------

m	Specifies how many sets of labels will be printed.
	$1 \leq m \leq 999999999$
n	Specifies how many copies should be printed for each particular label set.
	$1 \leq n \leq 999999999$



[Example]

SIZE 50 mm,25 mm  
 GAP 3 mm,0  
 DIRECTION 1  
 SET COUNTER @1 1  
 @1="0001"  
 CLS  
 TEXT 10,10,"3",0,1,1,@1  
 PRINT 3,2

[See Also]

SET COUNTER, DOWNLOAD

SOUND

[Description]

This command is used to control the sound frequency of the beeper. There are 10 levels of sounds. The timing control the sound can be set

by the “interval” parameter.

[Syntax] SOUND level, interval

Parameter	Description
level	Sound level: 0~9
interval	Sound interval: 1~4095

[Example] SOUND 5,200  
SOUND 3,200  
SOUND 3,200  
SOUND 4,200  
SOUND 2,200  
SOUND 2,200  
SOUND 1,200  
SOUND 2,200  
SOUND 3,200  
SOUND 4,200  
SOUND 5,200

## LIMITFEED

[Description] If the gap sensor is not set to a suitable sensitivity while feeding labels, the printer will not be able to locate the correct position of the gap. This command stops label feeding and makes the red LED flash if the printer does not locate gap after feeding the length of one label plus one preset value.

[Syntax] LIMITFEED n (inch, the English system)  
LIMITFEED n mm (mm, the metric system)  
LIMITFEED n dot(Dot measurement)

Parameter	Description
N	The maximum length for sensor detecting
Minpaper	The minimum length of paper
Maxgap	The maximum length of gap

[Note] The setting will remain resident in memory.  
For metric system, there must be a space between parameter n and mm.  
The default value is 10 inches when printer initializes.



Since V6.76 EZ, the default value for TDP-225 series printer is 14 inches when printer initializes.

The setting of parameters “mimpaper” and “maxgap” are using for calibrating the

preprinted label. This parameter has been supported since V6.98.7 EZ..

## SELFTEST

[Description] At this command, the printer will print out the printer information.

[Syntax] SELFTEST [page]

[Example] SELFTEST

# Label Formatting Commands

## BAR

[Description] This command draws a bar on the label format.

[Syntax] BAR x, y, width, height

Parameter	Description
x	The upper left corner x-coordinate in dot
y	The upper left corner y-coordinate in dot
width	The width of bar in dot
height	The height of bar in dot

**Note: 200 DPI: 1 mm = 8 dots**

**300 DPI: 1 mm = 12 dots**

**Recommended max. bar height is 12 mm at 4” width. Bar height over 12 mm may damage the power supply and affect the print quality.**

**Max. print ratio is different for each printer model. Desktop and industrial printer print ratio is limited to 20% and 30% respectively.**

[Example] SIZE 50 mm,25 mm

GAP 3 mm,0

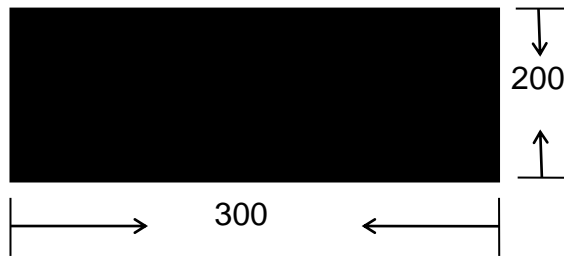
DIRECTION 1

CLS

BAR 80,80,300,100

PRINT 1,1

(100,100)



[See Also]      BOX

## BARCODE

[Description]      This command is used to print 1D barcodes on label form.

The available bar codes are listed below:

- Code 128
- Code 128M
- EAN 128
- Interleaved 2 of 5
- Interleaved 2 of 5 with check digit
- Code 39 standard
- Code 39full ASCII
- Code 39full ASCII with check digit
- Code 93
- EAN 13
- EAN 13 with 2 digits add-on
- EAN 13 with 5 digits add-on
- EAN 8
- EAN 8 with 2 digits add-on
- EAN 8 with 5 digits add-on
- Coda bar
- Post net
- UPC-A
- UPC-A with 2 digits add-on
- UPC-A with 5 digits add-on
- UPC-E
- UPC-E with 2 digits add-on
- UPC-E with 5 digits add-on
- MSI
- PLESSEY
- China POST
- ITF14

■ EAN14

[Syntax]      BARCODE X,Y,"code type",height,human  
readable,rotation,narrow,wide,[alignment,]"content ".

Parameter	Description
x	Specify the x-coordinate of the bar code on label
y	Specify the y-coordinate of the bar code on label
Codetype 128	Code 128, switching code subset A, B, C automatically.
128M	Code 128, switching code subset A, B, C manually

Controlcode	A	B	C
096	FNC3	FNC3	NONE
097	FNC2	FNC2	NONE
098	SHIFT	SHIFT	NONE
099	CODE C	CODE C	NONE
100	CODE B	FNC4	CODE B
101	FNC4	CODE A	CODE A
102	FNC1	FNC1	FNC1
103	Start (CODE A)		
104	Start (CODE B)		
105	Start (CODE C)		

Use “!” as a starting character for the control code followed by three control codes.  
If the start subset is not set, the default starting subset is B.

- EAN128Code 128, switching code subset A, B, C automatically
- 25Interleaved 2 of5
- 25C            Interleaved 2 of5 with check digits
- 39Auto switch full ASCII and standard code 39 for **PLUS** models.
- 39C            Code 39full ASCII with check digit

Code 39 standard with check digit

Auto switch full ASCII and standard code 39 for **PLUS** models.

- 39S            Code 39 standard
- 93            Code 93
- EAN 13EAN 13
- EAN 13+2EAN 13 with 2 digits add-on
- EAN 13+5EAN 13 with 5 digits add-on
- EAN 8EAN 8
- EAN 8+2EAN 8 with 2 digits add-on
- EAN 8+5EAN 8 with 5 digits add-on
- CODACodabar

- POST Post net
  - UPCAUPC-A
  - UPCA+2UPC-A with 2 digits add-on
  - UPCA+5UPC-A with 5 digits add-on
  - UPCEUPC-E
  - UPCE+2UPC-E with 2 digits add-on
  - UPCE+5 UPC-E with 5 digits add-on
  - CPOST China post code
  - MSIMSI code
  - MSIC
  - PLESSEYPLESSEY code
  - ITF 14ITF 14 code
  - EAN 14EAN 14 code
  - Height bar code height expressed by dot
  - human readable 0: human not readable
    - 1: human readable
    - 2: human readable aligns to center
    - 3: human readable aligns to right
  - rotation Rotate bar code clockwise in degrees
    - 0 non rotation
    - 90 rotate 90 degrees clockwise
    - 180 rotate 180 degrees clockwise
    - 270rotate 270 degrees clockwise
- Narrow
- width of narrow element in dot
- Wide
- width of wide element in dot

	narrow:wide 1:1	narrow: wide 1:2	narrow: wide 1:3	narrow: wide 2:5	narrow: wide 3:7
128	10x	N/A	N/A	N/A	N/A
EAN128	10x	N/A	N/A	N/A	N/A
25	N/A	10x	10x	5x	N/A
25C	N/A	10x	10x	5x	N/A
39	N/A	10x	10x	5x	N/A
39C	N/A	10x	10x	5x	N/A
93	N/A	N/A	10x	N/A	N/A
EAN13	8x	N/A	N/A	N/A	N/A
EAN13+2	8x	N/A	N/A	N/A	N/A
EAN13+5	8x	N/A	N/A	N/A	N/A
EAN8	8x	N/A	N/A	N/A	N/A
EAN8+2	8x	N/A	N/A	N/A	N/A
EAN8+5	8x	N/A	N/A	N/A	N/A
CODA	N/A	10x	10x	5x	N/A

POST	1x	N/A	N/A	N/A	N/A
UPCA	8x	N/A	N/A	N/A	N/A
UPCA+2	8x	N/A	N/A	N/A	N/A
UPCA+5	8x	N/A	N/A	N/A	N/A
UPCE	8x	N/A	N/A	N/A	N/A
UPCE+2	8x	N/A	N/A	N/A	N/A
UPCE+5	8x	N/A	N/A	N/A	N/A
CPOST	N/A	N/A	N/A	N/A	1x
MSI	N/A	N/A	10x	N/A	N/A
MSIC	N/A	N/A	10x	N/A	N/A
PLESSY	N/A	N/A	10x	N/A	N/A
ITF14	N/A	10x	10x	5x	N/A
EAN14	N/A	N/A	N/A	5x	N/A

code number the maximum number of digits of bar code content

Barcode type	Maximum bar Code length	Barcode type	Maximum bar Code length
128	—	POST	5,9,11
EAN128	—	UPCA	11
25	—	UPCA+2	13
25C	—	UPCA+5	16
39	—	UPCE	6
39C	—	UPCE+2	8
93	—	UPCE+5	11
EAN13	12	CPOST	—
EAN13+2	14	MSI	—
EAN13+5	17	MSIC	—
EAN8	7	PLESSY	—
EAN8+2	9	ITF14	13
EAN8+5	12	EAN14	13
CODA	—		

[Example]      BARCODE 100,100,"39",96,1,0,2,4,"1000"  
 BARCODE 10,10,"128M",48,1,0,2,2,"!104!096ABCD!101EFGH"  
 (The above example of code 128M encoded with CODE B start character. The next character will be the code 128function character FNC3 which is then followed by the ABCD characters and EFGH characters encoded as CODE A subset.

# BITMAP

[Description] This command is used to draw bitmap images (Not BMP graphic file).

[Syntax] BITMAP X, Y, width, height, mode, bitmap data...

		→ X Size 16 dot								1 dot							
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	3	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
	4	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
	5	0	0	0	1	0	0	0	1	1	1	1	1	1	1	1	1
	6	0	0	0	1	1	0	0	0	1	1	1	1	1	1	1	1
	7	0	0	0	1	1	1	0	0	0	1	1	1	1	1	1	1
	8	0	0	0	1	1	1	1	0	0	0	1	1	1	1	1	1
	9	0	0	0	1	1	1	1	1	0	0	0	1	1	1	1	1
	A	0	0	0	1	1	1	1	1	1	0	0	0	1	1	1	1
	B	0	0	0	1	1	1	1	1	1	1	0	0	0	1	1	1
	C	0	0	0	1	1	1	1	1	1	1	1	0	0	0	1	1
	D	0	0	0	1	1	1	1	1	1	1	1	1	0	1	1	1
	E	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
	F	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
		1 byte								1 byte							

Parameter	Description
x	Specify the x-coordinate of the bitmap image
y	Specify the y-coordinate of the bitmap image
width	The width of the image in bytes
height	The height of the image in dot
mode	Graphic mode is listed below:
0	OVERWRITE
1	OR
2	XOR
bitmap data	The bitmap data

ROW (Y- axis)	L-Byte		R-Byte	
	Binary	Hexadecimal	Binary	Hexadecimal
0	00000000	00	00000000	00
1	00000000	00	00000000	00
2	00000000	00	00000000	00
3	00000111	07	11111111	FF
4	00000011	03	11111111	FF
5	00010001	11	11111111	FF
6	00011000	18	11111111	FF
7	00011100	1C	01111111	7F
8	00011110	1E	00111111	3F
9	00011111	1F	00011111	1F
A	00011111	1F	10001111	8F
B	00011111	1F	11000111	C7
C	00011111	1F	11100011	E3
D	00011111	1F	11110111	F7
E	00011111	1F	11111111	FF
F	00011111	1F	11111111	FF

[Example]      SIZE 4,2  
                   GAP 0,0  
                   CLS  
                   BITMAP 200,200,2,16,0,  
                   PRINT 1,1  
 [See Also]     PUTBMP, PUTPCX

## BOX

[Description]    This command is used to draw rectangles on the label.  
 [Syntax]        BOX x,y,x\_end,y\_end,line thickness[,radius]

Parameter	Description
X_start	Specify x-coordinate of upper left corner in dot
Y_start	Specify y-coordinate of upper left corner in dot
X_end	Specify x-coordinate of lower right corner in dot
Y_end	Specify y-coordinate of lower right corner in dot
line thickness	Line thickness of the box in dot
radius	Optional. Specify the round corner. Default is 0. <b>*Since V5.28 EZ</b>

**Note: 200 DPI: 1 mm = 8 dots**

**300 DPI: 1 mm = 12 dots**

**Recommended max. thickness of box is 12 mm at 4" width. Thickness of box larger than 12 mm may damage the power supply and affect the print quality. Max. print ratio is different for each printer model. Desktop and industrial printer print ratio is limited to 20% and 30% respectively.**

[Example]      SIZE 4,1.1  
                  CLS  
                  BOX 60,60,610,210,4  
                  BOX 80,80,590,190,4  
                  BOX 100,100,570,170,4,20  
                  BOX 120,120,550,150,4,20  
                  PRINT 1

(100,100)



(200,200)

[See Also]      BAR

## ERASE

[Description]    This command is used to clear a specified region in image buffer.

[Syntax]        ERASE X\_start, Y\_start, X\_width, Y\_height

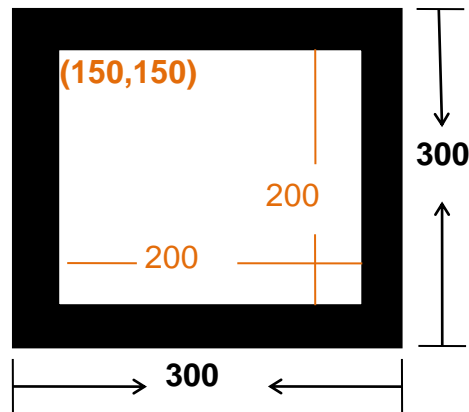
Parameter	Description
X_start	The x-coordinate of the starting point in dot
Y_start	The y-coordinate of the starting point in dot
X_width	The region width in x-axis direction in dot
Y_height	The region height in y-axis direction in dot

[Example]      SIZE 4,2.5  
                  GAP 0,0  
                  DIRECTION 1  
                  CLS  
                  BAR 100,100,300,300  
                  ERASE 150,150,200,200  
                  PRINT 1,1



[See Also]      CLS

(100,100)



## PUTBMP

[Description]      This command prints BMP format images. The grayscale printing is for direct thermal mode only. Support 1-bit (monochrome) and 8-bit (256-color) BMP graphic only.

[Syntax]            PUTBMP X, Y, "filename "[, bpp][, contrast]

Parameter	Description
X	The x-coordinate of the BMP format image
Y	The y-coordinate of the BMP format image
filename	The downloaded BMP filename.
bpp	Optional. Bits per pixel of grayscale graphic. Default is 1. *Since V6.91EZ. 1: 1-bit (monochrome) graphic 8: 8-bit (256-color) graphic
contrast	Optional. Contrast of grayscale graphic. Default is 80. Suggested range is from 60 to 100. *Since V6.91EZ.

[Example]

```

C:\BMP-PCX>DIR
Volume in drive C is WIN98
Volume Serial Number is 4140-4735

Directory of C:\BMP-PCX

06/08/2008    03:06 PM    <DIR>
06/08/2008    03:06 PM    <DIR>
06/08/2008    03:56 PM                12,430 GP. bmp
06/08/2008    03:10 PM                1,181 GP. pcx
                2 File(s)                13,611 bytes
                2 Dir(s)  8,802,189,312 bytes free

C:\BMP-PCX>COPY CON LPT1
DOWNLOAD "GP.BMP",12430,^Z
                1 file(s) copied.

C:\BMP-PCX>COPY GP.BMP/B LPT1
                1 file(s) copied.

C:\BMP-PCX>COPY CON LPT1
SIZE 3,2.5
GAP 0,0
CLS
PUTBMP 100,100,"GP.BMP"
PRINT 1,1
^Z
                1 file(s) copied.
C:\BMP-PCX>_

```

[See Also]      DOWNLOAD, BITMAP, PUTPCX

## PUTPCX

[Description]    This command prints PCX format images. TSPL language supports 2-color PCX format graphics. TSPL2 language supports 256-color PCX format graphics.

[Syntax]          PUTPCX X, Y, "filename"

Parameter	Description
X	The x-coordinate of the PCX image
Y	The y-coordinate of the PCX image
filename	The downloaded PCX filename. Case sensitive

[Example]

```

C:\BMP-PCX>DIR
Volume in drive C is WIN98
Volume Serial Number is 4140-4735

Directory of C:\BMP-PCX

06/08/2008    03:06 PM    <DIR>
06/08/2008    03:06 PM    <DIR>
06/08/2008    03:56 PM                12,430 GP. bmp
06/08/2008    03:10 PM                1,181 GP. pcx
                        2 File(s)            13,611 bytes
                        2 Dir(s)  8,802,189,312 bytes free

C:\BMP-PCX>COPY CON LPT1
DOWNLOAD "GP. PCX",12430,^Z
                1 file(s) copied.

C:\BMP-PCX>COPY GP. PCX/B LPT1
                1 file(s) copied.

C:\BMP-PCX>COPY CON LPT1
SIZE 3,2.5
GAP 0,0
CLS
PUTBMP 100,100,"GP. PCX"
PRINT 1,1
^Z
                1 file(s) copied.
C:\BMP-PCX>_

```

[See Also]      DOWNLOAD, BITMAP, PUTPCX

## QRCODE

[Description]      This command prints QR code.  
 [Syntax]            QRCODE x,y,ECC Level,cell  
                      width,mode,rotation,[model,mask,]"content"

Parameter	Description
X	The upper left corner x-coordinate of the QR code
Y	The upper left corner y-coordinate of the QR code
ECC level	Error correction recovery level L : 7% M : 15%

	Q : 25% H : 30%
cell width	1~10
mode	Auto / manual encode A : Auto M : Manual
rotation	0 : 0 degree 90 : 90 degree 180 : 180 degree 270 : 270 degree
model	M1: (default), original version M2: enhanced version (Almost smart phone is supported by this version.)
mask	S0~S8, default is S7
content	<p>The encodable character set is described as below,</p> <p>Encodable character set:</p> <p>1) Numeric data: (digits 0~9)</p> <p>2) Alphanumeric data</p> <p>Digits 0-9</p> <p>Upper case letters A-Z</p> <p>Nine other characters: space, \$ % * + - . / : )</p> <p>3) 8-bit byte data</p> <p>JIS 8-bit character set (Latin and Kana) in accordance with JIS X 0201</p> <p>4) Kanji characters</p> <p>Shift JIS values 8140 HEX ~9FFC HEX and E040 HEX ~EAA4 HEX . These are values shifted from those of JIS X 0208. Refer to JIS X 0208 Annex 1</p> <p>Shift Coded Representation for detail.</p> <p>Data characters per symbol (for maximum symbol size):</p> <p>Model 1 (Version 14-L)    Model 2 (Version 40-L)</p> <p>Numeric data    1,167 characters    7,089 characters</p> <p>Alphanumeric data    707 characters    4,296 characters</p> <p>8-bit byte data    486 characters    2,953 characters</p> <p>Kanji data    299 characters    1,817 characters</p> <p>* If "A" is the first character in the data string, then the following data after "A" is alphanumeric data.</p>

	<p>*If "N" is the first character in the data string, then the following data after "N" is numeric data.</p> <p>*If "B" is the first character in the data string, then the following 4 digits after "B" is used to specify numbers of data. After the 4 digits is the number of bytes of binary data to be encoded.</p> <p>*If "K" is the first character in the data string, then the following data after "K" is Kanji data.</p> <p>*If "!" is in the data string and follows by "N", "A", "B", "K" then it will be switched to specified encodable character set.</p> <p>Manual mode example:</p> <p>QRCODE 100,10,L,7,M,0,M1,S1,"ATHE FIRMWARE HAS BEEN UPDATED"</p> <p>(Where A: Alphanumeric data)</p> <p>QRCODE 100,10,M,7,M,0,M1,S2,"N123456"</p> <p>(Where N: Numeric data)</p> <p>QRCODE 100,10,Q,7,M,0,M1,S3,"N123456!ATHE FIRMWARE HAS BEEN UPDATED"</p> <p>(Where N: Numeric data ; !:Transfer char ; A: Alphanumeric data)</p> <p>QRCODE 100,10,H,7,M,0,M1,S3,"B0012Product name"</p> <p>(where B: Binary data ; 0012: 12 bytes )</p> <p>QRCODE 100,10,M,7,M,0,M1,S3,"K"</p> <p>(Where K: Kanji data)</p> <p>Auto mode example:</p> <p>QRCODE 100,10,M,7,A,0,"THE FIRMWARE HAS BEEN UPDATED"</p>
--	---

[Example]

SIZE 4,2.5

GAP 0,0

DIRECTION 1

CLS

QRCODE 10,10,H,4,A,0,"ABCabc123"

QRCODE 160,160,H,4,A,0,"123ABCabc"

QRCODE 310,310,H,4,A,0," 印表機 ABCabc123"

PRINT 1,1

## REVERSE

[Description] This command is used to reverse a region in image buffer.

[Syntax] REVERSE X\_start, Y\_start, X\_width, Y\_height

Parameter	Description
X_start	The x-coordinate of the starting point in dot
Y_start	The y-coordinate of the starting point in dot
X_width	The region width in x-axis direction in dot
Y_height	The region height in y-axis direction in dot

**Note:** 200 DPI: 1 mm = 8 dots

300 DPI: 1 mm = 12 dots

**Recommended max. height of reversed black area is 12mm at 4" width.  
Height of reversed area that is larger than 12 mm may damage the power  
supply and affect the print quality.**

**Max. print ratio is different for each printer model. Desktop and industrial  
printer print ratio is limited to 20% and 30% respectively.**

[Example] SIZE 4,2.5  
GAP 0,0  
DIRECTION 1  
CLS  
TEXT 100,100,"3",0,1,1,"REVERSE"  
REVERSE 90,90,128,40  
PRINT 1,1



## TEXT

[Description] This command is used to print text on label.

[Syntax] TEXT X, Y, "font", rotation, x-multiplication, y-multiplication, "content"

Parameter	Description
X	The x-coordinate of the text
Y	The y-coordinate of the text
Font	Font name

0	Monotype CG Triumvirate Bold Condensed, font width and height is stretchable.
1	8 x 12fixed pitch dot font
2	12 x 20fixed pitch dot font
3	16 x 24 fixed pitch dot font
4	24 x 32fixed pitch dot font
5	32 x 48 dot fixed pitch font
6	14 x 19 dot fixed pitch font OCR-B
7	21 x 27 dot fixed pitch font OCR-B
8	14 x25 dot fixed pitch font OCR-A
TST24.BF2	Traditional Chinese 24 x 24font
TSS24.BF2	Simplified Chinese 24 x 24font(GB)
K	Korean 24 x 24font (KS)
Rotation	The rotation angleof text
0	0 degree
90	90 degrees, in clockwise direction
180	180 degrees, in clockwise direction
270	270 degrees, in clockwise direction
X-multiplication:	Horizontal multiplication, upto10x. Available factors:1~10width(point)of true type font. 1 point=1/72 inch.
Y-multiplication:	Vertical multiplication, up to 10x. Available factors: 1~10 For true type font, this parameter is used to specify the height (point) of true type font. 1 point=1/72 inch.

**Note: The internal font (font #1~#5) pitch between TSPL and TSPL2 is different.**

**Font "0" and "ROMAN.TTF" internal True Type Fonts are available in TSPL2 language**

**printers, but not TSPL language printers.**

**If there is any double quote (") within the text, please change it to \".**

**If font "0" is used, the font width and font height is stretchable by x-multiplication and**

**y-multiplication parameter. It is expressed by pt (point). 1 point=1/72inch.**

**EPL2 and ZPL2 are emulating for Eltron ® and Zebra ® languages.**

[Example]      SIZE 3,2  
                  GAP 0,0  
                  CLS  
                  TEXT 100,100,"5",0,1,1,"["DEMO FOR TEXT\[""  
                  TEXT 100,200,"0,1,20,\"[\"True Type Font Test Print\[""  
                  PRINT 1,1

# Status Polling Commands(RS-232)

## <ESC>!?

[Description] This command obtains the printer status at any time, even in the event of printer error. An inquiry request is solicited by sending an <ESC> (ASCII 27, escape character) as the beginning control character to the printer. A one byte character is returned, flagging the printer status. A 0 signifies the printer is ready to print labels.

Bit	Status
0	Head opened
1	Paper jam
2	Out of paper
3	Out of ribbon
4	Pause
5	Printing
6	Cover opened (option)
7	Environment Temperature over range (option)

Hex Receive	Printer Status
00	Normal
01	Head opened
02	Paper Jam
03	Paper Jam and head opened
04	Out of paper
05	Out of paper and head opened
08	Out of ribbon
09	Out of ribbon and head opened
0A	Out of ribbon and paper jam
0B	Out of ribbon, paper jam and head opened
0C	Out of ribbon and out of paper
0D	Out of ribbon, out of paper and head opened
10	Pause
20	Printing

[Syntax] <ESC>! ?

[See Also] <ESC>! S



## <ESC>!R

[Description] This command resets the printer. The beginning of the command is an ESCAPE character (ASCII 27).The files downloaded in memory will be deleted. This command cannot be sent in dump mode.

[Syntax] <ESC>! R

Parameter	Description
N/A	N/A

[See Also] <ESC>! ?

## ~!@

[Description] This command is used to inquire the mileage of the printer. The integer part of mileage is returned (the decimal part of mileage is not return). It is returned to PC in ASCII characters. The ending character of mileage is 0x0D.

[Syntax] ~! @

Parameter	Description
N/A	N/A

[Example] ~! @

## ~!A

[Description] This command is used to inquire about the free memory of the printer. The number of bytes of free memory is returned in decimal digits, with 0x0d as ending code of PC.

[Syntax] ~!A

Parameter	Description
None	N/A

[Example] ~!A

[See Also] FILES

## ~!C

[Description] This command inquires the presence of Real Time Clock. One byte is return from the printer, indicating whether or not the RTC is installed. This command is only for the firmware before V6.xx.

[Syntax] ~!C

Parameter	Description
None	N/A

Return value	Description
0	RTC is not installed.
1	RTC is installed.

[Example] ~!C

[See Also] FILES

## ~!D

[Description] This command enters the printer into DUMP mode. In DUMP mode, the printer outputs code directly without interpretation.

[Syntax] ~!D

Parameter	Description
None	N/A

[Example] ~!D

## ~!F

[Description] This command inquires all about files resident in the printer memory, and fonts installed in the memory module. The filename are returned in ASCII characters. Each file name ends with 0x0D. The ending character is 0x1A. Entering this command multiple times will cycle through the files resident on memory.

[Syntax] ~!F

Parameter	Description
-----------	-------------

None	N/A
------	-----

[Example] ~!F  
[See Also] FILES

## ~!I

[Description] The command inquires the code page and country setting of the printer.  
[Syntax] ~!I

Parameter	Description
None	N/A

[Example] ~!I  
[See Also] COUNTRY, CODEPAGE

## ~!T

[Description] This command is used to inquire the model name and number of the printer. They are returned in ASCII characters.  
[Syntax] ~!T

Parameter	Description
None	N/A

[Example] ~!T

# FileManagement Commands

## DOWNLOAD

[Description] "DOWNLOAD" is a header of the file that is to be saved in the printer's memory. The downloaded files can be divided into two categories: program files and data files (including text data files, PCX graphic files and bitmap font files) The detailed descriptions regarding the download syntax for different files are as follows:

Maximum numbers of file saved in DRAM:

50 files for TSPL/TSPL2 language printers

Maximum numbers of file saved in Flash memory:

50 files for TSPL language printers

256 files for TSPL2 language desktop printers (TTP/TDP-245 / TTP-343 / TTP-244)

256 files for TSPL2 language industrial printers (TTP-246M/344M)

If "AUTO.BAS" exists in the printer memory, it will be automatically executed upon printer startup. To disable the auto execution function, please follow the procedures below.

For TTP-245/TTP-343/TDP-245 series, 245C/343C/244CE series

Hold the FEED key and power on the switch. The LED color will be changed as following pattern.

Orange      red (5 blinks)      orange (5 blinks)      green (5 blinks)  
solid green (for firmware version before V3.37)

Orange      red (5 blinks)      orange (5 blinks)      green (5 blinks)  
green and orange (5 blinks)      red and orange (5 blinks)      solid  
green (V3.37)

Release the FEED key while LED becomes solid green to prevent the printer from running "AUTO.BAS".

For TTP-246M/TTP-344M series

Hold the FEED key and power on the switch. The ERROR LED will be on. Printer is now ready to use.

For TTP-2410M/TTP-246M PLUS series

Hold the PAUSE and FEED keys and power on the switch. "AUTO.BAS" will not be executed after printer initialization, and will now be ready for use.

Alternatively, hold the PAUSE key and power on the switch. After sensor calibration, the "AUTO.BAS" will not be executed. Printer is now ready for use.

[Syntax]

1. Download a program file  
DOWNLOAD[n,]"FILENAME.BAS"

Parameter	Description
n	Specify the memory which is used to save the download files.

n is ignored	Download files to DRAM only. If you would like to save the files from DRAM to Flash memory before turning off power, you can issue MOVE command to printer to move the files from DRAM to Flash memory. F: Download files to main board flash memory.
FILENAME.BAS	The filename resident in printer memory.

**Note:**

- (1).The filename is case sensitive.**
- (2).The extension of the program file must be “.BAS”**
- (3). Filename format must be in 8.3 format.**
- (4). It should use with EOP command.**
- (5). If memory is not specified, all files will be downloaded to DRAM.**
- (6). The priority of AUTO.BAS in each memory device:**
  - A. DRAM > FLASH > CARD (Ext. FLASH) if firmware is before V6.80EZ**
  - B. DRAM > CARD (Ext. FLASH) > FLASH if firmware is after V6.80EZ (include).**
- (7). No Battery is used to back up files in DRAM. which will be lost in the event printer power is lost.**

2. Download a data file

DOWNLOAD[n,]“FILENAME”, DATA SIZE, DATA CONTENT...where

Parameter	Description
n	Specify the memory location to save the download files.
n is ignored	Download files to DRAM only. If you would like to save the files from DRAM to Flash memory before turning off power, you can issue MOVE command to printer to move the files from DRAM to Flash memory. F: Download files to main board flash memory.
FILENAME.BAS	The name of data file that will remain resident in the printer memory. It is case sensitive.
DATA SIZE	The actual size (numbers of byte) of the data file without header.

**Note:**

- (1).For text data file, CR (carriage return) 0x0D and LF (Line Feed) 0x0A is the separator of data.**
- (2).If memory is not specified, all files will be downloaded to DRAM.**  
**No Battery is used to backup DRAM.**
- (3).No Battery is used to back up files in DRAM. which will be lost in the**

*event printer power is lost.*

*(5).When writing a download program, "DOWNLOAD" header must be placed in the beginning of file, and "EOP" must be placed at the end of program.*

*(6).To run the program, call the main filename without BAS extension or use RUN command to start the download program.*

[Example]        The example program listed below will download to printer SDRAM.

```
DOWNLOAD "EXAMPLE.BAS"
SIZE 4,4
GAP 0,0
DIRECTION 1
SET TEAR ON
CLS
TEXT 100,100,"3",0,1,1,"EXAMPLE PROGRAM"
PRINT 1
EOP
```

[See Also]        EOP, RUN, PUTBMP, PUTPCX, INPUT, FILES, ~!F

## EOP

[Description]    End of program. To declare the start and end of BASIC language commands used in a program, the DOWNLOAD "FILENAME.BAS" must be added in the first line of the program, and "EOP" statement at the last line of program.

[Syntax]         EOP

[Example]        DOWNLOAD "DEMO.BAS"  
SIZE 4,4  
GAP 0,0  
DIRECTION 1  
SET TEAR ON  
CLS  
TEXT 100,100,"3",0,1,1,"DEMO PROGRAM"  
PRINT 1  
EOP

[See Also]        DOWNLOAD, INPUT, FILES, ~!F

## FILES

- [Description] This command prints out the total memory size, available memory size and files lists (or lists the files through RS-232) in the printer memory (both FLASH memory and DRAM).
- [Syntax] FILES
- [See Also] ~!F, KILL

## KILL

- [Description] This command deletes a file in the printer memory. The wild card (\*) will delete all files resident in specified DRAM or FLASH memory.
- [Syntax] KILL[n], "FILENAME"

Parameter	Description
n	Specify the memory location that files will be deleted.
n is ignored	Kill files saved in DRAM. F: Kill files from main board flash memory. E: Kill files from expansion memory module.
FILENAME	The name of data file that will delete in the printer memory (case sensitive)

### Note:

- (1). If optional parameter n is not specified, firmware will delete the file in DRAM.**

### Syntax example

1. KILL "FILENAME"
2. KILL "\*.PCX"
3. KILL "\*"
4. KILL F, "FILENAME"
5. KILL E, "\*.PCX"

- [Example] Users can use printer SELFTEST utility to list printer configurations and files saved in the printer memory, or use the FILES command to print the downloaded file list in printer. Follow the steps below to delete files in the printer memory via parallel port connection.

```
C:\>COPY CON LPT1<ENTER>
      FILES<ENTER>
      <CTRL><Z><ENTER>
C:\>COPY CON LPT1<ENTER>
      KILL "DEMO.BAS" <ENTER>
```

```
<CTRL><Z><ENTER>
C:\>COPY CON LPT1<ENTER>
FILES<ENTER>
<CTRL><Z><ENTER>
```

**Note:**

**<ENTER> stands for PC keyboard “ENTER” key.**

**<CTRL><Z> means to hold PC keyboard “CTRL” key then press the PC keyboard <Z> key**

[See Also] ~!F, FILES

## MOVE

[Description] This command is used to move downloaded files from DRAM to the FLASH memory.

[Syntax] MOVE

Parameter	Description
N/A	N/A

[See Also] DOWNLOAD, EOP

## RUN

[Description] This command executes a program resident in the printer memory. It is available for TSPL2 language printers only.

[Syntax] RUN “FILENAME.BAS”

[Example] C:\>COPY CON LPT1<ENTER>  
RUN “DEMO.BAS”<ENTER>  
<CTRL><Z><ENTER>  
C:\>

**Note: This command can be replaced to filename that without typing “.BAS”.**

[See Also] DOWNLOAD, EOP



# BASIC Commands and Functions

## ABS( )

[Description]	This function returns the absolute value of an integer, floating point or variable.	
[Syntax]	ABS (VARIABLE)	
[Example]	DOWNLOAD "TEST.BAS" SIZE 4,4 GAP 0,0 DIRECTION 1 SET TEAR ON CLS A=ABS(-100) B=ABS(-50.98) C=-99.99 TEXT 100,100,"3",0,1,1,STR\$(A) TEXT 100,150,"3",0,1,1,STR\$(B) TEXT 100,200,"3",0,1,1,STR\$(ABS(C)) PRINT 1 EOP RUN "TEST.BAS" [See Also]      DOWNLOAD, EOP	
[See Also]	DOWNLOAD, EOP	

## ASC( )

[Description]	This function returns the ASCII code of the character.	
[Syntax]	ASC("A")	
[Example]	DOWNLOAD "TEST.BAS" SIZE 4,4 GAP 0,0 DIRECTION 1 SET TEAR ON CLS CODE1=ASC("A") TEXT 100,100,"3",0,1,1,STR\$(CODE1) PRINT 1 EOP RUN "TEST.BAS"	
[See Also]	DOWNLOAD, EOP, STR\$()	

## CHR\$( )

[Description] This function returns the character that has the specified ASCII code.

[Syntax] CHR\$(n)

Parameter	Description
n	The ASCII code

[Example] DOWNLOAD "TEST.BAS"  
SIZE 4,4  
GAP 0,0  
DIRECTION 1  
SET TEAR ON  
CLS  
A=75  
WORD\$=CHR\$(A)  
TEXT 100,100,"3",0,1,1,WORD\$  
PRINT 1  
EOP  
RUN "TEST.BAS"

[See Also] DOWNLOAD, EOP, STR\$(), ASC\$()

## END

[Description] This command states the ending of program.

[Syntax] END

[Example] DOWNLOAD "DEMO.BAS"  
SIZE 4,2  
GAP 0,0  
DIRECTION 1  
CLS  
TEXT 200,60,"4",0,1,1,"END COMMAND TEST"  
X=300  
Y=200  
X1=500  
Y1=400  
GOSUB DR\_LINE  
PRINT 1  
END

```

:DR_LINE
FOR I=1 TO 100 STEP 10
BOX X+I,Y+I,X1-I,Y1-I,5
NEXT
RETURN
EOP
DEMO

```

[See Also]      DOWNLOAD, EOP, GOSUB

## EOF()

[Description]      This function is used to detect an opened download file to see whether it has reached the end of file.

[Syntax]            EOF (File Handle)

Parameter	Description
File handle	Either 0 or 1

Parameter	Description
None-zero	End of file
0	Not end of file

[Example]            DOWNLOAD "DATA",16,COMPUTER  
2000

```

DOWNLOAD "DEMO.BAS"
SIZE 3,3
GAP 0.0,0
DIRECTION 1
CLS
OPEN "DATA",0
SEEK 0,0
Y=110
TEXT 10,10,"3",0,1,1,"*****EOF TEST*****"
:A
Temp$=""
READ 0,ITEM$,P
TEXT
10,Y,"2",0,1,1,ITEM$+"$"+STR$(P)+"[EOF(0)="+STR$(EOF(0))+"]"
BARCODE 10,Y+25,"39",40,1,0,2,4,"PRICE-"+STR$(P)
Y=Y+100

```

```

IF EOF(0)=0 THEN GOTO A
PRINT 1
EOP
DEMO
[See Also] DOWNLOAD, EOP, OPEN, READ, SEEK

```

## OPEN

[Description] This command opens a downloaded file and establishes the file handle. Up to two file handles are supported, thus only up to two files can be opened simultaneously. The file to be opened should be downloaded prior to using this command. When opening a file, the firmware will search automatically to see if the file exists in the on board flash memory or extended memory card. \*Since V6.37 EZ, if the file doesn't exist, the printer will create this file in the onboard FLASH.

[Syntax] OPEN "Filename", File handle

Parameter	Description
Filename	The file downloaded in the printer memory.
File handle	Either 0 or 1.

[Example]

```

DOWNLOAD "DATA.DAT",18,Open file in DRAM.
DOWNLOAD F,"DATA.DAT",19,Open file in FLASH.
DOWNLOAD "TEST.BAS"
data1$=""
data2$=""
data3$=""
OPEN "DATA.DAT",0
READ 0,data1$
CLOSE 0
OPEN F,"DATA.DAT",0
READ 0,data2$
CLOSE 0
KILL F,"*"
OPEN "NEW.DAT",0
SEEK 0,0
WRITE 0,"Auto create a new file in FLASH."
SEEK 0,0
READ 0,data3$
CLOSE 0
SIZE 4,1
GAP 0,0
CLS

```

```

TEXT 10,10,"3",0,1,1,data1$
TEXT 10,60,"3",0,1,1,data2$
TEXT 10,110,"3",0,1,1,data3$
PRINT 1
EOP
TEST

```

[See Also]     DOWNLOAD, EOP, READ, WRITE, SEEK, CLOSE

## READ

[Description]     This command is used to read data from downloaded data file.

[Syntax]           READ file handle, variables

Parameter	Description
File handle	0 or 1.
variables	string, integer or float point variable

[Example]

```

DOWNLOAD "DATA1",20,COMPUTER
2000
12
DOWNLOAD "DATA2",16,Mouse
900
93
DOWNLOAD "DEMO.BAS"
SIZE 3,1
GAP 0,0
DIRECTION 1
I=0
Y=100
OPEN "DATA1",0
OPEN "DATA2",1
SEEK 0,0
SEEK 1,0
:Start
CLS
TEXT 10,10,"3",0,1,1,"*****READ COMMAND TEST*****"
TEXT 10,50,"3",0,1,1,"OPEN-READ DATA"+STR$(I+1)
ITEM$=""
READ I,ITEM$,P,Q
TEXT 10,Y,"2",0,1,1,ITEM$+"$"+STR$(P)
BARCODE
10,Y+25,"39",40,1,0,2,4,"PRICE*"+STR$(Q)+"="+STR$(P*Q)
Y=Y+100

```

```

PRINT 1
Y=100
IF I<=1 THEN
IF EOF(I)=1 THEN
I=I+1
GOTO Start
ELSE
GOTO Start
ENDIF
ELSE
END
ENDIF
EOP
DEMO

```

[See Also]      DOWNLOAD, EOP, OPEN, EOF, LOF, SEEK, FREAD\$()

## SEEK

[Description]      This command is used to shift the specified file pointer to a certain position.

[Syntax]            SEEK file handle, offset

Parameter	Description
File handle	0 or 1.
offset	the offset characters which are shifted to the beginning of a new position

[Example]

```

DOWNLOAD "DATA",12,1234567890
DOWNLOAD "TEST.BAS"
SIZE 4,1.5
GAP 0,0
DIRECTION 1
REFERENCE 0,0
CLS
OPEN "DATA",0
SEEK 0,4
READ 0,Num$
TEXT 100,10,"3",0,1,1,"SEEK COMMAND TEST"
BAR 100,40,300,4
TEXT 100,60,"3",0,1,1,"SHIFT 4 CHARACTERS"
TEXT 100,110,"3",0,1,1,Num$
BAR 100,140,300,4
SEEK 0,0

```

```

READ 0,Num$
TEXT 100,160,"3",0,1,1,"SHIFT 0 CHARACTERS"
TEXT 100,210,"3",0,1,1,Num$
PRINT 1
EOP
TEST
[See Also] DOWNLOAD, EOP, OPEN, READ, EOF, LOF, FREAD$()

```

## LOF( )

[Description] This function returns the size of the specified file.  
 [Syntax] LOF("FILENAME")

Parameter	Description
FILENAME	The file downloaded in the printer memory.

[Example]

```

DOWNLOAD "DATA1",10,1234567890
DOWNLOAD "DATA2",15,ABCDEFGHIJKLMNO
DOWNLOAD "LofTest.BAS"
SIZE 4,1.5
GAP 0,0
DIRECTION 1
CLS
OPEN "DATA1",0
OPEN "DATA2",1
TEXT 10,20,"4",0,1,1,"LOF() FUNCTION TEST"
J=LOF("DATA1")
K=LOF("DATA2")
TEXT 10,140,"3",0,1,1,"DATA1 IS: "+STR$(J)+" Bytes"
TEXT 10,200,"3",0,1,1,"DATA2 IS: "+STR$(K)+" Bytes"
PRINT 1
EOP
LofTestt
[See Also] DOWNLOAD, EOP, OPEN, READ, EOF, SEEK, FREAD$()

```

## FREAD\$( )

[Description] This function reads a specified number of bytes of data from a file.  
 [Syntax] FREAD\$ (file handle, byte)

Parameter	Description
-----------	-------------

File handle	Either 0 or 1.
byte	Number of bytes to be read

[Example]

```

DOWNLOAD "DATA1",10,1234567890
DOWNLOAD "DATA2",15,ABCDEFGHIJKLMNO
DOWNLOAD "OPEN2.BAS"
SIZE 4,1
GAP 0,0
DIRECTION 1
CLS
OPEN "DATA1",0
OPEN "DATA2",1
SEEK 0,0
SEEK 1,0
Y$=FREAD$(0,6)
Z$=FREAD$(1,6)
TEXT 10,100,"3",0,1,1,"FREAD$(0,6) IS: "+Y$
TEXT 10,150,"3",0,1,1,"FREAD$(1,6) IS: "+Z$
PRINT 1
EOP
OPEN2

```

[See Also] DOWNLOAD, EOP, OPEN, READ, EOF, LOF(), SEEK

## FOR...NEXT

[Description] Loop is used to execute one or more lines of program repetitively. A loop counter value specifies the number of executions. Nested loops are allowed (up to 39 nested loops) in this printer. Jumping out in the middle of the FOR...NEXT loop is prohibited. \

[Syntax]

```

FOR variable = start TO end STEP increment
    statement; start < end
[EXITFOR]
NEXT

```

Parameter	Description
variable	The variable name is up to 8 characters
start	Integer or floating point numbers
end	Integer or floating point numbers
increment	Integer or floating point, positive or negative.
EXITFOR	Exit for loop

[Example]

```

DOWNLOAD "TEST.BAS"
SIZE 4,2.5

```



```

GAP 0,0
CLS
FOR I=1 TO 10 STEP 1
TEXT 100,10+30*(I-1),"3",0,1,1,STR$(I)
NEXT
FOR I=1 TO 1000 STEP 100
TEXT 200,10+((I-1)/10)*3,"3",0,1,1,STR$(I)
NEXT
FOR I=110 TO 10 STEP -10
TEXT 300,10+(ABS(I-110))*3,"3",0,1,1,STR$(I)
NEXT
FOR I=1 TO 5 STEP 0.5
IF I-INT(I)=0 THEN Y=10+60*(I-1) ELSE Y=Y+30
TEXT 400,Y,"3",0,1,1,STR$(I)
NEXT
PRINT 1
EOP
TEST
[See Also] DOWNLOAD, EOP

```

## IF...THEN...ELSE...ENDIF

[Description]	Use IF...THEN block to execute one or more statements conditionally. Either a single-line syntax or multiple-line “block” syntax can be used:
[Syntax]	<p>IF condition THEN statement</p> <p>Notice that the single-line form of IF ...THEN does not use an ENDIF statement.</p> <p>Or</p> <pre> IF condition THEN     Statements ELSE     Statements ENDIF </pre> <p>Or</p> <pre> IF condition 1 THEN     Statement block 1 ELSEIF condition 2 THEN     Statement block 2 . . . ELSEIF condition n THEN     Statement block n ENDIF </pre> <p>The syntax of IF...THEN...ELSE requires that the command be typed in</p>

one single line in less than 255 characters.

Parameter	Description
condition	Available relational operator: <, >, =, <=, >=
statement	Only one statement is available in

```
[Example]  DOWNLOAD "DEMO.BAS"
           SIZE 3,3
           GAP 0.12,0
           SPEED 4
           DENSITY 8
           DIRECTION 1
           REFERENCE 0,0
           OFFSET 0.00
SET PEELOFF
           CLS
           A=0
           B=0
           C=0
           D=0
           E=0
           F=0
           G=0
           H=0
           J=0
           K=0
           L=0
FORI=1 TO 100
           IF I-INT(I/1)*1=0 THEN A=A+I
           IF I-INT(I/2)*2=1 THEN B=B+I ELSE C=C+I
           IF I-INT(I/3)*3=0 THEN
               D=D+I
           ENDIF
           IF I-INT(I/5)*5=0 THEN
               E=E+I
           ELSE
               F=F+I
           ENDIF
           IF I-INT(I/7)*7=0 THEN
               G=G+I
           ELSEIF I-INT(I/17)*17=0 THEN
               H=H+I
           ELSEIF I-INT(I/27)*27=0 THEN
               J=J+I
```

```

ELSEIFI-INT(I/37)*37=0 THEN
    K=K+I
ELSE
    L=L+I
ENDIF
NEXT

TEXT 100,110,"3",0,1,1,"(1) 1+2+3+...+100="+STR$(A)
TEXT 100,160,"3",0,1,1,"(2) 1+3+5+...+99="+STR$(B)
TEXT 100,210,"3",0,1,1,"(3) 2+4+6+...+100="+STR$(C)
TEXT 100,260,"3",0,1,1,"(4) 3+6+9+...+99="+STR$(D)
TEXT 100,310,"3",0,1,1,"(5) 5+10+15+...+100="+STR$(E)
TEXT 100,360,"3",0,1,1," (1)-(5)="+STR$(F)
TEXT 100,410,"3",0,1,1,"(6) 7+14+21+...+98="+STR$(G)
TEXT 100,460,"3",0,1,1,"(7) 17+34+51+...+85="+STR$(H)
TEXT 100,510,"3",0,1,1,"(8) 27+54+...+81="+STR$(J)
TEXT 100,560,"3",0,1,1,"(9) 37+74="+STR$(K)
TEXT 100,610,"3",0,1,1," (1)-(6)-(7)-(8)-(9)="+STR$(L)
PRINT 1,1
EOP

```

DOWNLOAD "IFTHEN.BAS"

SIZE 3,4

GAP 0,0  
 DENSITY 8  
 SPEED 3  
 DIRECTION 0  
 REFERENCE 0,0

SET PEELOFF

CLS

A=50

B=5

C\$=""

D\$=""

:L1

IF A>100 THENGOTO L1 ELSE A=A+10

C\$=STR\$(A)+"IS SMALLER THAN 100"

TEXT 100,10,"4",0,1,1,C\$

PRINT 1

END

:L2

```

A=A+B
D$=STR$(A)+"IS LARGER THAN 100"
TEXT 100,100,"4",0,1,1,D$
PRINT 1
GOTO L1
EOP

```

**Note:**

*If the result of the expression is nonzero, the statement following THEN will be executed. If the result of the expression is zero, and the statement following the ELSE is present, it will be executed. Otherwise the next line of statement is executed.*

*If there are block of statements in IF...THEN ...ELSE, ENDIF must be used at the end of the IF...THEN ...ELSE statement.*

**Limitations:**

*The total numbers of nested IF ...THEN ...ELSE statement in a program cannot exceed 40.*

The total numbers of nested IF ...THEN ...ELSE, FOR...NEXT, GOSUB RETURN in a program cannot exceed 40 loops.

[See Also]      DOWNLOAD, EOP

## GOSUB...RETURN

[Description]      This command will branch to a subroutine, executing statements until "RETURN" is reached.

[Syntax]            GOSUB LABEL  
                       Statement  
                       END  
                       :LABEL  
                       statement  
                       RETURN

Parameter	Description
LABEL	Beginning of the subroutine. The maximum length of the label is 8 characters.

[Example]            DOWNLOAD "GOSUB1.BAS"  
                       SIZE 4,3  
                       GAP 0,0  
                       DIRECTION 1

```

CLS
TEXT 10,10,"3",0,1,1,"GOSUB & RETURN COMMAND TEST"
GOSUB DR_BOX
PRINT 1
END
:DR_BOX
FOR I=21 TO 81 STEP 10
BOX 80+I,80+I,80+300-I,80+300-I,5
NEXT
RETURN
EOP
GOSUB1
[See Also] DOWNLOAD, EOP, END, GOTO

```

## GOTO

[Description] This command is used to branch to a specified label. The label can not exceed 8 characters in length.

[Syntax] GOTO LABEL  
:LABEL

Parameter	Description
LABEL	Beginning of the point. The maximum length of the label is 8 characters.

[Example] DOWNLOAD "GOTO1.BAS"

```

SIZE 4,3
GAP 0,0
DIRECTION 1
CLS
A=0
TOTAL=0
:START
IF A<100 THEN
GOTO SUM
ELSE
GOTO PRTOU
ENDIF
:SUM
A=A+1
TOTAL=TOTAL+A
GOTO START
:PRTOU

```

```

B$="THE SUMMATION OF 1..100 IS
"+STR$(TOTAL)
TEXT 10,100,"3",0,1,1,B$
PRINT 1
END
EOP
[See Also] DOWNLOAD, EOP, END, GOSUB...RETURN

```

## REM

[Description] Comment. Prefix is “REM” that will be ignored by the printer.

[Syntax] REM

[Example] REM \*\*\*\*\*

REM This is a demonstration program\*

REM \*\*\*\*\*

DOWNLOAD "REMARK.BAS"

SIZE 4,3

GAP 0,0

DIRECTION 1

CLS

TEXT 50,50,"3",0,1,1,"REMARK DEMO PROGRAM"

REM TEXT 50,100,"3",0,1,1,"REMARK DEMO PROGRAM"

PRINT 1,1

EOP

REMARK

[See Also] DOWNLOAD, EOP, END

## INT()

[Description] This function is used to truncate a floating point number.

[Syntax] INT (n)

Parameter	Description
n	n can be positive or negative integer, floating point number or mathematical expression.

[Example] DOWNLOAD "DEMO.BAS"

SIZE 4,1

GAP 0,0

DIRECTION 1

INPUT "Number: ",Num

```

CLS
REM **** To round up or down****
N=INT(Num+0.5)
IF N>Num THEN
TEXT 50,100,"3",0,1,1,"To round up= "+STR$(N)
ELSE
TEXT 50,100,"3",0,1,1,"To round down= "+STR$(N)
ENDIF
PRINT 1
EOP
56.2

```

[See Also]      DOWNLOAD, EOP, END, ABS(), ASC(), STR\$()

## LEFT\$( )

[Description]      This function returns the specified number of characters down from the initial character of a string.

[Syntax]            LEFT\$(X\$, n)

Parameter	Description
X\$	The string to be processed
n	The number of characters to be returned

[Example]            DOWNLOAD "TEST.BAS"

```

SIZE 4,1
GAP 0,0
DIRECTION 1
A$="BARCODE PRINTER DEMO PRINTING"
C$=LEFT$(A$,10)
CLS
TEXT 10,10,"3",0,1,1,A$
TEXT 10,100,"3",0,1,1,"10 LEFT 10 CHARS: "+C$
PRINT 1
EOP
TEST

```

[See Also]            DOWNLOAD, EOP, END, RIGHT\$(), MID\$(), LEN(), STR\$()

## LEN( )

[Description]      This function returns the length of a string.

[Syntax]            LEN (string)

Parameter	Description
string	The string whose length is to be measured.

[Example]      DOWNLOAD "DEMO.BAS"  
 SIZE 4,1  
 GAP 0,0  
 DIRECTION 1  
 A\$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"  
 B=LEN(A\$)  
 CLS  
 TEXT 10,10,"3",0,1,1,A\$  
 TEXT 10,50,"3",0,1,1,"STRING LENGTH="+STR\$(B)  
 PRINT 1  
 EOP  
 DEMO

[See Also]      DOWNLOAD, EOP, END, LEFT\$(), LEN(), RIGHT\$(), MID\$(), STR\$(), VAL()

## MID\$( )

[Description]      This function is used to get the specified number of characters down from the mth character of a string.

[Syntax]            MID\$(string, m, n)

Parameter	Description
string	The string to be processed.
m	The beginning of mth characters in the string. $1 \leq m \leq \text{string length}$
n	The number of characters to return.

[Example]      DOWNLOAD "DEMO.BAS"  
 SIZE 4,1  
 GAP 0,0  
 DIRECTION 1  
 A\$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"  
 E\$=MID\$(A\$,11,10)  
 CLS  
 TEXT 10,10,"3",0,1,1,A\$  
 TEXT 10,200,"3",0,1,1,"10 MIDDLE CHARS: "+E\$  
 PRINT 1  
 EOP



DEMO

[See Also] DOWNLOAD, EOP, END, LEFT\$(), LEN(), RIGHT\$(), STR\$(), VAL()

## RIGHT\$( )

[Description] This function returns the specified number of characters up from the end of a string.

[Syntax] RIGHT\$ (X\$, n)

Parameter	Description
X\$	The string to be processed
n	The number of characters to be returned from the right side (end) of the string.

[Example] DOWNLOAD "DEMO.BAS"  
SIZE 4,1  
GAP 0,0  
DIRECTION 1  
A\$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"  
D\$=RIGHT\$(A\$,10)  
CLS  
TEXT 10,10,"3",0,1,1,A\$  
TEXT 10,150,"3",0,1,1,"10 RIGHT CHARS: "+D\$  
PRINT 1  
EOP  
DEMO

[See Also] DOWNLOAD, EOP, END, LEFT\$(), LEN(), MID\$(), STR\$(), VAL()

## STR\$( )

[Description] This function converts a specified value or expression into corresponding string of characters.

[Syntax] STR\$ (n)

Parameter	Description
n	An integer, floating point number or mathematical expression.

[Example] DOWNLOAD "DEMO.BAS"  
SIZE 4,1  
GAP 0,0

```

DIRECTION 1
A$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
F=100
G=500
H$=STR$(F+G)
CLS
TEXT 10,10,"3",0,1,1,A$
TEXT 10,60,"3",0,1,1,"F="+STR$(F)
TEXT 10,110,"3",0,1,1,"G="+STR$(G)
TEXT 10,160,"3",0,1,1,"F+G="+H$
PRINT 1
EOP
DEMO

```

[See Also]      DOWNLOAD, EOP, END, LEFT\$(), LEN(), RIGHT\$(), MID\$(), VAL()

## VAL()

[Description]      This function is used to convert numeric character into corresponding integer or floating point number.

[Syntax]            VAL("numeric character")

Parameter	Description
numeric character	"0~9", "."

[Example]

```

DOWNLOAD "DEMO.BAS"
SIZE 4,1
GAP 0,0
DIRECTION 1
A$="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
F$="100"
G$="500"
CLS
H=VAL(F$)+VAL(G$)
I$=STR$(H)
TEXT 10,10,"3",0,1,1,A$
TEXT 10,60,"3",0,1,1,"F="+F$
TEXT 10,110,"3",0,1,1,"G="+G$
TEXT 10,160,"3",0,1,1,"F+G="+I$
PRINT 1
EOP
DEMO

```

[See Also]      DOWNLOAD, EOP, END, LEFT\$(), LEN(), RIGHT\$(), MID\$(), STR\$()

## BEEP

[Description] This command is used to issue a beep sound on portable keyboard.

[Syntax] BEEP

Parameter	Description
None	N/A

[Example] DOWNLOAD "DEMO.BAS"  
SIZE 4,4  
GAP 0,0  
DIRECTION 1  
BEEP  
INPUT "Text1 =",TEXT1\$  
CLS  
TEXT 100,100,"3",0,1,1,TEXT1\$  
PRINT 1  
EOP

# Device Reconfiguration Commands

## SET COUNTER

[Description] Counter can be a real counter or a variable. This setting sets the counter number in program and their increments. There are three different kind of counters: digit (0~9~0), lower case letter (a~z~a) and upper case letter (A~Z~A).

[Syntax] SET COUNTER @n step  
@n = "Expression"

Parameter	Description
@n	n: counter number. There are 61 counters available (@0 ~ @60) in the printer. @0 to @50 will be cleared while restarting the printer. @51 to @60 will be stored in printer until the printer is restored to factory default.@51~@55 were supported since V6.37 EZ.@56~@60 were supported since V6.74 EZ.

step	The increment of the counter, can be positive or negative. $-999999999 \leq \text{step} \leq 999999999$ If the counter is used as a fixed variable, please set the increment to 0.
Expression	Initial string. String length is 101 bytes

[Example] SET COUNTER @0+1  
SET COUNTER @1+0  
SET COUNTER @2-1  
SET COUNTER @3 1  
@0="0001"  
@1="0101"  
@2="000A"  
@3="1"  
SIZE 4,0.5  
GAP 0,0  
DIRECTION 1  
CLS  
TEXT 600,10,"3",0,1,1,3,"@0 @1 @2"  
TEXT 600,30,"3",0,1,1,3,"Label "+@3+" -----"  
TEXT 600,50,"3",0,1,1,3,@0+" "+@1+" "+@2  
PRINT 5

[See Also] PRINT, TEXT, BARCODE

## SET KEY1, SET KEY2

[Description] This setting is used to enable/disable the KEY1/KEY2function.The default function of KEY1 is “FEED” key, KEY2 is “PAUSE” key. Before setting KEY1/KEY2function otherwise, please disable KEY1/KEY2first. The setting will remain resident in the printer even when the printer is power off.

[Syntax] SET KEY1ON/OFF  
SET KEY2ON/OFF  
SET KEY3 ON/OFF

Parameter	Description
ON	Enable KEYn function
OFF	Disable KEYn function

**Note:** The setting will remain in the printer even if the printer is power off.

[Example]      DOWNLOAD "DEMO.BAS"  
SIZE 3,1  
GAP 0,0  
DENSITY 8  
SPEED 3  
DIRECTION 0  
REFERENCE 0,0  
[See Also]      OFFEST, PRINT

## SET PEEL

[Description]    This setting is used to enable/disable the self-peeling function. The default setting for this function is off. When this function is set on, the printer stops after each label printing, and does not print the next label until the peeled label is taken away. This setting will be saved in printer memory when turning off the power.

[Syntax]        SET PEELON/OFF

Parameter	Description
ON	Enable the self-peeling function
OFF	Disable the self-peeling function

[Example]        DOWNLOAD "DEMO.BAS"  
SIZE 3,1  
GAP 0,0  
DENSITY 8  
SPEED 3  
DIRECTION 0  
REFERENCE 0,0  
SET CUTTER OFF  
SET KEY1 OFF  
SET KEY2 OFF  
SET KEY3 OFF  
KEY1=0  
KEY2=0  
KEY3=0  
:START  
IF KEY1=1 THEN  
CLS  
TEXT 100,10,"3",0,1,1,"KEY1 (MENU key) is pressed!! "  
PRINT 1,1  
ELSEIF KEY2=1 THEN  
CLS

```

TEXT 100,10,"3",0,1,1,"KEY2 (PAUSE key) is pressed!!"
PRINT 1,1
ELSEIF KEY3=1 THEN
CLS
TEXT 100,10,"3",0,1,1,"KEY3 (FEED key) is pressed!!"
TEXT 100,60,"3",0,1,1,"End of test"
PRINT 1,1
SET KEY1 ON
SET KEY2 ON
SET KEY3 ON
END
ENDIF
GOTO START
EOP
DEMO
[See Also]   OFFEST, PRINT

```

## SET TEAR & SET STRIPPER

[Description] This command is used to enable/disable feeding label to gap/black mark position for tearing off. This setting will be saved in printer memory when turning off the power.

[Syntax] SET TEARON/OFF (TSPL2 language printers only)  
 SET STRIPER ON/OFF (TSPL language printers only)

Parameter	Description
ON	The label gap will stop at the tear off position after print.
OFF	The label gap will NOT stop at the tear off position after print. The beginning of label will be aligned to print head.

[Example]

```

REM ***TEAR FUNCTION ON***
SIZE 3,3
GAP 0.08,0
DENSITY 8
SPEED 4
DIRECTION 0
REFERENCE 0,0
SET CUTTER OFF
SET PEEL OFF
SET TEAR ON
CLS

```

TEXT 50,100,"3",0,1,1,"TEAR FUNCTION TEST"

PRINT 1

[See Also] SET PEEL, SET CUTTER

## SET HEAD

[Description] This setting is used to enable/disable head open sensor. If the head open sensor is turned off, an open printer head will not return an error message. This setting will be saved in printer memory. This command is only available for BPL2 printers.

[Syntax] SET TEARON/OFF

Parameter	Description
ON	Turn on the "HEADOPEN" sensor
OFF	Turn off the "HEADOPEN" sensor

[Example] SET HEADON  
SET HEADOFF

## SET RIBBON

[Description] This setting is used to enable/disable ribbon sensor detection. (Thermal Transfer Printing/Thermal Direct Printing) Printer will detect the presence of a ribbon to determine using either direct thermal or thermal transfer printing upon printer startup. This setting will NOT be saved in printer memory.

[Syntax] SET RIBBON ON/OFF/INSIDE/OUTSIDE

Parameter	Description
ON	Thermal transfer printing
OFF	Thermal direct printing
INSIDE	The ribbon is inside wound. For TTP-384M only. *Since V6.80EZ.
OUTSIDE	The ribbon is outside wound. For TTP-384M only. *Since V6.80EZ.

[Example] REM \*\*\*\*\*Disable ribbon detection sensor for direct thermal printing.  
SET RIBBON OFF  
SIZE 4,1  
GAP 0,0  
CLS

```

TEXT 10,10,"3",0,1,1,"Direct thermal printing."
PRINT 1
REM *****Enable ribbon detection sensor for thermal transfer printing.
SET RIBBON ON
SIZE 4,1
GAP 0,0
CLS
TEXT 10,10,"3",0,1,1,"Thermal transfer printing."
PRINT 1
REM *****For using ink-in ribbon in TTP-384M.
SET RIBBON INSIDE
SIZE 4,1
GAP 0,0
CLS
TEXT 10,10,"3",0,1,1,"TTP-384M is using ink-in ribbon."
PRINT 1
REM *****For using ink-out ribbon in TTP-384M.
SET RIBBON OUTSIDE
SIZE 4,1
GAP 0,0
CLS
TEXT 10,10,"3",0,1,1,"TTP-384M is using ink-out ribbon."
PRINT 1

```

## SET COM1

[Description] This setting defines communication parameters for printer serial port.  
 [Syntax] SET COM1 baud, parity, data, stop

Parameter	Description
baud	Baud rate, available baud rates are as listed : 24: 2400 bps 48: 4800 bps 96: 9600 bps 19: 19200 bps 38: 38400 bps 57: 57600 bps 115: 115200 bps
Parity	Parity check N: None parity check E: Even parity check O: Odd parity check



data	Data bit 8: 8 bits data 7: 7 bits data
stop	Stop bit 1: 1 stop bit 2: 2 stop bits

[Example] The parallel port is used to setup the printer serial port in this example via MS-DOS mode.

```
C:\>COPY CON LPT1<ENTER>
SET COM1 19,N,8,1<ENTER>
<CTRL><Z><ENTER>
C:\>
```

**Note:**

**<ENTER> stands for PC keyboard "ENTER" key.**

**<CTRL><Z> means to hold PC keyboard "CTRL" key then press PC keyboard <Z> key.**

## SETPRINTKEY

[Description] This command will print one label and feed label gap to tear bar position for tearing away. Press FEED button to print the next label or batch of labels. If label content includes serial text or barcode, it will change the serial number accordingly. This setting will be saved in printer memory.

[Syntax] SET PRINTEKYOFF/ON/AUTO/<num>

Parameter	Description
OFF	Disable this function
ON	Enable this function
AUTO	Enable this function
<num>	Numbers of labels will be printed if FEED button is pressed.

[Example]

```
SIZE 4,2.5
GAP 0.12,0
SET PRINTKEY ON
SET COUNTER @0 1
@0="0001"
CLS
TEXT 10,10,"5",0,1,1,@0
PRINT 1
```

Execute:

Syntax	Receive "PRINTm"	Print Out
SET PRINTKEYON or SET PRINTKEY AUTO	1.) PRINT 2	Label 1~2
	2.) Press FEED key	Label 3~4

Syntax	Receive "PRINT m,n"	Print Out
SET PRINTKEYON or SET PRINTKEY AUTO	1.) PRINT 1,2	Label 1, Label 1
	2.) Press FEED key	Label 2, Label 2

Syntax	Receive "PRINT -1,n"	Print Out
SET PRINTKEYON or SET PRINTKEY AUTO	1.) PRINT -1,2	Label 1, Label 1
	2.) Press FEED key	Label 1, Label 1

Syntax	Receive "PRINTm"	Print Out
SET PRINTKEY 5	1.) PRINT 2	Label 1~2
	2.) Press FEED key	Label 3~7

Syntax	Receive "PRINT m,n"	Print Out
SET PRINTKEY 5	1.) PRINT 1,2	Label 1, Label 1
	2.) Press FEED key	Label 2~6

Syntax	Receive "PRINT -1,n"	Print Out
SET PRINTKEY 5	1.) PRINT -1,2	Label 1, Label 1
	2.) Press FEED key	Label 1, Label 1

## SET REPRINT

[Description] This command will disable/enable a reprinting attempt subsequent to a "no paper", "no ribbon" or "carriage open" error.

[Syntax] SET REPRINTOFF/ON

Parameter	Description
OFF	Disable this function
ON	Enable this function

[Example] SET REPRINTON

## PEEL

[Description] This command is used to obtain status of the peel-off sensor. Its attribute is read only.

[Syntax] PEEL

Parameter	Description
0	Paper is not on top of peel sensor
1	Paper is on top of peel sensor

[Example] DOWNLOAD "DEMO.BAS"  
SIZE 4,1  
GAP 0,0  
SET PEEL OFF  
SET KEY1 OFF  
SET LED1 OFF  
SET LED3 OFF  
:START  
LED1=0  
LED3=0  
IF KEY1=1 THEN GOTO A  
GOTO START  
:A  
LED1=1  
CLS  
TEXT 10,10,"3",0,1,1,"PEEL Function Test!! "  
PRINT 1,1  
:B  
LED1=0  
IF PEEL=1 THEN  
LED3=1  
GOTO B  
ELSE  
CLS  
TEXT 10,10,"3",0,1,1,"The label is removed from the PEEL sensor!!"  
PRINT 1,1  
GOTO START  
ENDIF  
EOP  
DEMO

## KEY1, KEY2

[Description] This command is used to read the status of KEY1,KEY2 .

[Syntax] KEY m=n

Key	Return Value
KEY1 (FEED)	0: released 1: pressed
KEY2(PAUSE)	0: released 1: pressed

[Example] DOWNLOAD "DEMO.BAS"  
SIZE 3,1  
GAP 0,0  
SPEED 4  
DENSITY 8  
DIRECTION 1  
REFERENCE 0,0  
SET KEY1OFF  
:START  
IF KEY1=1 THEN  
    CLS  
    TEXT 100,10,"3",0,1,1,"KEY FUNCTION TEST"  
    PRINT 1,1  
GOTO START  
EOP  
DEMO

## Printer Global Variables

### @LABEL

[Description] This variable is used to count how many pieces of labels have been printed. It can't be initialized if the printer is reset. It will be memorized if the printer power is turned off.

[Syntax] Write attribute: @LABEL=n or @LABEL="n"  
Read attribute: A=LABEL or A\$=STR\$(LABEL)

Parameter	Description
n	Number of labels printed. $0 \leq n \leq 999999999$

[Example]

```
DOWNLOAD "DEMO.BAS"
SIZE 4,2.5
GAP 0,0
DIRECTION 1
CLS
TEXT 10,50,"3",0,1,1,@LABEL
TEXT 10,100,"3",0,1,1,"@LABEL="+STR$(LABEL)
TEXT 10,150,"3",0,1,1,"*****Statement 1*****"
IF LABEL>1000 THEN
TEXT 10,200,"3",0,1,1,"LABEL>1000"
ELSE
TEXT 10,200,"3",0,1,1,"LABEL<1000"
ENDIF
TEXT 10,250,"3",0,1,1,"*****Statement 1*****"
A=LABEL
IF A>1000 THEN
TEXT 10,300,"3",0,1,1,"A>1000"
ELSE
TEXT 10,300,"3",0,1,1,"A<1000"
ENDIF
TEXT 10,350,"3",0,1,1,"*****Statement 3*****"
A$=STR$(LABEL)
IF VAL(A$)>1000 THEN
TEXT 10,400,"3",0,1,1,"VAL(A$)>1000"
ELSE
TEXT 10,400,"3",0,1,1,"VAL(A$)<1000"
ENDIF
PRINT 1,1
EOP
DEMO
```