

1 DATECS

Command Reference **THERMAL PRINTER**

MODEL EP-1000

Rev. 1.00 Newly issued November 05, 2002

1. EP-1000 COMMAND SET

	Code	Command Function
1	HT	Horizontal Tab Command
2	LF	Printing and Paper Feed Command
3	CR	Print one line Command
4	ESC SP	Setting the right space amount of the character
5	ESC \$	Specifying the Absolute Positions
6	ESC %	Specifying/Canceling Download Character Set
7	ESC &	Define user characters
8	ESC !	Collective Specifying Printing Mode
9	ESC *	Specifying the Bit Image Mode
10	ESC –	Specifying/ Canceling Underline
11	ESC .	Printer self test
12	ESC 2	Specifying 1/6-inch line feed rate
13	ESC 3	Setting line feed rate of minimum pitch
14	ESC =	Data Input Control
15	ESC @	Initializing the Printer
16	ESC D	Setting Horizontal Tab Position
17	ESC E	Specifying/canceling highlighting
18	ESC G	Specifying/canceling Double Printing
19	ESC J	Printing and feeding paper n/216 inch
20	ESC M	Switching from font A to font B
21	ESC T	Printing Diagnostic information
22	ESC V	Specifying/Canceling 90°-right- turned Characters
23	ESC Y	Specifying print density
24	ESC Z	Returning diagnostic information
25	ESC \	Specifying the relative positions
26	ESC a	Aligning the characters
27	ESC c5	Enabling/Disabling Panel Switches
28	ESC d	Printing and Feeding the paper by n lines
29	ESC i	Cutting the paper
30	ESC m	Cutting the paper
31	ESC p	Generating of impulse opening the drawer
32	ESC v	Transmitting the printer status
33	ESC {	Specifying/Canceling the Inverted Characters
34	GS k	Printing the bar code
35	GS w	Selecting the horizontal size (Scale factor) of the Bar Code
36	GS h	Selecting the height of the Bar Code
37	GS H	Selecting of Printing Position of HRI Code
38	GS f	Selecting the font of HRI characters
39	GS *	Defining the Download Bit Image (LOGO)
40	GS /	Printing the Download, Bit Image
41	GS :	Starting/Ending Macro Definition
42	GS ^	Executing the Macro

2. COMMAND DETAILS

Description of Items

XXXX ALL

[Function] Command Function

[Code] A sequence of code constituting a command is represented in hexadecimal number for < >H, binary number for < >B, and decimal number for < >, respectively; []k represents a repeat count of k-times.

[Range] Indicates an argument value (setting range) for the command in hexadecimal number (partly in decimal number).

* When used for application other than defined in each control, an error may occur.

Always use in the defined range.

[Outline] Describes a command outline.

[Caution] Describes a caution as required.

[Default] Describes an initial value in hexadecimal number for the command when accompanied by an argument.

[See Also] Describes the associated commands for use.

[Sample Program] Describes a coding example in the Q-BASIC sample program.

* This example is only for your reference and differs depending on the language used, version, and so on. For details, see the manual for the language used.

HT

[Function] Horizontal Tab Command

[Code] <09>H

[Outline] Shifts the printing position to the next horizontal tab position.

• Ignored when the next horizontal tab position has not been set.

[Caution] • The horizontal tab position is set by ESC D.

• Initial setting of the horizontal tab position is each 8 characters in 9th, 17th, 25th, columns from FONT A.

[See Also] ESC D

[Sample Program]

```
PRINT #1, "0123456789012345678901";
```

```
PRINT #1, CHR$ (&HA);
```

```
PRINT #1, CHR$ (&H9) + "AAA";
```

```
PRINT #1, CHR$ (&H9) + "BBB";
```

```
PRINT #1, CHR$ (&HA);
```

```
PRINT #1, CHR$ (&H1B) + "D";
```

```
PRINT #1, CHR$ (3) + CHR$ (7) + CHR$ (14) + CHR$ (0);
```

```
PRINT #1, CHR$ (&H9) + "AAA";
```

```
PRINT #1, CHR$ (&H9) + "BBB";
```

```
PRINT #1, CHR$ (&H9) + "CCC" + CHR$ (&HA);
```

LF

[Function] Printing and Paper Feed Command

[Code] <0A>H

[Outline] Prints data inside the input buffer and feeds lines based on the line feed amount having been set.

- The head of the line becomes the next print starting position.

[See Also] ESC 2, ESC 3

[Sample Program]

```
PRINT #1, "AAA" + CHR$ (&HA);  
PRINT #1, "BBB" + CHR$ (&HA);  
PRINT #1, CHR$ (&HA);  
PRINT #1, "CCC" + CHR$ (&HA);
```

CR

[Function] Print one line Command

[Code] <0D>H

[Outline] This command is ignored or its action is the same as LF depending on the last execution of GS) 2 command (FLAG SETTING COMMAND).

If Flag 2 is 0, carriage return (CR) command is ignored.

If Flag 2 is 1, CR command will act as LF command.

[Default] CR command is ignored. If you want that it acts like LF command set flag 2 to 1 by GS) command.

[See Also] GS)

ESC SP n

[Function] Setting the right space amount of the character

[Code] <1B>H<20>H<n>

[Range] {0 =< n =< 20(Hex)}

[Outline] The rightward space amount is set in dot unit (1/203 inch unit).

[Caution] The rightward space amount in doublewide mode is made double of the set volume.

[Default] n = 0

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + " " + CHR$ (0);  
PRINT #1, "AAAAA" + CHR$ (&HA);  
PRINT #1, CHR$ (&H1B) + " " + CHR$ (1);  
PRINT #1, "AAAAA" + CHR$ (&HA);  
PRINT #1, CHR$ (&H1B) + " " + CHR$ (12);  
PRINT #1, "AAAAA" + CHR$ (&HA);
```

ESC \$ n1 n2

[Function] Specifying the Absolute Positions

[Code] <1B>H<24>H<n1><n2>

[Range] {0 =< n1 =< FF(Hex)}

{0 =< n2 =< 1(Hex)}

[Outline] The printing start position is specified in the number of dots (1/203 inch unit) from the beginning of line.

- The number of dots is divided by 256, whose quotient is taken as n2 and the residual as n1.
- Therefore, the printing start position is equal to $n1 + n2 \div 256$ from the beginning of line.

[Caution] Specifying beyond the line end is ignored.

[Default] The initial value is not specified.

[See Also] ESC \

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + "$";
PRINT #1, CHR$ (0) + CHR$ (0) + "A";
PRINT #1, CHR$ (&H1B) + "$";
PRINT #1, CHR$ (50) + CHR$ (0) + "B";
PRINT #1, CHR$ (&H1B) + "$";
PRINT #1, CHR$ (0) + CHR$ (1) + "C";
PRINT #1, CHR$ (&HA);
PRINT #1, CHR$ (&H1B) + "$";
PRINT #1, CHR$ (100) + CHR$ (0) + "A";
PRINT #1, CHR$ (&H1B) + "\";
PRINT #1, CHR$ (&HC2) + CHR$ (&HFF) + "B";
PRINT #1, CHR$ (&HA);
```

ESC % n

[Function] Specifying/Canceling Download Character Set

[Code] <1B>H<25>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] Specifying/canceling download characters.

Further, only the lowest bit (n0) is valid for n.

n (Hex)	Function
0	Selecting download character set
1	Canceling download character set

[Caution] This setting cannot be restored to initial value even with Initialize command or by turning power OFF. It can be restored to the initial value by ESC `_`, a command for returning to initial state.

[Default] n = 0

[See Also] ESC &

ESC & m n1 n2 [d] k

[Function] Define user characters

[Code] <1B>H<26>H<m><n1><n2>[<d>]k

[Range] {m = 0, 1, 2, 3(Hex)}

{20 =< n1 =< FF(Hex)}

{n1 =< n2 =< FF(Hex)}

{k=(n2-n1+1)*48} (m=2)

{k=(n2-n1+1)*16} (m=3)

[Outline] Defines a group of user characters.

m=0: Copy internal character set A to user character set A

(Parameters n1, n2 and d are omitted)

m=1: Copy internal character set B to user character set B

(Parameters n1, n2 and d are omitted)

m=2: For Font A (12 ? 24), n1 denotes Start character code and n2 End character code. Every character is 48 bytes, two bytes for each line. Only the first nibble of the second byte is used.

m=3: Define character group with ASCII codes between >= n1 and =< n2 for character set B (9 ? 16). Every character is 16 bytes.

[Caution] The data for character set A is composed from left to right and from top to bottom with two bytes for each horizontal line. The first byte contains the first 8 bits with the left most bit is MSB. From the second byte only the first nibble (the most significant 4 bits) is valid.

The data for character set B is composed from left to right and from top to bottom with only one byte for each horizontal line. The ninth bith is always 0.

Downloaded characters are valid even after switching off the printer.

Downloaded characters are valid even after switching off the printer or after issuing Initialize command.

ESC ! n

[Function] Collective Specifying Printing Mode

[Code] <1B>H<21>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] Printing mode is assigned.

Bit	Function	Value	
		0	1
0	Character Font	Font A	Font B
1	Undefined		
2	Undefined		
3	High-lighting	Canceled	Specified
4	Double height	Canceled	Specified
5	Double width	Canceled	Specified
6	Undefined		
7	Underline	Canceled	Specified

Each n bit indicates the following:

[Caution] • With double height and double width being specified simultaneously, double wide and double high characters are consisted.

• An underline is attached to the full character width, which, however, is not attached to the part having been skipped by the horizontal tab.

Neither is it attached to 90°-right-turned characters.

• The underline width is as having been specified by <ESC ->. (The default setting is 1 dot width.)

• n case that double wide character and normal character exist in same one line, the layout of underline is consistent one.

[Default] n = 0

[See Also] ESC E, ESC –

Bit Function

[Sample Program]

```
PRINT #1, CHR$(&H1B) + "!" + CHR$(&H00) + "H" ;
PRINT #1, CHR$(&H1B) + "!" + CHR$(&H01) + "H" ;
PRINT #1, CHR$(&H1B) + "!" + CHR$(&H08) + "H" ;
PRINT #1, CHR$(&H1B) + "!" + CHR$(&H10) + "H" ;
PRINT #1, CHR$(&H1B) + "!" + CHR$(&H20) + "H" ;
PRINT #1, CHR$(&H1B) + "!" + CHR$(&HB9) + "H" ;
PRINT #1, CHR$(&HA) ;
END
```

ESC * m n1 n2 [d] k

[Function] Specifying the Bit Image Mode

[Code] <1B>H<2A>H<m><n1><n2> [d] k

[Range] {m= 0, 1, 20, 21(Hex)}

{0 =< n1 =< FF(Hex)}

{0 =< n2 =< 1(Hex)}

{0 =< d =< FF(Hex)}

{k = n1 + FF(Hex) x n2} (m=0, 1(Hex)) Number of bytes containing image data

{k = (n1 + FF(Hex) x n2) x 3} (m=20, 21(Hex)) Number of bytes containing image data

[Outline] According to the number of dots specified in n1, n2, specify the bit image of mode m.

- The No. of dots printed is divided by 256, whose quotient is taken as n2 and residual as n1.
- Bit image data is sent starting from the top to bottom and from the left to right (vertical columns scanning). In modes m=0, and 1 only one byte per column is sent and in mode m=20h, 21h 3 bytes for each column are sent.
- d is bit image data, the bits subject to printing are taken as "1" and those not as "0".
- The bit image modes specified by m are shown as follows:

m (Hex)	Mode	Vertical Direction		Horizontal Direction	
		Dots	Dot Density	Dot Density	Max. Dots
0	8-dot single density	8	67 DPI	101 DPI	192
1	8-dot double density	8	67 DPI	203 DPI	384
20	24-dot single density	24	203 DPI	101 DPI	192
21	24-dot double density	24	203 DPI	203 DPI	384

[Caution] • When the values set in m (bit image mode) are out of the above range, the data following after n1 is processed as normal printing data.

- After completion of bit image printing, printer returns to normal data processing mode.
- Line space is automatically set to 0.

Mode m (Hex)

[Sample Program]

```
PRINT #1, CHR$(&H1B) + " * ";
PRINT #1, CHR$(0) + CHR$(20) + CHR$(0); IMG1:
GOSUB IMG1 PRINT #1, CHR$(&HFF);
PRINT #1, CHR$(&HA); FOR I=1 TO 18
PRINT #1, CHR$(&H1B) + " * "; PRINT #1, CHR$(&H85);
PRINT #1, CHR$(1) + CHR$(20) + CHR$(0); NEXT I
GOSUB IMG1 PRINT #1, CHR$(&HFF);
PRINT #1, CHR$(&HA); RETURN
PRINT #1, CHR$(&H1B) + " * "; IMG2:
PRINT #1, CHR$(32) + CHR$(20) + CHR$(0); PRINT #1, CHR$(&HFF);
GOSUB IMG2 PRINT #1, CHR$(&HFF);
PRINT #1, CHR$(&HA); PRINT #1, CHR$(&HFF);
PRINT #1, CHR$(&H1B) + " * "; FOR I=1 TO 18
PRINT #1, CHR$(33) + CHR$(20) + CHR$(0); PRINT #1, CHR$(&H80);
GOSUB IMG2 PRINT #1, CHR$(&H00);
PRINT #1, CHR$(&HA); PRINT #1, CHR$(&H05);
END NEXT I
PRINT #1, CHR$(&HFF);
PRINT #1, CHR$(&HFF);
PRINT #1, CHR$(&HFF);
RETURN
```

[

This command has one more version.

[Function] Specifying the Bit Image Mode (PCX format)

[Code] <1B>H<2A>H<m><n>{<a><00>H} [d]k

[Range] {m= 10, 11 or 12(Hex)}

{0 =< n =< 30(Hex)}

{0 =< a =< 18(Hex)}

{0 =< d =< FF(Hex)}

{0 =< k =< n x 18(Hex)}

[Outline] Designates a bit image of n*8 dots horizontal by 24 or a dots vertical.

Dot density is fixed at 203 dpi both horizontally and vertically.

Bit image mode selected by the value of m is shown in the following table.

m (Hex)	Type
10	Non-compressed data of 24 dots vertical (Argument "a" and 00(Hex) are not used.)
11	Compressed data of 24 dots vertical (Argument "a" and 00(Hex) are not used.)
12	Compressed data of a dots vertical

n denotes the number of bytes for horizontal size of a bit image.

d denotes a graphic data.

In 10(Hex) mode, data is in n ? 24 bytes.

For data in 11(Hex) mode, the size in non-compressed format shall be n ? 24 bytes.

For data in 12(Hex) mode, the size in non-compressed format shall be n ? a bytes.

Compression system in 11(Hex) and 12(Hex) is the same as that for PCX file format (monochrome). The printer receives the compressed data, reproduces it in the following format and prints as bit-image data.

A When any one of the two bytes of the input is 0, the byte is treated as graphic data regardless of compression.

When both of the upper two bytes of graphic data to be printed are 1, the format B shall always be taken.

B When both of the upper two bytes of input are 1, these bytes are not graphic data and the remaining 6 bits are treated as data for specifying the times the data is repeated. The number of times to be repeated is 0- 3F (Hex). The byte following this byte to specify the number of times repeated is treated as graphic data to be repeated.

Graphic data is arranged in the layout from left to right and top to bottom before printing.

[Caution] • An underline is attached to the full character width. It is, however, not attached to the part having been skipped by horizontal tab command.

• An underline is not attached to a 90°- right-turned characters.

[Default] n = 0

[See Also] ESC !

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + "—" + CHR$ (0);
```

```
PRINT #1, "AAAAA" ;
```

```
PRINT #1, CHR$ (&H1B) + "—" + CHR$ (1);
```

```
PRINT #1, "AAAAA" + CHR$ (&HA);
```

ESC – n

[Function] Specifying/ Canceling Underline

[Code] <1B>H<2D>H<n>

[Range] {0 =< n =< 02(Hex)}

[Outline] Specifying/canceling an underline.

• Types of underlines by n value are shown below:

n (Hex)	Type
0	Canceling an underline.
1	Specifying an underline for 1-dot width.
2	Specifying an underline for 2-dots width.

[Caution] • An underline is attached to the full character width. It is, however, not attached to the part having been skipped by horizontal tab command.

• An underline is not attached to a 90°- right-turned characters.

[Default] n = 0

[See Also] ESC !

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + “-” + CHR$ (0);  
PRINT #1, “AAAAA” ;  
PRINT #1, CHR$ (&H1B) + “-” + CHR$ (1);  
PRINT #1, “AAAAA” + CHR$ (&HA);
```

ESC .

[Function] Printer self test

[Code] <1B>H<2E>H

[Range] None

[Outline] Prints test page, downloaded bit image, and self-diagnostic information. The self-diagnostic information includes print density, print head temperature, battery voltage, Serial or IrDA interface, baud rate, and memory switch setting.

[See Also] ESC T

ESC 2

[Function] Specifying 1/6-inch line feed rate

[Code] <1B>H<32>H

[Outline] The line feed rate per line is specified by 1/6 inch.

[Sample Program]

```
PRINT #1, “AAAAA” + CHR$ (&HA);  
PRINT #1, CHR$ (&H1B) + “3” + CHR$ (0);  
PRINT #1, “AAAAA” + CHR$ (&HA);  
PRINT #1, CHR$ (&H1B) + “3” + CHR$ (50);  
PRINT #1, “AAAAA” + CHR$ (&HA);  
PRINT #1, CHR$ (&H1B) + “2”;  
PRINT #1, “AAAAA” + CHR$ (&HA);  
PRINT #1, “AAAAA”;  
PRINT #1, CHR$ (&H1B) + “J” + CHR$ (100);  
PRINT #1, “AAAAA” + CHR$ (&HA);  
PRINT #1, “AAAAA” + CHR$ (&HA);
```

ESC 3 n

[Function] Setting line feed rate of minimum pitch

[Code] <1B>H<33>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] The line feed rate per line is specified by n/203 inch.

[Default] n = 22(Hex) 1/6-inch

[Sample Program] See Sample Program and Print Results for ESC 2.

ESC = n

[Function] Data Input Control

[Code] <1B>H<3D>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] Selecting equipment in which data input from the host is effective.

• Each bit of n indicates as follows:

Bit	Equipment	Value	
		0	1
0	Printer	Invalid	Valid
1	Not defined		
2	Not defined		
3	Not defined		
4	Not defined		
5	Not defined		
6	Not defined		
7	Not defined		

• When the printer has not been selected, this printer abandons all the received data until it is selected by this command.

[Caution] • Even when the printer has not been selected, it can become BUSY state through printer operation.

• When the printer is deselected, this printer discards all the data until it is selected with this command.

[Default] The initial value of n is “1”.

[Sample Program]

```
PRINT #1, "AAAAA";
```

```
PRINT #1, CHR$ (&H1B) + "=" + CHR$ (0);
```

```
PRINT #1, "aaaaa" + CHR$ (&HA);
```

```
PRINT #1, CHR$ (&H1B) + "=" + CHR$ (1);
```

```
PRINT #1, "AAAAA" + CHR$ (&HA);
```

ESC @

[Function] Initializing the Printer

[Code] <1B>H<40>H

[Outline] Clears data stored in the print buffer and brings various settings to the initial state (Default state). Items not cleared are as shown below.

- Data inside the internal input buffer are not cleared.
- Does not clear any data in the internal input buffer.
- Does not clear downloaded characters.
- Does not clear defined downloaded characters.
- Does not clear defined downloaded bit image.
- Does not clear defined macro.
- Does not clear setting for print density.
- Does not clear baud rate setting of serial port.
- Does not clear valid/invalid setting for panel switch.
- Does not clear ASB mode setting.
- Does not clear setting for auto power-off time.
- Does not clear memory switch.

[Caution] At power off, the following settings are cleared in addition to the items cleared by this command.

- Clears input buffer.
- Clears macro.
- Clears ASB mode setting.
- Clears valid/invalid setting for panel switch.

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + " ! " + CHR$ (&H30) ;  
PRINT #1, CHR$ (&H1B) + "V" + CHR$ (1);  
PRINT #1, "AAA" + CHR$ (&HA);  
PRINT #1, CHR$ (&H1B) + "@";  
PRINT #1, "AAA" + CHR$ (&HA);
```

ESC D [n] k NUL

[Function] Setting Horizontal Tab Position

[Code] <1B>H<44>H [<n>] k<00>H

[Range] {0 =< n =< FF(Hex)}

{0 =< k =< 20(Hex)}

[Outline] Specifying a horizontal tab position.

- “n” indicates the no. of columns from the beginning to the horizontal tab position. At this time, n= set position – 1 is to be specified. For example, to set the position at 9th column, n=8 is to be specified.
- k denotes the number of horizontal tab positions you want to set.
- The tab position is set at position where it is “character width ? ½” from the line beginning. The character width, at this time, includes the rightward space amount. In double wide characters, it is made double of the ordinary case.
- Tab positions can be specified are maximum 32. Specifying exceeding this is ignored. <n> k, which denotes a setting position, is input in the increasing order and ends at <00> H.
- ESC D NUL clears all the set tab positions. Following clearing, horizontal tab command is ignored.

[Caution] When the data, <n> k, is equal to or smaller than its preceding data, <n> k-1, it is assumed that tab setting is finished. If this is the case, the next data onward will be processed as normal data. When the data, <n> k, exceeds a 1-line print area, set the horizontal tab position, assuming “Set digit position = Maximum print digits + 1.” The horizontal tab position does not change even if the character width is altered after setting the horizontal tab position.

[Default] Initial value is specified every eight characters (9th, 17th, 25th, and on) of Font A.

[See Also] HT

ESC E n

[Function] Specifying/canceling highlighting

[Code] <1B>H<45>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] Specifying/canceling the highlighting characters.

- “n” is valid only for the lowest bit (n0).

n (Hex)	Type
0	Specifying highlighting.
1	Canceling highlighting.

- This is effective only for font A (12 ? 24 dots)
- Dot configuration of a highlighted character includes one extra dot added at its side.

[Caution] The double printing and highlighted character printing commands provide exactly the same operation.

[Default] n = 0

[See Also] ESC !

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + “E” + CHR$ (0);
```

```
PRINT #1, “AAABBB” + CHR$ (&HA);
```

```
PRINT #1, CHR$ (&H1B) + “E” + CHR$ (1);
```

```
PRINT #1, “AAABBB” + CHR$ (&HA);
```

ESC G n

[Function] Specifying/canceling Double Printing

[Code] <1B>H<47>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] Specifying/canceling the double printing.

- “n” is valid only for the lowest bit (n0).
- Control by n is shown as follows.

n (Hex)	Type
0	Canceling double printing.
1	Specifying double printing.

- This is effective to font A (12 ? 24 dots)

[Caution] • The print result of Double printing and highlight character printing is completely same.

[Default] n = 0

[See Also] ESC E

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + "G" + CHR$ (0);  
PRINT #1, "AAABBB" + CHR$ (&HA);  
PRINT #1, CHR$ (&H1B) + "G" + CHR$ (1);  
PRINT #1, "AAABBB" + CHR$ (&HA);
```

ESC J n

[Function] Printing and feeding paper n/203 inch

[Code] <1B>H<4A>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] Prints data in the print buffer and feeds paper by n/203 inch.

- This function is temporary and does not affect the feed operation thereafter.
- The beginning of the line is to be considered as the next printing start position.

[Default] Initial value is not defined.

[Sample Program] See Sample Program and Print Results for ESC 2 on Page 18.

ESC M

[Function] Switches between small and normal font

[Code] [1Bh]+[4Dn]+n

[Range] {0 =< n =< 255}

Value	Font
0	A (12X24 p)
1	B (9X16 p)

ESC T

[Function] Printing Diagnostic information

[Code] <1B>H<54>H

[Outline] Prints current printer attributes and diagnostics.

[See Also] ESC .

ESC V n

[Function] Specifying/Canceling 90°-right- turned Characters

[Code] <1B>H<56>H<n>

[Range] {0 =< n =< 1(Hex)}

[Outline] Specifying/canceling characters 90°-right- turned character.

“n” means the followings.

n (Hex)	Condition
0	Canceling 90°-right- turned Characters
1	Specifying 90°-right- turned Characters

[Caution] No underlines are attached to 90°-right- turned characters.

[Default] The initial value of n is “0”.

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + “V” + CHR$ (0);
```

```
PRINT #1, “AAAAA”;
```

```
PRINT #1, CHR$ (&H1B) + “V” + CHR$ (1);
```

```
PRINT #1, “AAAAA” + CHR$ (&HA);
```

ESC Y n

[Function] Specifying print density

[Code] <1B>H<59>H<n>

[Range] {0 =< n =< 5(Hex)}

[Outline] Specifies the print density.

“n” means the followings.

n (Hex)	Condition
0	70 % density
1	80 % density
2	90 % density
3	100 % density
4	120 % density
5	150 % density

[Caution] Higher density may lead to slower printing.

Increasing the density allows Low battery to be detected easily.

[Default] n = 3 (100%)

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + “Y” + CHR$ (0);
```

```
PRINT #1, “AAAAA”;
```

```
PRINT #1, CHR$ (&H1B) + “Y” + CHR$ (5);
```

```
PRINT #1, “AAAAA” + CHR$ (&HA);
```

ESC Z

[Function] Returning diagnostic information

[Code] <1B>H<5A>H

[Outline] The printer will return 32 bytes of information with the following structure:

1-22: Printer name up to 22 characters.

23-25: Firmware version

26-27: Language version, described by two characters.

28-32: When the corresponding bit is 1, the function is supported and when 0, the function is not supported.

BYTE	Meaning
28.0	IrDA support
28.1	Mag stripe reader
28.2	Reserved
28.3	Katakana support (Characters after 80(Hex))
28.4	JIS and SHIFT-JIS Support
28.5	Prints with ESC . and ESC T and Returns with ESC ' the thermal head temperature in Farenheit.
28.6	Not in use
28.7	Reserved, always 1
29.0	Update via firmware interface
29.1	Not in use
29.2	Not in use
29.3	Not in use
29.4	Not in use
29.5	Not in use
29.6	Not in use
29.7	Reserved, always 1
30.0	Not in use
30.1	Not in use
30.2	Not in use
30.3	Not in use
30.4	Not in use
30.5	Not in use
30.6	Not in use
30.7	Reserved, always 1
31.1	Not in use
31.2	Not in use
31.3	Not in use
31.4	Not in use
31.5	Not in use
31.6	Not in use
31.7	Reserved, always 1
32.0	Not in use
32.1	Not in use
32.2	Not in use
32.3	Not in use
32.4	Not in use
32.5	Not in use
32.6	Not in use
32.7	Reserved, always 1

ESC \ n1 n2

[Function] Specifying the relative positions

[Code] <1B>H<5C>H<n1><n2>

[Range] {0 =< n1 =< 255}

{0 =< n2 =< 255}

[Outline] The printing start position is specified in the number of dots (1/203 inch unit) from the current position.

- Rightward direction is taken as plus and leftward direction as minus.
- To specify N dot in minus (left) direction, use a complement of N for assignment.
- $N \text{ dots} = 65536 - N$
- The number of dots is divided by 256, whose quotient is taken as n2 and the residual as n1.

[Caution] Specifying exceeding the top of line or the end of line is ignored.

[See Also] ESC \$

[Sample Program] See Sample Program and Print Results for ESC \$.

ESC a n

[Function] Aligning the characters

[Code] <1B>H<61>H<n>

[Range] {0 =< n =< 2(Hex)}

[Outline] All the printed data within one line are aligned in the specified position. Depending on n value, positional alignment is carried out as in the table below:

n (Hex)	Position
0	Left end alignment
1	Centering
2	Right end alignment

[Caution] • This is valid only when n is inputted at the beginning of line.

- The initial value of n is "0".

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + "a" + CHR$ (0);
```

```
PRINT #1, "AAAAA" + CHR$ (&HA);
```

```
PRINT #1, CHR$ (&H1B) + "a" + CHR$ (1);
```

```
PRINT #1, "AAAAA" + CHR$ (&HA);
```

```
PRINT #1, CHR$ (&H1B) + "a" + CHR$ (2);
```

```
PRINT #1, "AAAAA" + CHR$ (&HA);
```

ESC c5 n

[Function] Enabling/Disabling Panel Switches

[Code] <1B>H<63>H<35>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] Selecting the LF switch valid/invalid.

•“n” is valid only in the lowest bit (n0).

n (Hex) Condition

0 LFSW valid.

1 LFSW invalid.

[Caution] • When the panel switch is disabled with this command, the LF switch is disabled.

Therefore, the paper cannot be fed by operating the LF switch.

• When LF switch is set to Invalid, it cannot be cleared by the Initialize command (ESC @) but is cleared by turning power off.

[Default] The initial value of n is “0”.

[Sample Program]

PRINT #1, CHR\$ (&H1B) + “c5” + CHR\$ (0);When enabling the LF switch

PRINT #1, CHR\$ (&H1B) + “c5” + CHR\$ (1);When disabling the LF switch

ESC d n

[Function] Printing and Feeding the paper by n lines

[Code] <1B>H<64>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] Prints data inside the buffer and feeds paper by n lines.

Specified line does not remain.

The beginning of the line is to be considered as the next printing start position.

[Default] The initial value is not defined.

[Sample Program]

PRINT #1, “AAAAA”

PRINT #1, CHR\$ (&H1B) + “d” + CHR\$ (2);

PRINT #1, “AAAAA” + CHR\$ (&HA);

ESC i

[Function] Cutting the paper

[Code] <1B>H<69>H

The paper is moved so the last printed raw remains after the cutter.

The command is fulfilled only if it is put at the beginning of the raw.

ESC m

[Function] Cutting the paper

[Code] <1B>H<6D>H

Analogical as ESC i

ESC p

[Function] Generating of impulse opening the drawer

[Code] [1Bh] + [70h] + m + n1 + n2

The value of m is meaningless.

n1 and n2 are between 0 to 255.

The printer generates impulse $2 \cdot n1$ msec and $2 \cdot n2$ msec waiting interval after the impulse.

It is necessary $n2 > 4 \cdot n1$, otherwise impulse is not generated.

ESC v

[Function] Transmitting the printer status

[Code] <1B>H<76>H

[Outline] Current printer status is transmitted.

Status sent out consists of 1 byte whose content is as in the table below.

Bit	Function	Value	
		0	1
0	Not defined		
1	Not defined		
2	Paper or paper cover	With paper and cover closed	No paper or cover open
3	Not defined		
4	Not used	Fixed to 0	—
5	Not defined		
6	Not defined		
7	Not defined		

[Caution] In paper end (paper near end) status, this command may be unreceptible state due to BUSY state.

[Sample Program]

```
OPEN "COM1:N81NN" AS #1 ;  
PRINT #1, CHR$ (&H1B) ; "v" ;  
A$ = INPUT$ (1, #1) ;  
CLOSE #1  
END
```

ESC { n

[Function] Specifying/Canceling the Inverted Characters

[Code] <1B>H<7B>H<n>

[Range] {0 =< n =< FF(Hex)}

[Outline] Specifying/canceling inverted characters.

•“n” is valid only for the lowest bit (n0).

n (Hex)	Condition
0	Canceling inverted characters.
1	Specifying inverted characters.

[Caution] • Inverted-printing means printing the line at 180° turned.

• This is valid only when this is specified at the beginning of a line.

[Default] • The initial value of n is “0”.

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + “{” + CHR$ (0);  
PRINT #1, “AAAAA” + CHR$ (&HA);  
PRINT #1, “BBBBB” + CHR$ (&HA);  
PRINT #1, CHR$ (&H1B) + “{” + CHR$ (1);  
PRINT #1, “AAAAA” + CHR$ (&HA);  
PRINT #1, “BBBBB” + CHR$ (&HA);
```

GS k m [d1 dk] NUL

GS k m n [d1 dn]

[Function] Printing the bar code

[Code] (1) <1D>H<6B>H<m> [d1.....dk] NUL

(2) <1D>H<6B>H<m><n> [d1.....dn]

[Range] (1) 0 =<m=< 6 The definitions of "k" and "d" vary with the bar code system.

(2) 65 =<m=< 73 The definitions of "n" and "d" vary with the bar code system.

[Outline] Selects a bar code system and prints the bar code.

For (1):

m	Bar code system	Range of "n"	Range of "d"
0	UPC-A	11	48=< d=< 57
1	UPC-E	11	48 =<d=< 57
2	JAN13 (EAN)	12	48 =<d=< 57
3	JAN8 (EAN)	7	48 =<d=< 57
4	CODE39	–	48=<d=< 57, 65=< d=< 90 32, 36, 37, 43, 45, 46, 47
5	ITF	(Must be an even number)	48=< d=< 57
6	CODABAR (NW-7)	–	48=<d=< 57, 65=< d=< 68 36, 43, 45, 46, 47, 58

For (2):

m(2)	Bar code system	Range of "n"	Range of "d"
65	UPC-A	11	48 =<d=< 57
66	UPC-E	11	48 =<d=< 57
67	EAN13 (JAN)	12	48 =<d=< 57
68	EAN8 (JAN)	7	48 =<d=< 57
69	CODE39	–	48 =<d=< 57, 65=< d=< 90 32, 36, 37, 43, 45, 46, 47
70	ITF (Must be an 48=< d=<57even number)
71	CODABAR	–	48 =<d=< 57, 65=< d=< 68 36, 43, 45, 46, 47,58
72	CODE93	–	0 =<d=< 127
73	CODE128	–	0=<d=< 127

GS w n

[Function] Selecting the horizontal size (Scale factor) of the Bar Code

[Code] <1D>H<77>H<n>

[Range] {2 =< n =< 4(Hex)}

[Outline] Selecting bar code width.

n denotes the number of dots in fine element width.

[Default] The initial value of this width is "3".

[Sample Program]

```
PRINT #1, CHR$ (&H1D) + "h" + CHR$ (30);
PRINT #1, CHR$ (&H1D) + "w" + CHR$ (2);
GOSUB BC
PRINT #1, CHR$ (&H1D) + "h" + CHR$ (50);
PRINT #1, CHR$ (&H1D) + "w" + CHR$ (3);
GOSUB BC
PRINT #1, CHR$ (&H1D) + "h" + CHR$ (80);
PRINT #1, CHR$ (&H1D) + "w" + CHR$ (4);
GOSUB BC
END
BC:
PRINT #1, CHR$ (&H1D) + "k";
PRINT #1, CHR$ (4);
PRINT #1, "12" + CHR$ (0);
RETURN
```

GS h n

[Function] Selecting the height of the Bar Code

[Code] <1D>H<68>H<n>

[Range] {1 =< n =< FF(Hex)}

[Outline] Selecting bar code height.

n denotes the number of dots in the vertical direction.

[Default] n = a2(Hex) 162 dots

[Sample Program] See Sample Program and Print Results for GS w .

GS H n

[Function] Selecting of Printing Position of HRI Code

[Code] <1D>H<48>H<n>

[Range] {0 =< n =< 3(Hex)}

[Outline] Selecting printing position of HRI code in printing bar codes.

•“n” means the followings.

n (Hex)	Printing Position
0	No printing
1	Above the bar code
2	Below the bar code
3	Both above and below the bar code

The HRI code refers to the bar code-turned characters so that you can read them.

[Caution] The HRI code is printed in the font selected with GS f. Specify before the GS k command.

[Default] The initial value of n is “0”.

[See Also] GS f

[Sample Program]

```
PRINT #1, CHR$ (&H1B) + “3” + CHR$ (5);
PRINT #1, CHR$ (&H1D) + “h” + CHR$ (50);
PRINT #1, CHR$ (&H1D) + “H” + CHR$ (0);
GOSUB BC
PRINT #1, CHR$ (&H1D) + “H” + CHR$ (1);
GOSUB BC
PRINT #1, CHR$ (&H1D) + “H” + CHR$ (2);
GOSUB BC
PRINT #1, CHR$ (&H1D) + “H” + CHR$ (3);
GOSUB BC
END
BC:
PRINT #1, CHR$ (&H1D) + “k”;
PRINT #1, CHR$ (4);
PRINT #1, “12” + CHR$ (0);
PRINT #1, CHR$ (&HA);
RETURN
```

GS f n

[Function] Selecting the font of HRI characters

[Code] <1D>H<66>H<n>

[Range] {n = 0, 1(Hex)}

[Outline] Selecting the font of HRI characters in printing bar code.

The type of font can be selected with “n” as follows:

n (Hex)	Font
0	Font A (12 ? 24)
1	Font B (9 ? 16)

The HRI characters refer to the bar code-turned characters so that you can read them.

[Caution] The HRI characters are printed at the position specified with GS H.

[Default] n = 0

[See Also] GS H

[Sample Program]

```
PRINT #1, CHR$(&H1D) + "h" + CHR$(50);
PRINT #1, CHR$(&H1D) + "H" + CHR$(2);
PRINT #1, CHR$(&H1D) + "f" + CHR$(0);
GOSUB BC
PRINT #1, CHR$(&H1D) + "F" + CHR$(1);
GOSUB BC
END
BC:
PRINT #1, CHR$(&H1D) + "k";
PRINT #1, CHR$(4);
PRINT #1, "12" + CHR$(0);
PRINT #1, CHR$(&HA);
RETURN
```


GS * n1 n2 [d] n1 n2 D1 Dn

[Function] Defining the Download Bit Image (LOGO)

[Code] <1D>H<2A>H<n1><n2> [< d >]

[Range] {1 =< n1 =< 7F} Defines horizontal size of downloaded image.

{1 =< n2 =< F8} Defines the vertical size of downloaded image.

[Outline] Defines downloading bit images of the number of dots specified by n1 and n2.

The numbers of dots are n1 ? 8 in horizontal direction and n2 ? 8 in vertical direction.

The number of horizontal bytes can be up to 7F, but only the first 48 (30H) bytes will be printed. The rest will be rejected.

- d indicates bit image data.

- The downloaded bit image thus defined is valid till it is redefined or ESC _ is executed. After power off, the downloaded bit image is maintained even by the execution of Initialize command.

[Caution] • This downloaded bit image is also printed at the self-test printing.

- The maximum size of the Bit image cannot exceed 16 KB.

[See Also] GS /

[Sample Program]

```
GOSUB IMG
```

```
PRINT #1, CHR$ (&H1D) + "/" + CHR$ (0);
```

```
PRINT #1, CHR$ (&H1D) + "/" + CHR$ (1);
```

```
PRINT #1, CHR$ (&H1D) + "/" + CHR$ (2);
```

```
PRINT #1, CHR$ (&H1D) + "/" + CHR$ (3);
```

```
END
```

```
IMG:
```

```
n 1 = 10 : n 2= 50
```

```
PRINT #1, CHR$ (&H1D) + "***";
```

```
PRINT #1, CHR$ (n1) + CHR$ (n2);
```

```
FOR I=1 TO n2
```

```
FOR J=1 TO n1
```

```
PRINT #1, CHR$ (J);
```

```
NEXT I
```

```
NEXT J
```

```
RETURN
```

GS / m

[Function] Printing the Download, Bit Image

[Code] <1D>H<2F>H<m>

[Range] {0 =< m =< 03(Hex)}

[Outline] Prints download bit image in a mode specified by m.

- Modes can be selected by m are shown below.

m (Hex)	Name	Dot Density in Vertical Direction	Dot Density in Horizontal Direction
0	Normal mode	203 DPI	203 DPI
1	Double wide mode	203 DPI	101 DPI
2	Double high mode	101 DPI	203 DPI
3	Double wide/ double high mode	101 DPI	101 DPI

[Caution] • When data exist inside the print buffer, this command is ignored.

- When a download bit image has not been defined, this command is ignored.

- A portion of a download bit image exceeding one line length is not printed.

[See Also] GS *

[Sample Program] Refer to the programming example and print result of GS *.

GS :

[Function] Starting/Ending Macro Definition

[Code] <1D>H<3A>H

[Outline] Specifying starting/ending macro definition.

Means termination when received while defining a macro.

[Caution] • Maximum content available for macro definition is 3328 bytes.

- A portion exceeding 3328 bytes is not defined.
- Even with ESC @ (initialization of the printer) having been executed, defined content is not cleared. Therefore, it is possible to include ESC @ into the content of macro definition.
- Normal printing operation is carried out even while in macro definition.

[Default] Initially, Macro is not specified.

[See Also] GS ^

[Sample Program]

```
PRINT #1, CHR$ (&H1D) + “: “ ;  
PRINT #1, “+—+” + CHR$ (&HA);  
PRINT #1, “ | | “ + CHR$ (&HA);  
PRINT #1, “+—+” + CHR$ (&HA);  
PRINT #1, CHR$ (&H1D) + “: “;  
PRINT #1, CHR$ (&H1D) + “ ^ “;  
PRINT #1, CHR$ (2) + CHR$ (10);  
PRINT #1, CHR$ (0);
```

GS ^ n1 n2 n3

[Function] Executing the Macro

[Code] <1D>H<5E>H<n1><n2><n3>

[Range] {0 =< n1 =< FF(Hex)}

{0 =< n2 =< FF(Hex)}

{0 =< n3 =< 1(Hex)}

[Outline] Executing contents defined in macro.

The following items are set for parameters n1 to n3.

n1: The number of times of macro execution

n2: Waiting time on macro execution. Waiting time of n2 ? 100 msec is given for every execution.

n3: Macro execution mode

n (Hex)	Mode
0	Continuous execution
1	Execution by LF switch

Continuous execution: The Macro is executed n1 times continuously at the time intervals specified by n2.

Execution by LF switch: After waiting for lapse of time specified by n2, the LF switch is waited to be pressed. When it is pressed, the macro is executed once.

This action is repeated n1 times.

[Caution] • When this command is received while in macro definition, suspension of macro definition is indicated. At this time, the defined content is cleared.

- No execution takes place when macro is held undefined or n1=0.
- While in macro execution with n3=1, paper feed with the LF switch is not available.
- During execution of a macro, the power cannot be turned off.

[See Also] GS :

[Sample Program] See Sample Program and Print Results for GS .: