# Table of Contents

1. Foreword .........................................................................................................................4
   1-1 Image Buffer Configuration ........................................................................................5
   1-2 Information for calculating position on image buffer ..................................................6
   1-3 Command List ..............................................................................................................8
   1-4 Programming Considerations .......................................................................................10

2. Detail Description .............................................................................................................11
   2-1 Commands for Designing a Label ................................................................................11
      2-1-1 T (Text String) .....................................................................................................12
      2-1-2 V (Text String Vector Font) ................................................................................15
      2-1-3 B1 (1 Dimensional bar code) ..............................................................................18
      2-1-4 B2 (2 Dimensional bar code) ..............................................................................21
      2-1-5 B3 (Special Barcode) ...........................................................................................25
      2-1-6 BD (Block Draw) ..................................................................................................26
      2-1-7 CD (Circle Draw) ................................................................................................29
      2-1-8 CS (Character Set selection) ...............................................................................30
      2-1-9 P (Print) .............................................................................................................33
   2-2 Media & Buffer Related Commands ...........................................................................35
      2-2-1 ST (Set Printing Type) .........................................................................................36
      2-2-2 SM (Set Margin) ..................................................................................................36
      2-2-3 SF (Set Back-Feed Option) ................................................................................38
      2-2-4 SL (Set Length) ...................................................................................................39
      2-2-5 SW (Set Width) ....................................................................................................42
      2-2-6 SB (Set Buffer mode) .........................................................................................43
      2-2-7 CB (Clear Buffer) ................................................................................................43
   2-3 Printer Setting Commands ...........................................................................................44
      2-3-1 SS (Set Speed) .....................................................................................................45
      2-3-2 SD (Set Density) ..................................................................................................45
      2-3-3 SO (Set Orientation) ............................................................................................46
      2-3-4 SP (Set Port) .......................................................................................................47
      2-3-5 SA (Set Offset) ....................................................................................................48
      2-3-6 TA (Tear-off/Cutter Position Setting) ..................................................................48
   2-4 Variable Related Commands .......................................................................................49
      2-4-1 SC (Set Counter) ..................................................................................................50
      2-4-2 AC (Auto Counter) .............................................................................................51
      2-4-3 SV (Set Variable) ................................................................................................52
      2-4-4 ? (Get Variables) ...............................................................................................53
      2-4-5 PV (Print with Variables) ...................................................................................54
   2-5 Template Related Commands .......................................................................................55
      2-5-1 TS (Template store Start) ....................................................................................56
      2-5-2 TE (Template store End) .....................................................................................56
      2-5-3 TR (Template Recall) ..........................................................................................57
      2-5-4 TD (Template Delete) ........................................................................................58
      2-5-5 TI (Template Information) ...................................................................................58

Rev. 1.27
2-6 Image Related Commands ........................................................................59
2-6-1 IS (Image Store) .................................................................................60
2-6-2 IR (Image Recall) ................................................................................60
2-6-3 ID (Image Delete) ................................................................................61
2-6-4 II (Image Information) ......................................................................61
2-6-5 LD .........................................................................................................62
2-6-6 LC .........................................................................................................64
2-6-7 BMP ....................................................................................................66

2-7 Downloadable font Related Commands ...................................................67
2-7-1 DT (Download True Type Font) .........................................................68
2-7-2 DD (Downloaded font Delete) ..............................................................69
2-7-3 DI (Downloaded font Information) ......................................................70

2-8 The Others ............................................................................................71
2-8-1 @ (Initialize Printer) ...........................................................................72
2-8-2 PI (Printer Information) ...................................................................72
2-8-3 CUT (Auto-cutter Enable/Disable) ....................................................73
2-8-4 ^cp (Check Printer Status and Report 2 bytes) ...............................74
2-8-5 ^cu (Check Printer Status and Report 1 byte) .................................75
2-8-6 ^PI (Send Printer information to host) ..............................................76

3. Programming Example .............................................................................77
3-1 Example) T_resident ..............................................................................77
3-2 Example) T_Rotate4 ..............................................................................78
3-3 Example) V_resident ...........................................................................79
3-4 Example) V_Rotate4 ..............................................................................80
3-5 Example) Code39 ..................................................................................81
3-6 Example) BD1 .......................................................................................82
3-7 Example) BD3 .......................................................................................83
3-8 Example) BD4 .......................................................................................85
3-9 Example) BD5 .......................................................................................86
3-10 Example) Slope ...................................................................................87
3-11 Example) SW&SL ...............................................................................88
3-12 Example) TEST00_TS .......................................................................90
3-13 Example) TEST00_TR ......................................................................91
3-14 Example) TEST04_TS .......................................................................92
3-15 Example) TEST04_TR ......................................................................93
3-16 Example) IR1 ......................................................................................93
3-17 Example) TEST10_TS .......................................................................94
3-18 Example) TEST10_TR ......................................................................94
3-19 Example) TEST11_TS .......................................................................95
3-20 Example) TEST11_TR ......................................................................95
3-21 Example) SLCS_BIXOLON ................................................................96
1. Foreword

In this chapter, the basic concept of SLCS and some information necessary for the programmer to use SLCS will be explained. Please read this part before starting programming for efficient and easy use of BIXOLON Label Printers.

We at BIXOLON maintain ongoing efforts to enhance and upgrade the functions and quality of all our products. In following, product specifications and/or user manual content may be changed without prior notice.
1-1 Image Buffer Configuration

1) Maximum size
   A) When using Double Buffering Function
      832dots × 1216dots (104mm × 152mm) = 4 inch × 6 inch
   B) When using Single Buffering Function
      832dots × 2432dots (104mm × 304mm) = 4 inch × 12 inch

2) Dot size: 0.125mm(W) × 0.125mm(H) (203dpi)
1-2 Information for calculating position on image buffer

1) Relation between position and number of dots

<table>
<thead>
<tr>
<th>Inch</th>
<th>mm</th>
<th>dots</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.04</td>
<td>1</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>0.40</td>
<td>10.00</td>
<td>80</td>
<td></td>
</tr>
<tr>
<td>1.00</td>
<td>25.40</td>
<td>203</td>
<td>203 dpi</td>
</tr>
<tr>
<td>1.25</td>
<td>31.75</td>
<td>254</td>
<td></td>
</tr>
<tr>
<td>1.50</td>
<td>38.10</td>
<td>305</td>
<td></td>
</tr>
<tr>
<td>1.75</td>
<td>44.45</td>
<td>355</td>
<td></td>
</tr>
<tr>
<td>2.00</td>
<td>50.80</td>
<td>406</td>
<td></td>
</tr>
<tr>
<td>2.25</td>
<td>57.15</td>
<td>457</td>
<td></td>
</tr>
<tr>
<td>2.50</td>
<td>63.50</td>
<td>508</td>
<td></td>
</tr>
<tr>
<td>2.75</td>
<td>69.85</td>
<td>556</td>
<td></td>
</tr>
<tr>
<td>3.00</td>
<td>76.20</td>
<td>610</td>
<td></td>
</tr>
<tr>
<td>4.00</td>
<td>101.6</td>
<td>813</td>
<td></td>
</tr>
</tbody>
</table>

2) Font Information

<table>
<thead>
<tr>
<th>Font name</th>
<th>Width × Height (dots)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>09 ×15</td>
</tr>
<tr>
<td>1</td>
<td>12 ×20</td>
</tr>
<tr>
<td>2</td>
<td>16 ×25</td>
</tr>
<tr>
<td>3</td>
<td>19 ×30</td>
</tr>
<tr>
<td>4</td>
<td>24 ×38</td>
</tr>
<tr>
<td>5</td>
<td>32 ×50</td>
</tr>
<tr>
<td>6</td>
<td>48 ×76</td>
</tr>
<tr>
<td>7</td>
<td>22 ×34</td>
</tr>
<tr>
<td>8</td>
<td>28 ×44</td>
</tr>
<tr>
<td>9</td>
<td>37 ×58</td>
</tr>
<tr>
<td>Korean a</td>
<td>16 ×16(ascii:9×15)</td>
</tr>
<tr>
<td>Korean b</td>
<td>24 ×24(ascii:12×24)</td>
</tr>
<tr>
<td>Korean c</td>
<td>20 ×20(ascii:12×20)</td>
</tr>
<tr>
<td>Korean d</td>
<td>26 ×26(ascii:16×30)</td>
</tr>
<tr>
<td>Korean e</td>
<td>20 ×26(ascii:16×30)</td>
</tr>
<tr>
<td>Korean f</td>
<td>38 ×38(ascii:22×34)</td>
</tr>
<tr>
<td>GB2312 m</td>
<td>24 ×24(ascii:12×24)</td>
</tr>
<tr>
<td>BIG5 n</td>
<td>24 ×24(ascii:12×24)</td>
</tr>
<tr>
<td>Vector</td>
<td>Scalable</td>
</tr>
</tbody>
</table>
3) Example of text and barcode

![Example of text and barcode](image)

4) Example of rotation

![Example of rotation](image)
# 1-3 Command List

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Remarks</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Commands for Designing Label</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>Text</td>
<td>Draw text string on the image buffer</td>
<td></td>
</tr>
<tr>
<td>V</td>
<td>Text (Vector Font)</td>
<td>Draw text string on the image buffer</td>
<td></td>
</tr>
<tr>
<td>B1</td>
<td>1d barcode</td>
<td>Draw 1D Barcode on the image buffer</td>
<td></td>
</tr>
<tr>
<td>B2</td>
<td>2d barcode</td>
<td>Draw 2D Barcode on the image buffer</td>
<td></td>
</tr>
<tr>
<td>B3</td>
<td>Special barcode</td>
<td>Draw special barcode on the image buffer</td>
<td></td>
</tr>
<tr>
<td>BD</td>
<td>Block Draw</td>
<td>Draw line or box on the image buffer</td>
<td></td>
</tr>
<tr>
<td>CD</td>
<td>Circle Draw</td>
<td>Draw circle on the image buffer</td>
<td></td>
</tr>
<tr>
<td>CS</td>
<td>Character Set selection</td>
<td>Select international code table</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>Print</td>
<td>Start printing the content of image buffer</td>
<td></td>
</tr>
<tr>
<td><strong>2. Media &amp; Buffer related Commands</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ST</td>
<td>Set Print Type</td>
<td>Select Thermal Direct / Transfer printing</td>
<td></td>
</tr>
<tr>
<td>SM</td>
<td>Set Margin</td>
<td>Set the marginal value of the image buffer</td>
<td></td>
</tr>
<tr>
<td>SF</td>
<td>Set Back-feed Option</td>
<td>Set back-feeding option</td>
<td></td>
</tr>
<tr>
<td>SL</td>
<td>Set Label Length</td>
<td>Set length of label</td>
<td></td>
</tr>
<tr>
<td>SW</td>
<td>Set Label Width</td>
<td>Set length of label</td>
<td></td>
</tr>
<tr>
<td>SB</td>
<td>Set Buffer mode</td>
<td>Enable or Disable double buffering function</td>
<td></td>
</tr>
<tr>
<td>CB</td>
<td>Clear Buffer</td>
<td>Clear image buffer</td>
<td></td>
</tr>
<tr>
<td><strong>3. Printer Setting Commands</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SS</td>
<td>Set Speed</td>
<td>Set printing speed</td>
<td></td>
</tr>
<tr>
<td>SD</td>
<td>Set Density</td>
<td>Set printing density from level 0 to 20</td>
<td></td>
</tr>
<tr>
<td>SO</td>
<td>Set Orientation</td>
<td>Set printing direction</td>
<td></td>
</tr>
<tr>
<td>SP</td>
<td>Set serial Port</td>
<td>Set serial port configurations</td>
<td></td>
</tr>
<tr>
<td>SA</td>
<td>Set Offset</td>
<td>Set offset value</td>
<td></td>
</tr>
<tr>
<td>TA</td>
<td>Set Tear-off/Cut</td>
<td>Set Tear-off/Cut value</td>
<td></td>
</tr>
<tr>
<td><strong>4. Variable related Commands</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SC</td>
<td>Set Counter</td>
<td>Used in Template sequence</td>
<td></td>
</tr>
<tr>
<td>AC</td>
<td>Set Counter</td>
<td>Used in normal mode</td>
<td></td>
</tr>
<tr>
<td>SV</td>
<td>Set Variable</td>
<td>Used in Template sequence</td>
<td></td>
</tr>
<tr>
<td>?</td>
<td>Get variables</td>
<td>Get content of variables and counters</td>
<td></td>
</tr>
<tr>
<td>PV</td>
<td>Print with Variable</td>
<td>Use this command in Template</td>
<td></td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
<td>Remarks</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>---------</td>
<td></td>
</tr>
</tbody>
</table>
| **5. Template Related Commands**
| TS | Template store Start | All contents between these commands are saved in printer memory |
| TE | Template store End |
| TR | Template Recall | Load and reuse the stored Template |
| TD | Template Delete | Delete stored Template from printer memory |
| TI | Template Information | Print the list of currently stored Templates |

| **6. Image Data Related Commands**
| IS | Image Store | PCX format image file can be stored |
| IR | Image Recall | Load and reuse the stored image |
| ID | Image Delete | Delete stored image |
| II | Image Information | Print the list of currently stored images |
| LD | Bitmap data draw | Draw bitmap image data on the image buffer |
| LC | Compression bitmap data draw | Draw compression bitmap image data on specific position of image buffer |
| BMP | BMP format file draw | Draw BMP format file on the image buffer |

| **7. Downloadable Font Related Commands**
| DT | Download True Type font | Windows system font used |
| DD | Downloadable font Delete | Delete downloaded font |
| DI | Downloadable font Information | Print the list of currently stored images |

| **8. The Others**
| @ | Reset printer | Initialize the printer |
| PI | Printer Information | Print current setting of printer |
| CUT | Enable/Disable Cutter option | Cutting is executed after Printing is finished if cutting option is enabled by this command |
| ^cp | Check Printer Status | Return 2 bytes status values to host |
| ^cu | Check Printer Status | Return 1 byte status value to host |
| ^PI | Send Printer information | Send various information to host |
1-4 Programming Considerations

1) All commands are case-sensitive and some commands require one or more parameters and ‘Data’.

2) Command Conventions

```
Command
Tp1,p2,p3,p4,p5,p6,p7,p8,p9,'TEXT DATA'
```

Parameters  Command Specific Data

3) Each command line must be terminated with a ‘CR’(0Dh, 13) + ‘LF’(0Ah,10).

4) The commands which draw text, barcode, lines… just draw on the image buffer, they do not start printing. The printer will start printing when the P command comes.

---

! Caution

The ‘P’ command must be terminated by ‘CR’(0x0d). If not, the printer will not start printing until ‘CR’ comes.
2. Detail Description

2-1 Commands for Designing a Label

These commands are used to design a label by providing text, barcode, line, box... and to print content of image buffer on media.

1) T  
   Draw Text String on the image buffer.

2) V  
   Draw Text (Vector Font) String on the image buffer.

3) B1  
   Draw 1D Barcode on the image buffer.

4) B2  
   Draw 2D Barcode on the image buffer.

5) B3  
   Draw Special Barcode on the image buffer.

6) BD  
   Draw Line, Block, Box & Slope on the image buffer.

7) CD  
   Draw Circle on the image buffer.

8) CS  
   Set Code page and ICS(International Character Set).

9) P  
   Start printing the content of the image buffer.
2-1-1 T (Text String)

Description
Draw text string on the image buffer

Syntax
T p1,p2,p3,p4,p5,p6,p7,p8,p9,(p10),"DATA"

Parameters
p1 : Horizontal position (X) [dot]
p2 : Vertical position (Y) [dot]
p3 : Font selection

<table>
<thead>
<tr>
<th>Value</th>
<th>Font Size(pt)</th>
<th>Width × Height(dots)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>6</td>
<td>9 × 15</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>12 × 20</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>16 × 25</td>
</tr>
<tr>
<td>3</td>
<td>12</td>
<td>19 × 30</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>24 × 38</td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>32 × 50</td>
</tr>
<tr>
<td>6</td>
<td>30</td>
<td>48 × 76</td>
</tr>
<tr>
<td>7</td>
<td>14</td>
<td>22 × 34</td>
</tr>
<tr>
<td>8</td>
<td>18</td>
<td>28 × 44</td>
</tr>
<tr>
<td>9</td>
<td>24</td>
<td>37 × 58</td>
</tr>
<tr>
<td>a</td>
<td>KOREAN 1</td>
<td>16 × 16 (ascii 9×15)</td>
</tr>
<tr>
<td>b</td>
<td>KOREAN 2</td>
<td>24 × 24 (ascii 12×24)</td>
</tr>
<tr>
<td>c</td>
<td>KOREAN 3</td>
<td>20 × 20 (ascii 12×20)</td>
</tr>
<tr>
<td>d</td>
<td>KOREAN 4</td>
<td>26 × 26 (ascii 16×30)</td>
</tr>
<tr>
<td>e</td>
<td>KOREAN 5</td>
<td>20 × 26 (ascii 16×30)</td>
</tr>
<tr>
<td>f</td>
<td>KOREAN 6</td>
<td>38 × 38 (ascii 22×34)</td>
</tr>
<tr>
<td>m</td>
<td>GB2312</td>
<td>24 × 24 (ascii 12×24)</td>
</tr>
<tr>
<td>n</td>
<td>BIG5</td>
<td>24 × 24 (ascii 12×24)</td>
</tr>
<tr>
<td>j</td>
<td>Shift JIS</td>
<td>24 × 24 (ascii 12×24)</td>
</tr>
</tbody>
</table>

*A to Z are assigned to Downloadable font. Refer to DT command.*

p4 : Horizontal multiplier : 1 ~ 4
p5 : Vertical multiplier : 1 ~ 4
p6 : Right-side character spacing [dot]

Plus(+)/Minus(-) option can be used. Ex) 5, +3, -10...

p7 : Rotation

<table>
<thead>
<tr>
<th>Value</th>
<th>Rotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No Rotation</td>
</tr>
<tr>
<td>1</td>
<td>90 degrees</td>
</tr>
<tr>
<td>2</td>
<td>180 degrees</td>
</tr>
<tr>
<td>3</td>
<td>270 degrees</td>
</tr>
</tbody>
</table>

p8 : Reverse printing

N : Normal printing  R : Reverse printing
p9 : Bold
N : Normal
B : Bold
p10 : Text Alignment(Optional)
F : p1 means the position of the first character in text string - **Left alignment**
L : p1 means the position of the last character in text string - **Right alignment**
R: Write text string form right to left.
(BIXOLON → NOLOXIB)

- **This parameter is for alignment of text lines.**

DATA : The various data types can be used in the data field as followings.
1) Fixed text string : ' Text String'
2) Variables declared in template by SV command : Vnn
3) Counters declared by the SC command : Cn
- **1) , 2) and 3) can be mixed together**

**Example**

T50,100,3,1,1,0,0,N,N,' BIXOLON Label Printer'
T50,100,3,1,1,0,0,N,N,'Manufacturer : V00
T50,100,3,1,1,0,0,N,N,V00
T50,100,3,1,1,0,0,N,N,'Manufacturer : C0
T50,100,3,1,1,0,0,N,N,C0

- **If you want to print ' or \ then you must type like \ or \.**
Example

SM20,20
T26,20,0,0,0,0,N,N,'Font- 6 pt'
T26,49,1,0,0,0,N,N,'Font - 8 pt'
T26,81,2,0,0,0,N,N,'Font - 10 pt'
T26,117,3,0,0,0,N,N,'Font - 12 pt'
T26,156,4,0,0,0,N,N,'Font - 15 pt'
T26,200,5,0,0,0,N,N,'Font - 20 pt'
T26,252,6,0,0,0,N,N,'Font - 30 pt'
P1

Result

<table>
<thead>
<tr>
<th>Font - 6 pt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Font - 8 pt</td>
</tr>
<tr>
<td>Font - 10 pt</td>
</tr>
<tr>
<td>Font - 12 pt</td>
</tr>
<tr>
<td><strong>Font - 15 pt</strong></td>
</tr>
<tr>
<td><strong>Font - 20 pt</strong></td>
</tr>
<tr>
<td><strong>Font - 30 pt</strong></td>
</tr>
</tbody>
</table>
Description
Draw text (Vector Font) string on the image buffer

Syntax
\texttt{Vp1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11,p12, 'DATA'}

Parameters
\begin{itemize}
  \item \texttt{p1} : Horizontal position (X) [dot]
  \item \texttt{p2} : Vertical position (Y) [dot]
  \item \texttt{p3} : Font selection
    \begin{itemize}
      \item U: ASCII (1Byte code)
      \item K: KS5601 (2Byte code)
      \item B: BIG5 (2Byte code)
      \item G: GB2312 (2Byte code)
      \item J: Shift-JIS (2Byte code)
      \item a: OCR-A (1Byte code)
      \item b: OCR-B (1Byte code)
    \end{itemize}
  \item \texttt{p4} : Font width (W)[dot]
  \item \texttt{p5} : Font height (H)[dot]
  \item \texttt{p6} : Right-side character spacing [dot]
    \begin{itemize}
      \item Plus (+)/Minus (-) option can be used. Ex) 5, +3, -10…
    \end{itemize}
  \item \texttt{p7} : Bold
    \begin{itemize}
      \item N : Normal
      \item B : Bold
    \end{itemize}
  \item \texttt{p8} : Reverse printing
    \begin{itemize}
      \item N: Normal printing
      \item R: Reverse printing
    \end{itemize}
  \item \texttt{p9} : Text style
    \begin{itemize}
      \item N: Normal
      \item I: Italic
    \end{itemize}
  \item \texttt{p10} : Rotation
    \begin{itemize}
      \item \begin{tabular}{|c|c|}
          \hline
          \textbf{Value} & \textbf{Rotation} \\
          \hline
          0 & No Rotation \\
          1 & 90 degrees \\
          2 & 180 degrees \\
          3 & 270 degrees \\
          \hline
        \end{tabular}
    \end{itemize}
  \item \texttt{p11} : Text Alignment (Optional)
    \begin{itemize}
      \item L: \texttt{p1} means the position of the first character in the text string - Left alignment
      \item R: \texttt{p1} means the position of the last character in the text string - Right alignment
      \item C: \texttt{p1} means the position of the center character in the text string - Center alignment
    \end{itemize}
\end{itemize}
**p12:** Text string write direction

0: Write text string form left to right (BIXOLON)
1: Write text string form right to left (NOLOXIB)

**DATA:** The various data types can be used in the data field as follows.

1) Fixed text string: ‘Text String’
2) Variables declared in template by `SV` command: `Vnn`
3) Counters declared by the `SC` command: `Cn`

*1), 2), and 3) can be mixed together.*

<table>
<thead>
<tr>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>V50,100,U,25,25,+1,N,N,N,0,L,0,’BIXOLON Label Printer’</td>
</tr>
<tr>
<td>V50,200,U,35,35,-1,N,N,N,0,L,0,’Manufacturer :’ V00</td>
</tr>
<tr>
<td>V50,300,U,35,35,+1,B,R,I,0,L,0, V00</td>
</tr>
<tr>
<td>V50,400,U,45,25,+1,N,N,N,0,L,0,’Vector Font Test’ C0</td>
</tr>
<tr>
<td>V50,500,U,25,45,+1,N,N,N,0,L,0, C0</td>
</tr>
</tbody>
</table>

*To print ‘ or \, \ or \ must be typed.*
Example

V50,100,U,25,25,+1,N,N,N,0,L,0,'Vector Font Test'
V50,200,U,35,35,-1,N,N,N,0,L,0,'Vector Font Test'
V50,300,U,35,35,+1,B,R,I,0,L,0,'Vector Font Test'
V50,400,U,45,25,+1,N,N,N,0,L,0,'Vector Font Test'
V50,500,U,25,45,+1,N,N,N,0,L,0,'Vector Font Test'
V50,700,U,65,65,+1,N,N,N,0,L,0,'ABCDEFGHIJKLMNO'
V50,900,U,65,65,+1,N,N,N,0,L,0,'abcdefghijklmnopqrstuvwxyz'

P1

Result

Vector Font Test
Vector Font Test
Vector Font Test
Vector Font Test

ABCDEFGHIJKLMNOPQRSTUVWXYZ
abcdefghijklmnopqrstuvwxyz
2-1-3 B1 (1 Dimensional bar code)

Description
Draw 1D Barcode on the image buffer

Syntax
B1,p1,p2,p3,p4,p5,p6,p7,p8(,p9),'DATA'

Parameters
p1 : Horizontal position (X) [dot]
p2 : Vertical position (Y) [dot]
p3 : Barcode selection

<table>
<thead>
<tr>
<th>p3</th>
<th>Barcode</th>
<th>p3</th>
<th>Barcode</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Code39</td>
<td>5</td>
<td>UPC-A</td>
</tr>
<tr>
<td>1</td>
<td>Code128</td>
<td>6</td>
<td>UPC-E</td>
</tr>
<tr>
<td>2</td>
<td>I2of5</td>
<td>7</td>
<td>EAN13</td>
</tr>
<tr>
<td>3</td>
<td>Codabar</td>
<td>8</td>
<td>EAN8</td>
</tr>
<tr>
<td>4</td>
<td>Code93</td>
<td>9</td>
<td>UCC/EAN128</td>
</tr>
</tbody>
</table>

p4 : Narrow bar width [dot]
p5 : Wide bar width [dot]
p6 : Bar code height [dot]
p7 : Rotation

<table>
<thead>
<tr>
<th>Value</th>
<th>Rotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No Rotation</td>
</tr>
<tr>
<td>1</td>
<td>90 degrees</td>
</tr>
<tr>
<td>2</td>
<td>180 degrees</td>
</tr>
<tr>
<td>3</td>
<td>270 degrees</td>
</tr>
</tbody>
</table>

p8 : HRI (Human Readable Interpretation)
0 : Not printed
1 : Below the bar code(Font Size : 1)
2 : Above the bar code(Font Size : 1)
3 : Below the bar code(Font Size : 2)
4 : Above the bar code(Font Size : 2)
5 : Below the bar code(Font Size : 3)
6 : Above the bar code(Font Size : 3)
7 : Below the bar code(Font Size : 4)
8 : Above the bar code(Font Size : 4)

(p9) : quiet zone width(optional) : 0 ~ 20
△ Quiet zone is added to the front and end of the barcode for safe scanning. Because of the quiet zone, the barcode seems to be seen drawn in incorrect position. If p9 is not used, the printer automatically sets parameter to 0.

**Quiet zone with = p9 × narrow bar width(p4)**

‘DATA’ : The various data types can be used in the data field as followings.
1) Fixed text string : ‘Text String’
2) Variable declared in template by SV command : Vnn
3) Counter declared by the SC command : Cn
4) In the Code 128, when send data to printer if codeset selection commands (>A, >B, >C) will be used codeset can be selected.

   By using >A, Codeset will be set Codeset A.
   By using >B, Codeset will be set Codeset B.
   By using >C, Codeset will be set Codeset C.

If Codeset select command is not used, automatically set to Auto-mode.

△ 1), 2) and 3) can be used together

**Example**

<table>
<thead>
<tr>
<th>Code</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>B178,196,0,2,6,100,0,0,’1234567890’</td>
<td></td>
</tr>
<tr>
<td>B178,196,0,2,6,100,0,0,‘V00’</td>
<td></td>
</tr>
<tr>
<td>B178,196,0,2,6,100,0,0,’C0’</td>
<td></td>
</tr>
<tr>
<td>B178,196,1,2,6,100,0,0,’&gt;A1234567890’</td>
<td></td>
</tr>
<tr>
<td>B178,196,1,2,6,100,0,0,’&gt;B1234567890’</td>
<td></td>
</tr>
<tr>
<td>B178,196,1,2,6,100,0,0,’&gt;C1234567890&gt;A5’</td>
<td></td>
</tr>
</tbody>
</table>
Example

SM20,20

B178,196,0,2,6,100,0,0,’1234567890’  // Caution: The position is not (178,196) but (78,196)

B150,468,0,4,10,200,0,0,’1234567890’
P1

Result
2-1-4 B2 (2 Dimensional bar code)

**Description**
Draw 2D Barcode on the image buffer

**Syntax**
B2\(p_1,p_2,p_3\ldots \cdot \text{DATA}'

**Parameters**
- \(p_1\) : Horizontal position (X) [dot]
- \(p_2\) : Vertical position (Y) [dot]
- \(p_3\) : 2D barcode selection

<table>
<thead>
<tr>
<th>(p_3)</th>
<th>2D Barcode</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>MaxiCode</td>
</tr>
<tr>
<td>P</td>
<td>PDF417</td>
</tr>
<tr>
<td>Q</td>
<td>QR Code</td>
</tr>
<tr>
<td>D</td>
<td>Data Matrix</td>
</tr>
</tbody>
</table>

*Following parameters \((p_4, p_5 \ldots, \text{Data})\) are barcodes-specific. See the following pages for details of each 2D barcodes.*
Maxicode (When p3 is M)

- **p1**: Horizontal position (X) [dot]
- **p2**: Vertical position (Y) [dot]
- **p3**: M (means 'Maxicode')
- **p4**: Mode selection

<table>
<thead>
<tr>
<th>p4</th>
<th>Rotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Mode0</td>
</tr>
<tr>
<td>2</td>
<td>Mode2</td>
</tr>
<tr>
<td>3</td>
<td>Mode3</td>
</tr>
<tr>
<td>4</td>
<td>Mode4</td>
</tr>
</tbody>
</table>

‘DATA’ : Data format is dependent on ‘Mode’

<table>
<thead>
<tr>
<th>Mode</th>
<th>Data Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>'cl,co,pc,lpm'</td>
</tr>
<tr>
<td>2 or 3</td>
<td>'cl,co,pc,lpm'</td>
</tr>
<tr>
<td>4</td>
<td>'lpm'</td>
</tr>
</tbody>
</table>

- **cl**: Class Code (3 digits)
- **co**: Country Code (3 digits)
- **Mode2**: Numeric Characters
- **Mode3**: International Characters
- **pc**: Postal Code
- **lpm**: Low priority message (data)

**Example**

1) **Mode 0**

\[ \texttt{B2200.200,M,0,'999,840,06810,7317,THIS IS A TEST OF MODE 0 STRUCTURED CARRIER MESSAGE ENCODING. THIS IS AN 84 CHAR MSG'} \]

2) **Mode 2**

\[ \texttt{B2200.200,M,2,'999,840,06810,7317,THIS IS A TEST OF BIXOLON LABEL PRINTER SRP770. MODE 2 ENCODING. THIS IS AN 84 CHAR.'} \]

3) **Mode 3**

\[ \texttt{B2200.200,M,3,'999,056,B1050,7317,THIS IS A TEST OF BIXOLON LABEL PRINTER SRP770. MODE 3 ENCODING. THIS IS AN 84 CHAR.'} \]

4) **Mode 4**

\[ \texttt{B2200.200,M,4,'THIS IS A 93 CHARACTER CODE SET A MESSAGE THAT FILLS A MODE 4, UNAPPENDED, MAXICODE SYMBOL...'} \]
PDF417 (When p3 is P)

p1: Horizontal position (X) [dot]
p2: Vertical position (Y) [dot]
p3: P (means ‘PDF417’) p4: Maximum Row Count: 3 ~ 90
p5: Maximum Column Count: 1 ~ 30
p6: Error Correction level

<table>
<thead>
<tr>
<th>p6</th>
<th>EC Level</th>
<th>EC Codeword</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>16</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>32</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>64</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>128</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>256</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>512</td>
</tr>
</tbody>
</table>

p7: Data compression method

<table>
<thead>
<tr>
<th>p7</th>
<th>Data Type</th>
<th>Compression</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Text</td>
<td>2 Characters per codeword</td>
</tr>
<tr>
<td>1</td>
<td>Numeric</td>
<td>2.93 Characters per codeword</td>
</tr>
<tr>
<td>2</td>
<td>Binary</td>
<td>1.2 Bytes per codeword</td>
</tr>
</tbody>
</table>

p8: HRI

0: Not Printed
1: Below the barcode

p9: Barcode origin point

0: Center of barcode
1: Upper left corner of barcode (default)

p10: Module Width: 2 ~ 9
p11: Bar Height: 4 ~ 99
p12: Rotation

<table>
<thead>
<tr>
<th>Value</th>
<th>Rotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No Rotation</td>
</tr>
<tr>
<td>1</td>
<td>90 degrees</td>
</tr>
<tr>
<td>2</td>
<td>180 degrees</td>
</tr>
<tr>
<td>3</td>
<td>270 degrees</td>
</tr>
</tbody>
</table>

‘DATA’: ASCII data or Binary data.

Example

B2100,750,P,30,5,0,0,1,1,3,10,0,’BIXOLON Label Printer SRP770’ \h The position is (100,750)
QR Code (When p3 is Q)  

- **p1**: Horizontal position (X) [dot]  
- **p2**: Vertical position (Y) [dot]  
- **p3**: Q (means ‘QR Code’)  
- **p4**: MODEL selection  
  1: MODEL1  
  2: MODEL2  
- **p5**: ECC Level  

<table>
<thead>
<tr>
<th>p6</th>
<th>Recovery Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>7%</td>
</tr>
<tr>
<td>M</td>
<td>15%</td>
</tr>
<tr>
<td>Q</td>
<td>25%</td>
</tr>
<tr>
<td>H</td>
<td>30%</td>
</tr>
</tbody>
</table>

- **p6**: Barcode Size: 1~4  
- **p7**: Rotation  

<table>
<thead>
<tr>
<th>Value</th>
<th>Rotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No Rotation</td>
</tr>
<tr>
<td>1</td>
<td>90 degrees</td>
</tr>
<tr>
<td>2</td>
<td>180 degrees</td>
</tr>
<tr>
<td>3</td>
<td>270 degrees</td>
</tr>
</tbody>
</table>

‘DATA’: ASCII data or Binary data.

**Example**  

**B2**200,100,Q,2,M,4,0,’ABCDEFGHIJKLMNOP1234567890’ // The position is (200,100)

Data Matrix (When p3 is D)  

- **p1**: Horizontal position (X) [dot]  
- **p2**: Vertical position (Y) [dot]  
- **p3**: D (the ECC 200 data quality format)  
- **p4**: Barcode Size: 1 ~ 4;  
- **P5**: Reverse  
  - N: Normal  
  - R: Reverse (or Inverse) – Reverse Video or Negative image  
- **P6**: Rotation

<table>
<thead>
<tr>
<th>Value</th>
<th>Rotation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No Rotation</td>
</tr>
<tr>
<td>1</td>
<td>90 degrees</td>
</tr>
<tr>
<td>2</td>
<td>180 degrees</td>
</tr>
<tr>
<td>3</td>
<td>270 degrees</td>
</tr>
</tbody>
</table>

‘DATA’: ASCII data or Binary data.

**Example**  

**B2**200,100,D,2,N,’BIXOLON Label Printer’ // The position is (200,100)
2-1-5 B3 (Special Barcode)

Description
Draw Special Barcode on the image buffer

Syntax
\[ B3p1,p2,p3 \ldots \text{‘DATA’} \]

Parameters
\begin{itemize}
  \item \textbf{p1} : Horizontal position (X) [dot]
  \item \textbf{p2} : Vertical position (Y) [dot]
  \item \textbf{p3} : Special barcode selection
\end{itemize}

\begin{center}
\begin{tabular}{|c|c|}
  \hline
  \textbf{p3} & \textbf{Special Barcode} \\
  \hline
  I & IMB(Intelligent Mail Barcode) \\
  \hline
\end{tabular}
\end{center}


dd Following parameters (p4, p5 \ldots ,Data) are barcodes-specific. 
See the following pages for details of each special barcodes.

\textbf{IMB (p3 = I)}
\begin{itemize}
  \item \textbf{p1} : Horizontal position (X) [dot]
  \item \textbf{p2} : Vertical position (Y) [dot]
  \item \textbf{p3} : I (means ‘IMB’)
  \item \textbf{p4} : Rotation
\end{itemize}

\begin{center}
\begin{tabular}{|c|c|}
  \hline
  \textbf{Value} & \textbf{Rotation} \\
  \hline
  0 & No Rotation \\
  1 & 90 degrees \\
  2 & 180 degrees \\
  3 & 270 degrees \\
  \hline
\end{tabular}
\end{center}

\begin{itemize}
  \item \textbf{P5} : HRI :
    \begin{itemize}
      \item 0 : Not Printed
      \item 1 : Below the barcode
    \end{itemize}
\end{itemize}

\textbf{‘DATA’} : ASCII data or Binary data.

Example
\[ B3100,100,I,0,1,’0123456709498765432101234567891’ \] // The position is (100,100)
2-1-6 BD (Block Draw)

**Description**
Draw Line, Block, Box & Slope on the image buffer

**Syntax**
BD\(p_1,p_2,p_3,p_4,p_5(,p_6)\)

**Parameters**
p1 : Horizontal start position (X) [dot]
p2 : Vertical start position (Y) [dot]
p3 : Horizontal end position (X) [dot]
p4 : Vertical end position (Y) [dot]
p5 : Options

<table>
<thead>
<tr>
<th>p5</th>
<th>Type</th>
<th>Additional p6</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>Line Overwriting</td>
<td>Not necessary</td>
</tr>
<tr>
<td>E</td>
<td>Line Exclusive OR</td>
<td>Not necessary</td>
</tr>
<tr>
<td>D</td>
<td>Line Delete</td>
<td>Not necessary</td>
</tr>
<tr>
<td>S</td>
<td>Slope(a oblique line)</td>
<td>Thickness</td>
</tr>
<tr>
<td>B</td>
<td>Box</td>
<td>Thickness</td>
</tr>
</tbody>
</table>

* If p5 is S or B, then additional p6 must follow p5.
Example

1) Start and end position

\[(p_1, p_2) \quad \text{to} \quad (p_3, p_4)\]

2) Overwriting mode (when \(p_5\) is \(O\))

3) Exclusive OR mode (when \(p_5\) is \(E\))

4) Delete block mode (when \(p_5\) is \(D\))
5) Slope block mode (when p5 is S)

6) Draw box mode (when p5 is B)
2-1-7 CD (Circle Draw)

**Description**
Draw Circle on the image buffer

**Syntax**
CD\(p_1, p_2, p_3, p_4\)

**Parameters**
- \(p_1\) : Horizontal start position (X) [dot]
- \(p_2\) : Vertical start position (Y) [dot]
- \(p_3\) : Circle Size Selection

<table>
<thead>
<tr>
<th>Value</th>
<th>Diameter (mm)</th>
<th>Width (\times) Height(dots)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>40 (\times) 40</td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td>56 (\times) 56</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>72 (\times) 72</td>
</tr>
<tr>
<td>4</td>
<td>11</td>
<td>88 (\times) 88</td>
</tr>
<tr>
<td>5</td>
<td>13</td>
<td>104 (\times) 104</td>
</tr>
<tr>
<td>6</td>
<td>21</td>
<td>168 (\times) 168</td>
</tr>
</tbody>
</table>

- \(p_4\) : Multiplier : 1 ~ 4

**Example**
CD\(100,200,2,1\)

(100,200)

Circle size(diameter is 7mm)
2-1-8 CS (Character Set selection)

**Description**
To select international character set and code table.

**Syntax**

\[ \text{CS} p1,p2 \]

**Parameters**

\( p1 \) : International Character Set

<table>
<thead>
<tr>
<th>( p1 )</th>
<th>Country</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>U.S.A</td>
</tr>
<tr>
<td>1</td>
<td>France</td>
</tr>
<tr>
<td>2</td>
<td>Germany</td>
</tr>
<tr>
<td>3</td>
<td>U.K</td>
</tr>
<tr>
<td>4</td>
<td>Denmark I</td>
</tr>
<tr>
<td>5</td>
<td>Sweden</td>
</tr>
<tr>
<td>6</td>
<td>Italy</td>
</tr>
<tr>
<td>7</td>
<td>Spain I</td>
</tr>
<tr>
<td>8</td>
<td>Norway</td>
</tr>
<tr>
<td>9</td>
<td>Denmark II</td>
</tr>
<tr>
<td>10</td>
<td>Japan</td>
</tr>
<tr>
<td>11</td>
<td>Spain II</td>
</tr>
<tr>
<td>12</td>
<td>Latin America</td>
</tr>
<tr>
<td>13</td>
<td>Korea</td>
</tr>
<tr>
<td>14</td>
<td>Slovenia/Croatia</td>
</tr>
<tr>
<td>15</td>
<td>China</td>
</tr>
</tbody>
</table>
p2 : Code Pages

<table>
<thead>
<tr>
<th>p2</th>
<th>Code Table</th>
<th>Language</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>CP437</td>
<td>U.S.A</td>
</tr>
<tr>
<td>1</td>
<td>CP850</td>
<td>Latin1</td>
</tr>
<tr>
<td>2</td>
<td>CP 852</td>
<td>Latin2</td>
</tr>
<tr>
<td>3</td>
<td>CP 860</td>
<td>Portuguese</td>
</tr>
<tr>
<td>4</td>
<td>CP 863</td>
<td>Canadian French</td>
</tr>
<tr>
<td>5</td>
<td>CP 865</td>
<td>Nordic</td>
</tr>
<tr>
<td>6</td>
<td>WCP 1252</td>
<td>Latin 1</td>
</tr>
<tr>
<td>7</td>
<td>CP 865 + WCP 1252</td>
<td>European Combined</td>
</tr>
<tr>
<td>8</td>
<td>CP 857</td>
<td>Turkish</td>
</tr>
<tr>
<td>9</td>
<td>CP 737</td>
<td>Greek</td>
</tr>
<tr>
<td>10</td>
<td>WCP 1250</td>
<td>Latin 2</td>
</tr>
<tr>
<td>11</td>
<td>WCP 1253</td>
<td>Greek</td>
</tr>
<tr>
<td>12</td>
<td>WCP 1254</td>
<td>Turkish</td>
</tr>
<tr>
<td>13</td>
<td>CP 855</td>
<td>Cyrillic</td>
</tr>
<tr>
<td>14</td>
<td>CP 862</td>
<td>Hebrew</td>
</tr>
<tr>
<td>15</td>
<td>CP 866</td>
<td>Cyrillic</td>
</tr>
<tr>
<td>16</td>
<td>WCP 1251</td>
<td>Cyrillic</td>
</tr>
<tr>
<td>17</td>
<td>WCP 1255</td>
<td>Hebrew</td>
</tr>
<tr>
<td>18</td>
<td>CP 928</td>
<td>Greek</td>
</tr>
<tr>
<td>19</td>
<td>CP 864</td>
<td>Arabic</td>
</tr>
<tr>
<td>20</td>
<td>CP 775</td>
<td>Baltic</td>
</tr>
<tr>
<td>21</td>
<td>WCP1257</td>
<td>Baltic</td>
</tr>
<tr>
<td>22</td>
<td>CP858</td>
<td>Latin 1 + Euro</td>
</tr>
</tbody>
</table>

◆ Default Setting is U.S.A standard (p1=0 and p2=0).

◆ European Combined Page

<table>
<thead>
<tr>
<th>Address</th>
<th>Code Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x80</td>
<td>Euro Currency</td>
</tr>
<tr>
<td>0x81 ~ 0x9f</td>
<td>PC865</td>
</tr>
<tr>
<td>0xA0 ~ 0xff</td>
<td>PC1252</td>
</tr>
<tr>
<td>Country</td>
<td>International Character Set</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------</td>
</tr>
<tr>
<td>U.S.A</td>
<td>#$ @ [ \ ] ^ ' {</td>
</tr>
<tr>
<td>France</td>
<td>#$ à ° ç $ ^ ' é ü é</td>
</tr>
<tr>
<td>Germany</td>
<td>#$ $ Ä Ö Ü ^ ' ä ö ü β</td>
</tr>
<tr>
<td>U.K.</td>
<td>£ $ [ \ ] ^ ' {</td>
</tr>
<tr>
<td>Denmark I</td>
<td>#$ $ ∆ Ø Å ^ ' æ ø å ~</td>
</tr>
<tr>
<td>Sweden</td>
<td>#$ # É Ä Ö Å Ü é ä ö á ü</td>
</tr>
<tr>
<td>Italy</td>
<td>#$ $ @ ° \ é ^ ü à ö è ì</td>
</tr>
<tr>
<td>Spain</td>
<td>$ @ ° \ é ^ ü à ö è ì</td>
</tr>
<tr>
<td>Norway</td>
<td>#$ # É ∆ Ø Å Ü é æ ø å ~</td>
</tr>
<tr>
<td>Denmark II</td>
<td>#$ # É ∆ Ø Å Ü é æ ø å ~</td>
</tr>
<tr>
<td>Japan</td>
<td>$ @ [ \ ] ^ ' {</td>
</tr>
<tr>
<td>Spain II</td>
<td>#$ # á i Ñ ¿ é · i ñ ö ü</td>
</tr>
<tr>
<td>Latin America</td>
<td>#$ # á i Ñ ¿ é ü í ñ ö ü</td>
</tr>
<tr>
<td>Korea</td>
<td>$ @ [ \ ] ^ ' {</td>
</tr>
<tr>
<td>Slovenia/Croatia</td>
<td>#$ # Ž Š Đ Ć Č Ž š đ č č</td>
</tr>
<tr>
<td>China</td>
<td>$ @ [ \ ] ^ ' {</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ASCII Code</th>
<th>0 1 2 3 4 5 6 7 8 9 A B C D E F</th>
</tr>
</thead>
<tbody>
<tr>
<td>0~31 : Control Code</td>
<td>! &quot; # $ % &amp; ' ( ) * + , - . /</td>
</tr>
<tr>
<td>32~127 : Alphanumeric</td>
<td>0 1 2 3 4 5 6 7 8 9 : ; &lt; = &gt; ?</td>
</tr>
<tr>
<td></td>
<td>@ A B C D E F G H I J K L M N O</td>
</tr>
<tr>
<td></td>
<td>P Q R S T U V W X Y Z [ ] ^ _ `</td>
</tr>
<tr>
<td></td>
<td>a b c d e f g h i j k l m n o p</td>
</tr>
<tr>
<td></td>
<td>q r s t u v w x y z {</td>
</tr>
</tbody>
</table>

※ Refer to the “Code Pages Manual” for more extension code pages table.
2-1-9 P (Print)

Description
Let the printer start printing the content of image buffer

Syntax
\[ P\ p_1,\ [p_2] \]

Parameters
- \(p_1\) : Number of label sets : 1 ~ 65535
- \(p_2\) : Number of copies of each label : 1 ~ 65535

⚠️ The P command cannot be used in a template sequence. If printing command is needed in template sequence, then use the PV command (See the example of next page).

❗️ Caution
The ‘P’ command should be terminated by ‘CR’ (0x0d). If not, the printer will not start to print until ‘CR’ comes.
Example

(1) In case of Using P (P is used outside of template sequence)

```
TS'TPL_TST1'
SV00,15,N,'Model Name :'
T50,100,3,1,1,0,0,N,N,'Model Name :V00
TE

TR'TPL_TST1'

?  // Start Template Store
SV00,15,N,'Model Name :'
SV01,2,N,'# of set :'
SV02,2,N,'# of copies :'
T50,100,3,1,1,0,0,N,N,'Model Name :V00
PVV01,V02

TE

TR'TPL_TST1'

?  // Recall stored template ‘TPL_TST1’
SRP770  // Get content of variable used in recalled template
P3,2  // Content of variable V00

P  // when using P command, It must not be inside template,
   // but be used after recalling the template and entering the
   // contents of all variables.

// After P command, printer starts printing.
```

(2) In case of Using PV(PV is used inside of template sequence)

```
TS'TPL_TST1'
SV00,15,N,'Model Name :'
SV01,2,N,'# of set :'
SV02,2,N,'# of copies :'
T50,100,3,1,1,0,0,N,N,'Model Name :V00
PVV01,V02

TE

TR'TPL_TST1'

?  // Start Template Store
SV00,15,N,'Model Name :'
SV01,2,N,'# of set :'
SV02,2,N,'# of copies :'
T50,100,3,1,1,0,0,N,N,'Model Name :V00

PVV01,V02  // PV command can be used inside the template

TE

TR'TPL_TST1'

?  // Recall stored template ‘TPL_TST1’
SRP770  // Get content of variable used in recalled template
3  // Content of variable V00
2  // Content of variable V00

// As soon as all contents of variables are entered
// printer will starts printing.
2-2 Media & Buffer Related Commands

1) ST
   Select Thermal Direct/Transfer Printing.

2) SM
   Set marginal value in label(Image buffer)

3) SF
   Set back-feed option.

4) SL
   Set label(Image buffer) length

5) SW
   Set label(Image buffer) width

6) SB
   Set buffer mode(Enable or disable Double Buffering)

7) CB
   Clear Image Buffer
2-2-1 ST (Set Printing Type)

**Description**
Select Thermal Direct Printing or Thermal Transfer Printing.

**Syntax**
```plaintext
ST p1
```

**Parameters**
- **p1**: Direct Thermal / Thermal Transfer
  - d: Direct Thermal
  - t: Thermal Transfer

2-2-2 SM (Set Margin)

**Description**
Set marginal value of the image buffer. This command moves the origin point (0,0) to (p1,p2) and make (p1,p2) become the new origin.

**Syntax**
```plaintext
SM p1,p2
```

**Parameters**
- **p1**: Horizontal margin [dots]
- **p2**: Vertical margin [dots]

---

* The origin point is upper-left point of the image buffer
** When printing orientation is from top to bottom

(0,0) : Origin

(p1,p2) : New Origin

virtual (0,0)

Image Buffer

New Image Buffer

Label Feeding Direction

** When printing orientation is from bottom to top.

virtual (0,0)

New Origin : (p1,p2)

Label Feeding Direction
2-2-3 SF (Set Back-Feed Option)

Description
Set back-feed option.
This command decides whether printer does back-feed action before starting printing.

Syntax
SF\(p1,(p2)\)

Parameters
\(p1\) : Enable/Disable
- 0 : Disable back-feed option.
- 1 : Enable back-feed option (Default)
\(p2\) : Back feeding step quantity.
- This parameter is valid when \(p1\) is 1.
- The step quantity defined by user can’t exceed printer’s default feeding quantity.
- 0 means printer’s default feeding quantity.

- This option is useful for the continuous paper or black mark media with perforation line away from black mark.
- The printer’s default back feeding step quantity depends on the printer models and printer modes such as normal, peeler or cutter.

Examples)
SF0  → Disable Printer’s back-feeding option.
SF1  → Default quantity of Back feed is executed before printing.
SL1,0 → Default quantity of Back feed is executed before printing.
SL1,100 → 100 step’s Back feed is executed before printing.
SL0,100 → Back feed is disabled and p2(100) is ignored.
2-2-4 SL (Set Length)

Description
Set length of label and gap(or Black Mark) and specify media type.

Syntax
SLp1,p2(,p3)(,p4)

Parameters
p1 : Label length [dots] : Maximum 2432 dots(12 inch)
- Double buffering feature can be used only when label length(p1) is less than 1216(2432/2, 6inch) dots.
- If p1 is over 1216 dots, the double buffering feature will be automatically released.
- So if you don’t use double buffering feature, you can design maximum 2432 dots(12 inch) size label.
p2 : Gap length or thickness of black line [dots]
p3 : Media Type

<table>
<thead>
<tr>
<th>p3</th>
<th>Media type</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>Gap</td>
</tr>
<tr>
<td>C</td>
<td>Continuous</td>
</tr>
<tr>
<td>B</td>
<td>Black Mark</td>
</tr>
</tbody>
</table>
- If this parameter is not used, automatically set to G(Gap type).
- The default value of label length is 6 inch(1216 dots)
- This command sets the length of image buffer and the printer will print and form feed as much as the length set by this command.
- When using Continuous type media, the label length must be set.
p4 : Offset Length between Black Mark(or Gap) and perforation line [dots]
- This parameter is valid when p3 parameter is used.

Examples)
SL1200,20 → Gap media,Media length:1200dots,Gap length:20dots
SL1200,20,C → Continuous media,Media length:1200dots,Gap length:20dots
SL1200,20,G → Gap media,Media length:1200dots,Gap length:20dots
SL1200,20,B → Black Mark media,Media length:1200dots,Gap length:20dots

The perforation line is on the black mark.
SL1200,20,B,200 → Black Mark media,Media length:1200dots,Gap length:20dots

The perforation line is 200 dots behind from black mark.

- In the Gap Mode, the printer will form feed until meeting the next gap.
- In the Continuous Mode, the printer will form feed as much as label length set by SL.
- In the B/M Mode, the printer will form feed until meeting the next B/M.
Example – p1 & p2(Length)

SL406,20  // Set label length to 406 dots (2 inch, 50mm) and gap length to 20 dots (2.5mm)

p1
(In this example, 406)

P2
(In this example, 20)
Example – p3(Media Type)

1. Gap Type

2. Continuous Type

3. Black Mark Type

Example – p4(Offset Length)

- Black Mark
- Perforation Line
- p4 : Offset Length
2-2-5 SW (Set Width)

Description
Set label width.
Resize the image buffer to match the label size.

Syntax
SW p1

Parameters
p1 : Label width [dots]

- The default value of label width is 4.1 inch (832 dots) and that is the maximum printable width.
- SRP770 is the center aligned printer and media is positioned in the center of the head.

Example
SW406 // Set label width to 2 inch (406 dots)
2-2-6 SB (Set Buffer mode)

**Description**
Set double buffer mode

**Syntax**
SB \( p_1 \)

**Parameters**
- \( p_1 \) : Enable ‘Double Buffering’ function.
  - 0 : Disable double buffer mode
  - 1 : Enable double buffer mode (Default)

- Double buffering feature enables the printer to construct the image buffer for the next label while printing the current label.
- Double buffering feature can be used only if the label length set by SL is less than half of the maximum label length.

2-2-7 CB (Clear Buffer)

**Description**
Clear image buffer and be ready to make a new label.

**Syntax**
CB

**Example**
CB  // Clear Image Buffer
2-3 Printer Setting Commands

1) SS
   Set printer speed

2) SD
   Set printing density

3) SO
   Set printing orientation

4) SP
   Set serial port

5) SA
   Set Offset

6) TA
   Set Tear-off/Cut
2-3-1 SS (Set Speed)

**Description**
Set print speed

**Syntax**
SS\(_{p1}\)

**Parameters**
\(p1\) : Speed set value

<table>
<thead>
<tr>
<th>Value</th>
<th>Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2.5 ips</td>
</tr>
<tr>
<td>1</td>
<td>3.0 ips</td>
</tr>
<tr>
<td>2</td>
<td>4.0 ips</td>
</tr>
<tr>
<td>3</td>
<td>5.0 ips</td>
</tr>
<tr>
<td>4</td>
<td>6.0 ips</td>
</tr>
<tr>
<td>5</td>
<td>7.0 ips</td>
</tr>
<tr>
<td>6</td>
<td>8.0 ips</td>
</tr>
</tbody>
</table>

2-3-2 SD (Set Density)

**Description**
Set printing density

**Syntax**
SD\(_{p1}\)

**Parameters**
\(p1\) : Density Level
- 0 ~ 20 (0 is the lowest density)
2-3-3 SO (Set Orientation)

**Description**
Set printing direction

**Syntax**

```
SO p1
```

**Parameters**

- `p1`: Printing direction
  - `T`: Print from top to bottom (default)
  - `B`: Print from bottom to top

**Example**

- `SOT` // Print from top of the image buffer to bottom.
- `SOB` // Print from bottom of the image buffer to top.

---

**Printing Direction**

1. SOT (Print from Top to Bottom)

```
SRP770
From Top to Bottom
```

2. SOB (Print from Bottom to Top)

```
SRP770
From Bottom to Top
```
2-3-4 SP (Set Port)

**Description**
Set serial port.

**Syntax**
\[ \text{SP} p1, p2, p3, p4 \]

**Parameters**
- **p1**: Baud rate
  - **Value** | **Baud Rate (bps)**
    - 0 | 9,600
    - 1 | 19,200
    - 2 | 38,400
    - 3 | 57,600
    - 4 | 115,200

- **p2**: Parity
  - **Value** | **Parity**
    - O | Odd parity
    - E | Even parity
    - N | No parity (Default)

- **p3**: Number of data bits
  - **Value** | **Data bits**
    - 7 | 7 bit
    - 8 | 8 bits (Default)

- **p4**: Number of stop bits
  - **Value** | **Stop bits**
    - 1 | 1 bit (Default)
    - 2 | 2 bits
2-3-5 SA (Set Offset)

**Description**
Save (set) offset length between black marks (or gap) and dotted lines [dots]

**Syntax**
SA\(p_1\)

**Parameters**
\(p_1 : -100\text{~to~}100\)

- Offset values saved via the use of SA commands are stored permanently on the printer.
  (Offset values saved via the cf. SL command are reset after the power is turned off.)

2-3-6 TA (Tear-off/Cutter Position Setting)

**Description**
This function regulates the label cutting location after printing. Tear-off position or Cut position can adjust.

**Syntax**
TA\(p_1\)

**Parameters**
\(p_1 : -100\text{~to~}100\)

- Tear-off/Cutter Position values saved via the use of TA commands are stored permanently on the printer.
2-4 Variable Related Commands

1) SC
   Counters which is used in template sequence

2) AC(Auto Counter)
   Counters which is used in normal commands sequence
   (outside of template sequence)

3) SV
   Set variable

4) ?
   Get data for counter and variable

5) PV
   Print with variables
2-4-1 SC (Set Counter)

Description
Define one counter of total 10 counters.
Counters must be used in Template sequence and execute consecutive auto-numbering function.

Syntax
SC p1,p2,p3,p4,'Prompt'

Parameters
p1: Identity of Counter : 0 ~ 9
   ▶ Total 10 counters, from C0 to C9, are provided.
p2: The size of the field which displays the content of counter : 1 ~ 27
p3: Justification in field(Field size is p2)

<table>
<thead>
<tr>
<th>Value</th>
<th>Justification</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>No</td>
</tr>
<tr>
<td>R</td>
<td>Right</td>
</tr>
<tr>
<td>L</td>
<td>Left</td>
</tr>
<tr>
<td>C</td>
<td>Center</td>
</tr>
</tbody>
</table>

p4 : Step Value : ±1 ~ ±9
   ▶ + or – symbol must precede. Ex) –2 or +3
   ‘Prompt’: This text string is transmitted to host(PC) by serial interface in order to give information to host about the declared counter.

♠ The data field of T(Text) or B(Barcode) commands is used to print the contents of counter.

♠ SC should be used just in Template sequence. If you want to use counter function in normal mode(not in Template), use the AC(Auto Counter).

Example
SC0,7,N,+3,'Please Enter Serial Number’
2-4-2 AC (Auto Counter)

**Description**
Define one counter of total 10 counters.
Counters can be used in normal mode (not in Template) and execute consecutive auto-numbering.

**Syntax**
AC\(p1,p2,p3,\text{'Start Value'}\)

**Parameters**
\(p1\) : Identity of Counter : 0 ~ 9
- Total 10 counters, from C0 to C9, are provided.
\(p2\) : The size of the field which displays the content of counter : 1 ~ 27
\(p3\) : Step Value : ±1 ~ ±9
- + or – symbol must precede. Ex) –2 or +3

‘Start Value’ : Start value of auto-counting. Just digits can be used in this field

- The Auto-counter defined by AC command can be printed with T and B1 command.
- This function is useful to print serial number or serial barcode without using Template.
- AC can not be used in Template sequence. If you want to use counter function in Template sequence, use the SC command.

**Example**
AC0,3,+1,'123' \(//\) Please input the start value of counting between ‘ marks
AC1,7,+1,'1234567'
T100,100,3,1,1,0,0,N,N,C0
B1100,400,0,2,7,100,0,1,12,C1
P3,1
2-4-3 SV (Set Variable)

Description
Define variables for the text or barcode ‘data’ fields.

Syntax
SV{p1},{p2},{p3},'Prompt'

Parameters
p1: Identity of Variables: 00 ~ 99
p2: Maximum number of characters: 1 ~ 99
p3: Justification in field (Field size is p2)

<table>
<thead>
<tr>
<th>Value</th>
<th>Justification</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>No</td>
</tr>
<tr>
<td>R</td>
<td>Right</td>
</tr>
<tr>
<td>L</td>
<td>Left</td>
</tr>
<tr>
<td>C</td>
<td>Center</td>
</tr>
</tbody>
</table>

‘Prompt’: This ASCII text field is used to ask a value to be entered for the variable(p1) and is transmitted to the host by serial interface.

- The data field of T(Text) or B(Barcode) commands is used to print the contents of variable.
- Variable is entered to data field like V00 or V01.

Example
SV01,20,N,'Please Enter Product Code :'
2-4-4 ? (Get Variables)

**Description**
Use this command to get the content of variables or counters

**Syntax**

```
? 
Content of variable
```

* Data must be entered in ascending order

**Example**

```
TS 'Template1' // Template Store Start
SV00,20,N,'Enter Company Name : ' // Declare(Set) variable V00
SV01,15,N,'Enter Product Code : ' // Declare(Set) variable V01
T50,30,3,1,0,0,N,N,V00 // Use T command to print V00
T50,150,3,1,0,0,N,0,'Code : 'V01 // Use T command to print V01
TE // Template Store End

TR 'Template1' // Recall Template1
? // Start to get data for variables
SEM // data for V00
770 // data for V01
P1 // Start Printing when the P command comes

**Result**

```
SEM
Code : 770
```
2-4-5 PV (Print with Variables)

Description
This command is used in template sequence.
The parameters are given by variables.

Syntax
PVp1,[p2]

Parameters
p1 : Number of label sets : 1 ~ 65535
p2 : Number of copies of each label : 1 ~ 65535

Example

TS'Template1'
SV00,20,N,'Please Input the Name :'
SV01,5,N,'Input Number of label sets :'
SV02,5,N,'Input Number of label copies :'
T50,30,3,1,1,0,0,N,N,V00
PV V01,V02
TE

TR'Template1'
?
This is PV Test
2
1

*** Start Printing as soon as data for all variables(and counters) are entered. ***
2-5 Template Related Commands

Template(a certain format of label, sequence of SLCS commands) related commands

1) TS
   Indicate start of template sequence store.

2) TE
   Indicate end of template sequence store.

3) TR
   Recall and reuse stored template.

4) TD
   Delete stored template.

5) TI
   Print the list of all templates stored in memory.
2-5-1 TS (Template store Start)

**Description**
Start template sequence storing.
All the contents following ‘TS’ are stored in memory until meeting ‘TE’ Command.

**Syntax**
```text
TS 'Template name'
```

**Parameters**
- ‘Template name’: This name will be used when ‘Recall’ the stored template.
  - The name is allowed to be up to 10 characters long.
  - The ‘Template name’ is Case-Sensitive.

**Example**
```text
TS 'Template1'     // Start template storing
......
TE                  // End template storing
```

2-5-2 TE (Template store End)

**Description**
End template sequence storing

**Syntax**
```text
TE
```

**Example**
```text
TS 'Template1'     // Start template storing
......
TE                  // End template storing
```

* When storing is finished, the printer sends ‘!’ to the host to prompt end of storing.
2-5-3 TR (Template Recall)

Description
Recall the stored template from memory to make a label and print that.

Syntax
```
TR 'Template name'
```

Parameters
- **‘Template name’**: Indicate the template to be recalled.
  - ♦️ The name is allowed to be up to 10 characters long.
  - ♦️ The ‘Template name’ is **Case-Sensitive**.

Example
```
TR 'Template1'  // Recall 'Template1'
```

- ♦️ If recalled Template does not include any variable or counter, just ‘P’ command is enough to start printing.
- ♦️ If recalled Template includes variables or counters but not ‘PV’(Print with Variables), use ‘?’ command to get data for variables and counters and finally ‘P’ command is necessary to start printing.
- ♦️ If recalled Template includes PV commands, printing will start as soon as all data for variables and counters are entered.
2-5-4 TD (Template Delete)

**Description**
Delete stored template from memory

**Syntax**

```
TD 'Template name'
```

**Parameters**

- **‘Template name’**: Indicate the template to be deleted.
  - The name is allowed to be up to 10 characters long.
  - The ‘Template name’ is **Case Sensitive**.
  - By using *, all templates will be deleted from memory.

**Example**

```
TD 'Template1'  // Delete ‘Template1’
TD *             // Delete all currently stored templates
```

2-5-5 TI (Template Information)

**Description**
Print list of currently stored templates and available memory space

**Syntax**

```
TI
```

**Example**

```
TI
```

**Result**

```
Templates Information
========================
1. Template1
2. Template2
Available template memory : 53Kbyte
```
2-6 Image Related Commands

These commands provide functions to download and print graphic data. PCX and BMP format file are supported and bitmap image data can be printed directly.

1) IS
   Download PCX format image data to NV(Non Volatile) area of memory.

2) IR
   Recall and print downloaded image data.

3) ID
   Delete image data in NV memory.

4) II
   Print all images stored in memory.

5) LD
   Draw the bitmap image data directly on specific position on image buffer.

6) LC
   Draw compression bitmap image data on specific position of image buffer.

7) BMP
   Draw BMP format image file directly on specific position on image buffer.
2-6-1 IS (Image Store)

Description
Download PCX format Image file into the Printer Memory

Syntax
IS\(p_1\), 'Image name'\ DATA OF *.PCX

Parameters
- \(p_1\) : The size of image file in unit of byte.
- 'Image name' : This is the name that will be used when recalling the stored image data.
  - The name is allowed to be up to 10 characters long.
  - The name is case sensitive.

2-6-2 IR (Image Recall)

Description
Recall the stored image from memory and draw that on the image buffer.

Syntax
IR\(p_1,p_2\), 'Image name'

Parameters
- \(p_1\) : Horizontal position (X) [dot]
- \(p_2\) : Vertical position (Y) [dot]
- 'Image name' : Indicate the image data to be recalled.
  - Variable can be used in this field.
  - The name is allowed to be up to 10 characters long.
  - This name is Case Sensitive.

Example
IR30,100,'Image1' // Recall 'Image1'
IR30,100,V01 // Variable can be used in name field
2-6-3 ID (Image Delete)

**Description**
Delete stored image from memory

**Syntax**

```
ID 'Image name'
```

**Parameters**

- `'Image name'`: Indicate the Image in memory to be deleted.
  - The name is allowed to be up to 10 characters long.
  - This name is **Case Sensitive**.
  - By using `*`, all images in memory will be deleted.

**Example**

```
ID 'Image1' // Delete 'Image1'
ID * // Delete all currently stored images
```

2-6-4 II (Image Information)

**Description**
Print list of currently stored images in memory and available memory space

**Syntax**

```
II
```

**Example**

```
II
```

**Result**

```
Image Information
=====================
1. Image1
2. Image2
Available Images memory : 5.3Kbyte
```
Draw bitmap image data on specific position of image buffer.

**Syntax**

```
LDxL xH yL yH dhL dhH dvL dvH d1~dk
```

**Parameters**

- **xL**: Low byte of horizontal start position (X) [dot]
- **xH**: High byte of horizontal start position (X) [dot]
  
  → Start position in x direction = xH * 256 + xL

- **yL**: Low byte of vertical start position (Y) [dot]
- **yH**: High byte of vertical start position (Y) [dot]
  
  → Start position in y direction = yH * 256 + yL

- **dhL**: Low byte of the number of bytes in x-direction.
- **dhH**: High byte of the number of bytes in x-direction.
  
  → Number of data in x direction = dhH * 256 + dhL

- **dvL**: Low byte of the number of lines.
- **dvH**: High byte of the number of lines.
  
  → Number of data in y direction = dvH * 256 + dvL

- **d1~dk**: bitmap image data.
  
  → k = (dhH*256 + dhL) * (dvH*256 + dvL)

---

**! CAUTION**

There are no commas(,) and no space between each parameters.
Example

LD 0x11 0x02 0x40 0x02 0x08 0x00 0x20 0x00 0xFF ~ 0xFF

① x position : 0x02 * 0x100(256) + 0x11 = 0x211(529)
② y position : 0x02 * 0x100(256) + 0x40 = 0x240(576)
③ horizontal data number : 0x00 * 0x100(256) + 0x08 = 0x08(8)
④ vertical data number : 0x00 * 0x100(256) + 0x20 = 0x20(32)
⑤ bitmap data : total number = 8 * 32 = 256
Draw compression bitmap image data on specific position of image buffer

Syntax

\[ \text{LCp1p2xL xH yL yH dhL dhH dvL dvH d1~dk} \]

Parameters

- \( p1 \): Compression type
  - R: RLE

- \( p2 \): Color
  - 0x00: black
  - 0x01: Color(red or blue)

- \( xL \): Low byte of horizontal start position (X) [dot]
- \( xH \): High byte of horizontal start position (X) [dot]
  - \( \rightarrow \) Start position in x direction = \( xH \times 256 + xL \)

- \( yL \): Low byte of vertical start position (Y) [dot]
- \( yH \): High byte of vertical start position (Y) [dot]
  - \( \rightarrow \) Start position in y direction = \( yH \times 256 + yL \)

- \( dhL \): Low byte of the number of bytes in x-direction.
- \( dhH \): High byte of the number of bytes in x-direction.
  - \( \rightarrow \) Number of data in x direction = \( dhH \times 256 + dhL \)

- \( dvL \): Low byte of the number of lines.
- \( dvH \): High byte of the number of lines.
  - \( \rightarrow \) Number of data in y direction = \( dvH \times 256 + dvL \)

- \( d1 \sim dk \): Compression bitmap image data.
  - \( \rightarrow k = (dhH \times 256 + dhL) \times (dvH \times 256 + dvL) \)

\[ \text{! CAUTION} \]

There are no commas(,) and no space between each parameters.
Example

LC R 0x00 0x11 0x02 0x40 0x02 0x08 0x00 0x20 0x00 0xFF ~ 0xFF

① Compression type : R = RLE
② Color : 0x00 = Black
③ x position : 0x02 * 0x100(256) + 0x11 = 0x211(529)
④ y position : 0x02 * 0x100(256) + 0x40 = 0x240(576)
⑤ horizontal data number : 0x00 * 0x100(256) + 0x08 = 0x08(8)
⑥ vertical data number : 0x00 * 0x100(256) + 0x20 = 0x20(32)
⑦ bitmap data : total number = 8 * 32 = 256

RLE compression
This is the algorithm to compress the continuous data.
The compression is applied to 0x00 & 0xff data but not the others.
0xff 0x04 data is created if 0xff is repeated four times like 0x00 0x00 0x00 0x00.
In the same way, 0x00 0x04 is created by four times repeats of 0x00 such as 0x00 0x00 0x00 0x00.
Here is the example of compression.

Example) 0x78 0x78 0xff 0xff 0xff 0xff 0x22 0x00 0x00 0x00 0x00
          0x78 0x78 0xff 0x05 0x22 0x00 0x04
2-6-7 BMP

Send BMP format file directly to printer.
Just white/black BMP file is supported

Syntax

```
BMP p1, p2
```

Data string of *.bmp

Parameters

- **p1**: Horizontal position (X) [dot]
- **p2**: Vertical position (Y) [dot]

---

1. ↓ means ‘CR(+LF)’
2. There is comma(,) between p1 and p2.
3. After p2 (Before sending BMP data string) ‘CR(+LF)’ must follow.

---

Example

In dos mode,
```
COPY bmp.txt+image2.bmp+P.txt LPT1 /b
```

---

**Bmp.txt**

```
BMP200,200 ↓
```

**P.txt**

```
P1 ↓
```
2-7 Downloadable font Related Commands

Download fonts into the printer memory. Users can download special size or special design of ASCII font and use this font with T command.

1) DT
   Download True Type Font into Printer Memory

2) DD
   Delete downloaded fonts from memory

3) DI
   Print all downloaded fonts in memory and available memory space.
2-7-1 DT (Download True Type Font)

Description
Download windows system font into printer memory.

Syntax
\[ DT(p_1, p_2, 'Font Name', a_1 \ldots a_n, b_1 \ldots b_n, DATA_1 \ldots DATA_n) \]

Parameters
- \( p_1 \): Total number of characters to be downloaded: 0~255
- \( p_2 \): Font Height: 0~255
- Font name: A ~ Z

- \( a_n \): Character position in ASCII Table(0~255)
- \( b_n \): Font width(dots)
- \( (DATA_n) \): Character Bitmap Data

\[ \text{Total bytes of bitmap data} = p_2 \times (b_n + 7)/8 \text{ bytes} \]

Example
44 44 2a 0d 0a 44 54 60 14 27 41 27 \[ \text{DD}^*..DT..'A' \]

\[ p_1 \] \[ p_2 \] \[ Font name \]

20 0b \ldots
\[ a_1 \] \[ b_1 \] \[ DATA_1 : p_2 \times (b_1 + 7)/8 \text{ bytes} \]

21 0c \ldots
\[ a_2 \] \[ b_2 \] \[ DATA_2 : p_2 \times (b_2 + 7)/8 \text{ bytes} \]

\[ \ldots \]

Number of DATAn = 12 \times (8+7)/8 = 12 \text{ Bytes}
2-7-2 DD (Downloaded font Delete)

Description
Delete downloaded font from memory

Syntax
DD 'font name'

Parameters
'font name' : Indicate the Image in memory to be deleted.(A~Z)
- This name is Case Sensitive.
- By using *, all images in memory will be deleted.
- You can show the downloaded font list by DI command.

Example
DD'A' // Delete downloaded font A
DD*  // Delete all downloaded fonts in memory
2-7-3 DI (Downloaded font Information)

**Description**
Print list of downloaded font.

**Syntax**
```
DI
```

**Example**
```
DI
```

**Result**
```
<table>
<thead>
<tr>
<th>Name</th>
<th>w</th>
<th>h</th>
<th>c</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>16</td>
<td>25</td>
<td>128</td>
<td>6400</td>
</tr>
<tr>
<td>G</td>
<td>12</td>
<td>24</td>
<td>224</td>
<td>10752</td>
</tr>
</tbody>
</table>

Free Memory 179419
```

*w*: font width,  
*h*: font height,  
*c*: total number of characters
2-8 The Others

Commands not included in 1 to 7 categories.

1) @
   Printer initialization

2) PI
   Print information of printer configuration

3) CUT
   Enable/Disable Cutting Action

4) ^cp
   Check printer status and report 2bytes status data to host.

5) ^cu
   Check printer status and report 1byte status data to host.

6) ^PI
   Send various printer information to host.
2-8-1 @ (Initialize Printer)

**Description**
Initialize the printer

**Syntax**
@

2-8-2 PI (Printer Information)

**Description**
Print current printer setting.

**Syntax**
PI
2-8-3 CUT (Auto-cutter Enable/Disable)

**Description**
Enable or Disable Auto-cut action after printing by ‘P’ command.

**Syntax**
\[ \text{CUT} p1(p2) \]

**Parameters**
- **p1** : Cutting Action Enable/Disable
  - **y** : Enable cutter to act after printing is finished.
  - **n** : Disable cutter.
- **p2** : Cutting Period
  - Cutting Period means the number of pages between two cuttings.

⚠️ This command is not the cutting command itself but cutting enable/disable command.

⚠️ Cutting is executed immediately after printing is finished by P command if the cutter option is enabled by this CUT command.

⚠️ Last page is always cut.

---

**Example – p1(Cutter Enable/Disable)**

<table>
<thead>
<tr>
<th>Cutting is executed after Printing is finished</th>
<th>Cutting is not executed after Printing is finished</th>
</tr>
</thead>
<tbody>
<tr>
<td>T20... B130... BD... ... \text{CUT}y \text{P1}</td>
<td>T20... B130... BD... ... \text{CUT}n \text{P1}</td>
</tr>
</tbody>
</table>

**Example – p2(Cutting Period)**

- \text{CUT}y // Cut every page
- \text{CUT}y,1 // Cut every page
- \text{CUT}y,2 // Cut every 2 pages
- \text{CUT}y,4 // Cut every 4 pages
2-8-4 \(^{cp}\) (Check Printer Status and Report 2 bytes)

**Description**
Check printer status and report 2 bytes status data to host.

**Syntax**
\[^{cp}\]

**Return Value**

1. Format

\(<1^{\text{st}} \text{ Byte}> <2^{\text{nd}} \text{ Byte}>\)

2. Table

<table>
<thead>
<tr>
<th>Return Values Byte</th>
<th>Description</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Byte</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Paper Empty</td>
<td>0x80</td>
</tr>
<tr>
<td>6</td>
<td>Cover Open</td>
<td>0x40</td>
</tr>
<tr>
<td>5</td>
<td>Cutter jammed</td>
<td>0x20</td>
</tr>
<tr>
<td>4</td>
<td>Thermal Head(TPH) overheat.</td>
<td>0x10</td>
</tr>
<tr>
<td>3</td>
<td>Gap Detection Error(Auto-sensing failure)</td>
<td>0x08</td>
</tr>
<tr>
<td>2</td>
<td>Ribbon End Error</td>
<td>0x04</td>
</tr>
<tr>
<td>1</td>
<td>Not assigned</td>
<td>0x02</td>
</tr>
<tr>
<td>0</td>
<td>Not assigned</td>
<td>0x01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2nd Byte</th>
<th>Description</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>On building label to be printed in image buffer.</td>
<td>0x80</td>
</tr>
<tr>
<td>6</td>
<td>On printing label in image buffer</td>
<td>0x40</td>
</tr>
<tr>
<td>5</td>
<td>Issued label is paused in peeler unit.</td>
<td>0x20</td>
</tr>
<tr>
<td>4</td>
<td>Not assigned</td>
<td>0x10</td>
</tr>
<tr>
<td>3</td>
<td>Not assigned</td>
<td>0x08</td>
</tr>
<tr>
<td>2</td>
<td>Not assigned</td>
<td>0x04</td>
</tr>
<tr>
<td>1</td>
<td>Not assigned</td>
<td>0x02</td>
</tr>
<tr>
<td>0</td>
<td>Not assigned</td>
<td>0x01</td>
</tr>
</tbody>
</table>

3. Examples

<table>
<thead>
<tr>
<th>When Return Values are</th>
<th>Printer Status is</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Byte</td>
<td>2nd Byte</td>
</tr>
<tr>
<td>0x00</td>
<td>0x00</td>
</tr>
<tr>
<td>0x80</td>
<td>0x00</td>
</tr>
<tr>
<td>0x80</td>
<td>0x40</td>
</tr>
<tr>
<td>0x60</td>
<td>0x40</td>
</tr>
</tbody>
</table>
2-8-5 \(^{cu}\) (Check Printer Status and Report 1 byte)

**Description**
Check printer status and report 1 byte status data to host.

**Syntax**
\(^{cu}\)

**Return Value**
1. Format

\[
<1^{st} \text{ Byte}>
\]

2. Table

<table>
<thead>
<tr>
<th>Return Values</th>
<th>Description</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Byte</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Paper Empty</td>
<td>0x80</td>
</tr>
<tr>
<td>6</td>
<td>Cover Open</td>
<td>0x40</td>
</tr>
<tr>
<td>5</td>
<td>Cutter jammed</td>
<td>0x20</td>
</tr>
<tr>
<td>4</td>
<td>Thermal Head(TPH) overheat.</td>
<td>0x10</td>
</tr>
<tr>
<td>3</td>
<td>Gap Detection Error(Auto-sensing failure)</td>
<td>0x08</td>
</tr>
<tr>
<td>2</td>
<td>Ribbon End</td>
<td>0x04</td>
</tr>
<tr>
<td>1</td>
<td>Not assigned</td>
<td>0x02</td>
</tr>
<tr>
<td>0</td>
<td>Not assigned</td>
<td>0x01</td>
</tr>
</tbody>
</table>
2-8-6 ^PI (Send Printer information to host)

Description
Send various printer information such as model name, firmware version, statistics data or so to host.

Syntax
^PI(p1),(p2),(p3)

Parameters
p1 : items to be reported.
  0 : Model Name
  1 : Model Type : Disabled
  2 : F/W Version
  3: None
  4 : Mechanical conditions of printer

<table>
<thead>
<tr>
<th>p2</th>
<th>Item</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>TPH temperature</td>
<td>℃</td>
</tr>
<tr>
<td>1</td>
<td>Printing density (density)</td>
<td>–</td>
</tr>
<tr>
<td>2</td>
<td>Tear-off/cutter position</td>
<td>dot</td>
</tr>
</tbody>
</table>

Return Value Format

<table>
<thead>
<tr>
<th>Items</th>
<th>Return Format</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Name</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“SRP-770” + 0x0d + 0x0a</td>
</tr>
<tr>
<td>Model Type</td>
<td>Disabled</td>
<td></td>
</tr>
<tr>
<td>F/W Version</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“1.23” + 0x0d + 0x0a</td>
</tr>
<tr>
<td>TPH temperature</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“85” + 0x0d + 0x0a</td>
</tr>
<tr>
<td>Printing density</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“17” + 0x0d + 0x0a</td>
</tr>
<tr>
<td>Paper Width</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“832” + 0x0d + 0x0a</td>
</tr>
<tr>
<td>Paper Length</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“1200” + 0x0d + 0x0a</td>
</tr>
<tr>
<td>Gap Length</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“24” + 0x0d + 0x0a</td>
</tr>
<tr>
<td>Paper Horizontal Margin</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“10” + 0x0d + 0x0a</td>
</tr>
<tr>
<td>Paper vertical Margin</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“12” + 0x0d + 0x0a</td>
</tr>
<tr>
<td>Tear-off/cutter position</td>
<td>Character String + 0x0d + 0x0a</td>
<td>“+80” + 0x0d + 0x0a</td>
</tr>
</tbody>
</table>
3. Programming Example

3-1 Example) T_resident

SS3 // Set Speed to 5 ips
SD20 // Set Density level to 20
SW800 // Set Label Width 800
SOT // Set Printing Orientation from Top to Bottom
T26,20,0,1,1,0,0,N,N,'Font - 6 pt'
T26,49,1,1,0,0,N,N,'Font - 8 pt'
T26,81,2,1,0,0,N,N,'Font - 10 pt'
T26,117,3,1,0,0,N,N,'Font - 12 pt'
T26,156,4,1,0,0,R,N,'Font - 15 pt'
T26,200,5,1,0,0,N,N,'Font - 20 pt'
T26,252,6,1,0,0,N,N,'Font - 30 pt'
P1

Result

Font – 6 pt
Font – 8 pt
Font – 10 pt
Font – 12 pt
Font – 15 pt
Font – 20 pt
Font – 30 pt
3-2 Example) T_Rotate4

SS3
SW832
T300,500,4,1,1,0,0,N,N,'ABCDEFG'
T300,500,4,1,1,0,1,N,N,'ABCDEFG'
T300,500,4,1,1,0,2,N,N,'ABCDEFG'
T300,500,4,1,1,0,3,N,N,'ABCDEFG'
P1

Result
3-3 Example) V_resident

SS3
SD20
SW800
SOT

V50,100,U,25,25, +1,N,N,N,0,L,0,'Vector Font Test'
V50,200,U,35,35,-1,N,N,N,0,L,0,'Vector Font Test'
V50,300,U,35,35,+1,B,R,1,0,L,0,'Vector Font Test'
V50,400,U,45,25, +1,N,N,N,0,L,0,'Vector Font Test'
V50,500,U,25,45,+1,N,N,N,0,L,0,'Vector Font Test'
V50,700,U,65,65, +1,N,N,N,0,L,0,'ABCDEFGHJKLMNO'
V50,900,U,65,65,+1,N,N,N,0,L,0,'abcdefghijklmno'

Result

Vector Font Test

Vector Font Test

Vector Font Test

Vector Font Test

ABCDEFHJKLMNO

abcdefghijklmno
3-4 Example) V_Rotate4

SS3
SW832
V400,500,U,45,40,+1,N,N,N,0,L,0,'VECTOR FONT'
V400,500,U,45,40,+1,N,N,N,1,L,0,'VECTOR FONT'
V400,500,U,45,40,+1,N,N,N,2,L,0,'VECTOR FONT'
V400,500,U,45,40,+1,N,N,N,3,L,0,'VECTOR FONT'
P1

Result
3-5 Example) Code39

SM10,0

B178,196,0,2,6,100,0,0’1234567890’ // Caution : The position is not (178,196)
but (78,196).

B150,468,0,4,10,200,0,0’1234567890’
P1

Result

![Barcode Image]

![Barcode Image]
3-6 Example) BD1

SS3       // Set Speed to 5 ips
SD20      // Set Density level to 20
SW800     // Set Label Width to 800

BD50,50,750,500,B,20
T100,150,5,1,1,0,0,N,N,'Normal Mode'
T100,300,5,1,1,0,0,R,N,'Reverse Mode'

SOT
P1

Result

Normal Mode
Reverse Mode
3-7 Example) BD3

SS3  // Set Printing Speed to 5 ips
SD20 // Set Printing Density level to 20
SW800 // Set Label Width to 800

BD50,100,400,150,O  // Draw a block in Overwriting Mode
BD50,200,400,250,O
BD50,300,400,350,O
BD100,50,150,400,E  // Draw a block in Exclusive OR mode
BD200,50,250,400,E
BD300,50,350,400,E
BD500,200,700,400,O
BD510,210,670,370,D // Draw a block in Delete mode, namely Erase block area
BD100,600,350,1000,O
T50,700,5,1,1,0,0,N,N,'NORMAL' // Write Text data on image buffer
T50,800,5,1,1,0,0,N,N,'NORMAL'
BD110,780,340,900,E
T500,700,5,1,1,0,0,n,N,'TEST'
BD480,680,700,800,E

SOT  // Set Printing Orientation from Top to Bottom
P1  // Start Printing
Result

NORMAL

TEST

NORMAL
3-8 Example) BD4

SW800
SM10,0

BD100,300,550,330,O // Overwrite mode
BD200,200,250,430,O // Overwrite mode
BD400,200,450,430,E // Exclusive OR mode

P1

Result

![Diagram](attachment:image.png)
3-9 Example) BD5

CB
SW800
SM10.0
BD100,300,300,500,0
**BD400,300,700,500,B,30** // Box mode, additional parameter follows
P1

Result

![Diagram of square and rectangle](image-url)
3-10 Example) Slope

CB
SS3
SD20
SW8000

BD100,300,300,800,S,100 // Slope mode, additional parameter follows
BD600,300,400,800,S,100

P1

Result
3-11 Example) SW&SL

CB
SS3
SD20

SW800 // Set Label Width to 800
SL300,10,C // Continuous type
BD0,0,800,300,B,10
T30,40,4,1,1,0,0,N,N,'SW=800, SL=300, Continuous'
P1

SW600
SL500,10,C
BD0,0,600,500,B,10
T30,40,4,1,1,0,0,N,N,'SW=600, SL=500'
T30,100,4,1,1,0,0,N,N,'Continuous'
P1

SW400
SL800,10,C
BD0,0,400,800,B,10
T30,40,4,1,1,0,0,N,N,'SW=400'
T30,90,4,1,1,0,0,N,N,'SL=800'
T30,140,4,1,1,0,0,N,N,'Continuous'
P1
Result

SW = 800, SL=300, Continuous

SW = 600, SL=500, Continuous

SW = 400, SL = 800, Continuous
3-12 Example) TEST00_TS

TD 'Test00' // Template Delete
TS 'Test00' // Start Template Store

SV00,15,N,'Manufacturer :'
SV01,15,R,'Model Name :'

SV00,15,N,'Manufacturer :'

SV01,15,R,'Model Name :'

T50,100,3,1,0,0,0,N,N,'Manufacturer :V00' // Print variable V00 with some text string
T50,150,3,1,0,0,0,N,N,'Model Name :V01' // Print variable V01 with some text string
T50,300,3,1,0,0,0,N,N,V00 // Print variable V00 only
T50,350,3,1,0,0,0,N,N,V01 // Print variable V01 only

TE // End Template Store

TI // Print and show all templates in memory

Result

<table>
<thead>
<tr>
<th>Templates Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Test 1</td>
</tr>
<tr>
<td>2. Test0</td>
</tr>
</tbody>
</table>

Available template memory : 5.3Kbyte

Available template memory : 5.3Kbyte
3-13 Example) TEST00_TR

TR ‘Test00’ // Recall Stored template ‘Test00’

? // To get contents for variables used in ‘Test00’
SEM // Content for V00
SRP770 // Content for V01

P1 // Print

Result

<table>
<thead>
<tr>
<th>Manufacturer: SEM</th>
<th>No Justification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Name: SRP770</td>
<td>Right Justification</td>
</tr>
<tr>
<td>SEM</td>
<td>Right Justification</td>
</tr>
<tr>
<td>SRP770</td>
<td>No Justification</td>
</tr>
</tbody>
</table>
3-14 Example) TEST04_TS

TS'Test04'  // Start Template Store

CB  // Clear Image Buffer
SS3  // Set Speed to 5 ips
SD20  // Set Density level 20
SW800  // Set Label Width to 800
SOT  // Set Printing Orientation from Top to Bottom(Default)

SV00,15,L,'prompt'  // Declare variable V00, field size:15, Left justification
SV01,15,R,'prompt'  // Declare variable V01, field size:15, Right justification
SV02,15,C,'prompt'  // Declare variable V02, field size:15, Center justification
SV03,15,N,'prompt'  // Declare variable V03, field size:15, No justification
SV04,15,L,'prompt'  // Declare variable V04, field size:15, Left justification
SV05,15,R,'prompt'  // Declare variable V05, field size:15, Right justification
SV06,15,C,'prompt'  // Declare variable V06, field size:15, Center justification
SV07,15,N,'prompt'  // Declare variable V07, field size:15, No justification

T26,50,4,1,1,0,0,R,N,V00  // Print variable only
T26,100,4,1,1,0,0,R,N,V01
T26,150,4,1,1,0,0,R,N,V02
T26,200,4,1,1,0,0,R,N,V03
T26,250,4,1,1,0,0,R,N,'SRP770 :V04  // Print variable with fixed text data
T26,300,4,1,1,0,0,R,N,'SRP770 :V05
T26,350,4,1,1,0,0,R,N,'SRP770 :V06
T26,400,4,1,1,0,0,R,N,'SRP770 :V07

TE  // End Template Store
3-15 Example) TEST04_TR

TR'Test04' // Recall Template
?

// Start Get values for variables
A // data for variable V00
B // data for variable V00
C
D
E
F
G
H // data for variable V07

P1 // Start Printing

3-16 Example) IR1

IR130,400,'BIXOLON' // Recall stored image data
P1 // Printing

!!! Use the PCXDown utility when you download the pcx image file to printer memory.
Refer to IS command.

Result

BIXOLON®
3-17 Example) TEST10_TS

TS 'Test10' // Start Template Store

CB // Clear Image Buffer
SS3 // Set Speed to 5 ips
SD20 // Set Density to 20
SW800 // Set Label Width to 800
SOT // Set Printing Orientation from Top to bottom

SV00,15,C,'prompt' // Declare Variable 00
SV01,15,N,'prompt' // Declare Variable 01
SV02,10,N,'prompt' // Declare Variable 02

T130,250,5,1,0,0,R,N,V00 // Print Content of V00
T250,600,5,1,0,0,N,N,V01 // Print Content of V01
IR130,400,V02 // Use V02 as Image Name

TE // End Template Store

3-18 Example) TEST10_TR
(File location : CD\Testfile\Template\Test10\TEST10_TR.txt)

TR 'Test10' // Recall Template

? // Start Get data for variables
BIXOLON // data for V00
SRP770 // data for V01
BIXOLON // data for V02(Image Name)

P1 // Start Printing

BIXOLON

SRP770
3-19 Example) TEST11_TST

TS'Test11' // Start Template Store

CB // Clear Image Buffer
SS3 // Set Printing Speed to 5 ips
SD20 // Set Density to 20
SW800 // Set Label Width to 800
SOT // Set Printing Orientation from Top to Bottom
SC0,4,L,+1,’COUNTER1’ // Declare Counter 0, Field=4, Step:+1, Left Justi.
SC1,4,N,-1,’COUNTER2’ // Declare Counter 1, Field=4, Step:-1, No Justi.
T50,50,4,1,1,0,0,N,N,’Serial Number : ’C0 // Print Counter 0
T50,150,4,1,1,0,0,R,N,’Serial Number : ’C1 // Print Counter 1
TE // End of Template Store

Serial Number : 0001
Serial Number : 9999

Serial Number : 0002
Serial Number : 9998

3-20 Example) TEST11_TR

TR'Test11’ // Recall Template

? // Start Get values for variables
0001 // data for Counter 0
9999 // data for Counter 1
P3,1

Serial Number : 9999
Serial Number : 0001

? 9999
0001
P3,1

Serial Number : 0000
Serial Number : 0000
Serial Number : 9999
3-21 Example) SLCS_BIXOLON

SM10,21
SS3
SD20
SW832
SOT
CS0,0

BD18,14,798,164,O
T400,62,4,2,2,0,0,0,R,B,'BIXOLON'
T65,98,3,1,1,0,0,R,B,'BIXOLON Label'
T20,276,3,1,1,1,0,N,N,' BIXOLON'
T20,306,3,1,1,1,0,N,N,' Yeongtong Dong'
T20,336,3,1,1,1,0,N,N,' Sowon City,South Korea'
T22,218,4,1,1,0,0,N,B,'SHIP TO:'
BD18,410,784,415,O
BD553,197,558,413,O
B169,458,0,4,8,137,0,0,0,*1234567890*
T26,421,1,1,1,0,0,N,N,'POSTAL CODE:'
BD18,616,784,621,O
BD20,781,786,786,O
T503,798,1,1,1,0,0,N,N,'DESTINATION:'
T42,841,5,1,1,0,0,N,B,'30 Kg'
BD18,928,784,933,O
T25,798,1,1,1,0,0,N,N,'WEIGHT:'
T259,798,1,1,1,0,0,N,N,'DELIVERY NO:'
T23,630,1,1,1,0,0,N,N,'AWB:'
BD241,783,246,932,O
BD486,784,491,933,O
T274,841,5,1,1,0,0,N,B,'425518'
T104,627,3,1,1,0,0,N,N,'8741493121'
T565,841,5,1,1,0,0,N,B,'ICN'
B1127,672,4,4,8,90,0,0,0,'8741493121'
B2560,180,M,0,999,840,06810,7317,THIS IS A TEST OF MODE 0 STRUCTURED CARRIER MESSAGE ENCODING. THIS IS AN 84 CHAR MSG'
B280,960,P,30,10,0,0,0,1,3,14,0,'BIXOLON Label Printer SRP770, This is Test Printing.'
P1