

SATO

SATO Europe GmbH

PROGRAMMING MANUAL SATO RISC PRINTERS



**CL408/412, CL608/612 (VA), XL400/410,
M8400RV, M8459S, M8460S, M8485S, M8490S**

SATO Europe GmbH
Im Hülsenfeld 13
40721 Hilden
Germany
Tel.:+49 (0)2103 9592-0
Fax.:+49 (0)2103 55322

Issued June 1999

© Copyright 1999 SATO Europe GmbH

Contents

1. Introduction	7
1.1 The SATO RISC Programming Language	8
1.3 Selecting Protocol Control Codes	8
1.4 Using Basic	9
1.5 Printing with the Parallel Port	10
1.6 Printing with the RS232 Port	11
1.7 The Print Area	12
1.8 First Method: Using Base Reference Point	16
1.9 Second Method: Using Horizontal Offset Position	16
2. Command Codes	21
Bar Codes	22
Bar Codes, Expansion	27
Bar Codes, Variable Ratio	28
Base Reference Point	30
Characters, Custom-Designed	32
Character Expansion	34
Character, Fixed Spacing	36
Character Pitch	37
Character, Proportional Spacing	39
Clear Print Job(s) & Memory	40
Continuous Forms Printing	41
Copy Image Area	42
Cutter Command	44
Fonts U, S, M, OA, OB, XU, XS & XM	45
Font, Vector	47
Fonts WB, WL, XB & XL	49
Form Feed	51
Form Overlay, Recall	52
Form Overlay, Store	53
Graphics, Custom	54
Graphics, PCX	56
Job ID Store	57
Journal Print	58
Lines and Boxes	59

Line Feed	61
Mirror Image (From Firmware Version 1E2050, 1E3002 for 6xxVA)	63
Off-Line/Pause	72
Postnet	73
Print Darkness	74
Print Length, Expanded	75
Print Position	76
Print Quantity	78
Print Speed	79
Repeat Label	81
Replace Data (Partial Edit)	82
Reverse Image	84
Rotate, Fixed Base Reference Point	86
Rotate, Moving Base Reference Point	88
Sequential Numbering	90
Start/Stop Label	92
Calendar Option Commands	93
Calendar Increment	94
Calendar Print	96
Calendar Set	98
Memory Card Option Commands	99
Memory Card Function	
Clear Card Memory	100
Expand Memory Area	101
Fonts, TrueType Recall	102
Fonts, TrueType Store	103
Format/Field Recall	104
Format/Field Store	105
Graphics, Custom Recall	107
Graphics, Custom Store	109
Graphics, PCX Recall	111
Graphics, PCX Store	112
Initialize	113
Slot Select	114
Status	115

Custom Protocol Command Codes Download	117
3. Command Code Quick Reference	119
3.1 Instruction	119
3.2 Calendar Option Commands	129
3.3 Memory Card Option Commands	130
4. Bar Code Specifications	133
Codabar	134
Code 39	135
Interleaved Two of Five (I 2/5)	136
UPC-A/EAN-13	137
EAN-8	139
Industrial Two of Five	140
Matrix Two of Five	141
Code 128	142
MSI	145
Code 93	146
UPC-E	147
Bookland (UPC/EAN Supplements)	148
UCC-128	149
Postnet	151
Two-Dimensional Bar Codes	
Data Matrix, Data Format	152
Data Matrix, Print Data	154
Data Matrix, Sequential Numbering	155
Maxicode Vers. 3	157
PDF417	158
QR Code (Optional Special Firmware)	160
5. Interface Specifications	169
5.1 Overview	169
5.2 Interface Types	169
5.3 The Receive Buffer	170
5.4 RS232C Serial Interface	171
5.5 Centronics Parallel Interface	179
5.6 I/F Connector	180
6. Troubleshooting	187
6.1 Overview	187

6.2	Initial Checklist	188
6.3	Troubleshooting the Centronics (Parallel) Interface	188
6.4	Troubleshooting the RS232C (Serial) Interface	190
6.5	Error Signals M8400RV and M84S Series Printers (Some procedures are not NA for M-8459S)	191
6.6	Error Signals CL408/412	192
6.7	Error Signals CL608/612 (VA)	193
6.8	Error Signals XL400/410	194
7.	Custom Protocol Command Codes	197
7.1	Description	197
7.2	Download Command Structure	198
7.3	Download Procedure	199
8.	Custom Characters And Graphics	201
8.1	Custom-designed Character Example	201
8.2	PCX Graphics Example	211

1. Introduction

This manual presents the commands that are used with the SATO RISC Printers to produce labels with logos, bar codes and alphanumeric data. All of the RISC commands use the same syntax. Some commands reference a physical point on the label using horizontal and vertical dot reference numbers. The allowable range for these references is dependent upon the particular printer to accommodate different print widths and resolutions. These differences are noted in tables under the commands affected.

- Introduction
- Command Codes
- Command Codes Quick Reference
- Command Codes Specifications
- Interface Specification
- Troubleshooting
- Custom Protocol Command Codes
- Custom Characters And Graphics

1.1 The SATO RISC Programming Language

A programming language for a printer is a familiar concept to most programmers. It is a group of commands that are designed to use the internal intelligence of the printer. The commands, which are referred to as RISC Command Codes, contain non-printable ASCII characters (such as **<STX>**, **<ETX>**, **<ESC>**) and printable characters. These commands must be assembled into an organized block of code to be sent as one data stream to the printer, which in turn interprets the command codes and generates the desired label output. The programmer is free to use any programming language available to send the desired data to the SATO RISC printer.

The command codes used by the SATO RISC Printers are based upon "Escape" (1B hexadecimal) sequences. Typically there are four types of command sequences:

<ESC>{Command}

These commands generally tell the printer to perform a specific action, like "clear the memory."

<ESC>{Command} {Data}

Commands with this format tell the printer to perform a specific action which is dependent upon the following data, like "print X labels", where the value for X is contained in the data.

<ESC>{Command} {Parameter}

These commands set the operational parameters of the printer, like "set the print speed to 3."

<ESC> {Command} {Parameter} {Data}

Some commands can contain both Parameter and Data elements, such as "print a Code 39 symbol containing the data."

1.3 Selecting Protocol Control Codes

Protocol codes are the special control characters that prepare the printer to receive instructions. For example, the **<ESC>** character tells the printer that a command code will follow and the **<ENQ>** character asks for the printer status.

There are two pre-defined different sets of Protocol Control codes to choose from. Each set is made up of six special characters. The Standard Protocol Control codes are non-printable characters, and the Non-Standard Protocol Control codes are printable characters. The Non-Standard set may be useful on host computers using protocol converters or in an application where non-printable ASCII characters cannot be sent from the host. This manual uses the Standard Protocol Control codes for all of the examples. Alternately, the user may define and download a set of custom Protocol Control Codes.

The Protocol Control codes are selected by a DIP switch DSW2-7 (Refer to the Operation Manual.).

Control Character	Standard DSW2-7 OFF	Non-Standard DSW2-7 ON	Description
STX	02 Hex	7B Hex = {	Start of Data
ETX	03 Hex	7D Hex = }	End of Data
ESC	1B Hex	5E Hex = ^	Command Code to follow
NULL	7E Hex	7E Hex = ~	Cutter Command
ENQ	05 Hex	40 Hex = @	Get printer status, Bi-Com mode
CAN	18 Hex	21 Hex = !	Cancel print job, Bi-Com mode

1.4 Using Basic

It may be useful to test your printer using a BASIC program on a PC. You may also write your actual production programs in BASIC. Whatever the reason, if you will be working in BASIC, some of the following hints may help you get started:

1. Set the WIDTH of the output device to 255 characters to avoid automatically sending <CR> and <LF> characters after every line. The command string should be continuous and uninterrupted by <CR> and/or <LF> commands. The examples given in this manual are printed on separate lines because they will not fit on one line and do not contain any <CR> and/or <LF> characters. If these characters are needed, they are explicitly noted by the inclusion of <CR> and <LF> notations.
2. If you are using the printer's RS232 interface, it is necessary to set the COM port on the PC such that the CTS and DSR signals will be ignored. Send your OPEN "COM" statement in the following way: OPEN "COM1:9600,E,8,1,CS,DS" AS #1
3. This sets the RS232 communication parameters of the host PC's COM1 port for 9600 baud, Even parity, 8 Data bits, 1 Stop bit and directing the port to ignore the **CTS** and **DSR** control signals.

4. You may want to minimize keystrokes and program size by assigning the **<ESC>** character to a string variable since this character is used quite often.

The following two examples in BASIC show a typical example using these hints. Both of these examples use the Standard Protocol codes.

1.5 Printing with the Parallel Port

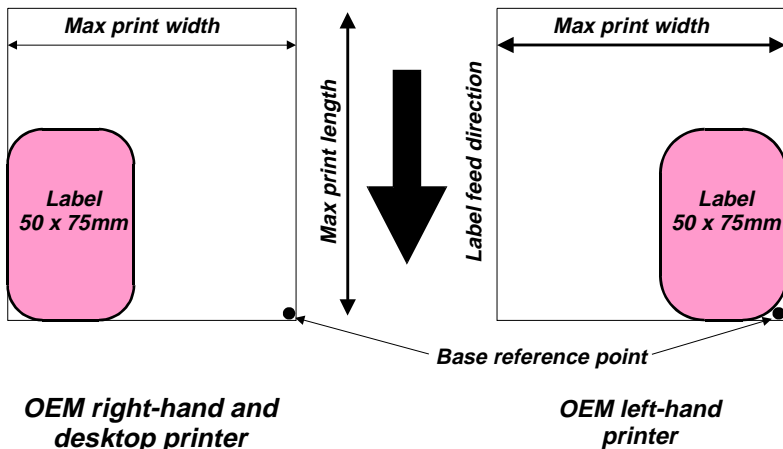
10	E\$=CHR\$(27)	Sets the "E\$" string as an <ESC> character
20	WIDTH "LPT1:",255	Sets the width of the output to 255 characters
30	LPRINT E\$;"A";	Sends an "<ESC>A" command code to the LPT1 parallel port
40	LPRINT E\$;"H400";E\$;"V100";E\$;"XL1SATO";	Sends the data "SATO" to be to be placed 400 dots horizontally and 100 dots vertically on the label and printed in the "XL" font.
50	LPRINT E\$;"Q1";	Instructs the printer to print one label.
60	LPRINT E\$;"Z";	Tells the printer that the last command has been sent. The printer can now create and print the job.

1.6 Printing with the RS232 Port

10	E\$=CHR\$(27)	Sets the "E\$" string as an <ESC> character.
20	OPEN "COM1:9600,N,8,1,CS,DS" AS #1;	Opens the COM1 port for output and sets the parameters as 9600 baud, No parity, 8 Data bits, 1 Stop bit and instructs the port to ignore the CTS and DSR control signals.
30	PRINT #1,CHR\$(2);	Sends an <STX> (ASCII Code a decimal "2") to the printer instructing it to prepare to receive a message.
50	PRINT #1,E\$;"A";	Sends an "<ESC>A" command code to Print Port #1 opened by statement 20 above.
60	PRINT #1, E\$;"H400",E\$;"V100";E\$;"XL1SATO";	Sends the data "SATO" to be placed 400 dots horizontally and 100 dots vertically on the label and printed in the "XL" autosmoothed font.
50	PRINT #1, E\$;"Q1";	Instructs the printer to print a quantity of one label.
60	PRINT #1, E\$;"Z";	Tells the printer that the last command has been sent. The printer can now create and print the job.
70	PRINT #1,CHR\$(3);	Sends an <ETX> (ASCII Code decimal "3") to the printer telling it that this is the end of the message.

1.7 The Print Area

The maximum print area for the various RISC printers are listed in the table on the next pages. Most of your label applications will not require tags/labels that fill the entire print area, therefore it is important to understand how to work with labels that do not use the entire print area. The goal is to help you avoid printing where no label exists, which may lead to print head damage, not to mention frustration when you cannot see the printed output.



The diagram above illustrates the maximum print area and a sample 50mm wide by 75mm long label placed within this area. As can be seen, your label will be oriented against the inside left edge of the printer as viewed from the front of the printer. The base reference point is located at the H1, V1 position of the print area in the normal print orientation (no rotation)

Print Area for CL Printers.

	CL 408	CL 412	CL 608 (VA)	CL 612 (VA)
Resolution	203 dpi 8 dot/mm	305 dpi 12 dot/mm	203 dpi 8 dot/mm	305 dpi 12 dot/mm
Max print width	832 dots 4.1 in. 104 mm	1248 dots 4.1 in. 104 mm	1216 dots 6.0 in. 152 mm	1984 dots 6.5 in. 165 mm
Max label width	5.0 in. 128 mm	5.0 in. 128 mm	7.0 in. 178 mm	7.0 in. 178 mm
Standard print length	1424 dots 7.0 in. 178 mm	2136 dots 7.0 in. 178 mm	1424 dots 7.0 in. 178 mm	2136 dots 7.0 in. 178 mm
Expanded print length using <ESC>AX	2848 dots 14.0 in. 356 mm	4272 dots 14.0 in. 356 mm	2848 dots 14.0 in. 356 mm	4272 dots 14.0 in. 356 mm
Expanded print length with memory card ¹				
128 Kb	1257 dots 6.2 in. 157 mm	838 dots 2.7 in. 69 mm	860 dots 4.2 in. 107 mm	528 dots 1.7 in. 44 mm
512 Kb	5038 dots 24.8 in. 629 mm	3359 dots 10.9 in. 278 mm	3447 dots 16.9 in. 430 mm	2113 dots 6.9 in. 176 mm
1 Mb	9999 dots ² 49.2 in. 1249 mm	6720 dots 22.0 in. 560 mm	6896 dots 33.9 in. 862 mm	4227 dots 13.6 in. 352 mm
2 Mb	9999 dots ² 49.2 in. 1249 mm	9999 dots ² 32.8 in. 833 mm	9999 dots ² 49.2 in. 1249 mm	8455 dots 27.7 in. 704 mm

1.) When a memory card is used to expand the print length, the card capacity is used instead, not in addition to, the internal memory.

2.) Limited by the digits in the command field.

Print Area for S-Type Printers

	M8485S	M4890S	M8460S	M8459S
Resolution	203 dpi 8 dot/mm	305 dpi 12 dot/mm	203 dpi 8 dot/mm	203 dpi 8 dot/mm
Max print width	1024 dots 5.0 in. 128 mm	1344 dots 4.4 in. 112 mm	1216 dots 6.0 in. 152 mm	896 dots 4.4 in. 112 mm
Max label width	5.2 in. 134 mm	5.2 in. 134 mm	6.4 in. 165 mm	5.2 in. 134 mm
Standard print length	1424 dots 7.0 in. 178 mm	2136 dots 7.0 in. 178 mm	1424 dots 7.0 in. 178 mm	1424 dots 7.0 in. 178 mm
Expanded print length using <ESC>AX	2848 dots 14.0 in. 356 mm	4272 dots 14.0 in. 356 mm	2848 dots 14.0 in. 356 mm	2848 dots 14.0 in. 356 mm
Expanded print length with memory card ¹				
128 Kb	1022 dots 5.0 in. 127 mm	778 dots 2.5 in. 64 mm	860 dots 4.2 in. 107 mm	1186 dots 5.7 in. 146 mm
512 Kb	4094 dots 20.1 in. 511 mm	3119 dots 10.1 in. 259 mm	3447 dots 16.9 in. 430 mm	4678 dots 23.0 in. 584 mm
1 Mb	8190 dots ² 40.3 in. 1023 mm	6240 dots 20.4 in. 520 mm	6896 dots 33.9 in. 862 mm	9360 dots 46.0 in. 1170 mm
2 Mb	9999 dots ² 49.2 in. 1249 mm	9999 dots ² 32.8 in. 833 mm	9999 dots ² 49.2 in. 1249 mm	9999 dots ² 49.2 in. 1249 mm

1.) When a memory card is used to expand the print length, the card capacity is used instead, not in addition to, the internal memory.

2.) Limited by the digits in the command field.

Print Area for XL and RV printers

	XL 400	XL410	M8400RV
Resolution	203 dpi 8 dot/mm	305 dpi 12 dot/mm	203 dpi 8 dot/mm
Max print width	800 dots 3.9 in. 100 mm	1200 dots 3.9 in. 100 mm	832 dots 4.1 in. 104 mm
Max label width	4.0 in. 102 mm	4.0 in. 102 mm	5.0 in. 128 mm
Standard print length	1920 dots 9.4 in. 240 mm	1920 dots 9.4 in. 240 mm	1424 dots 7.0 in. 178 mm
Expanded print length using <ESC>AX			2848 dots 14.0 in. 356 mm
Expanded print length with memory card ¹			
128 Kb	1168 dots 6.2 in. 146 mm	743 dots 6.2 in. 62 mm	1257 dots 6.2 in. 157 mm
512 Kb	4768 dots 24.8 in. 596 mm	2977 dots 24.8 in. 248 mm	5038 dots 24.8 in. 629 mm
1 Mb	9360 dots ² 49.2 in. 1170 mm	5956 dots 49.2 in. 496 mm	9999 dots ² 49.2 in. 1249 mm
2 Mb	9999 dots ² 49.2 in. 1249 mm	9999 dots ² 32.8 in. 833 mm	9999 dots ² 49.2 in. 1249 mm

- 1.) When a memory card is used to expand the print length, the card capacity is used instead, not in addition to, the internal memory.
- 2.) Limited by the digits in the command field.

There are two methods available to make sure your printed output will appear correctly on your label. They are as follows:

1.8 First Method: Using Base Reference Point

Send the Base Reference Point command as part of your data to the printer to set a new base reference point for your label. Calculate the distance (in dots) from the normal base reference point to the closest edge of the label. Issue the Base Reference Point command <ESC>A3 after the Start command in your data stream.

Input: <ESC>A

<ESC>A3HnnnnV0001.

nnnn: New horizontal Base Reference Point

Formula $nnnn = (\text{Max print width [mm]} - \text{Label width [mm]}) \times \text{Print head density [dots/mm]}$

Example CL612 printer with 50mm wide label

Max. print width: 165mm

Print head density: 12 dots/mm

$nnnn = (165\text{mm} - 50\text{mm}) \times 12 \text{ dots/mm}$

nnnn= 1380 dots

For the above escape sequence this results in

<ESC>A3H1380V0001

Note:

1. *This operation resets the reference point for all the following data in the non-volatile RAM*
2. *The <ESC>A3 Base Reference Point command can also shift the reference point in the negative direction, towards the outside edge of the label.*
3. *This calculation is not necessary for OEM left-hand printers.*

1.9 Second Method: Using Horizontal Offset Position

Use the normal base reference point from the print area and use the horizontal position for each field to properly locate it on the label. Calculate the distance (in dots) from the normal base reference point to the closest edge of the label.

Each <ESC>H command would have the value "nnnn" (see above example) added to it to correctly position each field.

>ESC>H position + nnnn

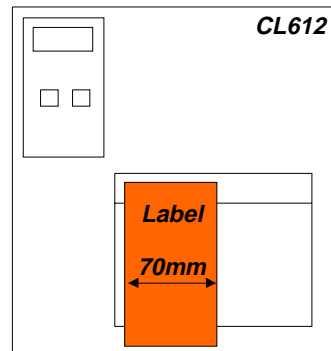
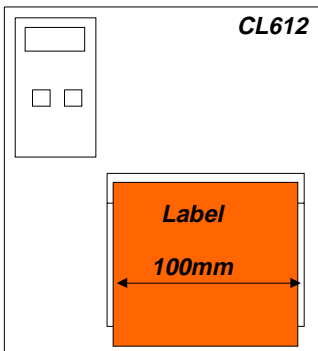
For CL612: <ESC>H position +1380

Note: *This calculation is not necessary for OEM left-hand printers.*

On the following 2 pages, the methods above are described with a sample tag/label output. These samples reflect how the printed information would appear on a 128mm wide label. If you want to test any of the sample tag/label outputs and are using tags/labels less than 128mm in width, we suggest that you add the Base Reference Point command to the data stream in order for the images to print on your tags/labels.

Warning *You must be careful not to print off the tag/label surface as the tag/label provides a heat sink for the print head elements. Doing so will cause irreparable damage to the head. This damage is not covered under the print head warranty. The addition of the base reference point command to the sample data stream may help to adjust the print for your tags/labels. See the following two examples or refer to the base reference point command description.*

For example, the following illustrates a sample data stream for a CL612 printer and the resulting label assuming a 70mm wide label:



If you are using a 70mm wide tag/label, the entire image may not appear on your tag/label. By adding the following Base Reference Point command to the second line of the data stream, the base reference point will be changed, causing the image to be shifted over toward the inside of the printer where it can be printed on the narrower label.

Example for first Method, using base reference point

Calculation: $nnnn = (165\text{mm} - 70\text{mm}) \times 12 \text{ dot/mm} = 1140$

<ESC>A

<ESC>A3H1140V0001

<ESC>H0070<ESC>V0100<ESC>L0202<ESC>XL0SATO

<ESC>H0050<ESC>V0200<ESC>B103100*SATO*

<ESC>H0170<ESC>V0310<ESC>L0101<ESC>XSSATO

<ESC Q1

<ESC>Z

Example for second Method, horizontal position offset

<ESC>A

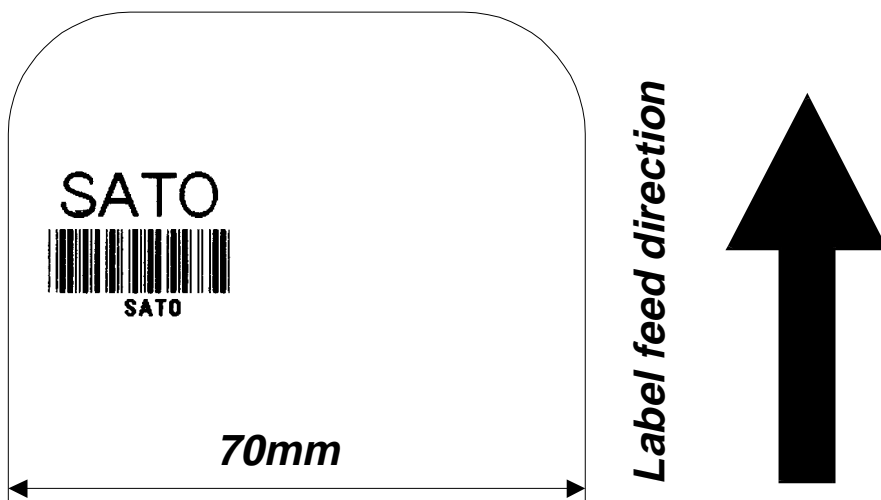
<ESC>H1210<ESC>V0100<ESC>L0202<ESC>XL0SATO

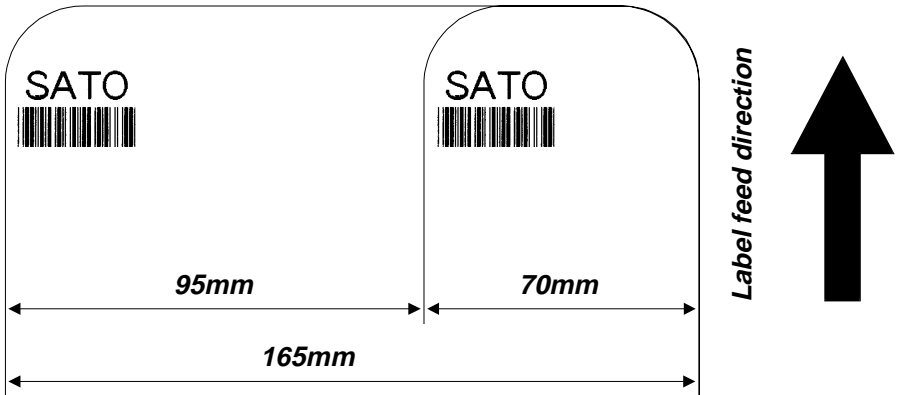
<ESC>H1190<ESC>V0200<ESC>B103100*SATO*

<ESC>H1310<ESC>V0310<ESC>L0101<ESC>XSSATO

<ESC Q1

<ESC>Z





The image is moved horizontally to the right 95mm (1140 dots) so that it can be printed on a 70mm wide label.

For more information, see the Base Reference Point command description.

This page is intentionally left blank.

2. Command Codes

This section contains all the RISC printer Command Codes. The commands must be sent to the printer in an organized fashion in order for the label(s) to print.

The objective of this chapter is to explain the different commands and provide examples of their usage, as well as to provide a detailed reference for programming the printers.

Each command begins on a separate page with its own heading. A uniform layout is used to help you find key information about each command. For each Command Code in this section, there will be a sample data input stream to the printer and the expected print output. By studying the examples, you can learn how to use the particular command within a whole block of printer code. Pay particular attention to the "Special Note" with each command to learn other important information.

The subject commands are highlighted in bold letters in the Reference Sheets. There are two parts of most, but not all, commands. The first is the command character which immediately follows the **<ESC>** code. It is always an upper case alpha or a special character (such as an "&" or a "%"). It is never a lower case alpha character. If the command requires additional variable information, it is represented by a group of lower case alpha characters immediately following the command character. For example, if an **aaaabb** is listed following the basic command, the printer will look for six characters immediately following the command. The first four would represent the value of **aaaa** and the next two the value of **bb**.

The maximum number of characters defined in a parameter is represented by the number of characters shown in the command structure. For example, a command followed by an **aaaa** can have up to four characters. In general, commands with only one parameter following the command can be entered without the leading zeroes. However, certain commands require the exact number of matching characters. A command with two parameters listed following the command code without a comma delimiter, such as **aaaabbbb** require the exact number of digits to be entered. If the value of **aaaa** is "800" and the value of **bbbb** is "300", then the parameters must be entered as "08000300". It is recommended that you make it a practice to always enter leading zeros to prevent any mistakes.

Note: *These examples assume the use of the Standard Protocol Command Codes and a parallel interface. The labels for all printers will be similar, but, because of different resolutions and print widths may be larger or scaled differently.*

Bar Codes

Command Structure:

1:3 narrow/wide bar ratio <ESC>**Babbcccd**

2:5 narrow/wide ratio <ESC>**BDabbcccd**

1:2 narrow/wide bar ratio <ESC>**Dabbcccd**

a: Bar Code Symbol

0: Codabar

1: Code 39

2: Interleaved 2 of 5 (I 2/5)

3: UPC-A / EAN-13

4: EAN-8

5: Industrial 2 of 5

6: Matrix 2 of 5

7: reserved

8: reserved

9: reserved

A: MSI

B: reserved

C: Code 93

D: reserved

E: UPC-E

F: Bookland

G: Code 128

I: UCC 128

bb: Number of dots (01-12) for narrow bar and narrow space

ccc: Bar height in dots (001-600)

d: UCC 128 only. Not used for other bar code types

0 No human readable text

1 Human readable at top

2 Human readable at bottom

Example: <ESC>BD103200

Placement: Immediately preceding data to be encoded

Default: None

Function:

To print bar code images on a label. With this command, there are 13 standard bar code symbologies available to be printed and three two dimensional symbols. Each of the bar codes are unique, and it is important to know the differences.

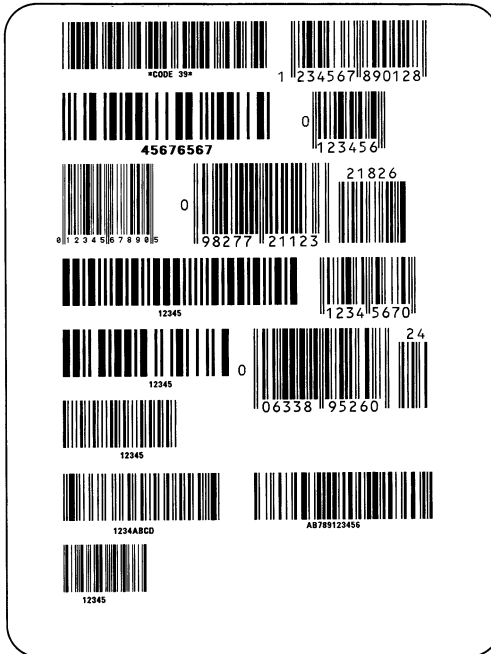
Input:

```
<ESC>A
<ESC>H0050<ESC>V0025<ESC>B103100*CODE 39*
<ESC>H0230<ESC>V0130<ESC>XS*CODE 39*
<ESC>H0050<ESC>V0175<ESC>BD20310045676567
<ESC>H0215<ESC>V0285<ESC>XM45676567
<ESC>H050<ESC>V0325<ESC>BD30215001234567890
<ESC>H050<ESC>V0525<ESC>BD50310012345
<ESC>H0250<ESC>V0635<ESC>XS12345
<ESC>H050<ESC>V0675<ESC>BD60310012345
<ESC>H0230<ESC>V0785<ESC>XS12345
<ESC>H050<ESC>V0825<ESC>BA03100123455
<ESC>H0170<ESC>V935<ESC>XS12345
<ESC>H050<ESC>V980<ESC>BC03100081234ABCD
<ESC>H0155<ESC>V1095<ESC>XS1234ABCD
<ESC>H050<ESC>V1130<ESC>B002100A12345B
<ESC>H090<ESC>V1240<ESC>XS12345
<ESC>H0530<ESC>V0025<ESC>BD303100123456789012
<ESC>H0590<ESC>V0525<ESC>BD4031001234567
<ESC>H0575<ESC>V175<ESC>DE03100123456
<ESC>H0550<ESC>V0225<ESC>OB0
<ESC>H0583<ESC>V0280<ESC>OB123456
<ESC>H0325<ESC>V0325<ESC>D30315009827721123
<ESC>L0101<ESC>H0295<ESC>V0400<ESC>OB0
<ESC>H0340<ESC>V0478<ESC>OB98277
<ESC>H0480<ESC>V0478<ESC>OB21123
<ESC>H630<ESC>V0365<ESC>BF0313021826
<ESC>H645<ESC>V0335<ESC>OB21826
<ESC>H0450<ESC>V0675<ESC>D30315000633895260
<ESC>L0101<ESC>H0415<ESC>V0750<ESC>OB0
<ESC>H0465<ESC>V0828<ESC>OB06338
```

<ESC>H0605<ESC>V0828<ESC>OB95260
<ESC>H0755<ESC>V0705<ESC>BF0314024
<ESC>H0770<ESC>V0675<ESC>OB24
<ESC>H0450<ESC>V0980<ESC>BG03100>GAB>B789>C123456
<ESC>H0560<ESC>V1085<ESC>XS AB789123456
<ESC>Q1<ESC>Z

Note: *Carriage Returns and Line Feeds have been added to the command listing for clarity and should not be included in the actual data stream*

Output:

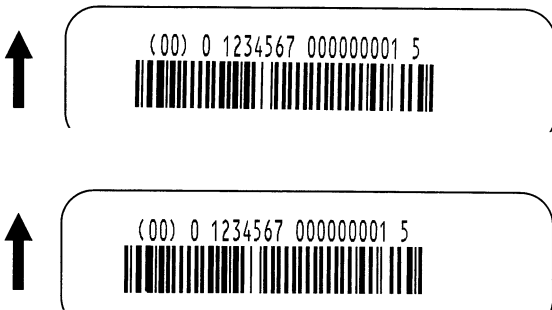


UCC-128 without Incrementing

Input:

```
<ESC>A  
<ESC>H0100<ESC>V0100<ESC>BI07150101234567000000001  
<ESC>Q2<ESC>Z
```

Output:

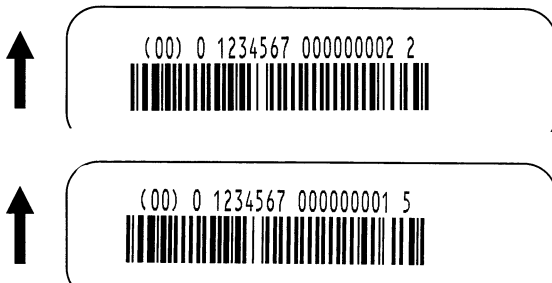


UCC-128 with Incrementing

Input:

```
<ESC>A  
<ESC>H0100<ESC>V0100<ESC>F001+001<ESC>BI0715010123456  
70000000001  
<ESC>Q2<ESC>Z
```

Output:



Note:

- 1. UPC and EAN bar codes are not affected by the different types of narrow to wide ratios. Instead, the <ESC>D command adds descender bars to these codes where needed to meet UPC specifications. The <ESC>BD command puts descender bars and human readable text below the symbol.**
- 2. The Code 128, UCC 128, MSI, and Code 93 bar codes are not affected by the narrow to wide ratios.**
- 3. The Codabar, Code 39, Industrial 2 of 5, and Matrix 2 of 5 bar codes are affected by the Character Pitch command. This command must be placed before the Bar Code command.**
- 4. Because of their unique characteristics, two-dimensional symbols are covered separately.**
- 5. For UCC128, the FNC1 code is automatically inserted and the Mod 10 and Mod 103 check digits are automatically calculated.**
- 6. For the MSI bar code, the check digit is not automatically calculated.**

Bar Codes, Expansion

Command Structure:

<ESC>**BW**aa**bbb**

aa: Expansion factor by which the width of all bars and spaces will be increased (01-12)

bbb: Bar height by dot (004-600 dots)

Example: <ESC>BW02100

Placement: Immediately follows the <ESC>BT command and precedes data to be encoded.

Default: None

Function:

This command works together with the <ESC>BT command to specify an expansion factor and the bar code height for the particular symbol being printed.

Input:

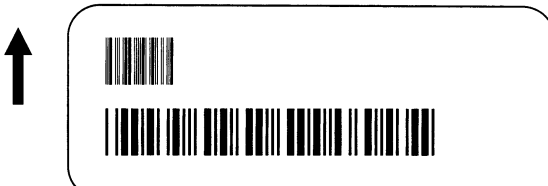
<ESC>A

<ESC>H0050<ESC>V0200<ESC>BD103100*M8400RV*

<ESC>H0050<ESC>V0050<ESC>BT001030103<ESC>BW03100123456

<ESC>Q1<ESC>Z

Output:



Note:

1. *This command must be preceded by the Variable Ratio Bar Codes <ESC>BT command.*
2. *The following bar codes will be affected by the Character Pitch command: Codabar, Code 39, Interleaved 2 of 5, Matrix 2 of 5.*

Bar Codes, Variable Ratio

Command Structure

<ESC>BTabbccddee

a: Bar Code Symbol
 0: Codabar
 1: Code 39
 2: Interleaved 2 of 5
 5: Industrial 2 of 5
 6: Matrix 2 of 5

bb: Narrow space in dots (01-99)

cc: Wide space in dots (01-99)

dd: Narrow bar in dots (01-99)

ee: Wide bar in dots (01-99)

Example: **<ESC>BT101030103**

Placement: Following print position commands and preceding
<ESC>BW

Default: Current setting

Function:

To print a bar code with a ratio other than those specified through the standard bar code commands (B,BD, and D). This is done through individual control of each of the bar code elements (bars, spaces) as shown above. Remember that this command only applies to the five bar code types shown.

Input:

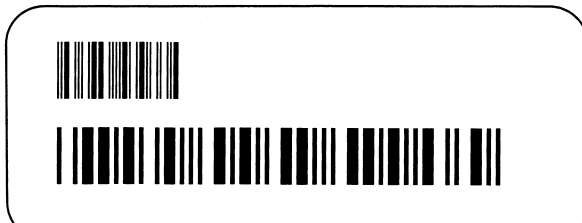
<ESC>A

<ESC>H0050<ESC>V0200<ESC>BD104100*M8400RV*

<ESC>H0050<ESC>V0050<ESC>BT001030103<ESC>BW03100123456

<ESC>Q1<ESC>Z

Output:



Note:

- 1. This command must be immediately followed by the <ESC>BW Bar Code Expansion command.**
- 2. You may use only one variable ratio bar code per label.**
- 3. If the data specified in this command is incorrect, the command is ignored and the ratio used will be based on the previous setting.**

Base Reference Point

Command Structure:

<ESC>A3H-aaaaVbbbb

-: This character is optional. When present, it specifies that the horizontal offset is in the negative direction. If it is left out the offset direction is positive.

aaaa: Horizontal Print Offset (see Note 5 for field range)

bbbb: Vertical Print Offset (see Note 5 for field range)

Example: **<ESC>A3H100V0050**

Placement: Preceding all images that are based on the new base reference point

Default: Current V and H offset setting in the printer configuration

Function:

To establish a new base reference point for the current label. The base reference point is the top left corner or "origin" from where all print position commands are based.

This command may be very helpful when using labels less than four inches wide to place images on the printable label surface. It may also be used to move images past preprinted fields on a label.

Input:

<ESC>A<ESC>L0202

<ESC>H0100<ESC>V0025<ESC>XMNORMAL REFERENCE POINT

<ESC>A3H0300V0075

<ESC>H0100<ESC>V0050<ESC>XMNEW REFERENCE POINT

<ESC>Q1<ESC>Z

Output:



NORMAL REFERENCE POINT

NEW REFERENCE POINT

Note:

1. Use of this command will set the Vertical/Horizontal Offset setting of the printer configuration until a new Base Reference Point command is issued or the setting is changed from the operator panel.
2. This command may be used more than once in a print job.
3. An alternative to using this command is to make changes to your current Horizontal and Vertical Print Position commands.

Example:

Let's say the current base reference point is H=1, V=1 and you wish to move all the fields on your label downward vertically by 150 dots. You could either (1) add the Base Reference Point command or (2) change all the vertical position commands by an additional 150 dots.

4. For a more detailed example of the Base Reference Point command, see "Print Area" in this section.
5. The allowable field ranges for this command are:

	CL 408	CL 412	CL 608 (VA)	CL 612 (VA)
Horizontal aaaa	-0832 to 0832	-1248 to 1248	-1216 to 1216	-1984 to 1984
Vertical bbbb	0001 to 1424	0001 to 2136	0001 to 1424	0001 to 2136

	M8485S	M8490S	M8460S	M8459S
Horizontal aaaa	-1248 to 1248	-1344 to 1344	-1216 to 1216	-0896 to 0896
Vertical bbbb	0001 to 1424	0001 to 2136	0001 to 1424	0001 to 2136

	XL 400	XL 410	M8400RV
Horizontal aaaa	-0800 to 0800	-1200 to 1200	-0832 to 0832
Vertical bbbb	0001 to 1920	0001 to 1920	0001 to 1920

Characters, Custom-Designed

Command Structure:

Store Command <ESC>**Tabcc**

Recall command <ESC>**Kab90cc**

a: Matrix size

1: 16x16 matrix

2: 24x24 matrix

b: Specifies the character encoding method for the data stream

H: Hexadecimal characters

B: Binary characters

cc: Memory location to store/recall the character. Valid memory locations are 21 to 52 (counting in Hex) or “!” to “R” in Binary

{data} Data to describe the character

Example: <ESC>T1H3F

<ESC>K1H903F

Placement: The Store command is typically sent in its own data stream to the printer, between the Start/Stop commands. The Recall command is sent in a secondary data stream to print the character, and follows any necessary position or size commands.

Default: None

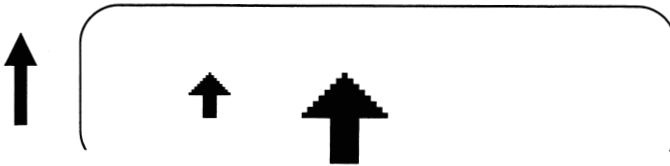
Function:

To allow for the creation, storage, and printing of custom characters, such as special fonts or logos. Up to 50 individual characters may be stored in the custom character volatile memory.

Input:

```
<ESC>T1H3F
0100038007C00FE01FF03FF87FFCFFFE07C007C007C007C007C00
7C007C007C0
<ESC>Z
<ESC>A
<ESC>H150<ESC>V100<ESC>L0505<ESC>K1H903F
W2434SH
```

Output:



Note:

1. *When printing the custom character using the Recall command, the character is affected by the following commands:*
2. *Character Expansion*
3. *Character Pitch*
4. *Line Feed*
5. *Rotate, Fixed Base Reference Point*
6. *Rotate, Moving Base Reference Point*
7. *The characters are stored in volatile memory and must be reloaded if the printer power is lost.*
8. *Do not use ASCII <CR> or <LF> characters (carriage return or line feed) as line delimiters within the graphic data or the actual image will not be printed as specified.*

Character Expansion

Command Structure:

<ESC>Laabb

aa: Multiple to expand horizontally (01-12)

bb: Multiple to expand vertically (01-12)

Example: <ESC>L0305

Placement: Preceding the data to be expanded

Default: <ESC>L0101

Function:

To expand characters independently in both the horizontal and vertical directions. The command allows you to enlarge the base size of each font (except the vector font) up to 12 times in either direction. Expanded characters are typically used for added emphasis or for long distance readability.

Input:

<ESC>A

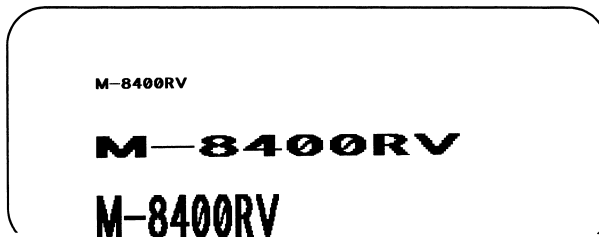
<ESC>A<ESC>H0100<ESC>V0100<ESC>XMM-8400RV

<ESC>H0100<ESC>V0200<ESC>L0402<ESC>XMM-8400RV

<ESC>H0100<ESC> V0300<ESC>L0204<ESC>XMM-8400RV

<ESC>Q1<ESC>Z

Output:



Note:

1. *This command will expand the following fonts: Fonts U, S, M, XU, XS, XM, OA & OB and fonts WB, WL, XB and XL.*
2. *This command will also affect the following commands:
Character Pitch
Characters, Custom-Designed*
3. *The Character Expansion value is in effect for the current print job until a new expansion command is specified.*
4. *The Line and Box command, if used within the data stream, may return all subsequent text to the default expansion of 1x1. Therefore, either send the Character Expansion command before all printed data, or send Line and Box commands last, preceding the <ESC>Q Quantity command.*

Character, Fixed Spacing

Command Structure:

<ESC>PR

Example: See Above

Placement: Preceding the data

Default: The default is Proportional Spacing.

Function:

To reset proportional spacing and place the printer to fixed spacing.

Input:

<ESC>A

<ESC>H0025<ESC>V0050<ESC>PS

<ESC>L0202<ESC>XMPROPORTIONAL SPACING

<ESC>H0025<ESC>V0130<ESC>PR

<ESC>L0202<ESC>XMFIXED SPACING

<ESC>Q1<ESC>Z

Output:



PROPORTIONAL SPACING
FIXED SPACING

Note:

1. *This command only works with the proportionally spaced fonts XU, XM, XS, XL and XB.*
2. *Once this command is sent in the data stream, it is in effect until the end of the print job or a reset command is sent.*

Character Pitch

Command Structure:

<ESC>**Paa**

aa: Number of dots between characters (00-99)

Example: <ESC>P03

Placement: Preceding the text to be printed

Default: <ESC>P02

Funktion:

To designate the amount of spacing (in dots) between characters.
This command provides a means of altering character spacing for label constraints or to enhance readability.

Input:

<ESC>A

<ESC>H0025<ESC>V0025<ESC>L0202<ESC>XB1M-8400RV

<ESC>H0025<ESC>V0125<ESC>L0 202<ESC>P20<ESC>XB1M-8400RV

<ESC>H0025<ESC>V0225<ESC>L0202<ESC>P40<ESC>XB1M-8400RV

<ESC>Q1<ESC>Z

Output:



M-8400RV
M - 8 4 0 0 R V
M - 8 4 0 0 R V

Note:

1. *This command is affected by the <ESC>L Character Expansion command . The character pitch is actually the product of the current horizontal expansion multiple and the designated pitch value.*

Example:

<ESC>L0304

<ESC>P03

Pitch = (03) x (03) = 9 dots

2. *To avoid confusion, you may want to include the <ESC>L Character Expansion command and this command together in your program.*
3. *This command affects fonts U, S, M, XU, XS, XM, OA & OB , fonts WB, WL, XB and XL, and the vector font.*
4. *Character Pitch will always revert to the default value unless it is specified before each new font command in the data stream.*
5. *This command also affects Codabar, Code 39 and Industrial 2 of 5 bar codes.*

Character, Proportional Spacing

Command Structure:

Reset to proportional spacing <ESC>**PS** (default)
Set to fixed spacing <ESC>**PR**
Example: See above
Placement: Preceding the data to be proportional spaced
Default: <ESC>**PS**

Function:

To specify the printing of proportional or fixed spacing for proportionally spaced fonts.

Input:

```
<ESC>A  
<ESC>H0025<ESC>V0050<ESC>PS  
<ESC>L0202<ESC>XMPROPORTIONAL SPACING  
<ESC>H0025<ESC>V0130<ESC>PR  
<ESC>L0202<ESC>XMFIXED SPACING  
<ESC>Q1<ESC>Z
```

Output:



PROPORTIONAL SPACING
FIXED SPACING

Clear Print Job(s) & Memory

Command Structure:

<ESC>*a

a: specifies which part of the memory has to be cleared. If the "a" parameter is not included with this command and the printer is in the multi-buffer mode, this command clears all print jobs in the printer memory, including the current print job.

If "a" is included with this command, it specifies the internal memory section to be cleared

T To clear the custom character memory

& To clear the form overlay memory

X To clear all internal memory

Example: <ESC>*

<ESC>*X

Placement: This command should be sent to the printer as an independent data stream.

Default: None

Function:

To clear individual memory or buffer areas of the printer.

Input:

<ESC>A

<ESC>*

<ESC>Z

Output:

There is no printer output as a result of this command. The current print job in the buffer will be terminated and all other print jobs in the buffer cleared.

Note:

1. ***See memory card functions for command variations used to clear data from the memory card.***
2. ***It is not necessary to clear the printer's memory between each print job.***
3. ***The primary purpose of this command is to clear all print jobs in the multi-buffer mode. The "a" parameter can be used in either the multi-buffer or single job mode to clear specific parts of the memory.***
4. ***When the "a" parameter is used, the section of memory specified will not be cleared until the label is printed.***

Continuous Forms Printing

No Command Structure

The printer locates the end of an adhesive label by sensing the backing between labels or through the use of an eye-mark (black rectangle on the reverse side of the backing). It locates the end of a tag from a notch, eye-mark, or a hole between tags. Both sensors should be disabled when printing continuous forms by placing the Label Sensor Selection switch (DSW3-3) in the ON position. See Operation Manual: Printer Configuration for instructions on configuring the printer using the front panel DIP switch array.

If you will be using continuous labels or tags, the printer must be told to stop feeding in another manner. The length is determined by the position of the last printed image on the label or tag. The printer will stop feeding when this last field is finished printing. The length may be increased with printed spaces (20 hexadecimal) if necessary. There is no command code to control label length.

Copy Image Area

Command Structure:

<ESC>WDHaaaaVbbbbXccccYdddd

aaaa: Horizontal position of the top left corner of the area to be copied

bbbb: Vertical position of the top left corner of the area to be copied

cccc: Horizontal length of the image area to be copied

dddd: Vertical length of the image area to be copied

Example: **<ESC>WDH0100V0050X0600Y0400**

Placement: Anywhere within the data stream, after specifying the location of the duplicate image.

Default: None

Function:

To copy an image from one location to another on the same label. This may be useful for duplicating individual fields or entire sections of the label with only one command.

Input:

<ESC>A

<ESC>H0050<ESC>V0050<ESC>E010<ESC>XM

M-8400RVM-8400RVM-8400RV

M-8400RVM-8400RVM-8400RV

M-8400RVM-8400RVM-8400RV

M-8400RVM-8400RVM-8400RV

<ESC>H0180<ESC>V0250<ESC>WDH0165V0050X0400Y0200

<ESC>Q1<ESC>Z

Output:



M-8400RV M-8400RV M-8400RV
M-8400RV M-8400RV M-8400RV
M-8400RV M-8400RV M-8400RV
M-8400RV M-8400RV M-8400RV

RV M-8400RV M-8400RV
RV M-8400RV M-8400RV
RV M-8400RV M-8400RV
RV M-8400RV M-8400RV

Note:

1. Use the *Print Position* commands (*V* and *H*) to locate the new area for the duplicate image.
2. Position of the new target area must not be inside the original image.
3. If you use the *Rotate* command, *V*, *H*, *X* and *Y* axis will be reversed.
4. If the reference area of the target image exceeds the print area, it will not be printed.
5. The allowable field ranges for this command are:

	CL 408	CL 412	CL 608 (VA)	CL 612 (VA)
Horizontal aaaa	0 to 0832	0 to 1248	0 to 1216	0 to 1984
Vertical bbbb	0001 to 1424	0001 to 2136	0001 to 1424	0001 to 2136

	M8485S	M8490S	M8460S	M8459S
Horizontal aaaa	0 to 1248	0 to 1344	0 to 1216	0 to 0896
Vertical bbbb	0001 to 1424	0001 to 2136	0001 to 1424	0001 to 2136

	XL 400	XL 410	M8400RV
Horizontal aaaa	0 to 0800	0 to 1200	0 to 0832
Vertical bbbb	0001 to 1920	0001 to 1920	0001 to 1920

Cutter Command

Command Structure:

<ESC><NUL>**aaaa**

aa: Number of labels to print between each cut (01-9999)

Example: <ESC><NUL>0002

Placement: Following the Print Quantity command <ESC>Q

Default: <ESC><NUL>0001 (if cutter enabled)

Function:

To control the cutting of labels when using a SATO cutter unit with the printer. This command allows the cutting of a multi-part tag or label at a specified interval within a print job.

Input:

<ESC>A

<ESC>H0020<ESC>V0020<ESC>WB1TEST LABEL<ESC>Q3

<ESC><NUL>0002

<ESC>Z

Output:

This set of commands will print 6 labels (3 x 2) with two labels between each cut.

Note:

- 1. You must have the optional printer Cutter to use this function. Contact your SATO representative for more information.**
- 2. To use this command, the printer configuration must have the cutter option enabled.**
- 3. If the cutter option has been enabled in the printer configuration and the cut value (aa = 00), the cutter is inactive. The <NUL> represents the ASCII 00 Hex character.**
- 4. When using the Cutter command, the total number of labels printed is the product of the cut value and the print quantity. For example, if the cut value is 2, and the print quantity is 20, then 20 sets of two labels will be printed.**

Fonts U, S, M, OA, OB, XU, XS & XM

Command Structure:

Font XU	<ESC> XU
Font U	<ESC> U
Font XS	<ESC> XS
Font S	<ESC> S
Font XM	<ESC> XM
Font M	<ESC> M
Font OA	<ESC> OA
Font OB	<ESC> OB
Example:	See above
Placement:	Preceding the data to be printed
Default:	None

Function:

To print text images on a label. These are eight of the built-in fonts available on the printer. All matrices include descenders.

Dot Matrix for Non-Proportional Fonts

U	5W x 9H dot matrix
S	8W x 15H dot matrix
M	13W x 20H dot matrix
OA	15W x 22H dot matrix
OB	20W x 24H dot matrix

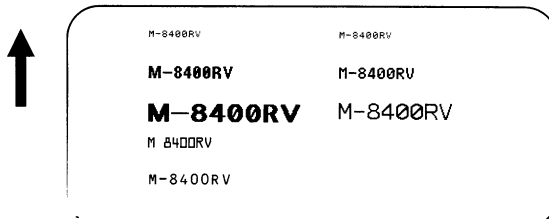
Dot Matrix for Proportional Fonts

XU	5W x 9H dot matrix
XS	17W x 17H dot matrix
XM	24W x 24H dot matrix

Input:

```
<ESC>A<ESC>PS
<ESC>H0001<ESC>V0100<ESC>L0202<ESC>XUM-8400RV
<ESC>H0001<ESC>V0175<ESC>L0202<ESC>XSM-8400RV
<ESC>H0001<ESC>V0250<ESC>L0202<ESC>XMM-8400RV
<ESC>H0001<ESC>V0325<ESC>L0101<ESC>OAM-8400RV
<ESC>H 0001<ESC>V0400<ESC>L0101<ESC>OBM-8400RV
<ESC>H0300<ESC>V0100<ESC>L0202<ESC>UM-8400RV
<ESC>H0300<ESC>V0175<ESC>L0202<ESC>SM-8400RV
<ESC>H0300<ESC>V0250<ESC>L02 02<ESC>MM-8400RV
<ESC>Q1<ESC>Z
```

Output:



Note:

1. Characters may be enlarged through the use of the Character Expansion command.
2. Character spacing may be altered through the use of the Character Pitch command. The default is 2 dots between characters. It is recommended to use a spacing of 5 dots for OCR-A and 1 dot for OCR-B.
3. You may also create custom characters or fonts. See the <ESC>T Custom-Designed Characters command.
4. A font must be defined for each field to be printed. There is no default font.
5. Fonts U, S, M, OA and OB are identical to fonts U, S, M, OA and OB on the SATO M-8400 printer.
6. The proportionally spaced fonts XU, XS, XM, XL and XA can be printed with fixed spacing using the <ESC>PS Proportional Space command.
7. The matrices of the OA and OB fonts are scaled so that they will remain a constant size according to the OCR-A and OCR-B specifications when printed on different resolution printers.

	8 dot /mm	203 dpi	12 dot/mm	305 dpi
OA-Font	15 dots W x 22 dots H		22 dots W x 33 dots H	
OB-Font	20 dots W x 24 dots H		30 dots W x 36 dots H	

Font, Vector

Command Structure:

Specify Vector Font <ESC>**\$a,b,c,d**

Data for Vector Font <ESC>**\$(data)**

- a:** Font spacing
 A Helvetica Bold (proportional spacing)
 B Helvetica Bold (fixed spacing)
- b:** Font width (50-999)
- c:** Font height (50-999 dots)
- d:** Font variation (0-9) as follows:
 0 Standard
 1 Standard open (outlined)
 2 Gray (mesh) pattern 1
 3 Gray (mesh) pattern 2
 4 Gray (mesh) pattern 3
 5 Standard open, shadow 1
 6 Standard open, shadow 2
 7 Standard mirror image
 8 Italic
 9 Italic open, shadow

Example: <ESC>**\$A,100,200,0**<ESC>**\$(123456)**

Placement: Immediately preceding data to be printed.

Default: None

Function:

To specify printing of the unique SATO vector font. The vector font allows large characters to be printed with smooth, round edges. Each character is made of a number of vectors (or lines), and will require slightly more printer compiling time.

Input:

```
<ESC>A
```

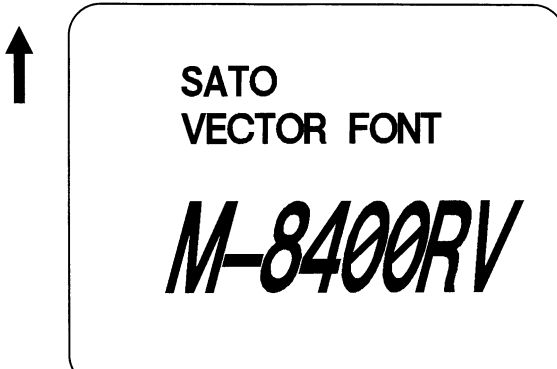
```
<ESC>H0100<ESC>V0100<ESC>$A,100,100,0<ESC>$(SATO
```

```
<ESC>H0100<ESC>V0200<ESC>$(VECTOR FONT
```

```
<ESC>$A,200,300,8<ESC>H0100<ESC>V0350<ESC>$(M-8400RV
```

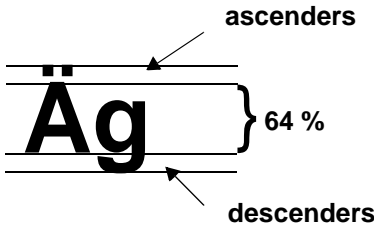
```
<ESC>Q1<ESC>Z
```

Output:



Note:

- 1. *If the font size designation is out of the specified range, a default value of 50 is used.*
- 2. *The Pitch command can be used with Vector fonts.*
- 3. *The font width and height values include ascenders, descenders and other space.*



- 4. *A font must be defined for each field to be printed. There is no default font.*

Fonts WB, WL, XB & XL

Command Structure:

Font WB	<ESC> WB a
Font XB	<ESC> XB a
Font WL	<ESC> WL a
Font XL	<ESC> XL a
a:	0 Disables auto-smoothing of font 1 Enables auto-smoothing of font (see Note below)
Example:	<ESC>WB1123456
Placement:	Preceding the data to be printed
Default:	None

Function:

To print text images on a label. These are the four auto-smoothing fonts available on the printer.

Dot Matrix for Non-Proportional Fonts

WB	18W x 30H dot matrix
WL	28W x 52H dot matrix

Dot Matrix for Proportional Fonts

XB	48W x 48H dot matrix
XL	48W x 48H dot matrix

Input:

```
<ESC>A<ESC>PS
<ESC>H0001<ESC>V0100<ESC>WB0M-8400RV
<ESC>H0001<ESC>V0185<ESC>WB1M-8400RV
<ESC>H0001<ESC>V0270<ESC>WL0M-8400RV
<ESC>H0001<ESC>V0355<ESC>WL1 M-8400RV
<ESC>H0300<ESC>V0100<ESC>XB0M-8400RV
<ESC>H0300<ESC>V0185<ESC>XB1M-8400RV
<ESC>H0300<ESC>V0270<ESC>XL0M-8400RV
<ESC>H0300<ESC>V0355<ESC>XL1M-8400RV
<ESC>Q1<ESC>Z
```

Output:



M-8400RV	M-8400RV
M-8400RV	M-8400RV
M-8400RV	M-8400RV
M-8400RV	M-8400RV

Note:

1. *Auto-smoothing (when enabled) is only effective if the character expansion rate is at least (3) times in each direction.*
2. *Characters may be enlarged through the use of the <ESC>L Character Expansion command.*
3. *Character spacing may be altered through the use of the <ESC>A Character Pitch command.*
4. *A font must be defined for each field to be printed. There is no default font.*
5. *The proportionally spaced fonts XU, XS, XM, XL and XB can be printed with fixed spacing using the <ESC>PR Proportional Space command.*

Form Feed

Command Structure:

<ESC>**A(space)**

<ESC>**Z**

Example: See above

Placement: Separate data stream sent to printer

Default: None

Function:

To feed a blank tag or label, which is the equivalent of a “form feed”

Input:

<ESC>A (space)

<ESC>Z

Output:

Blank label or tag

Form Overlay, Recall

Command Structure:

<ESC>/

Example: See above

Placement: Must be preceded by all other data and placed just before the Print Quantity command (<ESC>Q)

Default: None

Function:

To recall the label image from the form overlay memory for printing. This command recalls a stored image from the overlay memory. Additional or different data can be printed with the recalled image.

Input:

<ESC>A

<ESC>H0001<ESC>V0125

<ESC>STHIS IS THE STORED IMAGE WITH A BARCODE

<ESC>H0001<ESC>V0165<ESC>B103100*12345*

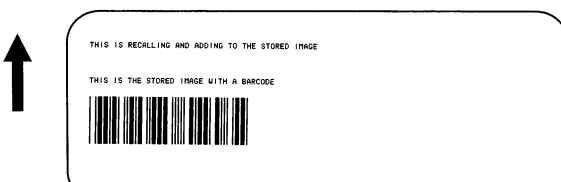
<ESC>&<E SC>Z

<ESC>A<ESC>H0001<ESC>V0050

<ESC>STHIS IS RECALLING AND ADDING TO THE STORED IMAGE<ESC>/

<ESC>Q1<ESC>Z

Output:



Note: *The overlay is stored using the <ESC>& Form Overlay Store command.*

Form Overlay, Store

Command Structure:

<ESC>&

Example: See above

Placement: Must be preceded by all other data and placed just before the Stop command (<ESC>Z)

Default: None

Function:

To store a label image in the volatile form overlay memory. Only one label image may be stored in this memory area at a time.

Input:

<ESC>A

<ESC>H0001<ESC>V0125

<ESC>STHIS IS THE STORED IMAGE WITH A BARCODE

<ESC>H0001<ESC>V0165<ESC>B103100*12345*

<ESC>&

<ESC>Z

Output:

There is no output from this command. It stores the label image in the overlay buffer.

Note:

- 1. Remember that this storage is volatile. Therefore, if the printer loses power, the overlay must be sent again.**
- 2. The overlay is recalled using the <ESC>/ Form Overlay Recall command.**
- 3. Form overlays do not have to be recompiled each time they are called to be printed and therefore may result in much faster print output.**

Graphics, Custom

Command Structure:

<ESC>Gabbbccc(data)

a: Specifies format of data stream to follow

B Binary format

H Hexadecimal format

bbb: Number of horizontal 8 x 8 blocks (001-248)

ccc: Number of vertical 8 x 8 blocks (001-267)

(data): Hex data to describe the graphic image

Example: <ESC>GH006006

Placement: May be placed anywhere within the data stream after the necessary position commands.

Default: None

Function:

To create and print custom graphics (logos, pictures, etc.) on a label. The graphic image may be printed along with other printed data to enhance label appearance or eliminate the need for preprinted label stock. Using a dot-addressable matrix, design the graphic image in 8 dot by 8 dot blocks, then send it in a binary format to the printer.

Input:

<Esc>A

<Esc>H100<Esc>V100

<Escsc>GH006006

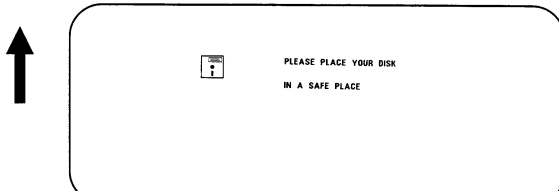
```

FFFFFFFFFFFFFF FFFFFFFFFFFFFFFF C000000000003
C00000000003 C000FFFFFFFFF3 C00080000013
C00080000013 C0009FFFFFF13 C00080000013
C00080000013 C0009FFFFFF13 C00080000013
C00080000013 C0009FFFFFF3 C00000000003
C00000000003 C00000000003 C00000000003
C00000000003 C00000000003 C00003C00003
C00007E00003 C0000FF00003 C0000FF00003
C0000FF00003 C0000FF00003 C00007E00003
C00003C00003 C00000000003 C00000000003
C00000000003 C00000000003 C00001800003
C00003C00003 C00003C00003 C00003C00003
C00003C00003 C00003C00003 C00003C00003

```

```
C00003C00003 C00003C00003 C00003C00003
C00003C00003 C00001800003 C00000000003
C00000000003 FFFFFFFFFFFFFFFF FFFFFFFFFFFFFFFF
<ESC>H0300<ESC>V0100<ESC>XSPLEASE PLACE YOUR DISK
<ESC>H0300<ESC>V0150<ESC>XSIN A SAFE PLACE
<ESC>Q1<ESC>Z
```

Output:



Note:

1. **Do not use ASCII <CR> or <LF> characters (carriage return or line feed) as line delimiters within the graphic data or the actual image will not be printed as specified.**
2. **A custom graphic can be enlarged by the <ESC>L Character Expansion command.**
3. **Use an optional Memory Card to expand the print length.**
4. **To store graphic images in an optional memory card, see the Memory Card Functions section.**
5. **The binary format reduces the transmission time by 50%.**

Graphics, PCX

Command Structure:

<ESC>GPaaaa,(data)

aaaa: Number of bytes to be downloaded

Example: <ESC>GP32000, ... data...

Placement: Anywhere within the job data stream

Default: None

Function:

To allow the creation and printing of graphic images using a PCX file format.

Input:

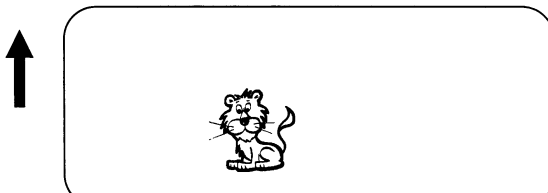
<ESC>A

<ESC>V0150<ESC>H0100<ESC>GP03800,(...Data...)

<ESC>Q1

<ESC>Z

Output:



Note:

1. **The maximum number of bytes that can be downloaded is 32K (compressed). The number specified by this command includes the PCX header information. The maximum size of the uncompressed PCX file is 64K. If the uncompressed file exceeds 64K, the graphic will not print.**
2. **Only black and white PCX files can be downloaded.**
3. **The image created by this command cannot be rotated.**
4. **The file size specified by this command is the DOS file size in bytes.**

Job ID Store

Command Structure:

<ESC>IDaa

aa: Job ID assigned (01 to 99)

Example: <ESC>ID09

Placement: Immediately following the <ESC>A in the job data stream.

Default: None

Function:

To add an identification number to a job. The status of the job can then be determined using the ENQ command in the Bi-Com status mode (See Interface Specifications for more information).

Input:

```
<ESC>A
<ESC>ID02
. . . Job . . .
<ESC>Z
```

Output:

There is no printer output as a result of this command.

Note:

1. *Works only in Bi-Com mode. The Job ID number must be stored before Bi-Com status mode can be used.*
2. *If more than one ID number is sent in a single job, the last number transmitted will be used, i.e.*

```
<ESC>A
<ESC>ID01
. . . . .
<ESC>ID02
. . . . .
```

Journal Print

Command Structure:

<ESC>J

Example: See above

Placement: Immediately following <ESC>A

Default: None

Function:

To print text in a line by line format on a label. By specifying this command, you automatically select Font XS with a Character Expansion of 2x2. You also establish a base reference point of H2,V2. The character pitch is 2 dots and the line gap is 16 dots. Simply issue an ASCII <CR> at the end of each text line.

Input:

<ESC>A

<ESC>J WITH THE JOURNAL FEATURE

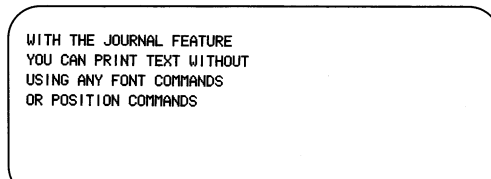
YOU CAN PRINT TEXT WITHOUT

USING ANY FONT COMMANDS

OR POSITION COMMANDS

<ESC>Q1<ESC>Z

Output:



Note:

1. *Journal mode assumes a maximum label width . Otherwise, you may print where there is no label and damage your print head.*
2. *It is effective only for the current print job.*

Lines and Boxes

Command Structure:

Line: <ESC>**FWaabcccc**

aa: Width of horizontal line in dots (01-99)

b: Line orientation
 H Horizontal line
 V Vertical Line

cccc: Length of line in dots (see Note for max length)

Box: <ESC>**FWaabbVccccHdddd**

aa: Width of horizontal side in dots (01-99)

bb: Width of vertical side in dots (01-99)

cccc: Length of vertical side in dots (see Note 2 for max length)

dddd: Length of horizontal side in dots (see Note 2 for max length)

Example: <ESC>FW02H0200

Placement: Following the necessary positioning commands

Default: None

Function:

To print horizontal lines, vertical lines, and boxes as images on the label.

Input:

<ESC>A

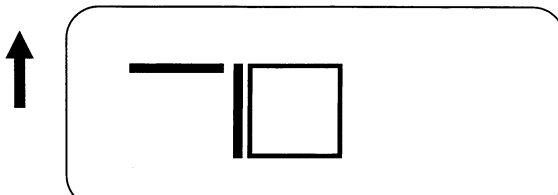
<ESC>H0100<ESC>V0100<ESC>FW20H0200

<ESC>H0320<ESC>V0100<ESC>FW20V0200

<ESC>H0350<ESC>V0100<ESC>FW1010H0200V0200

<ESC>Q1<ESC>Z

Output:



Note:

1. *It is recommended that all lines and boxes be specified in the normal print direction.*
2. *The maximum allowable lengths are as follows:*

	CL 408	CL 412	CL 608 (VA)	CL 612 (VA)
Horizontal aaaa	0 to 0832	0 to 1248	0 to 1216	0 to 1984
Vertical bbbb	0001 to 1424	0001 to 2136	0001 to 1424	0001 to 2136

	M8485S	M8490S	M8460S	M8459S
Horizontal aaaa	0 to 1248	0 to 1344	0 to 1216	0 to 0896
Vertical bbbb	0001 to 1424	0001 to 2136	0001 to 1424	0001 to 2136

	XL 400	XL 410	M8400RV
Horizontal aaaa	0 to 0800	0 to 1200	0 to 0832
Vertical bbbb	0001 to 1920	0001 to 1920	0001 to 1920

Line Feed

Command Structure:

`<ESC>Eaaa`

aaa: Number of dots (001-999) between the bottom of the characters on one line to the top of the characters on the next line

Example: `<ESC>E010`

Placement: Preceding the text that will use the line feed function

Default: None

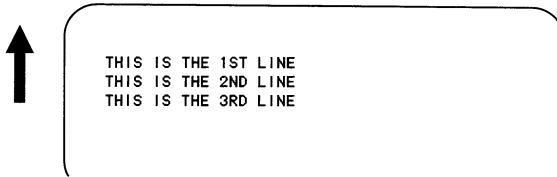
Function:

To print multiple lines of the same character size without specifying a new print position for each line. With the Line Feed command, specify the number of dots you want between each line. Then, send an ASCII `<CR>` at the end of each line of text. The printer automatically identifies the size of the last character, moves down the number of dots specified, and begins printing the next line.

Input:

```
<ESC>A
<ESC>E010<ESC>H0050<ESC>V0050<ESC>L0202<ESC>S
THIS IS THE 1ST LINE
THIS IS THE 2ND LINE
THIS IS THE 3RD LINE
<ESC>Q1<ESC>Z
```

Output:



```
THIS IS THE 1ST LINE
THIS IS THE 2ND LINE
THIS IS THE 3RD LINE
```

Note:

1. *This command can be used for text as well as for bar codes.*
2. *It is effective only for the current data stream.*
3. *When printing lines or boxes in the same data stream with the Line Feed command, the Lines and Boxes command should be specified last, preceding <ESC>Q Quantity command.*
4. *This command is invalid only if the value specified is zero.*
5. *The rotation command can be used with this command.*
6. *Following this command with a <CR> character will allow you to print with auto line feed. In this case, the print position will be determined from the value specified in the command and the H value set in the printer. However, if you specify several H values after this command, the print position will be determined by the H value last specified. You must redefine the font to be used after each H command.*

Mirror Image

(From Firmware Version 1E2050, 1E3002 for 6xxVA)

Command Structure:

<ESC>**RM**aaaa, bbbb

aaaa: Horizontal position of the top left corner of the image area to be mirrored 0008 ~ 9999 dot.

bbbb: Vertical position of the top left corner of the image area to be mirrored 0008 ~ 9999 dot.

Example: <ESC>RM0150, 0050

Function/Placement:

This command allows you to print a data as mirrored, not only whole or part of data.

If you specify the image area with horizontal <H> and vertical <V> position, both specified before <RM> command are valid for start position.

A data with <RM> command parameter over the printable area is not mirrored as treated as a command error. Also, a print job with <RM> command without print date is treated as a command error.

<RM> command can be used without command parameters for horizontal and vertical position. In this case, data placed before this command are mirrored.

This command can be used with <%> command, while do not use with <R> command regardless of its effect.

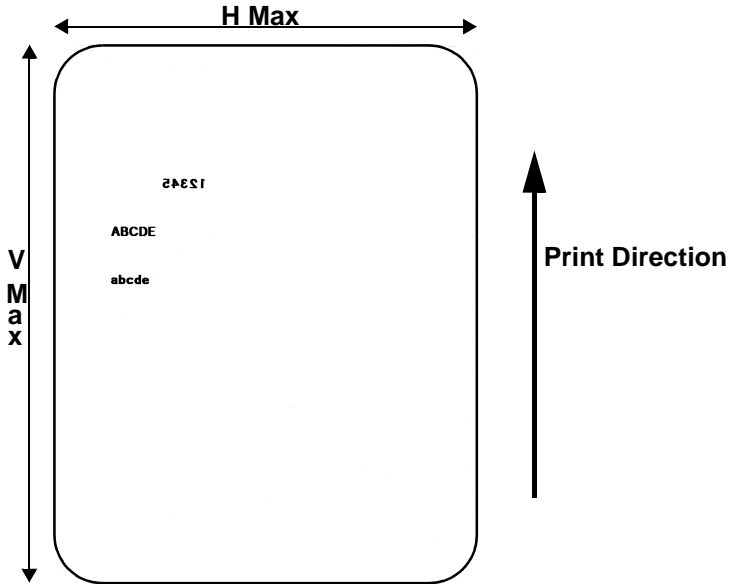
Note:

- 1. This command can not be used with commands requiring re-editing print image, like sequential numbering, RTC, Copy image or so. Also, it can not be used with commands for data storage, like graphic, format, etc.***
- 2. This command should not be used with bar codes to avoid the scanning problem caused by refractive index through the glass or the window.***
- 3. Do not specify this command multiple times in a job.***

Input:

```
<ESC>A  
<ESC>V100<ESC>H100<ESC>XM12345  
<ESC>V200<ESC>H100<ESC>XMABCDE  
<ESC>V300<ESC>H100<ESC>XMabcde  
<ESC>V100<ESC>H100<ESC>RM0200, 0080  
<ESC>Q1<ESC>Z
```

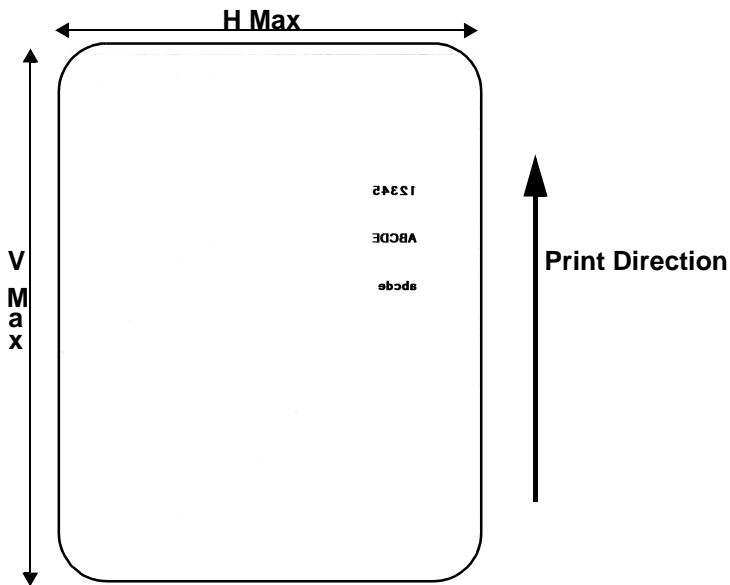
Output:



Input:

```
<ESC>A  
<ESC>V100<ESC>H100<ESC>XM12345  
<ESC>V200<ESC>H100<ESC>XMABCDE  
<ESC>V300<ESC>H100<ESC>XMabcde  
<ESC>RM  
<ESC>Q1<ESC>Z
```

Output:



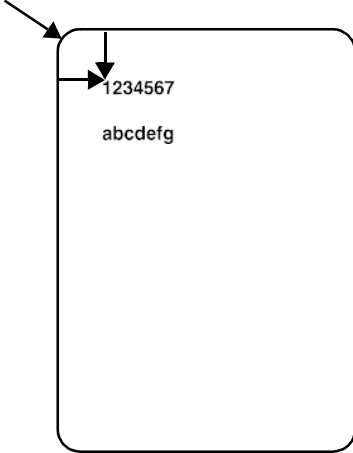
1. Normal Direction

Input:

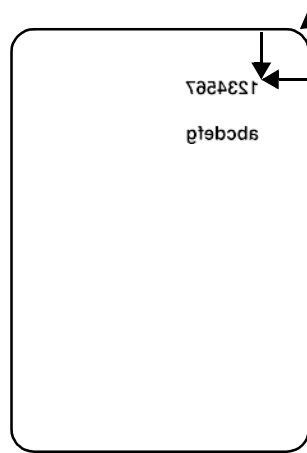
```
<ESC>A  
<ESC>V100<ESC>H100<ESC>XM12345  
<ESC>V300<ESC>H100<ESC>XMabcdefgh  
<ESC>RM  
<ESC>V200<ESC>H100<ESC>XMABCDEF  
<ESC>Q1<ESC>Z
```

Output:

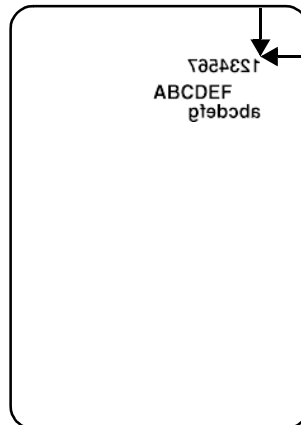
Base reference point w/o <RM>



Base reference point w <RM>



Mirror position as normal direction



2. Mirror Image with Rotation

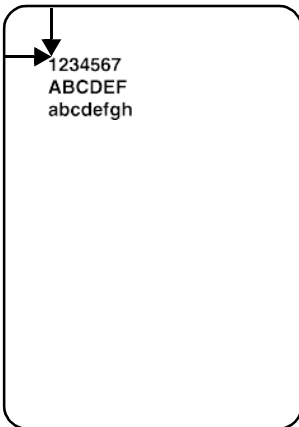
Mirror position is changed if <RM> is specified with <%> command.

Input:

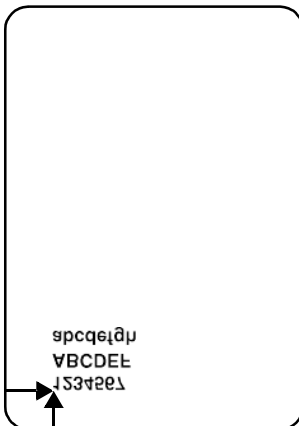
```

<ESC>A
<ESC>V100<ESC>H100<ESC>XM12345
<ESC>V200<ESC>H100<ESC>XMABCDEF
<ESC>V300<ESC>H100<ESC>XMabcdefgh
<ESC>%1<ESC>RM
<ESC>Q1<ESC>Z
    
```

Output:



Mirror Position

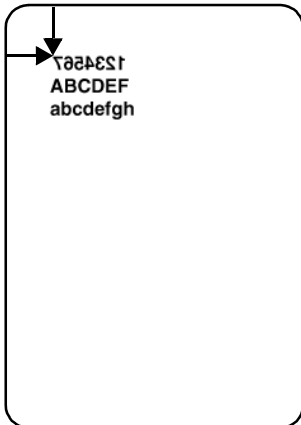


3. Mirror Image with specifying area

Input:

```
<ESC>A
<ESC>V100<ESC>H100<ESC>XM12345
<ESC>V100<ESC>H100<ESC>RM0150, 0050
<ESC>V200<ESC>H100<ESC>XMABCDEF
<ESC>V300<ESC>H100<ESC>XMabcdefgh
<ESC>Q1<ESC>Z
```

Output:

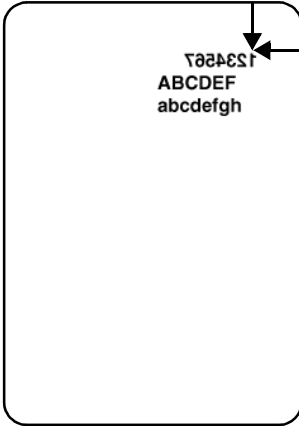


4. Specifying of a mirror command between print fields

Input:

```
<ESC>A
<ESC>V100<ESC>H100<ESC>XM12345
<ESC>RM
<ESC>V200<ESC>H100<ESC>XMABCDEF
<ESC>V300<ESC>H100<ESC>XMabcdefgh
<ESC>Q1<ESC>Z
```

Output:

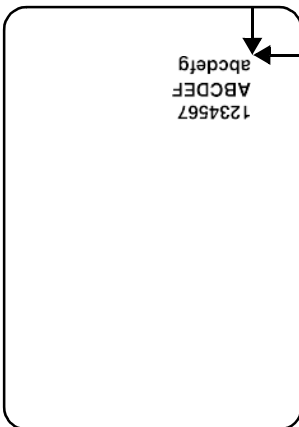


5. In case of specifying mirror command after the last print field.

Input:

```
<ESC>A  
<ESC>V100<ESC>H100<ESC>XM12345  
<ESC>V200<ESC>H100<ESC>XMABCDEF  
<ESC>V300<ESC>H100<ESC>XMabcdefgh  
<ESC>RM  
<ESC>Q1<ESC>Z
```

Output:



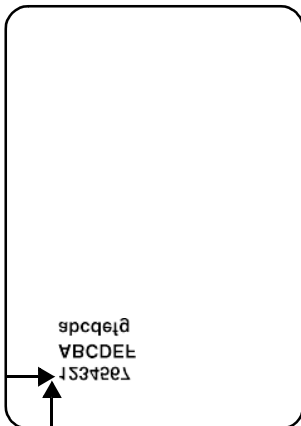
6. <%> specifying effects on <RM> command only

In this case a base mirror position is only changed

Input:

```
<ESC>A
<ESC>V100<ESC>H100<ESC>XM12345
<ESC>V200<ESC>H100<ESC>XMABCDEF
<ESC>V300<ESC>H100<ESC>XMabcdefgh
<ESC>%1<ESC>RM
<ESC>Q1<ESC>Z
```

Output:

**7. <%> specifying effects on <RM> command and print fields**

#1

Input:

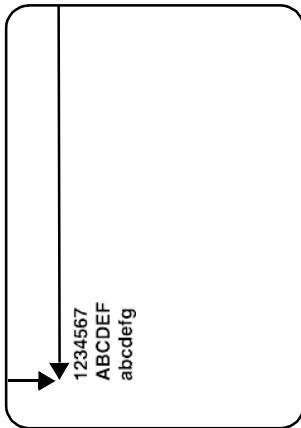
```
<ESC>A
<ESC>%1
<ESC>V800<ESC>H100<ESC>XM12345
<ESC>V800<ESC>H100<ESC>XMABCDEF
<ESC>V800<ESC>H100<ESC>XMabcdefgh
<ESC>RM
<ESC>Q1<ESC>Z
```

#2

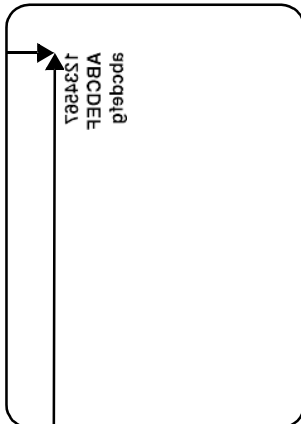
Input:

```
<ESC>A  
<ESC>%1  
<ESC>V800<ESC>H100<ESC>XM12345  
<ESC>V800<ESC>H100<ESC>XMABCDEF  
<ESC>V800<ESC>H100<ESC>XMabcdefgh  
<ESC>RM  
<ESC>Q1<ESC>Z
```

Output:



Mirror position



Off-Line/Pause

Command Structure:

<ESC>@,nn...n

nn...n: Optional message to be displayed on the LCD.
Maximum of 32 characters

Example: See above

Placement: Anywhere in the print job between <ESC>A and
<ESC>Z

Default: None

Function:

To specify the printer to come to an off-line state. When used within a print job, the printer goes off-line after finishing the print job.

Input:

<ESC>A

<ESC>@,Load Blue Labels and place printer On-Line

. . . Job . . .

<ESC>Z

Output:

There is no printer output for this command. The printer is placed in the Off-Line mode as soon as the current print job is finished.

Note:

- 1. You must press the LINE key on the front panel to return the printer to an On-Line status.**
- 2. Remember, when using this command, that if the print job specifies <ESC>Q10, all ten labels will print before the printer will go off-line.**

Postnet

Command Structure:

<ESC>**BPn...n**

n...n: 5 digits (Postnet-32 format)
 6 digits (Postnet-37 format)
 9 digits (Postnet-52 format)
 11 digits (Postnet-62, Delivery Point format)

Example: <ESC>BP123456789

Placement: Immediately preceding the data to be encoded

Default: None

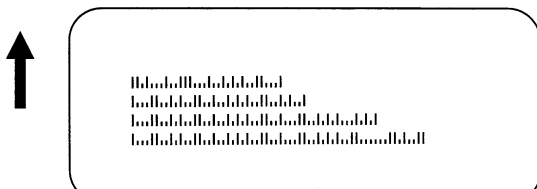
Function:

To print Postnet bar codes

Input:

```
<ESC>A
<ESC>H0100<ESC>V0120<ESC>BP94089
<ESC>H0100<ESC>V0160<ESC>BP123456
<ESC>H0100<ESC>V0200<ESC>BP1 23456789
<ESC>H0100<ESC>V0240<ESC>BP12345678901
<ESC> Q1<ESC>Z
```

Output:



Note:

1. *If the number of data digits does not match those listed, the command is ignored.*
2. *Only numeric data will be accepted.*

Print Darkness

Command Structure:

<ESC>#Ea

a: Print darkness value
(see Note for allowable setting)

Example: <ESC>#E2

Placement: Must be placed immediately after <ESC>A and immediately before <ESC>Z in its own separate data stream

Default: See Note

Function:

To specify a new print darkness setting. This command allows software control of the darkness setting for unique media and ribbon combinations.

Input:

<ESC>A

<ESC>#E2

<ESC>Z

Output:

There is no printer output for this command.

Note:

1. *This becomes the new setting in the printer configuration for all subsequent print jobs, unless changed. The setting is stored in non-volatile memory and is not affected by cycling power.*
2. *This command will overwrite the Display Setup.*
3. *The allowable Print Darkness settings are as follows:*

	M8459S, CL 408/CL 412 M8400RV	M8485S, M8490S, M8460S, CL 608/612 (VA), XL400/410
Darkness Settings	1, 2, <u>3</u> , 4 or 5	1, <u>2</u> or 3
	Default = 3 Minimum = 1 Maximum = 5	Default = 2 Minimum = 1 Maximum = 3

Print Length, Expanded

Command Structure:

Normal length <ESC>**AR**

Expanded length <ESC>**AX**

AX: Sets the print length to 14" (356 mm)

AR: Resets the maximum print length to 7" (178 mm)

Example: See above

Placement: Must follow the Start Code command and be in it's own separate data stream.

Default: <ESC>**AR**

Function:

To double the maximum print length (in feed direction) for a label.

Input:

<ESC>**A**

<ESC>**AX**

<ESC>**Z**

<ESC>**A**

<ESC>**H0050**<ESC>**V0100**<ESC>**WB1EXPAND TO:**

<ESC>**H0050**<ESC>**V2700**<ESC>**WB114 INCHES**

<ESC>**Q1**<ESC>**Z**

<ESC>**A**

<ESC>**AR**

<ESC>**Z**

Note:

- 1. AX is effective until AR is sent to reset the printer to its standard print length, or until the printer is repowered.**
- 2. It may be included in an independent data stream to specify the size of the maximum print area.**

Print Position

Command Structure:

Horizontal Position <ESC>Haaa

Vertical Position <ESC>Vbbb

aaaa: Number of dots horizontally from the base reference point (1 to maximum)

bbbb: Number of dots vertically from the base reference point (1 to maximum)

Example: <ESC>H0020<ESC>V0150

Placement: Preceding any printed field description of lines/boxes, fonts, bar codes or graphics.

Default: <ESC>H0001
 <ESC>V0001

Function:

The Horizontal and Vertical commands specify the top left corner of a field or label, using the current base reference point as an origin. They also establish a reference point for subsequent fields until the next horizontal and/or vertical print position command is issued.

Input:

<ESC>A

<ESC>H0025<ESC>V0050<ESC>L0303<ESC>MM-8400RV

<ESC>H0100<ESC>V0150<ESC>MM-8400RV

<ESC>Q1<ESC>Z

Output:



M-8400RV
M-8400RV

Note:

- 1. The print position of a field is affected by both the Rotate (<ESC>R and <ESC>A3) commands.**
- 2. For print lengths greater than 14 inches an optional Memory Card must be used.**
- 3. If any part of an image is placed past the maximum number of dots for standard length and or the capacity of the memory card, that part of the image will be lost.**
- 4. If any part of an image is placed past maximum allowable dots across the label, that part of the image will be lost.**
- 5. If you attempt to print where there is no paper, you may damage the print head.**
- 6. For these commands, the leading zeroes do not have to be entered. The command V1 is equivalent to V0001.**

Print Quantity

Command Structure:

<ESC>Qaaaaaa

aaaaaa: Total number of labels to print (1-999999)

Example: <ESC>Q500

Placement: Just preceding <ESC>Z, unless <ESC>NUL exists, then preceding that. This command must be present in every print job.

Default: None

Function:

To specify the total number of labels to print for a given print job.

Input:

<ESC>A

<ESC>H0100<ESC>V0100<ESC>WB1M-8400RV

<ESC>Q3

<ESC>Z

Output:

Three labels containing the data "M-8400RV" will be printed.

Note:

1. *To pause during a print job, you must press the LINE key on the Operator Panel.*
2. *To cancel a print job, you must turn off the printer, or you may send the <CAN> code if using the Bi-Com mode. Multi-Buffer jobs can be cleared with the <ESC>* Clear Print Job(s) and Memory command.*
3. *When used with the <ESC>F Sequential Numbering command, the Print Quantity value should be equal to the total number of labels to be printed.*
4. *If you do not specify a Print Quantity, the printer will not print a label.*
5. *For this command, leading zeroes do not have to be entered. The command Q1 is equivalent to Q000001.*

Print Speed

Command Structure:

<ESC>CSa

a: Designates the speed selection
See Note for allowable settings

Example: <ESC>CS6

Placement: Must be placed immediately after <ESC>A and immediately before <ESC>Z in its own separate data stream

Default: As previously set in the printer configuration

Function:

To specify a unique print speed through software for a particular label. This allows flexibility in finding the best performance and quality for the particular label format, media, and ribbon. All subsequent labels will print at this speed unless the speed is changed with this command or through the Operator Panel.

Input:

<ESC>A

<ESC>CS5

<ESC>Z

Output:

There is no printer output for this command. It sets the print speed of the printer.

Note:

1. *This becomes the new setting in the printer configuration for all subsequent print jobs, unless changed. The setting is stored in non-volatile memory and is not affected by cycling the power.*
2. *This command will overwrite the Display Settings.*
3. *The allowable Print Speed settings are as follows:*

Printer	a	Speed
CL 408	1	2 ips, 50 mm/s
CL 412	2	3 ips, 75 mm/s
	3	4 ips, 100 mm/*s
	4	5 ips, 125 mm/s
	5	6 ips, 150 mm/s
CL 608 (VA)	4	4 ips, 100 mm/s
CL 612 (VA)	6	6 ips, 150 mm/s*
M8490S	8	8 ips, 200 mm/s
M8460S		

M8459S	2	2 ips, 50 mm/s
	3	3 ips, 75 mm/s
	4	4 ips, 100 mm/s*
	5	5 ips, 125 mm/s
M8485S	4	4 ips, 100 mm/s
	6	6 ips, 150 mm/s*
	8	8 ips, 200 mm/s
	10	10 ips, 250 mm/s
M8400RV	1	2 ips, 50 mm/s
	2	4 ips, 100 mm/s
	3	6 ips, 150 mm/s*
	4	8 ips, 200 mm/s
	5	10 ips, 250 mm/s
XL 410	4	4 ips, 100 mm/s
	5	5 ips, 125 mm/s*
	6	6 ips, 150 mm/s
XL 400	5	5 ips, 125 mm/s
	6	6 ips, 150 mm/s*
	7	7 ips, 175 mm/s
	8	8 ips, 200 mm/s

* = default

Repeat Label

Command Structure:

<ESC>**C**

Example: See above

Placement: Must be placed immediately after <ESC>A and immediately before <ESC>Z in its own separate data stream

Default: None

Function:

To print duplicate of the last label printed

Input:

<ESC>A

<ESC>C

<ESC>Z

Output:

A duplicate of the previous label will be printed.

Note: *This command will have no effect if the power to the printer was cycled off and back on since printing the previous label.*

Replace Data (Partial Edit)

Command Structure:

<ESC>**0** (<ESC>**zero**)

Example: See above

Placement: Must follow <ESC>A and precede all other print data

Default: None

Function:

To replace a specified area of the previous label with new data. This command will cause the previous label to print along with any changes specified within the current data stream.

Input:

<ESC>A

<ESC>H0025<ESC>V0020<ESC>W0BCL 608

<ESC>H0025<ESC>V0085<ESC>WB1M8485S

<ESC>H0025<ESC>V0150<ESC>WL0M8490S

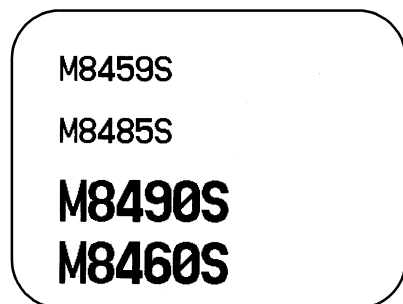
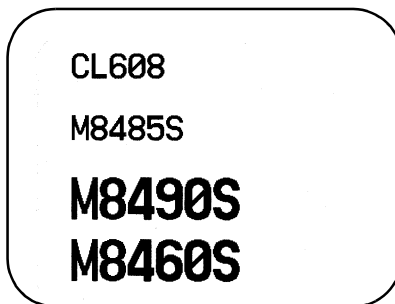
<ESC>H0025<ESC>V0215 <ESC>WL1M8460S

<ESC>Q1<ESC>Z

<ESC>A

<ESC>0<ESC>H0025<ESC>V0020<ESC>WB0M8459S<ESC>Q1<ESC>Z

Output:



Note:

1. **Specify the exact same parameters for the image to be replaced as were specified in the original data stream, including rotation, expansion, pitch, etc. This will ensure that the new data will exactly replace the old image. If the replacement data contains**

- fewer characters than the old data, then the characters not replaced will still be printed.*
- 2. This command will not function if the power has been cycled off and back on since the last label was printed.*
 - 3. Proportional Pitch text cannot be used with this command.*

Reverse Image

Command Structure:

<ESC>(aaaa,bbbb

a: Horizontal length in dots of reverse image area

b: Vertical height in dots of reverse image area.

See Note 6 for field ranges

Example: <ESC>(100,50

Placement: This command must be preceded by all other data and be placed just before <ESC>Q

Default: None

Function:

To reverse an image area from black to white and vice versa. Use the Print Position commands (<ESC>H and <ESC>V) to locate the top left corner of the reverse image area.

Input:

<ESC>A

<ESC>H0050<ESC>V0120<ESC>L0202<ESC>WB1REVERSE

<ESC>H0250<ESC>V0300<ESC>L0202<ESC>WB1HALF

<ESC>H0040<ESC>V0110<ESC>(370,100

<ESC>H0240<ESC>V0290<ESC>(220,47

<ESC>Q1<ESC>Z

Output:



Note:

1. A reverse image area is affected by the rotate commands. Therefore, always assume the printer is in the normal print orientation when designing and sending the Reverse Image command...

2. *If using reverse images with the form overlay, place this command before the Form Overlay command in the data stream.*
3. *If the Rotate commands are used with this command, the V and H parameters are reversed.*
4. *If the height and width to be reversed contain other than alpha-numeric data, the area is not printed.*
5. *If the values specified exceed the maximum ranges, the reverse image is not created.*
6. *The maximum allowable settings are:*

	CL 408	CL 412	CL 608 (VA)	CL 612 (VA)
Horizontal aaaa	-0832 to 0832	-1248 to 1248	-1216 to 1216	-1984 to 1984
Vertical bbbb	0001 to 1424	0001 to 2136	0001 to 1424	0001 to 2136

	M8485S	M8490S	M8460S	M8459S
Horizontal aaaa	-1248 to 1248	-1344 to 1344	-1216 to 1216	-0896 to 0896
Vertical bbbb	0001 to 1424	0001 to 2136	0001 to 1424	0001 to 2136

	XL 400	XL 410	M8400RV
Horizontal aaaa	-0800 to 0800	-1200 to 1200	-0832 to 0832
Vertical bbbb	0001 to 1920	0001 to 1920	0001 to 1920

7. *The Reverse Image is not working in combination with Sequential Numbering.*

Rotate, Fixed Base Reference Point

Command Structure:

<ESC>%a

- a:**
- 0 Sets print to normal direction
 - 1 Sets print to 90°CCW
 - 2 Sets print to 180° rotated (upside down)
 - 3 Sets print to 270° CCW

Example: <ESC>%3

Placement: Preceding any printed data to be rotated

Default: <ESC>%0

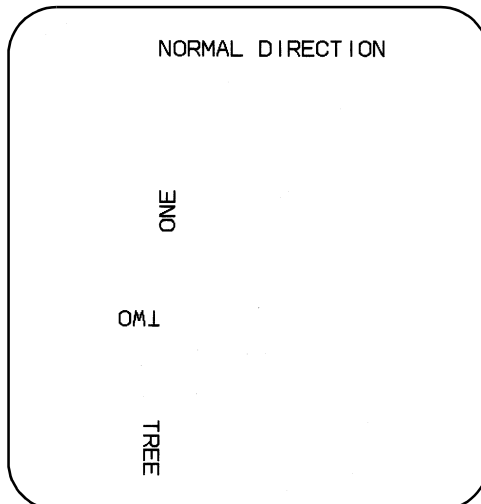
Function:

To rotate the print direction in 90° increments without changing the location of the base reference point. The diagram below illustrates the use of the <ESC>% Rotate command. Note that the entire print area is shown, but your label will probably not be as large as the entire area.

Input:

```
<ESC>A
<ESC>%0<ESC>H0200<ESC>V0100<ESC>MNORMAL DIRECTION
<ESC>%1<ESC>H0200<ESC>V0300<ESC>MONE
<ESC>%2<ESC>H0200<ESC>V0400<ESC>MTWO
<ESC>%3<ESC>H0200<ESC>V0500<ESC>MTHREE
<ESC>Q1<ESC>Z
```

Output:



Note:

- 1. Do not combine this command and the <ESC>R Rotate command in the same data stream.**
- 2. The specified values are valid until another Rotate (<ESC>%) command is received.**
- 3. Receipt of a Stop Print (<ESC>Z) command will reset the setting to the default value.**

Rotate, Moving Base Reference Point

Command Structure:

Normal Direction: <ESC>N

Rotated Direction: <ESC>R

Example: See above

Placement: Preceding any printed data to be rotated

Default: <ESC>N

Function:

The <ESC>R command rotates the printing of all subsequent images in a print job by 90° counterclockwise each time it is used. It also moves the base reference point to a different corner of the print area.

The <ESC>N command returns to the original base reference point and returns printing to the normal orientation.

Input:

<ESC>A

<ESC>N<ESC>H0100<ESC>V0010<ESC>MNORMAL DIRECTION

<ESC>R<ESC>H0100<ESC>V0100<ESC>MONE

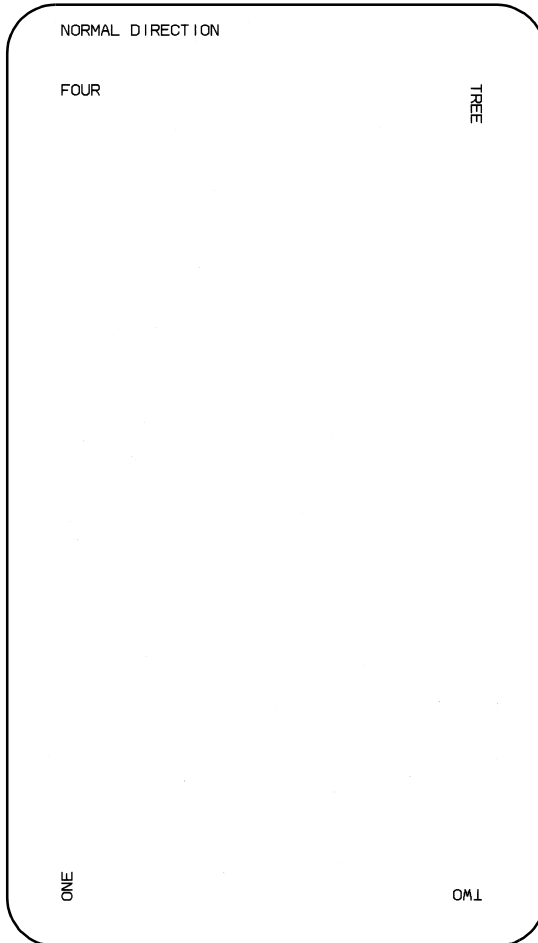
<ESC>R<ESC>H0100<ESC>V0100<ESC>MTWO

<ESC>R<ESC>H0100<ESC>V0100<ESC>MTHREE

<ESC>R<ESC>H0100<ESC>V0100<ESC>MFOUR

<ESC>Q1<ESC>Z

Output:



Note:

1. **Do not combine this command and the <ESC>% rotate command in the same data stream.**
2. **A custom graphic is not affected by this command. Therefore, always design and locate your graphic image to print in the appropriate orientation.**
3. **The specified values are valid until another Rotate (<ESC>R) command is received.**
4. **Receipt of a Stop Print (<ESC>Z) command will reset the setting to the default value.**

Sequential Numbering

Command Structure:

<ESC>Faaaabccccc,dd,ee

aaaa: Number of times to repeat the same data (0001-9999)

b: Plus or minus symbol (+ for increments; - for decrements)

cccc. Value of step for sequence (0001-9999)

,dd: Number of digits for sequential numbering (01-99). The first incrementing character position starts after the positions exempted from sequential numbering as specified in ee.

If these digits are left out, the default is 8.

,ee: Number of digits free from sequential numbering (00-99) starting with the right most position. If these digits are left out, the default is 0.

Example: **<ESC>F001-001,04,03**

Decrementing

004321321

Free from Decrementing

In this example, the right most (least significant) three digits would not decrement and the next four would decrement.

Placement: Preceding the starting value to be incremented or decremented.

Default: None

Function:

To allow the ability to print sequential fields (text, bar codes) where all incrementing is done within the printer. Up to eight different sequential fields can be specified per label. Sequencing is effective for up to 99-digit numeric data within each field.

Input:

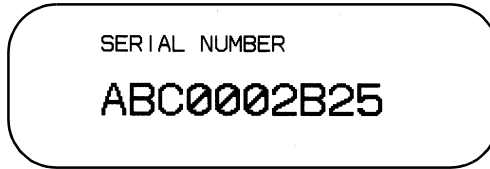
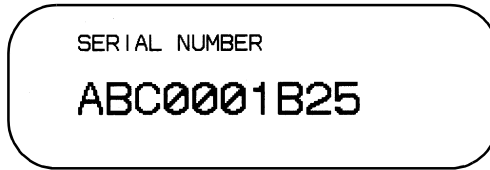
<ESC>A<ESC>H0100<ESC>V0050<ESC>MSERIAL NUMBER

<ESC>H0100<ESC>V0100<ESC>F001+001,04,03

<ESC>L0202<ESC>MABC0001B25

<ESC>Q2<ESC>Z

Output:



Note:

1. *The value specified for Print Quantity should be equal to the number of different sequential values desired multiplied by the number of repeats specified.*

Example:

To print 2 sets each of the numbers 1001-1025 on separate labels, we need 50 total labels. The commands would be as follows:

<ESC>A

<ESC>H0100<ESC>V0100<ESC>F002+001<ESC>XM1001

<ESC>Q50

<ESC>Z

2. *It is necessary to specify the print position for each sequential field on a label.*
3. *Up to eight different sequential fields can be specified per label.*
4. *This command ignores alpha characters in the sequential number field.*
5. *This command can not be used with the following commands:
Copy Image,
Reverse Image,
Line Feed.*

Start/Stop Label

Command Structure:

Start Command:<ESC>A

Stop Command:<ESC>Z

Example: See above

Placement: <ESC>A must precede data
<ESC>Z must follow data

Default: None

Function:

For all print jobs, the Start command must precede the data, and the Stop command must follow. The print job will not run properly if these are not in place.

Input:

<ESC>A

<ESC>H0001<ESC>V0100<ESC>WB1SATO

<ESC>H0130<ESC>V0200<ESC>B103150*M-8400RV*

<ESC>H0170<ESC>V0360<ESC>L0202<ESC>S*M-8400RV*

<ESC>Q1<ESC>Z

Output:

There is not output for these commands they are not accompanied by other label printing commands. However, these commands must precede and follow each print job sent to the printer.

Calendar Option Commands

The following commands in this section require the Calendar option.
(SEG No. 4449 Real Time Clock)

Calendar Increment

Command Structure:

<ESC>WPabbb

a: Y Years
 M Months
 D Days
 h Hours

bbb: Numeric data: Years (0-9), Months (00-99),
 Days (000-999), Hours (000-999),

Example: <ESC>WPM03

Placement: Anywhere within the data stream

Default: None

Function:

To add a value to the printer's current date and/or time, which may then be printed on the label. This command does not change the printer's internal clock setting.

Input:

```
<ESC>A
<ESC>H0010<ESC>V0100<ESC>XB1Current Date is:
<ESC>XB1<ESC>WADD/MM/YYYY*Y*/JJJ*/ww*
<ESC>WPM06
<ESC>H0010<ESC>V0200<ESC>XB1Expiration Date is:
<ESC>XB1<ESC>WA/DD/MM/YYYY*Y*/ww*
<ESC>Q1<ESC>Z
```

Output:

Current date is: 16/06/1999/167/24*

Expiration date is: 16/12/1999/350/50*

**Note: From Firmware 1E2050, 1E3002 for CL6xxVA*

Note:

- 1. This command requires the Calendar Option. See your SATO representative for more details**
- 2. Once the year increments past "99" it will wrap back to "00".**
- 3. This command can only be used once per data stream.**
- 4. The printer's internal clock may be set through the Calendar Set command.**
- 5. If a print quantity of more than one label per job is used, the same time and date will be on each label of the entire print job.**

Calendar Print

Command Structure:

<ESC>WA aaa...a

*YYYY 4 digit Year: 1981 ~ 2080

YY last 2 digit Year: 00 ~ 99

MM Month: 01 ~ 12

DD Day: 01 ~ 31

hh 12 Hour: 00 ~ 23

mm Minute: 00 ~ 59

*ss Second: 00 ~ 59

*TT AM/PM: AM, PM (*Do not use for numeric barcode*)

*HH 24 Hour: 00 ~ 11

*JJJ Julian Date: 001 ~ 366

*WWWeek: 00 ~ 53

*ww Week: 01 ~ 54

Example: <ESC>XB1<ESC>WADD/MM/YYYY/WW

Placement: Anywhere within the data stream .

Default: None

Function:

To specify the printing of a date and/or time field from the printer's internal clock. This may be used to date/time stamp your labels.

Input:

<ESC>A

<ESC>H0010<ESC>V0100<ESC>XB1The current date is:

<ESC>XB1<ESC>WADD/MM/YY*Y*/WW*

<ESC>H0010<ESC>V0200<ESC>XB1The current time is:

<ESC>XB1<ESC>WAhh:mm:ss*

<ESC>H0010<ESC>V0300<ESC>XB1The current time is:

<ESC>XB1<ESC>WAHH*:mm:ss*:TT*

<ESC>Q1<ESC>Z

*** = from firmware version 1E2050, 1E3002 for CL6xxVA**

Output:

The current date is: 16/06/1999/167/23*

The current time is: 21:55:21*

The current time is: 09:55:21* PM*

Note:

- 1. This function requires the Calendar Option. See your SATO representative for details.**
- 2. The date and time elements may be placed in any order for printing.**
- 3. Use a slash (/) to separate date elements and a colon (:) to separate time elements.**
- 4. The font for the date/time elements must be specified before this command.**
- 5. The printer's internal clock may be set through the Calendar Set command.**
- 6. This command can be used up to six times per job.**
- 7. The Copy or Reverse Image commands cannot be used with this command.**
- 8. Up to 16 characters can be used with this command.**

Calendar Set

Command Structure:

<ESC>WTaabbccdde

aa Year (01-99)

bb Month (01-12)

cc Day (01-31)

dd Hour (00-23)

ee Minute (00-59)

Example: <ESC>WT9101311200

Placement: This command must be sent in an independent data stream.

Default: None

Function:

To set the time and date of the printer's internal clock.

Input:

<ESC>A

<ESC>WT9312251300

<ESC>Z

Output:

There is no printer output for this command. It sets the current date to December 25, 1993 and the current time to 1:00 PM in the printer.

Note: *This command requires the Calendar Option. See your SATO representative for details.*

Memory Card Option Commands

These commands require the Memory Card Option.

Note: *Before a Memory Card can be used for the first time, it must be initialized using the <ESC>BJF command. If it is not initialized, the printer will not recognize the card and respond as if no card was installed.*

Memory Card Function

Clear Card Memory

Command Structure:

<ESC>*a,bb

a: Memory card section to be cleared
 G To clear SATO graphic files from memory card
 P To clear PCX graphic files
 F To clear formats from the memory card
 O To clear TrueType fonts

bb: Memory Card storage area to be cleared
 01 to 99 for Graphics, PCX or Formats
 00 to 09 for TrueType fonts

Example: <ESC>*G,01

Placement: This command should be sent to the printer immediately following the <ESC>CC Slot Select command.

Default: None

Function:

To clear individual memory areas in the Memory Card.

Input:

<ESC>A

<ESC>CC1<ESC>*O,09

<ESC>Z

Output:

There is no printer output as a result of this command.

Note:

1. *To clear everything in the memory card, use the <ESC>BJF Memory Card Initialize command.*
2. *This command is ignored if there is no data to be cleared.*
3. *This command is ignored if a memory card is not installed in the printer.*

Memory Card Function

Expand Memory Area

Command Structure:

<ESC>EXa

a: Memory Designation
0 Use Printer Memory
1 Use Memory Card Slot 1
2 Use Memory Card Slot 2

Example: <ESC>EX1

Placement: In its own data stream immediately after powering on.

Default: None

Function:

This command expands the memory available to image labels by using the Memory Card.

Input:

<ESC>A

<ESC>EX2

<ESC>Z

Output: There is no printer output as a result of this command.

Note:

- 1. You must have the optional Memory Card to use this command. Call your local SATO representative for details.**
- 2. When the printer is turned off, the the Memory Card is reset to normal operation.**
- 3. If the Memory Card specified already contains data, it cannot be used for memory expansion.**
- 4. The printer will reserve the specified Memory Card for expanded memory until it is turned off or receives another <ESC>EX Expand Memory Area command.**
- 5. Use care with Line and Box commands as excessively long lines can damage the print head.**
- 6. The maximum vertical position that can be specified by the <ESC>V vertical position command is shown in the table below:**
- 7. If a job contains elements out of the memory range, it is ignored.**
- 8. This command cannot be used with the <ESC>AX and <ESC>AR Expanded Print Length commands or the <ESC>R Rotate, Moving Base Reference Point command.**
- 9. If the Forms Overlay command <ESC>& is used with a Memory Card to expand the print area, the Form Overlay length is still limited to 14".**

Memory Card Function Fonts, TrueType Recall

Command Structure:

<ESC>**BJRabbccddeeeeff...f**

a: Font ID (0 thru 9)

bb: Horizontal Expansion (01 thru 12)

cc: Vertical Expansion (01 thru 12)

dd: Character Pitch (01 thru 99)

eeee: Number of characters to be printed using the font

f f f f: Data to be printed

Example: <ESC>BJR1020201000004SATO

Placement: Immediately following the <ESC>CC Slot Select command.

Default: None

Function:

This command recalls previously stored TrueType fonts from a Memory Card.

Input:

<ESC>A

<ESC>V0100<ESC>H0100<ESC>CC1<ESC>BJR1020201000004SATO

<ESC>Q1<ESC>Z

Note: *This command requires the Memory Card option. See your SATO representative for details.*

Memory Card Function Fonts, TrueType Store

Command Structure:

Begin Download <ESC>**BJ**(aa...abb..b
 Download <ESC>**BJDccccccddddee...e**
 End Download<ESC>**BJ**)
aa..a: 40 byte font description
bb..b: 10 byte date information
cccc: Memory offset (hexadecimal)
dddd: Number of data bytes to be stored (0001-2000)
ee...e: Font data to be downloaded
 Example: <ESC>BJ({50 byte header}
 <ESC>BJD{5 byte hex memory offset}{data}
 <ESC>BJ)
 Placement: Immediately following the <ESC>CC Slot Select command.
 Default: None

Function:

This command allows TrueType fonts to be stored in a Memory Card.

Input:

With software: Memory Card Manager (Call local dealer).
 The download data stream is very complex and it is recommended that the TrueType Download utility program be used instead of manually creating the required command and data stream.

Output:

There is no printer output as a result of this command. See <ESC>BJR TrueType Font Recall command.

Note:

1. *This command requires the Memory Card option. See your SATO representative for details.*
2. *The SATO TrueType Download utility program can be used to automate the download process from a computer running Windows 3.1 or above. A copy of this utility program is included as a part of the Memory Card Option.*

Memory Card Function

Format/Field Recall

Command Structure:

<ESC>YR,aa <ESC>/D,bb,cc...c

aa: Format number to be recalled (01-99)

bb: Field number to be recalled (01-99)

cc...c: Data to be placed in recalled field.

Example: <ESC>YR,01<ESC>/D,01,99

Placement: Immediately after <ESC>CC Slot Select command

Default: None

Function:

To recall a field from a stored format and place new data in the field.

Input:

<ESC>A

<ESC>CC1

<ESC>YR,02<ESC>/D,01,TWO FIELDS OF<ESC>/D,02,VARIABLE
DATA

<ESC>Q1<ESC>Z

Note:

1. *This command requires the Memory Card option. See your SATO representative for details.*
2. *Only one format can be recalled at a time. However, multiple fields may be recalled from the same format.*
3. *The number of data characters contained in the "cc...c" field cannot exceed the value designated in the <ESC>/N Field Store command. If it does, the data will be truncated to fit the field length defined in the Field Store Command.*

Memory Card Function Format/Field Store

Command Structure:

<ESC>YS,aa<ESC>/N,bb,cc{.....}

aa: Format number to be stored (01-99)

bb: Field number to be stored (01-99)

cc: Length of field to be stored (01-99)

{.....}: Command stream describing the field to be stored.

Example: <ESC>YS,01<ESC>/N,01,05

Placement: Immediately after <ESC>CC Slot Select command.

Default: None

Function:

To store a format field description in the memory card.

Input:

<ESC>A

<ESC>CC1

<ESC>YS,02<ESC>/N,01,13<ESC>V0100<ESC>H0100<ESC>XB1

<ESC>/N,02,13<ESC>V0200<ESC>H0200<ESC>XB1

<ESC>Z

Output:

There is no printer output as a result of this command. See <ESC>YR Format/Field Recall command.

Note:

1. ***This command requires the Memory Card option. See your SATO representative for details.***
2. ***Each job should be sent individually. If more than one job is sent in a data stream, only the first one will be accepted and the remainder ignored.***
3. ***The following commands cannot be stored in a format:***

<ESC>CS	Print Speed
<ESC>C	Repeat Label
<ESC>NULL	Cut Label
<ESC>Q	Print Quantity
<ESC>/D	Recall Field
<ESC>EX	Expanded Label Storage
<ESC>T	Custom Characters
<ESC>&	Store Form Overlay

<ESC>@	Off Line
<ESC>#E	Print Darkness
<ESC>BJ	TrueType Fonts
<ESC>ID	Store Job ID
<ESC>G	Store Custom Graphics
<ESC>*	Clear Memory & Buffer
<ESC>BT	Variable Ratio Bar Codes
<ESC>PI	Store PCX Graphics
<ESC>O	Partial Edit

Memory Card Function Graphics, Custom Recall

Command Structure:

<ESC>**GRaaa**

aaa: Graphics storage number (001-999)

Example: <ESC>GR111

Placement: The Recall command is sent in a secondary data stream to print the graphic, and follows any necessary position or size commands.

Default: None

Function:

Use the Recall command any time you want to print a graphic image on a label along with other printed data.

Input:

Non Rotated Graphic

<ESC>A<ESC>CC1

<ESC>V0100<ESC>H0080<ESC>L0505

<ESC>GR001

<ESC>Q1<ESC>Z

Graphic Rotated 90°

<ESC>A<ESC>CC1<ESC>%1

<ESC>V0180<ESC>H0250<ESC>L0505

<ESC>GR001

<ESC>Q1<ESC>Z

Graphic Rotated 180°

<ESC>A<ESC>CC1<ESC>%2

<ESC>V0180<ESC>H0500<ESC>L0505

<ESC>GR001

<ESC>Q1<ESC>Z

Graphic Rotated 270°

<ESC>A<ESC>CC1<ESC>%3

<ESC>V0100<ESC>H0700<ESC>L0505

<ESC>GR001

<ESC>Q1<ESC>Z

Note:

- 1. *The graphic image to be stored cannot be rotated before it is stored. It can be rotated when it is recalled.***
- 2. *Graphic images cannot be stored as part of a label format.***
- 3. *See the <ESC>GI Custom Graphic Store command.***

Memory Card Function Graphics, Custom Store

Command Structure:

<ESC>**Glabbbccddd{data}**

a: Specifies character format of the data

H Hex data

B Binary data

bbb: Number of horizontal 8 x 8 blocks (001-248)

ccc: Number of vertical 8 x 8 blocks (001-267)

ddd: Graphics storage number (001-999)

{data}: Hex or binary data to describe the graphic image

Placement: Immediately following the <ESC>CC Slot Select command.

Default: None

Function:

To provide similar functionality to the <ESC>G Custom Graphic command, but allows for the graphic image to be stored in a Memory Card. Use the Store command to send the graphic data to the printer, which is held in the optional memory card, even if printer power is lost.

Input:

<ESC>A

<ESC>CC1<ESC>GIH002002001

0100038007C00FE01FF03FF87FFCFFFE07C007C007C007C007C00
7C007C007C0

<ESC>Z

Output:

There is no printer output as a result of this command. See <ESC>GR Recall Custom Graphics command.

Note:

1. *You must have the optional Memory Card to use this command. Call your SATO representative for details.*
2. *The maximum storage capacity is 999 graphics, up to the capacity of the memory card used.*
3. *If a data transmission error occurs, the printer will beep and the "ERROR" LED will come on. You must then retransmit the image.*
4. *Each graphic to be stored must be sent in its own data stream.*

Example of correct data stream:

<ESC>A

<ESC>GIHaaabbb001(DATA)

<ESC>Z

<ESC>A

<ESC>GIHaaabbb002(DATA)

<ESC>Z

Example of incorrect data stream:

<ESC>A

<ESC>GIHaaabbb001(DATA)

<ESC>GIHaaabbb002(DATA)

<ESC>Z

5. *Do not use ASCII <CR> or <LF> characters (carriage return or line feed) as line delimiters within the graphic data or the actual image will not be printed as specified.*
6. *The graphics storage number (ddd) must be specified with this command.*

Memory Card Function Graphics, PCX Recall

Command Structure:

<ESC>**PY**aa

aa: Storage area number (001 thru 099)

Example: <ESC>PY001

Placement: This command must be placed within its own data stream specifying the placement of the graphic.

Default: None

Function:

To recall for printing a graphic file previously stored in a PCX format in the Memory Card.

Input:

Normal Rotation

```
<ESC>A<ESC>CC1
```

```
<ESC>V0100<ESC>H0000<ESC>PY001
```

```
<ESC>Q1<ESC>Z
```

Rotate Base Reference Point

```
<ESC>A<ESC>CC1<ESC>%1
```

```
<ESC>V0330<ESC>H0180<ESC>PY001
```

```
<ESC>Q1<ESC>Z
```

2nd Rotation, Base Reference Point

```
<ESC>A<ESC>CC1<ESC>%2
```

```
<ESC>V0330<ESC>H0600<ESC>PY001
```

```
<ESC>Q1<ESC>Z
```

3rd Rotation, Base Reference Point

```
<ESC>A<ESC>CC1<ESC>%3
```

```
<ESC>V0100<ESC>H0800<ESC>PY001
```

```
<ESC>Q1<ESC>Z
```

Note:

1. *This command requires Memory Card option. See your SATO representative for details.*
2. *See the <ESC>PI Store PCX Graphics command.*

Memory Card Function

Graphics, PCX Store

Command Structure:

<ESC>Plaaa,bbbb,{data}

aaa: Storage area number (001 thru 999)

bbbb: Size of PCX file in bytes

{data}: Data

Example: <ESC>PI001,32000,{data}

Placement: This command must be placed within its own data stream

Default: None

Function:

To store for later printing a PCX graphic file in the Memory Card.

Input:

BASIC Program to Download a PCX file to Memory Card #1, Location #1

```
OPEN "C:\WIZARD\GRAPHICS\LION.PCX" FOR INPUT AS #2
```

```
DA$ = INPUT$(3800,#2)
```

```
E$ = CHR$(27)
```

```
WIDTH "LPT1:",255
```

```
LPRINT E$;"A";E$;"CC1";
```

```
LPRINT E$;"PI001,03800,";DA$
```

```
LPRINT E$;"Z";
```

```
CLOSE #2
```

Output:

There is no printer output as a result of this command. See <ESC>PY PCX Graphics Recall command.

Note:

1. *This command requires Memory Card option. See your SATO representative for details.*
2. *Graphics cannot be stored as part of a format.*
3. *Only black and white PCX files can be stored.*
4. *The file size specified by this command is the DOS file size in bytes.*

Memory Card Function Initialize

Command Structure:

<ESC>**BJF**aaaaaaaa

aaaaaaaa: Eight character alphanumeric user ID

Example: <ESC>BJFsatocard

Placement: Immediately following the <ESC>CC Slot Select command.

Default: None

Function:

This clears all of the data from Memory Card in the specified slot and prepares the card to accept data.

Input:

<ESC>A

<ESC>CC2<ESC>BJFsatocard

<ESC>Z

Output:

There is no printer output as a result of this command.

Note:

- 1. You must have the optional memory card to use this command. Call your local SATO representative for information.**
- 2. All Memory Cards must be initialized before they can be used for the first time.**
- 3. Care should be exercised when using this command as it destroys any data previously written to the card. It will clear all data from the card and assign the new ID ("satocard" in the above example).**

Memory Card Function

Slot Select

Command Structure:

<ESC>**CCa**

a: Memory Card Slot

1 Slot 1

2 Slot 2

Example: <ESC>CC1

Placement: Immediately following the <ESC>A Start Code.

Default: Last selected Memory Card Slot.

Function:

Selects the card slot to be used for following Memory Card commands.

Input:

<ESC>A

<ESC>CC1

{commands}

<ESC>Z

Output:

There is no printer output as a result of this command.

Note: *This command requires the Memory Card option. See your SATO representative for more information.*

Memory Card Function Status

Command Structure:

<ESC>**BJJ**

Example: <ESC>BJJ

Placement: After the <ESC>CC Slot Select command.

Default: None

Function:

Casues the printer to print the card status.

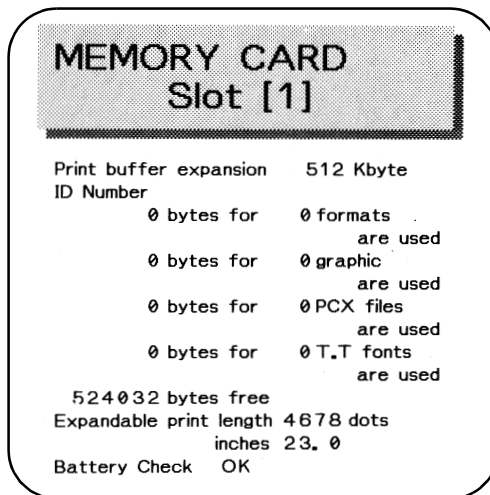
Input:

<ESC>A

<ESC>CC1<ESC>BJJ

<ESC>Z

Output:



Note:

1. *This command requires the Memory Card option. See your SATO representative for more information*
2. *The following information is provided on the status label:*
 - Line 1: Memory size of the card in Kbytes*
 - Line 2: The ID number assigned with the <ESC>BJJ command*
 - Line 3: Number of formats stored and bytes used*
 - Line 4: Number of graphics stored and bytes used*

Line 5: Number of PCX files and bytes used

Line 6: Number of TT fonts stored and bytes used

Line 7: Remaining free memory

Line 8: Max expandable print length using the card

Line 9: Battery check results

Custom Protocol Command Codes

Download

Command Structure:

<ESC>LD,a,b,c,d,e,f,g,h,i

- a:** Replacement character for STX in ASCII or hex format
- b:** Replacement character for ETX in ASCII or hex format
- c:** Replacement character for ESC in ASCII or hex format
- d:** Replacement character for ENQ in ASCII or hex format
- e:** Replacement character for CAN in ASCII or hex format
- f:** Replacement character for NULL in ASCII or hex format
- g:** Replacement character for OFFLINE in ASCII or hex format
- h:** Auto-Online. Printer powers up in the On Line mode.
0 = Yes
1 = No
- i:** Zero Slash. Places a slash through the "0" character.
0 = Yes
1 = No

Example: **<ESC>LD,{,},%,#,&,*~,0,0**

Placement: Immediately following the **<ESC>A** Start command and in an independent data stream.

Default: Standard Protocol command Codes

Function:

Allows the user to defines custom Protocol Command codes.

Input:

<ESC>A

<ESC>LD, { , } , % , # , & , * , ~ , 0 , 0

<ESC>Z

Output:

A Protocol Command code status label will be printed as a result of the a successful download of a custom set of Protocol Command codes.

Note

- 1. Commas must be used to separate the parameters. If a parameter is omitted between two commas, the default Non-Standard Protocol Command codes for that parameter will be used.**
- 2. This command must be sent as an independent data stream immediately following the <ESC>A Start code and immediately preceding the <ESC>Z Stop code. No other commands can be included in the data stream.**
- 3. If more or less than nine commas are included in the command, the entire command sequence will be ignored. The command must contain exactly nine commas.**
- 4. If two characters are specified for a parameter, it will be interpreted as a hex value. For example:**
Command Parameter Resulting Command Code
2B+
++
If a combination of characters are outside the hexadecimal range, the entire command sequence will be ignored.
- 5. Downloading Auto Online and Zero Slash settings will overwrite the values selected using the LCD panel. If these settings are changed using the LCD panel, they will overwrite any previously downloaded settings.**

3. Command Code Quick Reference

3.1 Instruction

A	Start Code. Begins all print jobs.
A(space)Z	Form Feed. Feeds a blank tag or label.
AR	Normal Print Length. This command resets the printer to the Standard print length (7 inches).
AX	Expanded Print Length. This command sets the printer to the Expanded print length (14 inches).
A3H-aaaa Vbbbb	Base Reference Point. Establishes a new base reference point position in dots for the current label. Units of measurement are dots.
Babbcccd	Bar Codes. Prints a 1:3 ratio bar code.
a =	0 Codabar
	1 Code 39
	2 Interleaved 2 of 5 (I 2/5)
	3 UPC-A/EAN-13
	4 EAN-8
	5 Industrial 2 of 5
	6 Matrix 2 of 5
	7 reserved
	8 reserved
	9 reserved
	A MSI
	B reserved
	C Code 93
	D reserved
	E UPC-E
	F Bookland
	G Code 128
	I UCC 128
bb =	Number of dots (01-12) for narrow bar and narrow space
ccc=	Bar height in dots (001-600)
d =	UCC 128 only
	0 No human readable text
	1 Human readable at top
	2 Human readable at bottom

Dabbcccd

Bar Codes. Prints a 2:5 ratio bar code, except for UPC, EAN, Code 93, Code 128 and UCC128 symbols, which are fixed width bar codes. For values a, bb, ccc and d see instructions for Babbcccd.

For UPC/EAN bar codes, this command puts descender bars and humand readable text below the symbol.

**BKaabbccdd
eeeffnn...n**

PDF417. Prints PDF417 2-D symbols.

aa = Minimum module dimension (03-09 dots). Will not print for values of 01, 02 or >= 10.

bb = Minimum module pitch dimension (04-240 dots). Will not print for values of 01, 02, 03 or greater than 25.

c = Security level (1-8).

dd = Code words per line (01-30). If 00 is specified for dd and ee, printer will automatically optimize settings.

ee = Rows/symbol (00 or 03). If 00 is specified for dd and ee, printer will automatically optimize settings.

fff = Number of characters to be encoded (0001-2700)

nn..n= Data to be printed.

BPn...n

Postnet. Prints Postnet bar codes.

n..n = 5 digit ZIP (Postnet-32 format)

6 digits (Postnet-37 format)

9 digit ZIP+4 (Postnet -52 format)

11 digit ZIP+4+DPC (Postnet-62, Delivery Point format).

BTabbccddee

Bar Codes. Variable Ratio. provides the ability to print a bar code with a ratio other than those specified through the standard bar code commands (B, BD, and D).

a = Bar code option:

0 Codabar

1 Code 39

2 Interleaved 2 of 5

5 Industrial 2 of 5

6 Matrix 2 of 5

bb = Narrow space in dots (01-99)
 cc = Wide space in dots (01-99)
 dd = Narrow bar in dots (01-99)
 ee = Wide bar in dots (01-99)

**BVaaa,bbb,
 cccccc,dddd,
 ee...e**

Maxicode. Prints Version 0 2-D Maxicode symbols.

aaa = Service class, numeric only (001-999).
 bbb = Country code, numeric only (001-999).
 cccccc= ZIP code, alphanumeric (000000-999999).
 dddd = Extended ZIP code, numeric only (0001-9999)
 ee...e = Low priority message, alphanumeric, 84 characters.

BWaabbb

Bar Codes - Expansion. Works together with the BT command to specify an expansion factor and the bar code height for the particular symbol being printed.

aa = Expansion factor by which the width of all bars and spaces is increased (01-12)
 bbb = Bar height by dot (004-600 dots)

**BXaabbccdd
 eefffghh**

Data Matrix - Data Format. Specifies the format of the Data Matrix 2-D symbology.

aa = Format ID (01-16, The values 07 and 17 will not be accepted)
 bb = Error correction level (00, 01, or 04-14, the values 02, 03 or >= 15 will not be accepted).
 cc = Horizontal cell size (03-12 dots/cell)
 dd = Vertical cell size (03-12 dots per cell)
 eee = Cells per line. Must use 000 for optimized symbol.
 fff = Cell lines. Must use 000 to optimize.
 g = Mirror image
 0 Normal Print
 1 Reverse Print
 hh = Guide cell thickness (01-15) 01 indicates normal type.

C	Repeat Label. Prints a duplicate of the last label printed.
CSa	Print Speed Selection. Specifies a unique print speed in in./sec. through software for a particular label.
Dabbcccd	Bar Codes. Prints 1:2 ratio bar code. For UPC and EAN bar codes, this will add descender bars. For values a, bb, ccc and d see instructions for Babbcccd.
DCxx...x	Data Matrix. Print Data. Prints data using Data Matrix format specified in BX Data Format command. xx...x = Data to be printed. Cannot exceed 500 characters.
Eaaa	Line Feed. Provides the ability to print multiple lines of the same character size without specifying a new print position for each line. aaa = Number of dots (1-999) between the bottom of the characters on one line to the top of the characters on thenext line.
Faaaabcccc ddee	Sequential Numbering. Allows the printing of sequencing fields (text, bar codes) where all incrementing is done within the printer. aaaa = Number of times to repeat the same data (0001-9999) b = Plus or minus symbol (+ for increments; - for decrements) cccc = Value of step for sequence (001-9999) dd = No. of digits for sequential numbering (01-99, default = 8) ee = No. of digits free from sequential numbering (01-99, default = 0)
FWaaHbbbb	Horizontal Line. Prints a horizontal line. Units of measurement are dots.
FWaabbVccc Hdddd	Box. Prints a box. For values aa, bbbb, cc, and dddd, see instructions for horizontal and vertical lines. Units of measurement are dots.
FWccVddd	Vertical Line. Prints a vertical line. Units of measurement are dots.

FXaaabccc dddeee	Data Matrix. Sequential Numbering. Prints sequential numbered Data Matrix 2-D symbols. aaa = Number of duplicate labels (001-999) b = Increment or decrement + Increment - Decrement ccc = Increment/decrement steps (001-999) ddd = Sequential numbering start position (001-999). Referenced to left side. eee = Incremented data length (001-999). Measured from start position.
Gabbccc (data)	Custom Graphics. Allows the creation and printing of graphic images using a dot-addressable matrix. a = Specifies format of data stream to follow B Binary H Hexadecimal bbb = Number of horizontal 8 x 8 blocks (001-152) ccc = Number of vertical 8 x 8 blocks (001-178 or 001-356 for 14" label) data = Data to describe the graphic image
GPaaaaa	PCX File. Downloads PCX file to the internal graphics image memory. aaaaa = No. of bytes to be downloaded (max DOS file size = 32K)
Haaaa	Horizontal Position. Specifies a field's horizontal location across the width of the label from the current base reference point. The units of measurement are dots.
IDaa	Store Job ID. Stores the Job ID number. aa = Job ID number assigned (01-99)
J	Journal Print. Provides the ability to print text line by line. Fixed spacing between lines and characters.
Kab90cc	Recall Custom Designed Characters. Recalls for printing a custom character stored by the Tabcc(data) command. a = 1 16 x 16 matrix

	2 24 x 24 matrix
b	= Indicates the format that data stream was stored in B Binary H Hexadecimal
bb	= Memory location where the character was stored. Valid locations are 21 to 52 or “!” to “R” in hex values.
Laabb	Character Expansion. Expands characters in both directions. aa = Multiple to expand horizontally (01-12) bb = Multiple to expand vertically (01-12)
LD,a,b,c,d, e,f,g,i	Download Protocol Command Codes. Downloads a user defined set of Alternate Protocol Command Codes. See Appendix E for details on the proper usage of this command.
M	Font type. Specifies the 13W x 20H dot matrix font (including descenders).
N	Rotate. Moving Base Reference Point. Sets the original base reference point and returns printing to normal orientation.
OA	Font type. Specifies the OCR-A font with dot matrix.
OB	Font type. Specifies the OCR-B font dot matrix.
Paa	Character Pitch. Designates the number of dots between characters. aa = Number of dots between characters (01-99)
PR	Fixed Font Spacing. Returns the printer to fixed character spacing mode.
PS	Proportional Font Spacing. Places the printer in the proportional character spacing mode. Will not work with U Font.
Qaaaaaa	Print Quantity. Specifies the total number of labels to print. aaaaaa =Total number of labels to print for the job (000001-999999)

- R** **Rotate, Moving Base Reference Point.** Rotates the printing of all subsequent images by 90 degrees counterclockwise each time it is used. Also moves the base reference point.
- S** **Font type.** Specifies the 8W x 15H dot matrix font (including descenders).
- Tabcc(data)** **Store Custom Designed Characters.** To create and store custom characters or images in the printer's volatile memory. See Kab90cc to recall the character for printing.
- a = 1 16 x 16 matrix
 2 24 x 24 matrix
- b = Specifies data stream format to follow
 B Binary
 H Hexadecimal
- cc = Memory location to store the character.
 Valid locations are 21 to 52 or "!" to "R" in hex values.
- (data) = Data to describe the character.
- U** **Font type.** Specifies a 5W x 9L dot matrix font (including descenders).
- Vbbbb** **Vertical Position.** Specifies a field's vertical location down the length of the label from the current base reference point. Units of measurement are dots.
- WBa** **Font type.** Specifies the 18W x 30L dot matrix font (including descenders).
- a = 0 Disables auto-smoothing of font
 1 Enables auto-smoothing if expansion is greater than 3
- WDHaaaaVbbbb**
XccccYdddd **Copy Image Area.** To copy an image to another location of the label.
- aaaa= Horizontal position of the top left corner of image area
- bbbb= Vertical position of the top left corner of image area
- cccc= Horizontal length of image area
- dddd= Vertical length of image area

WLa	Font type. Specifies the 28W x 52H dot matrix font (including descenders). a = 0 Disables auto-smoothing of font 1 Enables auto-smoothing if expansion is greater than 3
XBa	Font type. Specifies the 48W x 48L dot matrix font (including descenders). a = 0 Disables auto-smoothing of font 1 Enables auto-smoothing if expansion is greater than 3
XLa	Font type. Specifies the 428W x 48H dot matrix font (including descenders). a = 0 Disables auto-smoothing of font 1 Enables auto-smoothing if expansion is greater than 3
XM	Font type. Specifies the 24W x 24H dot matrix font (including descenders).
XS	Font type. Specifies the 17W x 17H dot matrix font (including descenders).
XU	Font type. Specifies the 5W x 9H dot matrix font (including descenders).
XWa	Font type. Specifies the 48W x 48H dot matrix font (including descenders). a = 0 Disables auto-smoothing of font 1 Enables auto-smoothing if expansion is greater than 3
Z	Stop Code. Ends all print jobs.
%a	Rotate - Fixed Base Reference Point. Rotates printing in 90° increments without changing the base reference point. a = 0 Sets print to normal direction 1 Sets print to 90° CCW 2 Sets print to 180° rotated (upside down) 3 Sets print to 270° CCW (90° CW)
\$a,b,c,d	Vector font. Specifies printing of the unique SATO vector font. a = A Helvetica Bold (proportional spacing) B Helvetica Bold (fixed spacing)

	b = Font width (50-999 dots*)
	c = Font height (50-999 dots*)
	d = Font variation (0-9) as follows:
	0 Standard
	1 Standard open (outlined)
	2 Gray (mesh) pattern 1
	3 Gray (mesh) pattern 2
	4 Gray (mesh) pattern 3
	5 Standard, shadow 1
	6 Standard, shadow 2
	7 Standard mirror image
	8 Italic
	9 Italic open (outlined)
\$(data)	Data for Vector font.
#Ea	Print Darkness. Specifies a new print darkness setting. The lightest setting is "1".
(aaaa,bbbb	Reverse Image. Reverse image from black to white and vice versa. Units of measure are dots.
&	Store Form Overlay. Stores a specified label image in the printer's volatile form overlay memory.
/	Recall Form Overlay. Recalls the label image from the printer's form overlay memory for printing.
0 (zero)	Replace Data (Partial Edit). Provides the ability to replace a specified area of the previous label with new data.
*a	Clear Print Job(s) and Memory. Clears individual memory and buffers.
	a = When not included in command, clears print jobs in Multi-Buffer mode.
	a = If included in command, specifies memory section to be cleared
	T Custom character memory, printer
	& Form overlay memory, printer
	X Clears all memory all memory and buffers
@, nn...n	Off-Line/Pause. Signals the printer to go off-line after the completion of a print job.
	nn..n = Optional 32 character message to be displayed on the LCD.

~aaaa

Cutter Command. Controls the cutting of labels when using the optional SATO cutter unit. (A <NULL> can be used in place of the “~”.)

aaaa = Number of labels to print between each cut
(0000-9999)

3.2 Calendar Option Commands

WA(elements) **Calendar Print.** Prints the date and/or time field (up to 16 characters) from the printer's internal clock. Use slash to separate date elements and colon to separate time elements.

elements = YYYY 4 digit Year: 1981 ~ 2080
 YY last 2 digit Year: 00 ~ 99
 MM Month: 01 ~ 12
 DD Day: 01 ~ 31
 hh 12 Hour: 00 ~ 23
 mm Minute: 00 ~ 59
 ss Second: 00 ~ 59
 TT AM/PM: AM, PM (*Do not use for numeric barcode*)
 HH 24 Hour: 00 ~ 11
 JJJ Julian Date: 001 ~ 366
 WW Week: 00 ~ 53
 ww Week: 01 ~ 54

WPabbb **Calendar Increment.** To add a value to the printer's current date and/or time. Does not change the printer's internal time setting.

a = Y Years
 M Months
 D Days
 h Hours

bbb = Numeric data, Years (1-9), Months (01-99), Days (001-999), Hours (001-999).

WTaabbccdee **Calendar Set.** To set the time and date of the printer's internal clock.

aa = Year (00-99)
 bb = Month (01-12)
 cc = Day (01-31)
 dd = Hour (00-23)
 ee = Minute (00-59)

3.3 Memory Card Option Commands

- BJ(aa..abb..b** **Start TrueType Font Storage.** Prepares the Memory Card to accept TrueType font data.
aa...a = 40 byte font description
bb...b = 10 byte date field
- BJDccccccddd**
ee...e **Download TrueType Font Data.** Downloads the TrueType font data to the memory area specified.
cccc = Memory Offset (hexadecimal)
dddd = Data size in bytes (max = 2000)
ee...e = Font data to be downloaded
- BJ)** **End TrueType Font Storage.** Ends the TrueType font storage process
- BJFaaaaaaaa** **Initialize Memory Card.** Initializes the Memory Card and formats it for use. Should be preceded by the Slot Select command for the card to be initialized.
aaaaaaaa = 8 character alphanumeric password
- BJRabbccdd**
eeeeff..f **TrueType Font Recall.** Recalls a previously stored TrueType font for use.
a = Font ID (1-9)
bb = Horizontal Expansion (01-12)
cc = Vertical Expansion (01-12)
dd = Character pitch (01-99)
eeee = Number of characters
ff...f = Data to be printed using font
- BJS** **Memory Card Status.** Reports the status of the currently active Memory Card to the host by printing a status label.
- CCa** **Slot Select.** Selects the Memory Card slot for all following Memory Card commands.
a = 1 Slot 1
b = 2 Slot 2
- EXa** **Expand Memory Area.** Expands the memory area used by the printer to image the label.
a = 0 Return to using internal printer RAM
1 Use Memory Card in Slot 1
2 Use Memory Card in Slot 2

**Glabbccdd
ee...e**

Store Custom Graphics. Stores a graphic image in the memory card to be called later for printing on a label.

a = Specifies format of data stream to follow
B Binary
H Hexadecimal

bbb = Number of horizontal 8 x 8 blocks (001-248)

ccc = Number of vertical 8 x 8 blocks
(001-267 for 7" label)

dd = Graphics storage number (01-99)

ee...e = Data to describe the graphic image

GRcc

Recall Custom Graphics. Recalls for printing the graphic image stored by the GI command.

cc = Storage number (01-99)

**Plaa,bbbb,
cc...c**

Store PCX Graphics File. Stores a PCX graphic file.

aa = Storage number (01-99)

bbbb = Number of bytes in the file to be stored.

PYaa

Recall PCX Graphics File. Recalls a PCX graphics file.

aa = The storage number assigned to the file (01-99)

**YR,aa
/D,bb,cc...c**

Recall Format/Field. To recall a field from a format previously stored in the memory card.

aa = Number of format to be recalled (01 to 99)

bb = Number of field to be recalled (01-99)

cc...c = Data to be placed in field.

**YS,aa
/Nbb,cc**

Store Format/Field. To store a field in a format in the memory card.

aa = Format number

bb = Field number (01-99)

cc = Number of characters in the field

***abb**

Clear Card Memory. Clears individual memory and buffer areas.

a = Memory section to be cleared

G SATO graphic files (01-99)

- P PCX graphic file (01-99)
 - F Stored formats (01-99)
 - O TrueType fonts, memory card (01-09)
- bb = Storage number

4. Bar Code Specifications

This section contains detailed information on the printing of bar codes on the SATO RISC printers. Information on printing the following bar code symbologies is provided:

- Codabar
- Code 39
- Interleaved 2 of 5
- UPC-A/EAN-13
- EAN-8
- Industrial 2 of 5
- Matrix 2 of 5
- Code 128
- MSI
- Code 93
- UPC-E
- UPC Supplements (Bookland)
- UCC-128
- Postnet
- Data Matrix
- Maxicode Vers. 3
- PDF417
- QR Code (Optional, special firmware)

Codabar

Command Structure:

1:3 ratio	<ESC> B0bbcccd (data) d
2:5 ratio	<ESC> BD0bbcccd (data) d
1:2 ratio	<ESC> D0bbcccd (data) d
bb:	Width of narrow element in dots (01-12)
ccc:	Bar height in dots (001-600)
d:	Required Start and Stop character (A, B, C, or D)
(data):	Bar code data (alphanumeric)

Character Set:

0-9, -, \$, :, /, +, .

A, B, C, D (Start/Stop characters)

Density Table:

Printer Resolution	Narrow/ Wide Ratio	Value of "bb"	"X" Dimension (mils)	Density (char/inch)
12 dot/mm	1 : 3	01	3.3	25.0
	1 : 3	02	6.7	12.6
	2 : 5	01	6.7	13.6
	1 : 2	02	3.3	15.1
8 dot/mm	1 : 3	01	5.0	16.9
	1 : 3	02	10.0	8.5
	2 : 5	01	10.0	9.2
	1 : 2	02	5.0	10.2

Input:

<ESC>H0400<ESC>V0025<ESC>B002100A12345B

<ESC>H0440<ESC>V0135<ESC>XS12345

Output:



Note: *You must add the appropriate (A, B, C or D) Start and Stop characters to the data string. The printer does not automatically add them when printing.*

Code 39

Command Structure:

- 1:3 ratio: <ESC>B**bb**ccc* (data) *
- 2:5 ratio: <ESC>BD1**bb**ccc* (data) *
- 1:2 ratio: <ESC>D1**bb**ccc* (data) *
- bb**: Width of narrow element in dots (01-12)
- ccc**: Bar height in dots (001-600)
- ***: Required Start and Stop character (asterisk)
- (data)**: = Bar code data (alphanumeric)

Character Set:

- 0-9, A-Z, Space, \$, %, +, -, ., /
- * (Start/Stop character)

Density Table:

Printer Resolution	Narrow/Wide Ratio	Value of "bb"	"X" Dimension (mils)	Density (char/inch)
12 dot/mm	1 : 3	01	3.3	18.8
	1 : 3	02	6.7	9.5
	2 : 5	01	3.3	10.3
	1 : 2	01	6.7	23.1
	1 : 2	02	3.3	11.5
8 dot/mm	1 : 3	01	5.0	12.7
	1 : 3	02	10.0	6.4
	2 : 5	01	10.0	7.0
	1 : 2	01	5.0	15.6
	1 : 2	02	10.0	7.8

Input:

```
<ESC>H0100<ESC>V0025<ESC>B103100*CODE 39*
<ESC>H0230<ESC>V0130<ESC>XS*CODE 39*
```

Output.



Note: You must add the "*" Start/Stop characters to the data stream. The printer does not add them automatically.

Interleaved Two of Five (I 2/5)

Command Structure:

:3 ratio: <ESC>**B2bbccc (data)**
 2:5 ratio: <ESC>**BD2bbccc (data)**
 1:2 ratio: <ESC>**D2bbccc (data)**
bb: Width of narrow element in dots (01-12)
ccc: Bar height in dots (001-600)
(data): Bar code data (numeric); must be an even number of digits or else the printer will add a leading zero; start and stop code are provided by the printer

Character Set:

0-9 (numeric only)

Density Table:

Printer Resolution	Narrow/Wide Ratio	Value of "bb"	"X" Dimension (mils)	Density (char/inch)
12 dot/mm	1 : 3	01	3.3	33.4
	1 : 3	02	6.7	16.7
	2 : 5	01	3.3	18.8
	1 : 2	01	6.7	21.4
	1 : 2	02	3.3	14.3
8 dot/mm	1 : 3	01	5.0	22.6
	1 : 3	02	10.0	11.3
	2 : 5	01	10.0	12.7
	1 : 2	01	5.0	14.5
	1 : 2	02	10.0	9.7

Input:

<ESC>H0100<ESC>V0100<ESC>B20310045676567
 <ESC>H0140<ESC>V0210<ESC>XM45676567

Output:



45676567

Note: *To add horizontal guard bars to the top and bottom of the bar code, use the Line and Box command.*

UPC-A/EAN-13

Command Structure:

<ESC>**B3bbccc (data)**

<ESC>**D3bbccc (data)**

<ESC>**BD3bbccc (data)**

bb: Width of narrow element in dots (01-03)

ccc: Bar height in dots (001-600)


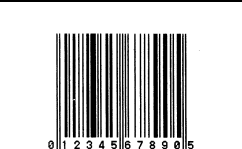
(data): Bar code data (numeric); must be exactly 13 digits. For UPC-A, the first digit must be a zero and the last 11 digits are the actual UPC-A data followed by a check digit.


To select UPC-A, 11 digits of data is sent. The printer adds a "0" and automatically generates the check digit. If 12 digits of data are sent, the printer assumes an EAN-13 symbol and automatically generates the check digit. The last digit of the bar code data is a modulo 10 check digit. If 13 digits of data are sent to the printer, the check digit is not created and must be supplied by the programmer. It must be the last character in the 13 digit string and can be determined by using the calculations outlined below.

Character Set:

0-9 (numeric only)

Density table and Output:

Printer Resolution	Value of "bb"	Narrow Bar Width (mils)	Narrow Bar Width (mm)	Magnification Factor	Example output (scaled 70%)
12 dot/mm	04	13.0	0.333	100%	
8 dot/mm	02	10.0	0.25	75%	

Printer Resolution	Value of "bb"	Narrow Bar Width (mils)	Narrow Bar Width (mm)	Magnification Factor	Example output (scaled 70%)
8 dot/mm	03	15.0	0.375	112%	 0 1 2 3 4 5 6 7 8 9 0 5

Input:

```
<ESC>H0100<ESC>V0375<ESC>BD30215001234567890
```

Note: *D3 provides guide bars that extend longer than the rest of the bar code. BD3 provides guide bars and the human readable text below the symbol.*

Calculating the Mod 10 Check Digit:

If you wish to encode the UPC-A data "01234567890", follow these steps to find the correct check digit.

ODD	0		2		4		6		8		0	
EVEN		1		3		5		7		9		CD

1. First add all the numbers in the ODD positions.
i.e., $0+2+4+6+8+0 = 20$
2. Multiply the result of Step 1 by 3.
i.e., $20 \times 3 = 60$
3. Add up all the numbers in the EVEN positions.
i.e., $1+3+5+7+9 = 25$
4. Add the result of Step 2 to that of Step 3.
i.e., $60 + 25 = 85$
5. Subtract the result of Step 4 from the next highest increment of 10.
i.e., $90 - 85 = 5$
6. The correct Modulo 10 check digit for the 11 digit string "01234567890" is 5.

EAN-8

Command Structure:

<ESC>**B4bbccc (data)**

<ESC>**D4bbccc (data)**

bb: Width of narrow element in dots (01-03)

ccc: Bar height in dots (001-600)

(data): Bar code data (numeric); must be exactly 8 digits.

Character Set:

0-9 (numeric only)

Density table:

Printer Resolution	Value of "bb"	Narrow Bar Width (mils)	Magnification Factor
12 dot/mm	02	6.7	Below Minimum
	03	10.0	75%
	04	13.0	100%
8 dot/mm	02	10.0	75%
	03	15.0	112%
	04	20.0	150%

Input:

<ESC>H0400<ESC>V0375<ESC>BD4031001234567

Output:



Note:

1. *D4 provides guide bars that extend longer than the rest of the bar code and the human readable text below the symbol.*
2. *The check digit is automatically calculated for EAN-8.*

Industrial Two of Five

Command Structure:

1:3 ratio:	B5bbccc (data)
2:5 ratio:	BD5bbccc (data)
1:2 ratio:	D5bbccc (data)
bb:	Width of narrow element in dots (01-12)
ccc:	Bar height in dots (001-600)
(data):	Bar code data (numeric); must be an even number of digits or else the printer will add a leading zero

Character Set:

0-9 (numeric only)

Input:

```
<ESC>H0100<ESC>V0600<ESC>BD50310012345
<ESC>H0300<ESC>V0710<ESC>XS12345
```

Output.



Note: *To add horizontal guard bars to the top and bottom of the bar code, use the Line and Box command.*

Matrix Two of Five

Command Structure:

- 1:3 ratio: <ESC>**B6bbccc (data)**
2:5 ratio: <ESC>**BD6bbccc (data)**
1:2 ratio: <ESC>**D6bbccc (data)**
bb: Width of narrow element in dots (01-12)
ccc: Bar height in dots (001-600)
(data): Bar code data (numeric only); must be an even number of digits or else the printer will add a leading zero.

Character Set:

0-9 (numeric only)

Input:

```
<ESC>H0100<ESC>V0775<ESC>BD60310012345  
<ESC>H0230<ESC>V0885<ESC>XS12345
```

Output:



Note: *To add horizontal guard bars to the top and bottom of the bar code, use the Line and Box command.*

Code 128

Command Structure:

<ESC>**BGbbccdd (data)**

bb: Width of narrow element in dots (01-12)

ccc: Bar height in dots (001-600)

dd: Start code to specify initial subset of bar code data

>G Subset A Start code

>H Subset B Start code

>I Subset C Start code

(data): Includes bar code data and subset shift codes; Shift codes are used to change the subset type within the bar code data.

Shift codes:

>E Subset A Shift code

>D Subset B Shift code

>C Subset C Shift code

Density Table:

Printer Resolution	Value of "bb"	"X" Dimension (mils)	Density (char/inch)	
			Subset A, B	Subset C
12 dot/mm	01	3.3	27.3	54.7
	02	6.7	13.6	27.2
	03	10.0	9.1	18.3
8 dot/mm	01	5.0	18.2	36.5
	02	10.0	9.1	18.3
	03	15.0	13.8	12.2

Input:

The following will start in Subset A for the characters "AB", shift to Subset B for "789", then shift to Subset C for "123456".

```
<ESC>H0200<ESC>V0550<ESC>BG03100>GAB>B789>C123456
```

```
<ESC>H0310<ESC>V655<ESC>XSAB789123456
```

Output:



Character Set:

VALUE	SUBSET A	SUBSET B	SUBSET C	VALUE	SUBSET A	SUBSET B	SUBSET C
0	SP	SP	00	36	D	D	36
1	!	!	01	37	E	E	37
2	"	"	02	38	F	F	38
3	#	#	03	39	G	G	39
4	\$	\$	04	40	H	H	40
5	%	%	05	41	I	I	41
6	&	&	06	42	J	J	42
7	'	'	07	43	K	K	43
8	((08	44	L	L	44
9))	09	45	M	M	45
10	*	*	10	46	N	N	46
11	+	+	11	47	O	O	47
12	,	,	12	48	P	P	48
13	-	-	13	49	Q	Q	49
14	.	.	14	50	R	R	50
15	/	/	15	51	S	S	51
16	0	0	16	52	T	T	52
17	1	1	17	53	U	U	53
18	2	2	18	54	V	V	54
19	3	3	19	55	W	W	55
20	4	4	20	56	X	X	56
21	5	5	21	57	Y	Y	57
22	6	6	22	58	Z	Z	58
23	7	7	23	59	[[59
24	8	8	24	60	\	\	60
25	9	9	25	61]]	61
26	:	:	26	62	^	^	62
27	;	;	27	63	—	—	63
28	<	<	28	64	NUL >(space)	' >(space)	64
29	=	=	29	65	SOH >	a >	65
30	>	>	30	66	STX >"	b >"	66
31	?	?	31	67	ETX >#	c >#	67
32	@	@	32	68	EOT >\$	d >\$	68
33	A	A	33	69	ENQ >%	e >%	69
34	B	B	34	70	ACK >&	f >&	70
35	C	C	35	71	BEL >'	g >'	71

Character Set:

VALUE	SUBSET A	SUBSET B	SUBSET C	VALUE	SUBSET A	SUBSET B	SUBSET C
72	BS >(h >(72	89	EM >9	y >9	89
73	HT >)	i >)	73	90	SUB >:	z >:	90
74	LF >*	j >*	74	91	ESC >;	{ >;	91
75	VT >+	k >+	75	92	FS ><	><	92
76	FF >,	l >,	76	93	GS >=	} >=	93
77	CR >-	m >-	77	94	RS >>	~ >>	94
78	SO >.	n >.	78	95	US >?	DEL >?	95
79	SI >/	o >/	79	96	FNC3 >@	FNC3 >@	96
80	DLE >0	p >0	80	97	FNC2 >A	FNC2 >A	97
81	DC1 >1	q >1	81	98	SHIFT >B	SHIFT >B	98
82	DC2 >2	r >2	82	99	Subset C >C	Subset C >C	99
83	DC3 >3	s >3	83	100	Subset B >D	FNC4 >D	Subset B >D
84	DC4 >4	t >4	84	101	FNC4 >E	Subset A >E	Subset A >E
85	NAK >5	u >5	85	102	FNC1 >F	FNC1 >F	FNC1 >F
86	SYN >6	v >6	86	103	SUBSET A START CODE >G		
87	ETB >7	w >7	87	104	SUBSET B START CODE >H		
88	CAN >8	x >8	88	105	SUBSET C START CODE >I		

MSI

Command Structure:

1:3 ratio:	<ESC> B Abbccc (data) d
2:5 ratio	<ESC> BD Abbccc (data) d
1:2 ratio	<ESC> D Abbccc (data) d
bb:	Width of narrow element in dots (01-12)
ccc:	Bar height in dots (001-600)
(data):	Bar code data (numeric); maximum of 15 digits
d:	Required check digit

Character Set:

0-9 (numeric only)

Input:

```
<ESC>H0100<ESC>V0950<ESC>BA03100123455  
<ESC>H0170<ESC>V1060<ESC>XS12345
```

Output:



Code 93

Command Structure:

1:3 ratio:<ESC>**BCbbccdd (data)**

bb: Width of narrow element in dots (01-12)

ccc: Bar height in dots (001-600)

dd: Length of data (number of digits, 00-99)

(data): Bar code data (alphanumeric); length must match value of parameter "dd"; check digit is supplied by printer

Character Set:

0-9, A-Z, -, ., Space, \$, /, +, %

Density Table:

Printer Resolution	Narrow/Wide Ratio	Value of "bb"	"X" Dimension (mils)	Density (char/inch)
12 dot/mm	1 : 3	01	3.3	33.3
	1 : 3	02	6.7	16.7
	1 : 3	03	10.0	11.1
8 dot/mm	1 : 3	01	5.0	22.5
	1 : 3	02	10.0	11.3
	1 : 3	03	15.0	7.5

Input:

<ESC>H0100<ESC>V1125<ESC>BC03100081234ABCD

<ESC>H0155<ESC>V1240<ESC>XS1234ABCD

Output:



1 234ABCD

UPC-E

Command Structure:

<ESC>**BEbbccc (data)**

<ESC>**DEbbccc (data)**

bb: Width of narrow element in dots (01-03)

ccc: Bar height in dots (001-600)

(data): Bar code data (numeric); must be exactly 6 digits

Character Set:

0-9 (numeric only)

Density table:

Printer Resolution	Value of "bb"	Narrow Bar Width (mils)	Magnification Factor
12 dot/mm	02	6.7	Below Minimum
	03	10.0	75%
	04	13.0	100%
8 dot/mm	02	10.0	75%
	03	15.0	112%
	04	20.0	150%

Input:

<ESC>H0400<ESC>V0550<ESC>DE03100123456

<ESC>H0375<ESC>V0600<ESC>OB0

<ESC>H0408<ESC>V0655<ESC>OB123456

Output:



Note: *Command DE provides guide bars that extend longer than the rest of the bar code.*

Bookland (UPC/EAN Supplements)

Command Structure:

<ESC>**BFbbccc (data)**

bb: Width of narrow element in dots (01-03)

ccc: Bar height in dots (001-600)

(data): Bar code data (numeric); must be exactly 2 or 5 digits

Character Set:

0-9 (numeric only)

Density table:

Printer Resolution	Value of "bb"	Narrow Bar Width (mils)	Magnification Factor
12 dot/mm	02	6.7	Below Minimum
	03	10.0	75%
	04	13.0	100%
8 dot/mm	02	10.0	75%
	03	15.0	112%
	04	20.0	150%

Input:

<ESC>H0325<ESC>V0725<ESC>D30315009827721123

<ESC>L0101<ESC>H0295<ESC>V0800<ESC>OB0

<ESC>H0340<ESC>V0878<ESC>OB98277

<ESC>H0480<ESC>V0878<ESC>OB21123

<ESC>H640<ESC>V0760<ESC>BF0313021826

<ESC>H655<ESC>V0730<ESC>OB21826

Output:



UCC-128

Command Structure:

<ESC>**Blbbcccd (data)**

bb: Width of the narrow elements in dots (01 to 12)

ccc: Bar height in dots (001 to 600)

d: Placement of human readable text

0 None

1 Text at top of bar code

2 Text at bottom of bar code

(data): 17 digits made up of the following:

1st digit = Container type

digits 2-8, Shipper identification

digits 9-17, Container Sequential number

Note: The Container Sequential number is not automatically sequenced by the printer.

Density Table and Character Set: See Code 128

Input:

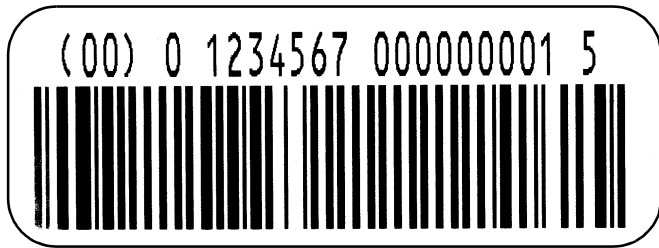
Without incrementing

<ESC>A

<ESC>H0100<ESC>V0100<ESC>BI04150101234567000000001

<ESC>Q2<ESC>Z

Output:



With incrementing

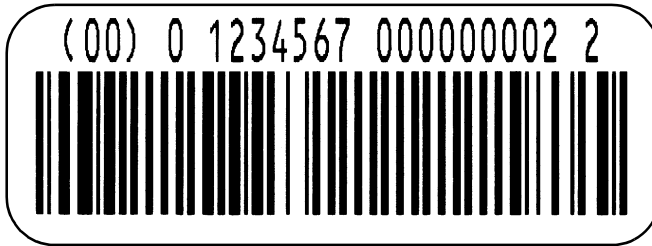
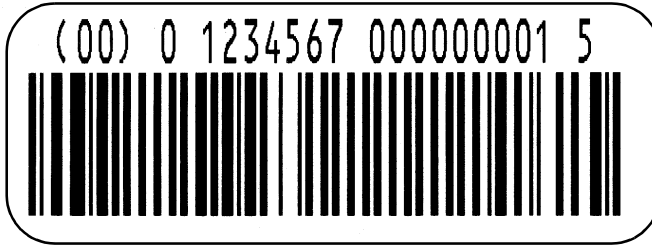
<ESC>A

<ESC>H0100<ESC>V0100<ESC>F001+001

<ESC>BI041501012345670000000001

<ESC>Q2<ESC>Z

Output:



Note:

1. *The Start, Function, Stop and Extension codes will be created by the printer and added automatically.*
2. *The internal Modulo 10 check character will be automatically created and added by the printer. The overall Code 128 symbol check character will be automatically created by the printer and added.*
3. *The automatically created human readable text will be created according to the following rules:*
 - *The spacing between the bar code and the text is fixed at 10 dots (.050 inches).*
 - *If the width of the human readable text is wider than the bar code, it will start at the same position as the bar code and extend past the right of the bar code.*
 - *If the width of the human readable text is less than the bar code, it will be centered on the bar code.*
 - *The automatically generated human readable font is OCR-B.*
 - *If any part of the human readable text extends outside the printable area, none of it will be printed. Care should be exercised when placing the bar code to allow for any automatically created human readable text.*

Postnet

Command Structure:

<ESC>**BP (data)**

- data:**
- 5 digits ZIP
 - 6 digits for Postnet 37
 - 9 digits for ZIP+4
 - 11 digits for Dlelivery Point Bar Code

Character Set:

1-9 (numeric only)

Input:

```
<ESC>H0100<ESC>V0120<ESC>BP94089
<ESC>H0100<ESC>V0160<ESC>BP123456
<ESC>H0100<ESC>V0200<ESC>BP123456789
<ESC>H0100<ESC>V0240<ESC>BP12345678901
```

Output:



Note:

1. **Frame bits and check digits added automatically by printer.**
2. **Bar code width and height are fixed and cannot be changed.**
3. **If the number of digits sent to the printer as data does not match one of the formats specified above (i.e. 5, 6, 9 or 11), the command is ignored and nothing will be printed.**
4. **If a “-” is included in the data stream (i.e. 84093-1565), it is ignored.**

Two-Dimensional Bar Codes

Data Matrix, Data Format

Command Structure:

<ESC>BXaabbccddeeffghh

- aa:** Format ID. 01-06 or 11-16. The values 07 and 17 will not be accepted by the printer.
- bb:** Error correction level. 00 ,01, or 04-14. The values 02, 03 or values of 15 or greater will be processed as a 00.
- cc:** Horizontal cell size. 03 - 12 dots/cell.
- dd:** Vertical cell size. 03 - 12 dots/cell.
- eee:** Number of cells in one line. Must use 000 to optimize.
- fff:** Number of cell lines. Must use 000 to optimize.
- g:** Mirror Image
 0 =Normal Print
 1 =Reverse Print
- hh:** Guide Cell Thickness. 01-15. 01 indicates normal type.
- Example: <ESC>BX03080505000000001
 Placement: Immediately preceding data to be encoded
 Default: None

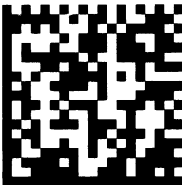
Function:

To designate the format for a Data Matrix two-dimensional bar code image on a label.

Input:

```
<ESC>A
<ESC>%0<ESC>V0100<ESC>H0100<ESC>BX05051010000000001
<ESC>DCDATA MATRIX DATA MATRIX
<ESC>Q1<ESC>Z
```

Output:



Note:

1. *If any of the parameters entered are outside the valid range, a symbol will not be printed when the <ESC>DC Print Data command is sent to the printer.*
2. *The number of cells per line (eee) and the number of cell lines (fff) should be specified as all zeroes, allowing the printer to automatically calculate the optimum configuration for the symbol.*
3. *The Reference Point for the Data Matrix symbol is the upper-left corner. If an <ESC>R Rotate command is used to rotate the symbol, it will rotate in the counter-clockwise direction.*
4. *The Format ID specified for "aa" is defined by the following table. The printer only supports the Format ID's defined in the table.*

ID Number	CHARACTER SET	ENCODING SCHEME
16 Bit CRC		
01	Numeric, Space	Base 11
02	Upper Case Alpha, Space	Base 27
03	Upper Case Alpha, Space, Comma, Period, Slash, Minus	Base 41
04	Upper Case Alphanumeric, Space	Base 37
05	ASCII 7-bit, Full Keyboard (20H -7FH)	ASCII
06	ISO 8-bit, International (20H -FFH)	8-Bit
32 Bit CRC		
11	Numeric, Space	Base 11
12	Upper Case Alpha, Space	Base 27
13	Upper Case Alpha, Space, Comma, Period, Slash, Minus	Base 41
14	Upper Case Alphanumeric, Space	Base 37
15	ASCII 7-bit, Full Keyboard (20H -7FH)	ASCII
16	ISO 8-bit, International (20H -FFH)	8-Bit

5. *The maximum number of data characters that can be specified for either the 16-Bit or 32-Bit CRC modes is 500.*

Two-Dimensional Bar Codes

Data Matrix, Print Data

Command Structure:

<ESC>DCxx...x

xx...x: Data, maximum of 500 characters

Example: <ESC>DC00006000

Placement: Immediately following the <ESC>BC Data Format designation command or the <ESC>FX Sequential Numbering command.

Default: None

Function:

To print a Data Matrix two-dimensional bar code image on a label.

Input:

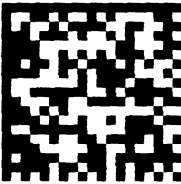
<ESC>A

<ESC>%0<ESC>V0100<ESC>H0100<ESC>BX05051010000000001

<ESC>DCDATA MATRIX DATA MATRIX

<ESC>Q1<ESC>Z

Output:



Note:

1. *The maximum amount of data that can be printed with this command is 500 characters.*
2. *If an <ESC>BX Data Format designation command contains any parameters out of the valid range, no symbol will be printed when this command is sent.*

Two-Dimensional Bar Codes

Data Matrix, Sequential Numbering

Command Structure:

<ESC>FXaaabccdddeee

aaa: Number of duplicate labels to be printed (001 -999)

b: Increment or Decrement

+ = Increment

- =Decrement

ccc: Increment/Decrement Steps (001 - 999)

ddd: Sequential numbering start position (001 - 999)

Referenced to left side.

eee: Incremented data length measured from start position (001 - 999)

Placement: Immediately following the <ESC>BX Data Format designation command and preceding the <ESC>DC Print Data Command.

Default: Non

Function:

To print sequential numbered Data Matrix symbols.

Input:

<ESC>A

<ESC>V0100<ESC>H0100

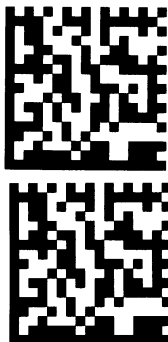
<ESC>BX0308101000000001

<ESC>FX002+001005003<ESC>DC000060000

<ESC>Q4<E SC>Z

Output:

Label Set #1



Label Set #2

**Note:**

1. *The maximum number of <ESC>FX Sequential Numbering commands that can be used in one job is eight.*
2. *In the example above four total labels will be printed (<ESC>FX002+005003<ESC>DC00006000), the sequential numbering will start at position 5 and the three digits "600" will be incremented in steps of 1. A total of two sets of labels will be printed, the first set of two labels with the value "00006000" and the next two label set with the value "00006010".*

<i>Label Set #1</i>	<i>Label Set #2</i>
<i>1st Label 00006000</i>	<i>3rd Label 00006010</i>
<i>2nd Label 00006000</i>	<i>4th Label 00006010</i>
3. *The <ESC>Q Label Quantity command must be set for the total number of labels to be printed. In the above example, the value for the <ESC>Q command should be 2 sets x 2 labels/set = 4. If, in the above example, it was set to a value of "1", only the first label would be printed.*

Two-Dimensional Bar Codes

Maxicode Vers. 3

Command Structure:

<ESC>**BVa,b,c,ddddddddd,eee,fff,DDD**

a: Symbol Number 1 - 8

b: Number Symbol 1 - 8

c: Mode 2: Structured Carrier Message
3: Structured Carrier Message
4: Standard Symbol
6: Reader Programming

ddd..d: Postal Code

ee...e: Country Code

fff: Service Class

DDD: Data Message (**Do not enter Null Code!**)

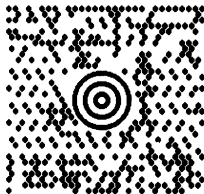
Function:

To print a Maxicode two-dimensional bar code image on a label.

Input:

```
<ESC>A<ESC>V0100<ESC>H0100
<ESC>BV1,1,2,152382802,840,001
[ ]><RS>01<GS>951Z00004951<GS>UPSMAXICODE
<GS>06X610
<GS>UPS<GS>PA<RS><EOT>
<ESC>Q1<ESC>Z
```

Output:



Note:

1. *The Secondary Message field (ee...e) must contain exactly 84 characters. If a smaller message is specified, the field must be padded with "exclamation point" character(s).*
2. *<Rs> represents Hex 1E, <Gs> represents Hex 1D, <Eot> represents Hex 04, <ESC> represents Hex 1B and <Sp> represents Hex 20.*

Two-Dimensional Bar Codes

PDF417

Command Structure:

<ESC>BKaabbcddeefffn...n

- aa:** Minimum module dimension (03-09 dots). Will not print if values of 01, 02 or greater than 10 are specified.
- bb:** Minimum module pitch dimension (04-24 dots). Will not print if values of 01, 02, 03 or greater than 25 are specified.
- c:** Security (error detection) Level (1-8).
- dd:** Code words per line (01-30). If 00 is specified for both dd and ee, the printer automatically optimizes the number of rows per symbol.
- ee:** Rows per symbol (00 or 03-40). If 00 is specified for both dd and ee, the printer automatically optimizes the number of rows per symbol.
- fff:** Number of characters to be encoded (0001-2700).
- nn...n:** Data to be printed.
- Example: <ESC>BK0304400000021
- Placement: Immediately preceding data to be encoded
- Default: None

Function:

To print a PDF417 two-dimensional bar code image on a label.

Input:

<ESC>A

<ESC>V0100<ESC>H0100<ESC>BK0607400000021PDF417 PDF417
PDF417

<ESC>Q1<ESC>Z

Output:



Note:

1. *When the code words per line and the number of rows per symbol (“dd” and “ee”) are set to all zeroes, the printer will calculate the optimum configuration.*
2. *If the product of the values entered for “dd” and “ee” are not equal to or less than the value of “fff” (i.e., “fff” is greater than “dd” x “ee”), an error will occur and the symbol will not be printed. It is recommended that these values each be set to “000” and the printer be allowed to automatically calculate the optimum values.*
3. *The values for “dd” and “ee” need to be made larger if the security level is increased.*
4. *The maximum data length is 2700 characters, but may be less depending upon:*
 - *the minimum module dimension (“aa”)*
 - *the security level specified by “c”.*
 - *the number of data characters*
5. *The Reference Point of the PDF417 symbol is the upper-left corner.*
6. *The <ESC>F Sequential Numbering command cannot be used with this command.*
7. *The <ESC>E Line Feed command cannot be used with this command.*
8. *The Macro and Truncated PDF417 symbols are not supported.*
9. *The values 00H thru 1FH can be specified as print data.*
10. *This command can be stored in a format.*
11. *The <ESC>R Rotate command can be used.*
12. *The print height of the symbol will vary depending upon the data specified; numeric only, alpha only or alphanumeric.*

Two-Dimensional Bar Codes

QR Code (Optional Special Firmware)

Available for CL 412 only (other printers on demand)

Limited factor without Maxicode and Datamatrix

Command Structure:

<ESC>2D30, a,bb,c,d (ee, ff, gg)

a: Error Correction Level L: 7%, M: 15%, Q: 25%, H: 30%

bb: Cell size 01 - 32 dots/cell

c: Connection Mode 0 = Normal, 1 = Nonnection
(The parameter for "ee", "ff", & "gg" will be used with "1" if the file is split into several blocks as independent symbol).

ee: Total Connection Number 01 - 16

ff: Connection Number for each symbol encoded as an independent one 01 - 16.

gg: Connection Mode Parity Data 00 - FF hex

<ESC>DS k, nn...n

k: Input Data Type
1 = Numeric, 2 = Alphanumeric, 3 = Kanji (Shift JIS code)

nn...n: Data

<ESC>DN IIII, nn...n

IIII: Data Size 0001 - 7089 bytes

This should be specified with either automatic code or manual mode with binary.

nn...n: Data

Character Set:

Binary

				S1								S0									
				b8	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
				b7	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	
				b6	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	
				b5	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	
b4	b3	b2	b1		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	0	0	0	0			SP	0	@	P	'	p									
0	0	0	1	1			!	1	A	Q	a	q									
0	0	1	0	2			"	2	B	R	b	r									
0	0	1	1	3			#	3	C	S	c	s									
0	1	0	0	4			\$	4	D	T	d	t									
0	1	0	1	5			%	5	E	U	e	u									
0	1	1	0	6			&	6	F	V	f	v									
0	1	1	1	7			'	7	G	W	g	w									
1	0	0	0	8			(8	H	X	h	x									
1	0	0	1	9)	9	I	Y	i	y									
1	0	1	0	A			*	:	J	Z	j	z									
1	0	1	1	B			+	;	K	[k	{									
1	1	0	0	C			,	<	L	\	l										
1	1	0	1	D			-	=	M]	m	}									
1	1	1	0	E			.	>	N	^	n	~									
1	1	1	1	F			/	?	O	_	o	DEL									

Alphanumeric

				S1							S0									
				b8	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
				b7	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
b4	b3	b2	b1		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	0	0	0			SP	0		P										
0	0	0	1	1			!	1	A	Q										
0	0	1	0	2			"	2	B	R										
0	0	1	1	3			#	3	C	S										
0	1	0	0	4			\$	4	D	T										
0	1	0	1	5			%	5	E	U										
0	1	1	0	6				6	F	V										
0	1	1	1	7				7	G	W										
1	0	0	0	8				8	H	X										
1	0	0	1	9				9	I	Y										
1	0	1	0	A			*	:	J	Z										
1	0	1	1	B			+		K											
1	1	0	0	C					L											
1	1	0	1	D			-		M											
1	1	1	0	E			.		N											
1	1	1	1	F			/		O											

Numeric					S1							S0								
				b8	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
				b7	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
b4	b3	b2	b1		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	0	0	0					0											
0	0	0	1	1					1											
0	0	1	0	2					2											
0	0	1	1	3					3											
0	1	0	0	4					4											
0	1	0	1	5					5											
0	1	1	0	6					6											
0	1	1	1	7					7											
1	0	0	0	8					8											
1	0	0	1	9					9											
1	0	1	0	A					:											
1	0	1	1	B																
1	1	0	0	C																
1	1	0	1	D																
1	1	1	0	E																
1	1	1	1	F																

Input (Quick Basic Example):

QR barcode test for CL412 1Y0200A/2E2000

CLS

E\$ = CHR\$(27)

WIDTH "LPT1:", 255

OPEN "lpt1:" FOR OUTPUT AS #1

Print Data

DATA1\$ = "0123456789"

DATA2\$ = " \$%*+-. /0123456789:ABCDEFGHIJKLMNQRSTUUVWXYZ"

DATA3\$ = ""

```
FOR i = 0 TO 31
DATA3$ = DATA3$ + CHR$(i)
NEXT i
```

PRINT #1, E\$; "A";

Manual Mode with NUMERIC

```

PRINT #1, E$; "H050"; E$; "V010"; E$; "XSMANUAL MODE WITH
NUMERIC"
PRINT #1, E$; "H050"; E$; "V050"; E$; "2D30,L,06,0,0"; E$;
"DS1,"; DATA1$;
PRINT #1, E$; "H350"; E$; "V050"; E$; "2D30,M,06,0,0"; E$;
"DS1,"; DATA1$;
PRINT #1, E$; "H650"; E$; "V050"; E$; "2D30,Q,06,0,0"; E$;
"DS1,"; DATA1$;
PRINT #1, E$; "H950"; E$; "V050"; E$; "2D30,H,06,0,0"; E$;
"DS1,"; DATA1$;

```

Manual Mode with ALPHANUMERIC

```

PRINT #1, E$; "H050"; E$; "V310"; E$; "XSMANUAL MODE WITH ALPHA-
NUMERIC";
PRINT #1, E$; "H050"; E$; "V350"; E$; "2D30,L,06,0,0"; E$;
"DS2,"; DATA2$;
PRINT #1, E$; "H350"; E$; "V350"; E$; "2D30,M,06,0,0"; E$;
"DS2,"; DATA2$;
PRINT #1, E$; "H650"; E$; "V350"; E$; "2D30,Q,06,0,0"; E$;
"DS2,"; DATA2$;
PRINT #1, E$; "H950"; E$; "V350"; E$; "2D30,H,06,0,0"; E$;
"DS2,"; DATA2$;

```

Manual Mode with BINARY

```

PRINT #1, E$; "H050"; E$; "V610"; E$; "XSMANUAL MODE WITH
BINARY";
PRINT #1, E$; "H050"; E$; "V650"; E$; "2D30,L,06,0,0"; E$;
"DN0033,"; DATA3$;
PRINT #1, E$; "H350"; E$; "V650"; E$; "2D30,M,06,0,0"; E$;
"DN0033,"; DATA3$;
PRINT #1, E$; "H650"; E$; "V650"; E$; "2D30,Q,06,0,0"; E$;
"DN0033,"; DATA3$;
PRINT #1, E$; "H950"; E$; "V650"; E$; "2D30,H,06,0,0"; E$;
"DN0033,"; DATA3$;

```

Multiple Data

```
PRINT #1, E$; "H050"; E$; "V910"; E$; "XSMULTIPLE DATA";
PRINT #1, E$; "H050"; E$; "V950"; E$; "2D30,H,06,0,0";
PRINT #1, E$; "DS1,"; DATA1$;
PRINT #1, E$; "DS2,"; DATA2$;
PRINT #1, E$; "DN0033,"; DATA3$;
```

Connection Mode

```
DATA$ = DATA1$ + DATA2$
X = LEN(DATA$)
XORDATA = ASC(MID$(DATA$, 1, 1))
  FOR i = 2 TO X
    XORDATA = XORDATA XOR ASC(MID$(DATA$, i, 1))
  NEXT i
XORDATA$ = HEX$(XORDATA)
```

Manual Mode

```
PRINT #1, E$; "H050"; E$; "V1410"; E$; "XS MANUAL MODE";
PRINT #1, E$; "H050"; E$; "V1450"; E$; "2D30,Q,06,0,1,03,01,";
XORDATA$;
PRINT #1, E$; "DS1,"; DATA1$;
PRINT #1, E$; "H500"; E$; "V1450"; E$; "2D30,Q,06,0,1,03,02,";
XORDATA$;
PRINT #1, E$; "DS2,"; DATA2$;
```

Automatic Mode

```
PRINT #1, E$; "H050"; E$; "V1710"; E$; "XSAUTOMATIC MODE";
PRINT #1, E$; "H050"; E$; "V1750"; E$; "2D30,Q,06,1,1,03,01,";
XORDATA$;
PRINT #1, E$; "DN0010,"; DATA1$;
PRINT #1, E$; "H500"; E$; "V1750"; E$; "2D30,Q,06,1,1,03,02,";
XORDATA$;
PRINT #1, E$; "DN0045,"; DATA2$;

PRINT #1, E$; "H800"; E$; "V2000"; E$; "Xsf:BASIC\QBA-
SIC\CL412\QR_CODE.BAS";

PRINT #1, E$; "Q1";
PRINT #1, E$; "Z";
```

Output:

MANUAL MODE WITH NUMERIC



MANUAL MODE WITH ALPHANUMERIC



MANUAL MODE WITH BINARY



MULTIPLE DATA



MANUAL MODE



AUTOMATIC MODE



Note:

- 1. The data command should be used according to the input mode or data type. Please refer to the attached programm sample and print sample.**
- 2. With automatic mode, the data for 80 - 9F hex or E0 - FF hex will be handled for Kanji data, not Binary data.**
- 3. With manual mode, the multiple data fields for Numeric, Alphanumeric, Kanji and Binary can be specified in one job. Also, the maximum data size should be less than 7000 bytes.**

This page is intentionally left blank.

5. Interface Specifications

5.1 Overview

This section explains the interface specifications for the SATO RISC printers. These specifications include detailed information on how to properly interface your printer with your host system and includes data about the following:

- Interface Types
- Using The Receive Buffer
- RS232C Serial Interface
- Centronics Parallel Interface
- I/F Connector

5.2 Interface Types

In order to provide flexibility in communicating with a variety of host computer systems, the SATO RISC printers can be configured for operation with either parallel or serial data transfers. Both a parallel Centronics and a serial RS232 interface are supplied with the standard printer.

The Centronics Parallel interface will probably be the most useful in communicating with IBM PCs and compatibles. The RS232C Serial interface allows connectivity to a number of other hosts. For instructions on how to properly configure the printer for either of these interface types, see the printer configuration instructions in Section 2 of your service manual.

NOTE: *Both the Centronics and RS232C interfaces are active at the same time, i.e. data can be received on either one, however no provision is made for port contention. If data is transmitted to both ports simultaneously, it will cause the data in the receive buffer to be corrupted.*

WARNING: *Never connect or disconnect interface cables or use a switch box with power applied to either the host or the printer. This may cause damage to the interface circuitry in the printer/host and is not covered by warranty.*

5.3 The Receive Buffer

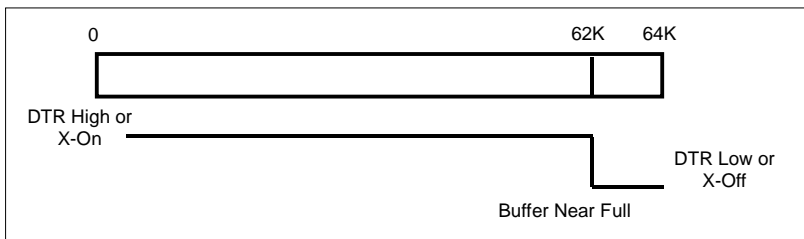
The SATO RISC printers have the ability to receive a data stream from the host in one of two ways. The receive buffer may be configured to accept one print job at a time or multiple print jobs. The single job print buffer is generally used by software programs that wish to maintain control of the job print queue so that it can move a high priority job in front of ones of lesser importance. The multiple job buffer, on the other hand prints all jobs in the order they are received by the printer, and the order of printing cannot be changed.

Single Job Buffer

The printer receives and prints one job at a time. Each job must not exceed 64 K bytes.

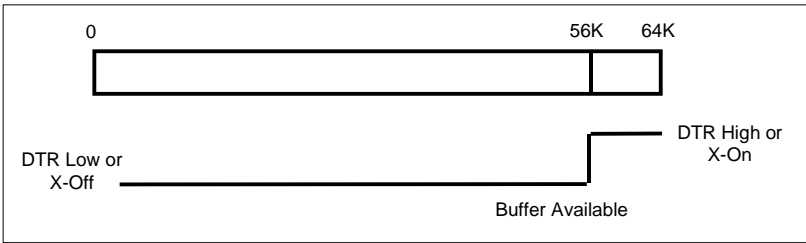
Multi Job Buffer

The printer is able to continuously receive print jobs, compiling and printing other jobs at the same time. It acts much like a “print buffer” to maximize the performance of the host and the printer. The Multi Job Buffer mode is selected with DSW2-5. When using the RS232 Serial interface, the Multi Job Buffer uses either the **Ready/Busy** with **DTR** (pin 20) or **X-On/X-Off** flow control protocols. See these sections for more details. With an empty receiving buffer, the status of **DTR** is “high” (or an **X-On** status if using **X-On/X-Off**), meaning the printer is ready to receive data. When the receive buffer is holding 62K bytes of data (2K bytes from being full), **DTR** will go “low” (or an **X-Off** is sent) indicating the printer can no longer receive data. This condition is called “Buffer Near Full”.



The receiving buffer will not be able to receive more data again until a “Buffer Available” condition occurs. This takes place when the receiving buffer has emptied so that only 56K bytes of data are being held (8K bytes

from being full). At this time, **DTR** will go “high” or an **X-On** is sent to tell the host that it can again receive data.



All printer error conditions (i.e., label out, ribbon out) will cause the printer to go busy (**DTR “low” or X-Off**) until the problem is corrected and the printer is placed on-line. The printer will also be busy if taken off-line from the front panel.

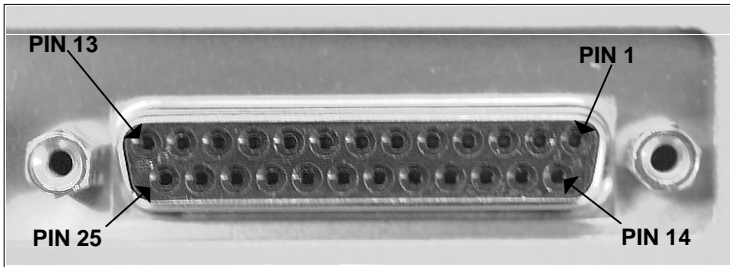
5.4 RS232C Serial Interface

General Specifications

Asynchronous ASCII	Half-duplex communication. Ready/Busy Hardware Flow Control Pin 20, DTR Control Pin 4, RTS Error Condition X-On/X-Off Software Flow Control. Bi-Directional communication (ENQ/Response)
Data Transmission Rate	2400, 4800, 9600 and 19200 bps
Character Format	1 Start Bit (fixed) 7 or 8 data bits (selectable) Odd, Even or No Parity (selectable) 1 or 2 Stop bits (selectable)

Electrical Specifications

Connector	DB-25S (Female)
-----------	-----------------



Cable	DB-25P (Male), 50 ft. maximum length. For cable configuration, refer to cable requirements appropriate to the RS232C protocol chosen.
Signal Levels	High = +5V to +12V Low = -5V to -12V

Pin Assignments

RS232C Interface Signals

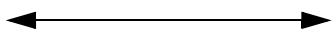
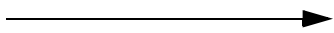
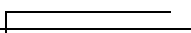
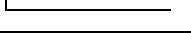
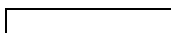
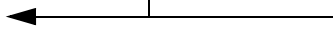
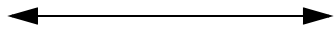
PIN	DIRECTION	SIGNAL DESCRIPTION
1	Reference	FG (Frame Ground)
2	To Host	TD (Transmit Data) - Data from the printer to the host computer. Sends X-On/X-Off characters or status data (Bi-Directional protocol).
3	To Printer	RD (Receive Data) - Data to the printer from the host computer.
4	To Host	RTS (Request to Send) Used with Ready/Busy flow control to indicate an error condition. RTS is high and remains high unless the print head is open, (in this case, RTS would return to the high state after the print head is closed and the printer is placed back on-line) or an error condition occurs during printing (e.g., ribbon out, label out).

5	To Printer	CTS (Clear to Send) - When this line is high, the printer assumes that data is ready to be transmitted. The printer will not receive data when this line is low. If this line is not being used, it should be tied high (to pin 4).
6	To Printer	DSR (Data Set Ready) - When this line is high, the printer will be ready to receive data. This line must be high before data is transmitted. If this line is not being used, it should be tied high (to pin 20).
7	Reference	SG (Signal Ground).
20	To Host	DTR (Data Terminal Ready) - This signal applies to Ready/Busy flow control. The printer is ready to receive data when this pin is high. It goes low when the printer is off-line, either manually or due to an error condition, and while printing in the Single Job Buffer mode. It will also go low when the data in the buffer reaches the Buffer Near Full level.

Ready/Busy Flow Control

Ready/Busy is the hardware flow control for the serial interface on the printer. By raising/lowering the voltage level on Pin 20 of the RS232 port, the printer notifies the host when it is ready to receive data. Pin 4 (RTS) and Pin 20 (DTR) are the important signals on the printer for this method of flow control. The host must be capable of supporting this flow control method for it to function properly.

Cable Requirements

HOST	INTERCONNECTION	PRINTER
FG		1 FG (Frame Ground)
TD		3 RD (Receive Data)
		4 RTS (Request to Send)
		5 CTS (Clear to Send)
		6 DSR (Data Set Ready)
*		20 DTR (Data Terminal Ready)
SG		7 SG (Signal Ground)

* This connection at the host side of the interface would depend upon the pin that is being used as the Ready/Busy signal by the driving software. Typically on a PC, it would be either CTS (pin 5) or DSR (pin 6) on a DB-25 connector.

Data Streams

Once the flow control method has been chosen for the RS232C interface, the data stream must be sent in a specific manner. The **STX** and **ETX** control characters must frame the data stream.

```
<STX><ESC>A . . Job#1 . . <ESC>Z<ETX><STX><ESC>A . . Job#n . .
<ESC>Z<ETX>
```

NOTE: All characters, including **STX**, **ESC** and **ETX** are in **ASCII**.

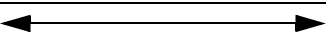
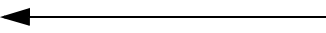
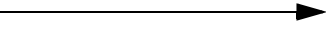
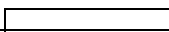
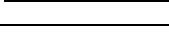
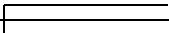
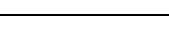
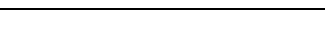
X-On/X-Off Flow Control

X-On/X-Off flow control must be used whenever hardware (Ready/Busy) flow control is not available or desirable. Instead of a voltage going high/low at pin 20, control characters representing "Printer Ready" (X-On = 11 hexadecimal) or "Printer Busy" (X-Off = 13 hexadecimal) are transmitted by the printer on pin 2 (Transmit Data) to the host. In order for this method of flow control to function correctly, the host must be capable of supporting it. **X-On/X-Off** operates in a manner similar to the function of pin 20 (**DTR**) as previously explained. When the printer is first switched on and goes

on-line, an **X-On** is sent out. In the Single Job Buffer mode, when the printer receives a print job, it transmits an **X-Off** and begins printing. When it is done printing, it transmits an **X-On**. In the Multi Job Buffer mode, the printer sends an **X-Off** when the "Buffer Near Full" level is reached and a **X-On** when the data level of the buffer drops below the "Buffer Available" mark. When the printer is taken off-line manually, it transmits an **X-Off** indicating it cannot accept data. When it is placed back on line manually, it sends an **X-On**, indicating it is again available for receipt of data. If an error occurs during printing (paper out, ribbon out), the printer sends nothing in the Single Job Buffer mode since the last character transmitted was an **X-Off**. When the error is cleared and the printer resumes printing, no **X-On** is sent until the current job is completed and the printer is once again read to receive the next job. If it is in the Multi Job Buffer mode, it sends an **X-Off** as soon as an error condition is detected. When the error is cleared and the printer is placed back on-line, it transmits an **X-On** indicating it is again ready to accept data.

Upon switch up, if no error conditions are present, the printer will continually send **X-On** characters at five millisecond intervals until it receives a transmission from the host.

Cable Requirements

HOST	INTERCONNECTION	PRINTER
FG		1 FG (Frame Ground)
RD		2 TD (Transmit Data)
TD		3 RD (Receive Data)
		4 RTS (Request to Send)
		5 CTS (Clear to Send)
		6 DSR (Data Set Ready)
		20 DTR (Data Terminal Ready)
SG		7 SG (Signal Ground)

Data Streams

The data streams for **X-On/X-Off** are constructed in the same way as they are for Ready/Busy flow control. The **STX** and **ETX** control characters must frame the data stream. (NOTE: All characters, including **STX**, **ESC** and **ETX** are in ASCII).

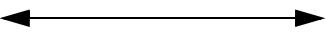
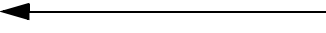
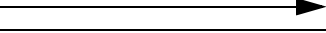
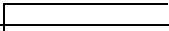
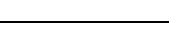
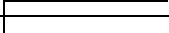
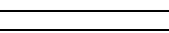
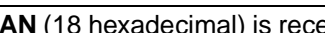
<STX><ESC>A .. Job#1 .. <ESC>Z<ESC>A .. Job#n ..
 <ESC>Z<ETX>

Example: <STX>A .. Job#1 .. <ESC>Z<ETX>XXXX

Bi-Directional Communications

This is a two-way communications protocol between the host computer and the printer, thus enabling the host to check printer status. When this protocol is selected, there is no busy signal from the printer (pin 20, **DTR**, is always high). The host must request the complete status from the printer, including ready/busy. Whenever the host requests printer status, it transmits an **ENQ** to the printer and the printer will respond with its status within five milliseconds. If printing, it will respond upon finishing the current label, then resume printing. In order for this protocol to work properly, pin 6 (**DTR**) and pin 5 (**CTS**) must be held high by the host. One way to ensure these pins are always in the correct state is to tie pin 20 (**DTR**) to pin 6 (**DSR**) and pin 4 (**RTS**) to pin 5 (**CTS**) at the printer end of the cable.

Cable Requirements

HOST	INTERCONNECTION	PRINTER
FG		1 FG (Frame Ground)
RD		2 TD (Transmit Data)
TD		3 RD (Receive Data)
		4 RTS (Request to Send)
		5 CTS (Clear to Send)
		6 DSR (Data Set Ready)
		20 DTR (Data Terminal Ready)
SG		7 SG (Signal Ground)

If a **CAN** (18 hexadecimal) is received by the printer, it will cancel the current print job and clear all data from the receive buffer.

Status Response

The Bi-Com protocol is an advanced version of bi-directional communications where the printer can also report the number of labels remaining to be printed for the current print job. Upon receipt of an **ENQ** command, the printer responds with nine bytes of status information bounded by an **STX/ETX** pair. The Bi-Com protocol works only in the Multi Job Buffer mode. The status information is defined as follows:

<STX>{ 2 Byte ID}{1 Status Byte}{6 Byte Label Remaining}<ETX>

ID - This is a two byte number identifying the current print job ID. The print job ID is defined using the **<ESC>ID** Job ID command transmitted with the print job (see Job ID Store in the command listing for more information on how to use this command). The range is from 00 to 99.

Status - A single byte defining the current status of the printer.

Label Remaining - Six bytes defining the number of labels remaining in the current print job. The range is from 000000 to 999999 labels.

If an **ENQ** is received after the print job specified in the ID bytes has been completed, or there is no data in the buffer, the printer will respond with two "space" characters (20 hexadecimal) for the ID number and six "zero" characters (30 hexadecimal) in the Remaining Labels bytes.

If a **CAN** (18 hexadecimal) command is received, it will stop the print job and clear all data from the receive and print buffers. A delay of five milliseconds or more is required before any new data can be downloaded. The **CAN** command is effective immediately upon receipt, even if the printer is off-line or in an error condition. The printer will return an **ACK** (06 hexadecimal) if there is no printer error condition and a **NAK** (15 hexadecimal) if an error condition exists.

Upon receipt of a valid print job (<ESC>A . . . <ESC>Z), and **ACK** (06 hexadecimal) will be returned by the printer if there are no errors and a **NAK** (16 hexadecimal) if a printer error exists.

Status Byte Definition, Bi-Com Protocol

ASCII	HEX	DEFINITION
OFF-LINE		
0	30	No Errors
1	31	Ribbon Near End
2	32	Buffer Near Full
3	33	Ribbon Near End and Buffer Near Full
ON-LINE, WAITING FOR DATA		
A	41	No Errors
B	42	Ribbon Near End
C	43	Buffer Near Full
D	44	Ribbon Near End and Buffer Near Full
ON-LINE, PRINTING		
G	47	No Errors
H	48	Ribbon Near End
I	49	Buffer Near Full
J	4A	Ribbon Near End and Buffer Near Full
ON-LINE, WAITING TO DISPENSE A LABEL		
M	4D	No Errors
N	4E	Ribbon Near End
O	4F	Buffer Near Full
P	50	Ribbon Near End and Buffer Near Full
ON-LINE, COMPILING PRINT JOB		
S	53	No Errors
T	54	Ribbon Near End
U	55	Buffer Near Full
V	56	Ribbon Near End and Buffer Near Full
OFF-LINE, ERROR CONDITION		
a	61	Receive Buffer Full
b	62	Head Open
c	63	Paper End
d	64	Ribbon End
e	65	Media Error
f	66	Sensor Error
g	67	Head Error
i	68	Memory Card Error
j	6A	Cutter Error
k	6B	Other Error Condition

5.5 Centronics Parallel Interface

Electrical Specifications

Printer Connector	AMP 57-40360 (DDK) or equivalent
Cable Connector	AMP 57-30360 (DDK) or equivalent
Cable Length	10 ft. or less
Signal Level	High = +2.4V to +5.0V Low = 0V to -0.4V

Centronics Parallel Interface Pin Assignments

PIN	SIGNAL	DIRECTION	PIN	SIGNAL	DIRECTION
1	STROBE	To Printer	19	STROBE Return	Reference
2	DATA 1	To Printer	20	DATA 1 Return	Reference
3	DATA 2	To Printer	21	DATA 2 Return	Reference
4	DATA 3	To Printer	22	DATA 3 Return	Reference
5	DATA 4	To Printer	23	DATA 4 Return	Reference
6	DATA 5	To Printer	24	DATA 5 Return	Reference
7	DATA 6	To Printer	25	DATA 6 Return	Reference
8	DATA 7	To Printer	26	DATA 7 Return	Reference
9	DATA 8	To Printer	27	DATA 8 Return	Reference
10	ACK	To Printer	28	ACK Return	Reference
11	BUSY	To Host	29	BUSY Return	Reference
12	PTR ERROR	To Host	30	PE Return	Reference
13	SELECT	To Host	31		
14			32	FAULT	To Host
15			33		
16			34		
17	FG	Frame Ground	35		
18	+5V (Z=24K ohm)		36		

DATA STREAMS

Single Job Buffer: The Single Job Buffer mode is not available when using the Centronics interface.

Multi Job Buffer: <STX><ESC>A . . Job#1 . . <ESC>Z<ETX>
<STX><ESC>A . . Job#n . . <ESC>Z<ETX>

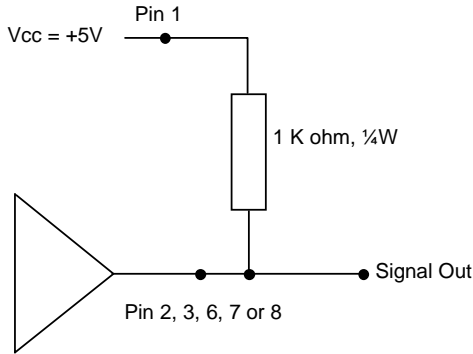
Note: For parallel communications, the STX and ETX characters are not required .

5.6 I/F Connector**PIN Assignments S-Types and M8400RV**

PIN	DIRECTION	SIGNAL DESCRIPTION
1	To Host	Vcc +5V
2	To Host	Ribbon Near End - This pin goes high when the amount of ribbon on the unwind shaft is approximately 46 feet (14 m). The output will be low when the ribbon is completely out.
3	To Host	Error - This pin goes low when the printer detects an error condition such as head open or receiving buffer full.
4	To Printer	Reprint - The last label will be reprinted when this signal is received.
5	To Printer	Print Start - The printer will print one label when this pin is pulled to ground. This signal must be enabled by placing switch DSW3-5 on the Control Panel in the OFF position.
6	To Host	End Print - It is used to drive an applicator or other external device requiring synchronisation with the print cycle. You may choose between four types of output signals using control panel DSW3-6 and DSW3-7 selections.
7	To Host	Label Out - This pin goes low (OV) when a label out error exists.
8	To Host	Ribbon Out - This pin goes low when the ribbon is out.
9	Reference	Signal Ground

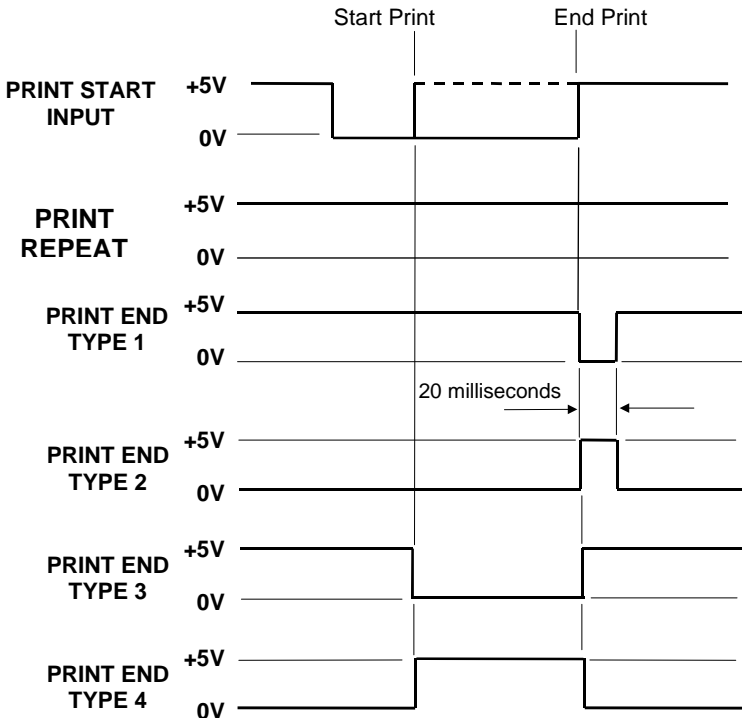
NOTE: The signals on pins 2, 3, 6, 7 and 8 each have an open collector output. These pins normally measure +.07V maximum when a true condition exists. If a false condition occurs, the voltage will drop to 0V. To achieve a signal level of +5V, you must add a 1K ohm, ¼ W pull-up resistor between the open collector output pin and Vcc (pin 1) as illustrated. This will provide a signal level of +5V for a true con-

dition and 0V when a false condition exists. The maximum voltage that can be applied to these pins is +50V and the maximum current they can sink is 500 milliamps.

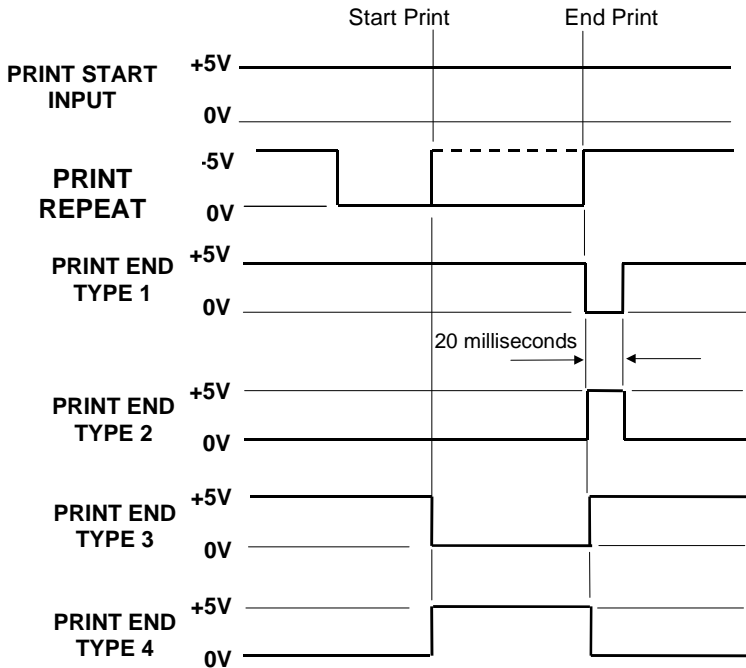


External Output Signal Types (Pin #6)

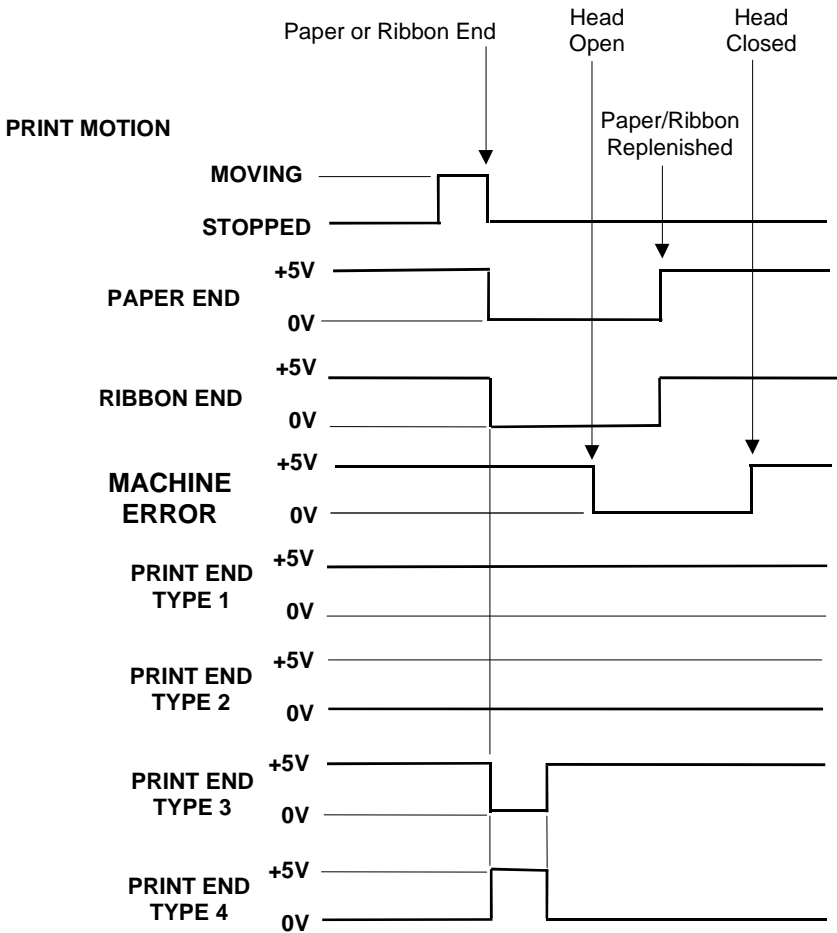
STANDARD OPERATION



REPEAT PRINT



ERROR SIGNALS

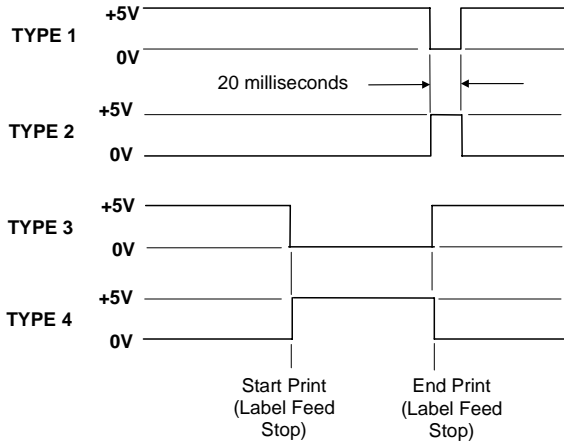
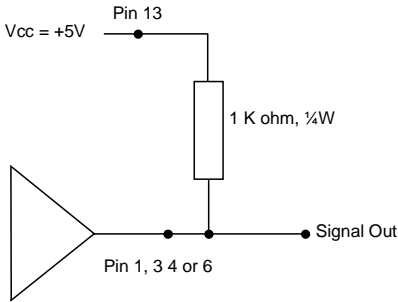


PIN Assignments CL 408/412 and CL608/612 (VA)

PIN	DIRECTION	SIGNAL DESCRIPTION
1	To Host	Label Out - This pin goes low (0V) when a label out error exists.
2	Reference	Signal Ground
3	To Host	Ribbon Out - This pin goes low when the ribbon is out.
4	To Host	Error - This pin goes low when the printer detects an error condition such as head open or receiving buffer full.
5	To Printer	Print Start - The printer will print one label when this pin is pulled to ground. This signal must be enabled by placing switch DSW3-5 on the Control Panel in the ON position.
6	To Host	End Print - It is used to drive an applicator or other external device requiring synchronization with the print cycle. You may choose between four types of output signals using control panel DSW3-6 and DSW3-7 selections.
7	To Printer	Print Repeat - The printer repeatedly prints the current label in the print buffer immediately after receiving this signal. DSW3-8 must be ON.
10	To Host	+12V - Used to power accessory items.
12	To Host	+24V - Used to power accessory items.
13	To Host	Vcc - +5V
14	Reference	Frame Ground

NOTE: *The signals on pins 1, 3 4 and 6 each have an open collector output. These pins normally measure +.07V maximum when a true condition exists. If a false condition occurs, the voltage will drop to 0V. To achieve a signal level of +5V, you must add a 1K ohm, ¼ W pull-up resistor between the open collector output pin and Vcc (pin 13) as illustrated. This will provide a signal level of +5V for a true condition and 0V when a false condition exists. The maximum voltage*

that can be applied to these pins is +50V and the maximum current they can sink is 500 milliamps.



PIN Assignments XL400/410

PIN	DIRECTION	SIGNAL DESCRIPTION
1		No Connection
2	Reference	Signal Ground
3	To Rewinder	Enables the rewinder
4		No Connection
5	To Printer	Full switch - Stops the printer when the stacker or rewinder is full
6	To Stacker	Enables the stacker
7		No Connection
10		No Connection
12	To Host	+24V - Used to power accessory items.
13	To Host	Vcc - +5V
14	Reference	Frame Ground

6. Troubleshooting

6.1 Overview

The design of SATO RISC printers is based upon proven technology and reliable components. When a problem occurs, the solution can be easily traced using the troubleshooting tables in this section. This table list symptoms, probable causes, and suggested corrective actions. Both print quality and general operational problems are listed in the troubleshooting table.

Make sure the basics have been checked before deciding you are unable to proceed any further. To help you this section has been divided into the following parts:

- Initial Checklist
- Centronics Parallel Interface
- RS232C Serial Interface
- Error Signals

6.2 Initial Checklist

If you are unable to produce output on your printer, check the following before deciding you are unable to proceed any further.

1. Is the printer switched up and ON-LINE?
2. Do any of the Front Panel LEDs indicate an error condition? If this light is ON, it may indicate the print head assembly is open.
3. Is the Print Head and the Label Hold Down in the down and latched position?

Other areas that may need looking at include:

6.3 Troubleshooting the Centronics (Parallel) Interface

1. Is the IBM parallel printer cable connected securely to your parallel port (DB-25S Female) on the PC and to the Centronics connector on the printer?

Warning: Never connect or disconnect interface cables (or use a switch box) with power applied to either the printer or the host. This may cause damage to the interface circuitry and is not covered by warranty.

2. Is there more than one parallel interface port on your PC (LPT1, LPT2, etc.)? If so, make sure you are sending data out the correct port.
3. When you send the print job to the printer, and it does not respond, do you get an error message on your PC that says "Device Fault" or something similar?

This may mean that the computer doesn't know the printer is there. Verify that:

- a) Both ends of the cable are securely inserted into their respective connectors.
- b) The printer is ON-LINE.
- c) The cable is not defective. There are other things that can cause this error message on your computer, but at this stage, a defective cable may be one of the reasons.

4. When you send the print job to the printer and it does not respond, and there is no error message on the PC:

a) Check your data stream for some of the basics. Is your job framed as follows?

<ESC>A—DATA---<ESC>Z

b) Verify that you've included all required parameters in the data stream.

c) Verify the following:

- You have not typed a "0" (zero) for an "O" (letter) or vice-versa.
- You have not missed any <ESC> characters where they're needed.
- Make sure all printer command codes are capital letters.
- Your protocol codes are set for Standard or Non-Standard and your data stream is consistent with these.

5. If you've checked all of the above and the printer still isn't printing, you may want to try a Receive Buffer Hex Dump to determine what (if anything) the printer is receiving from your computer.

The Centronics port is now listening for incoming data. Send your print job. The printer will now print (only once) a Hexadecimal (Hex) Dump of everything it received from the host computer. Each 2-digit hexadecimal character represents a character the printer received. It may be tedious, but now you can analyse and troubleshoot the data stream.

6. While checking the Hex Dump printout, if you notice 0D 0A (Carriage Return and Line Feed) characters throughout. The command string should be continuous. CR or LF characters are not allowed between the Start Command (<ESC>A) and the Stop Command (<ESC>Z). If you are using BASIC, it may be adding these characters automatically as the line wraps. Adding a "width" statement to your program can help to suppress these extra 0D 0A characters by expanding the line length up to 255 characters.

If you're not programming in BASIC, check to see if you have an equivalent statement in the language you're using to suppress extra carriage returns and line feeds from your data being sent out to the printer. We want the data stream to be one complete line going to the printer.

6.4 Troubleshooting the RS232C (Serial) Interface

1. Is the RS232C Serial cable connected securely to your serial port on the PC (DB-25S Male) and to the RS232C connector on the printer?

Warning: Never connect or disconnect interface cables (or use a switch box) with power applied to either the printer or the host. This may cause damage to the interface circuitry and is not covered by warranty.

2. Is the cable defective? At the very least, you should be using a “Null Modem Cable,” which crosses pins in a specific manner. This should enable your printer to print. We recommend that you use a cable built to specifications described in Section 3, Interface Specifications.
3. Check for obvious errors in the data stream. Remember that all print jobs for serial data must be framed by an STX and ETX.
4. If after sending your job to the printer, it only “beeps” (or displays a Framing Error message on the LCD panel), you may have a configuration problem. There may be some inconsistencies with the Baud Rate, Parity, Data Bits, or Stop Bits in relation to your host computer. If you are confused as to what the printer’s current RS232 settings are, you may choose the SATO defaults (all DIP switches in the OFF position) to achieve 9600 baud, no parity, 8 data-bits, and 1 stop bit).
5. If you still are unable to get printer output, try the Hex Dump as described in Step 5 under the Centronics Interface troubleshooting. In this case, the printer monitors its RS232C interface for incoming data.
6. From the Hex Dump, if you are seeing extra 0D 0A (CR and LF) characters, and are using BASIC, refer to the beginning of the Command Code section in the Operator and Technical Reference Manual.

6.5 Error Signals M8400RV and M84S Series Printers

(Some of the following procedures are not NA for M-8459S)

LED	LCD MESSAGE	AUDIBLE BEEP	ERROR CONDITION	TO CLEAR
Error On	Machine Error	1 Long	Machine Error	Switch power ON/OFF
Error On	EEPROM Error	1 Long	EEPROM Read/Write	Switch power ON/OFF
Error On	Head Error	1 Long	Head	Switch power ON/OFF
Error On	Sensor Error	3 Short	Sensor	Switch power ON/OFF
Error Blinks	Card R/W Error	1 Long	Memory Card Read/Write	Switch power ON/OFF
Error Blinks	Card Low Battery	1 Long	Memory Card Battery Low	Switch power ON/OFF
Error Blinks	Head Open	3 Short	Head Open	Close head lever
Error Blinks	Cutter Error	3 Short	Cutter	Switch power ON/OFF
Error On Line Blinks	PARITY ERROR	3 Short	RS232 Parity Error	Switch power ON/OFF
Error On Line Blinks	Overrun Error	3 Short	RS232 Overrun Error	Switch power ON/OFF
Error On Line Blinks	Framing Error	3 Short	RS232 Framing Error	Switch power ON/OFF
Error On Line Blinks	Buffer Over	3 Short	Buffer Overflow	Switch power ON/OFF
Error Blinks Label On	Paper End	3 Short	Label End	Open/close Head Lever Open/close Label Hold down
Error Blinks Ribbon On	Ribbon End	3 Short	Ribbon End	Open/close Head Lever Open/close Label Hold down
Error Blinks Label Blinks	Media Error	3 Short	Media Error	Open/Close Head Lever
Ribbon Blinks		None	Ribbon Near End	Replace ribbon with full roll
Line Blinks		None	Buffer Near Full	Slow down transmission rate

6.6 Error Signals CL408/412

LED	LCD MESSAGE	AUDIBLE BEEP	ERROR CONDITION	TO CLEAR
Red	Machine Error	1 Long	Machine Error	Switch power ON/OFF
Red	EEPROM Error	1 Long	EEPROM Read/Write	Switch power ON/OFF
Red	Head Error	1 Long	Head	Switch power ON/OFF
Red	Sensor Error	3 Short	Sensor	Switch power ON/OFF
Red Blinking	Card R/W Error	1 Long	Memory Card Read/Write	Switch power ON/OFF
Red Blinking	Card Low Battery	1 Long	Memory Card Battery Low	Switch power ON/OFF
Red Blinking	Head Open	3 Short	Head Open	Close Head Lever
Red Blinking	Cover Open	3 Short	Front/Top Door Open	Close Doors
Red Blinking	Parity Error	3 Short	RS232 Parity Error	Switch power ON/OFF
Red Blinking	Overrun Error	3 Short	RS232 Overrun Error	Switch power ON/OFF
Red Blinking	Framing Error	3 Short	RS232 Framing Error	Switch power ON/OFF
Red Blinking	Buffer Over	3 Short	Buffer Overflow	Switch power ON/OFF
Red Blinking	Paper End	3 Short	Label End	Open/Close Head
Red Blinking	Ribbon End	3 Short	Ribbon End	Open/Close Head
Red Blinking	Cutter Error	3 Short	Cutter Mal-function	Open/Close Head

6.7 Error Signals CL608/612 (VA)

LED	LCD MESSAGE	AUDIBLE BEEP	ERROR CONDITION	TO CLEAR
Error on	Maschine Error	1 Long	Maschine Error	Cycle power on/off
Error on	EEPROM Error	1 Long	EEPROM Read/Write	Cycle power on/off
Error on	Head Error	1 Long	Head	Cycle power on/off
Error on	Sensor Error	3 Short	Sensor	Cycle power on/off
Error Blinks	Card R/W Error	1 Long	Memory Card Read/Write	Cycle power on/off
Error Blinks	Card Low Battery	1 Long	Memory Card Battery Low	Cycle power on/off
Error Blinks	Head Open	3 Short	Head Open	Close Head Lever
Error Blinks	Cutter Error	3 Short	Cutter	Cycle power on/off
Error On Error Blinks	PARITY ERROR	3 Short	RS232 Parity Error	Cycle power on/off
Error On Error Blinks	Overrun Error	3 Short	RS232 Overrun Error	Cycle power on/off
Error On Error Blinks	Framing Error	3 Short	RS232 Framing Error	Cycle power on/off
Error On Error Blinks	Buffer Over	3 Short	Buffer Overflow	Cycle power on/off
Error Blinks Label On	Paper End	3 Short	Label End	Open/Close Head Lever or Label Holddown
Error Blinks Ribbon On	Ribbon End	3 Short	Ribbon End	Open/Close Head Lever or Label Holddown
Error Blinks Label Blinks	Media Error	3 Short	Media Error	Open/Close Head Lever
Ribbon Blinks		None	Ribbon Near End	Replace Ribbon with full roll
Line Blinks		None	Buffer Near Full	Slow down transmission rate

6.8 Error Signals XL400/410

LED	LCD	Beep	Error Condition	To Clear
Error On	Machine Error	1 Long	Machine Error	Cycle power ON/OFF
Error On	EEPROM Error	1 Long	EEPROM Read/Write	Cycle power ON/OFF
Error On	Head Error	3 Short	Print Head is damaged	Replace Print Head Cycle power ON/OFF
Error On	Sensor Error	3 Short	Media wandering Mis-selected sensor type	Check media guides Cycle power ON/OFF
Error Blinks	Card R/W Error	1 Long	Memory Card Read/Write	Format Memory Card Cycle power ON/OFF
Error Blinks	Card Low Battery	1 Long	Memory Card Battery Low	Replace MC battery Cycle power ON/OFF
Error Blinks	Head Open	3 Short	Head Open	Close head lever
Error On Line Blinks	Parity Error	3 Short	RS232 Parity Error	Cycle power ON/OFF
Error On Line Blinks	Overrun Error	3 Short	RS232 Overrun Error	Cycle power ON/OFF
Error On Line Blinks	Framing Error	3 Short	RS232 Framing Error	Cycle power ON/OFF
Error On Line Blinks	Buffer Full	3 Short	Buffer Overflow	Cycle power ON/OFF
Error Blinks Label On	Paper End	3 Short	Media End or Mis-selected media type	Replenish media Select correct media type Open/close Head Lever Open/close Media Hold down
Error Blinks	Ribbon End	3 Short	Ribbon End Ribbon Broken	Replace Ribbon Open/close Head Lever Open/close Media Hold down
Error Blinks	Media Error	3 Short	Media pitch is different than specified by command	Install proper media Open/close Head Lever

Error Blinks	Stacker Full	3 Short	Stacker is full of cut media	Empty Stacker
Error Blinks	Rewinder Full	3 Short	Rewinder is full	Remove roll from rewinder
Error Blinks	Cutter Error	3 Short	Cutter is jammed	Clear Cutter Cycle power ON/Off
Error Blinks	Cutter Sensor Error	3 Short	Media in cutter sensor Wrong cutter sensor position	Remove media from cutter sensor Select correct sensor

This page is intentionally left blank.

7. Custom Protocol Command Codes

7.1 Description

This section contains information on creating custom Protocol Command Codes for operating the RISC printer. The Protocol Command codes are used to tell the printer that a specific type of information is being transmitted to the printer. As an example, the Standard Protocol Command Code specifies the use of an <ESC> character to tell the printer that the following character(s) will represent a specific command. Sometimes the host computer is unable to generate the character or it uses the <ESC> character to control another function. In this case, an Alternate Protocol Command Code set can be selected for use by placing DIP switch 2-7 in the ON position. When the Alternate set is selected, the <ESC> character is not used and is instead replaced with a "carrot" (^) character. A command stream would then start with an "^A" instead of an "<ESC>A". These two sets of Protocol Command Codes are adequate for the majority of all applications, but occasionally situations occur where conflicts exist when using the Alternate set. In these cases, the user can define and download a custom set of Protocol Command Codes that are stored in EEPROM memory in the printer. After these are downloaded, they replace the Alternate Command Code set when DIP switch DS2-7 is in the ON position. When DIP switch DS2-7 is in the OFF position, the Standard Protocol Command Codes are used.

7.2 Download Command Structure

The command for downloading a new set of Protocol Command Codes takes the form of <ESC>LD,a,b,c,d,e,f,g,h,i. The parameters specified for "a" through "i" can be transmitted in either ASCII characters or hex notation, allowing a complete 128 character (except for the ",") set to be used for selecting the custom code.

PARAMETER	STANDARD SETTING	ALTERNATE SETTING (DEFAULT)
a	STX	{
b	ETX	}
c	ESC	^
d	ENQ	@
e	CAN	!
f	~	~
g	OFFLINE)
h (Auto ONLINE)	No	0 = YES 1 = NO
i (Zero Slash)	No	0 = YES 1 = NO

If the custom Protocol Command codes are incorrect or if the printer does not respond to commands using the custom set, the Alternate Protocol Control Codes can be restored by the following procedure:

1. Turn the printer off.
2. Place DIP switch DS2-7 in the ON position.
3. Turn power on while simultaneously pressing the **FEED** and **LINE** switches.
4. When the message "ALT PROTOCOL DEFAULT COMPLETED" appears on the display turn the printer off.

**ALT. PROTOCOL
DEFAULT COMPLETE**

For printers that do not have a display panel and the completion of the Protocol Default is signified by a single "beep". Once this beep is heard, then turn the printer off.

5. When the printer is powered up again, the Alternate Protocol Command Code set will be active. All previous custom settings will be lost.

7.3 Download Procedure

The procedure for downloading a custom Protocol Command Code set is:

1. Reset the printer to the default settings using the Reset procedure.
2. Place DIP switch DS2-7 in the ON position.
3. Turn the POWER switch ON while simultaneously pressing the **LINE** switch. This places the printer in the USER DOWNLOAD mode as signified by a "User Download" displayed on the LCD panel.

USER DOWNLOAD

For printers that do not have an LCD panel, you will hear a single "beep" signifying the printer is in the User Download mode.

4. Set DIP switch DS2-7 in the position to accept the Protocol Control codes to be used for downloading (i.e. DS2-7 = OFF for Standard codes and DS2-7 ON to use the Alternate set).
5. Press the **LINE** key to place the printer in the On-Line mode. The printer is ready to receive the download command data stream.
6. After the command has been sent, the unit will beep and print a status label. If it does not beep and print the label, the printer did not accept the data.
7. If the printer does not beep and print a setting label, turn the printer off, check your download command stream for errors and start the download process over at step 1.
8. If the custom codes are correct, press the **FEED** key to accept them and terminate the download process. If they are incorrect, turn the unit off without pressing the **FEED** key and begin the download process again at step 1.

```
STX=7B ETX=7D ESC=5E
ENQ=40 CAN=21 NULL=7E
OFFLINE=29
AUTO ONLINE YES
ZERO SLASH YES
```

Press the "FEED" key to activate the User Defaults or power the printer off to ignore them.

Custom Protocol Command Download for Sample command Stream:

```
WIDTH "LPT1:" , 255
E$ = CHR$(27)
LPRINT E$; "A";
LPRINT E$; "LD,{},^,@,!~,),0,0";
LPRINT E$; "Z";
```


8. Custom Characters And Graphics

8.1 Custom-designed Character Example

The following example is presented to help understand the use of the Custom-Designed Characters command. It demonstrates the design and printing of an "arrow" in a 16 x 16 matrix.

1. Determine which matrix size to use
 - 16 dot x 16 dots
 - 24 dots by 24 dots
2. Lay out a grid and draw the image on the grid.
 - Each square represents one dot
 - Blacken squares for each printed dot

	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
1								■								
2							■	■	■							
3						■	■	■	■	■						
4				■	■	■	■	■	■	■	■					
5				■	■	■	■	■	■	■	■	■				
6			■	■	■	■	■	■	■	■	■	■	■			
7		■	■	■	■	■	■	■	■	■	■	■	■	■		
8	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	
1																
2																
3																
4																
5																
6																
7																
8																

3. Transfer the image into two bit map representations and then into hexadecimal or binary format.

ROW	BIT MAP				HEX
1	0000	0001	0000	0000	01 00
2	0000	0011	1000	0000	03 80
3	0000	0111	1100	0000	07 C0
4	0000	1111	1110	0000	0F E0
5	0001	1111	1111	0000	1F F0
6	0011	1111	1111	1000	3F F8
7	0111	1111	1111	1100	7F FC
8	1111	1111	1111	1110	FF FE
9	0000	0111	1100	0000	07 C0
10	0000	0111	1100	0000	07 C0
11	0000	0111	1100	0000	07 C0
12	0000	0111	1100	0000	07 C0
13	0000	0111	1100	0000	07 C0
14	0000	0111	1100	0000	07 C0
15	0000	0111	1100	0000	07 C0
16	0000	0111	1100	0000	07 C0

4. To store the custom designed character in memory using a hexadecimal data stream, the command would be:

```
<ESC>A
```

```
<ESC>T1H3F0100038007C00FE01FF03FF87FFCFFFE07C007C007C  
007C007C007C007C007C0<ESC>Z
```

Note: This should be a continuous data string without any CR or LF characters.

5. To recall the custom character from memory, send the following code to the printer. Note that you can print other data as well. Also note how the character size was expanded using the <ESC>L command.

```
<ESC>A
```

```
<ESC>L0505<ESC>H0150<ESC>V100<ESC>K1H903F
```

```
<ESC>L0505<ESC>H0600<ESC>V100<ESC>K1H903F
```

```
<ESC>L0303<ESC>H0125<ESC>V0250<ESC>MTHIS SIDE UP !
```

```
<ESC>Q1
```

```
<ESC>Z
```

6. To store the custom designed character in memory using a binary data stream, the command would be:

```
<ESC>A
<ESC>T1B3F 01H 00H 03H 80H 07H C0H 0FH E0H 1FH
F0H 3FH F8H 7FH FCH FFH FEH 07H C0H 07H C0H
07H C0H 07H C0H 07H C0H 07H C0H 07H C0H
<ESC>Z
```

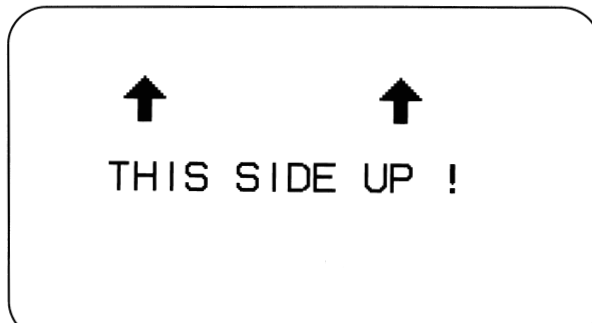
Note: Spaces are shown between hexadecimal values in the above example for clarity only and are not included in the data string.

Note that the data stream is only half as long as the hexadecimal format. This is because we can send the binary equivalent of "11111111" (represented above in its hexadecimal value of FFH), for example, using one eight bit word while it takes two eight bit words to transmit the hexadecimal equivalent "F" and "F". To send binary characters using BASIC, the expression "CHR (&HFF) will send the binary equivalent of FF (i.e., 11111111).

7. To recall the custom character from memory, send the following code to the printer:

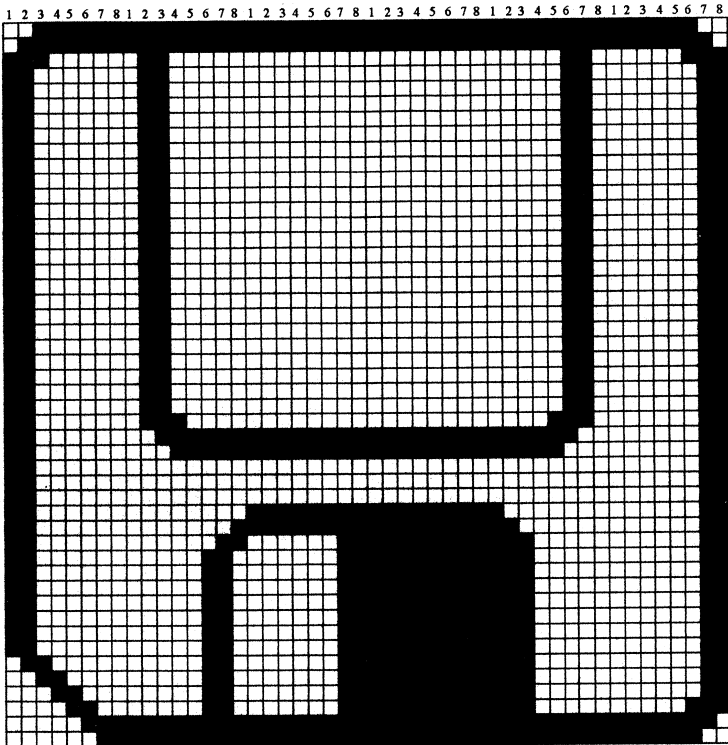
```
<ESC>A
<ESC>L505<ESC>H0150<ESC>V100<ESC>K1B903F
<ESC>L505<ESC>H0600<ESC>V100<ESC>K1B903F
<ESC>L0303<ESC>H0125<ESC>V0250<ESC>XMTHIS SIDE UP !
<ESC>Q1
<ESC>Z
```

The printer output for both the hexadecimal and binary format examples is:



The following example is presented to help you understand the use of the Custom Graphics command. It demonstrates the design and printing of a "diskette" in a 48 x 48 matrix.

1. Determine the matrix size for the graphic. It must be in 8 dot by 8 dot blocks. The example here has six blocks horizontally and six blocks vertically (48 x 48 dots).
2. Lay out a grid and draw the image on the grid.
 - Each square represents one dot
 - Blacken squares for each printed dot



3. Transfer the image into a bit map representation and then into hexadecimal format:

Binary:

```

1  00111111  11111111  11111111  11111111  11111111  11111100
2  01111111  11111111  11111111  11111111  11111111  11111110
3  11100000  01100000  00000000  00000000  00000110  00000111
4  11000000  01100000  00000000  00000000  00000110  00000011
5  11000000  01100000  00000000  00000000  00000110  00000011
6  11000000  01100000  00000000  00000000  00000110  00000011
7  11000000  01100000  00000000  00000000  00000110  00000011
8  11000000  01100000  00000000  00000000  00000110  00000011
1  11000000  01100000  00000000  00000000  00000110  00000011
2  11000000  01100000  00000000  00000000  00000110  00000011
3  11000000  01100000  00000000  00000000  00000110  00000011
4  11000000  01100000  00000000  00000000  00000110  00000011
5  11000000  01100000  00000000  00000000  00000110  00000011
6  11000000  01100000  00000000  00000000  00000110  00000011
7  11000000  01100000  00000000  00000000  00000110  00000011
8  11000000  01100000  00000000  00000000  00000110  00000011
1  11000000  01100000  00000000  00000000  00000110  00000011
2  11000000  01100000  00000000  00000000  00000110  00000011
3  11000000  01100000  00000000  00000000  00000110  00000011
4  11000000  00111111  11111111  11111111  11111100  00000011
5  11000000  00011111  11111111  11111111  11111000  00000011
6  11000000  00000000  00000000  00000000  00000000  00000011
7  11000000  00000000  00000000  00000000  00000000  00000011
8  11000000  00000000  00000000  00000000  00000000  00000011
1  11000000  00000000  11111111  11111111  10000000  00000011
2  11000000  00000001  11111111  11111111  11000000  00000011
3  11000000  00000011  00000011  11111111  11100000  00000011
4  11000000  00000110  00000011  11111111  11100000  00000011
5  11000000  00000110  00000011  11111111  11100000  00000011
6  11000000  00000110  00000011  11111111  11100000  00000011
7  11000000  00000110  00000011  11111111  11100000  00000011
8  11000000  00000110  00000011  11111111  11100000  00000011
1  11000000  00000110  00000011  11111111  11100000  00000011
2  11000000  00000110  00000011  11111111  11100000  00000011
3  01100000  00000110  00000011  11111111  11100000  00000011
4  00110000  00000110  00000011  11111111  11100000  00000011
5  00011000  00000110  00000011  11111111  11100000  00000011
6  00001100  00000110  00000011  11111111  11100000  00000011
7  00000110  11111111  11111111  11111111  11111111  11111110
8  00000011  11111111  11111111  11111111  11111111  11111100
    
```

Hex:

1	3F	FF	FF	FF	FF	FC
2	7F	FF	FF	FF	FF	FE
3	E0	60	00	00	06	07
4	C0	60	00	00	06	03
5	C0	60	00	00	06	03
6	C0	60	00	00	06	03
7	C0	60	00	00	06	03
8	C0	60	00	00	06	03
1	C0	60	00	00	06	03
2	C0	60	00	00	06	03
3	C0	60	00	00	06	03
4	C0	60	00	00	06	03
5	C0	60	00	00	06	03
6	C0	60	00	00	06	03
7	C0	60	00	00	06	03
8	C0	60	00	00	06	03
1	C0	60	00	00	06	03
2	C0	60	00	00	06	03
3	C0	60	00	00	06	03
4	C0	60	00	00	06	03
5	C0	60	00	00	06	03
6	C0	60	00	00	06	03
7	C0	60	00	00	06	03
8	C0	60	00	00	06	03
1	C0	60	00	00	06	03
2	C0	60	00	00	06	03
3	C0	60	00	00	06	03
4	C0	3F	FF	FF	FC	03
5	C0	1F	FF	FF	F8	03
6	C0	00	00	00	00	03
7	C0	00	00	00	00	03
8	C0	00	00	00	00	03
1	C0	00	FF	FF	80	03
2	C0	01	FF	FF	C0	03
3	C0	03	03	FF	E0	03
4	C0	06	03	FF	E0	03
5	C0	06	03	FF	E0	03
6	C0	06	03	FF	E0	03
7	C0	06	03	FF	E0	03
8	C0	06	03	FF	E0	03
1	C0	06	03	FF	E0	03
2	C0	06	03	FF	E0	03
3	60	06	03	FF	E0	03
4	30	06	03	FF	E0	03
5	18	06	03	FF	E0	03
6	0C	06	03	FF	E0	07
7	06	FF	FF	FF	FF	FE
8	03	FF	FF	FF	FF	FC

4. Using the hexadecimal data, send the following code to print the graphic image as designed.

```

E$ = CHR$(27)
WIDTH "lpt1:", 255
LPRINT E$; "A";
LPRINT E$; "V0100"; E$; "H0100"; E$; "GH006006";
LPRINT "3FFFFFFFFFC"; "7FFFFFFFFFE"; "E06000000607";
LPRINT "C06000000603"; "C06000000603"; "C06000000603";
LPRINT "C06000000603"; "C06000000603"; "C06000000603";
LPRINT "C06000000603"; "C06000000603"; "C06000000603";
LPRINT "C06000000603"; "C06000000603"; "C06000000603";
LPRINT "C06000000603"; "C06000000603"; "C06000000603";
LPRINT "C06000000603"; "C06000000603"; "C06000000603";
LPRINT "C06000000603"; "C06000000603"; "C06000000603";
LPRINT "C03FFFFFFFFC03"; "C01FFFFFFFF803"; "C00000000003";
LPRINT "C00000000003"; "C00000000003"; "C000FFFF8003";
LPRINT "C001FFFFFFFF03"; "C00303FFE003"; "C00603FFE003";
LPRINT "C00603FFE003"; "C00603FFE003"; "C00603FFE003";
LPRINT "C00603FFE003"; "C00603FFE003"; "C00603FFE003";
LPRINT "600603FFE003"; "300603FFE003"; "180603FFE003";
LPRINT "0C0603FFE007"; "06FFFFFFFFFE"; "03FFFFFFFFFC";
LPRINT E$; "H0200"; E$; "V0100"; E$; "XMSATO Printer Driver"
LPRINT E$; "H0200"; E$; "V0150"; E$; "XMDisk1"
LPRINT E$; "Q1";
LPRINT E$; "Z";

```

Note: Spaces shown in the hexadecimal listing above are for emphasis only. Spaces must not be encoded within the graphic portion of the data stream to the printer. Also, CR and LF characters to separate the lines must not be encoded in the data stream.

5. To send the data in binary format, the software must convert the data into binary format before transmitting it to the printer. Using the BASIC programming language for example, this is done by notation "CHR\$(&HC0)" which sends the hexadecimal value of "C0" as binary data (11000000). The BASIC program listing for sending this graphic to the printer (using the RS232 port) in binary format is:

```
E$ = CHR$(27)
OPEN "lpt1:" FOR RANDOM AS #1
PRINT #1, E$; "A";
PRINT #1, E$; "V0100"; E$; "H0100"; E$; "GB006006";
PRINT #1, CHR$(&H3F); CHR$(&HFF); CHR$(&HFF); CHR$(&HFF);
CHR$(&HFF); CHR$(&HFC);
PRINT #1, CHR$(&H7F); CHR$(&HFF); CHR$(&HFF); CHR$(&HFF);
CHR$(&HFF); CHR$(&HFE);
PRINT #1, CHR$(&HE0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H7);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H60); CHR$(&H0); CHR$(&H0);
CHR$(&H6); CHR$(&H3);
```



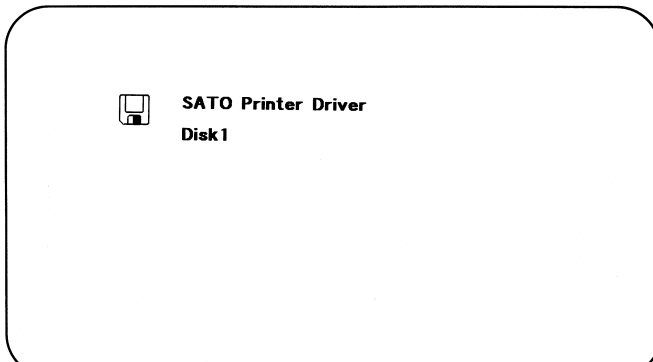
```
PRINT #1, CHR$( &HC0 ); CHR$( &H60 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H6 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H60 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H6 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H60 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H6 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H60 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H6 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H60 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H6 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H60 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H6 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H60 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H6 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H3F ); CHR$( &HFF ); CHR$( &HFF );  
CHR$( &HFC ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H1F ); CHR$( &HFF ); CHR$( &HFF );  
CHR$( &HF8 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H0 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H0 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H0 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H0 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H0 ); CHR$( &H0 ); CHR$( &H0 );  
CHR$( &H0 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H0 ); CHR$( &HFF ); CHR$( &HFF );  
CHR$( &H80 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H1 ); CHR$( &HFF ); CHR$( &HFF );  
CHR$( &HC0 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H3 ); CHR$( &H3 ); CHR$( &HFF );  
CHR$( &HE0 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H6 ); CHR$( &H3 ); CHR$( &HFF );  
CHR$( &HE0 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H6 ); CHR$( &H3 ); CHR$( &HFF );  
CHR$( &HE0 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H6 ); CHR$( &H3 ); CHR$( &HFF );  
CHR$( &HE0 ); CHR$( &H3 );  
PRINT #1, CHR$( &HC0 ); CHR$( &H6 ); CHR$( &H3 ); CHR$( &HFF );  
CHR$( &HE0 ); CHR$( &H3 );
```

```

PRINT #1, CHR$(&HC0); CHR$(&H6); CHR$(&H3); CHR$(&HFF);
CHR$(&HE0); CHR$(&H3);
PRINT #1, CHR$(&HC0); CHR$(&H6); CHR$(&H3); CHR$(&HFF);
CHR$(&HE0); CHR$(&H3);
PRINT #1, CHR$(&H60); CHR$(&H6); CHR$(&H3); CHR$(&HFF);
CHR$(&HE0); CHR$(&H3);
PRINT #1, CHR$(&H30); CHR$(&H6); CHR$(&H3); CHR$(&HFF);
CHR$(&HE0); CHR$(&H3);
PRINT #1, CHR$(&H18); CHR$(&H6); CHR$(&H3); CHR$(&HFF);
CHR$(&HE0); CHR$(&H3);
PRINT #1, CHR$(&HC); CHR$(&H6); CHR$(&H3); CHR$(&HFF);
CHR$(&HE0); CHR$(&H7);
PRINT #1, CHR$(&H6); CHR$(&HFF); CHR$(&HFF); CHR$(&HFF);
CHR$(&HFF); CHR$(&HFE);
PRINT #1, CHR$(&H3); CHR$(&HFF); CHR$(&HFF); CHR$(&HFF);
CHR$(&HFF); CHR$(&HFC);
PRINT #1, E$; "H0200"; E$; "V0100"; E$; "XMSATO Printer
Driver"
PRINT #1, E$; "H0200"; E$; "V0150"; E$; "XMDisk1"
PRINT #1, E$; "Q1";
PRINT #1, E$; "Z";

```

The printer output for both the hexadecimal and binary format examples is:



8.2 PCX Graphics Example

A graphics file in a PCX format may also be transmitted to the printer. The file must not be larger than 32K bytes (DOS file size reported in a DIR listing). For example, the Lion.PCX image shown below has a file size of 3800 bytes.



The uncompressed size (PCX is a compressed file) of the file must not be greater than 64K bytes. Generally this is not a problem unless the graphic image is surrounded by large amount of white space which the PCX algorithm can compress very efficiently. If this is the case, the file should be recaptured to eliminate the surrounding white space as much as possible.

The following basic program will send and print this file:

```
OPEN "c:\dynamic\graphics\lion.pcx" FOR INPUT AS #1
WIDTH "LPT1:" , 255
DA$=INPUT$(3800, #1)
E$=CHR$(27)
LPRINT E$;"A";
LPRINT E$;"V150";E$;"H100";E$;"GP3800,";DA$;
LPRINT E$;"Q1";E$;"Z";
```

This page is intentionally left blank.

Free Manuals Download Website

<http://myh66.com>

<http://usermanuals.us>

<http://www.somanuals.com>

<http://www.4manuals.cc>

<http://www.manual-lib.com>

<http://www.404manual.com>

<http://www.luxmanual.com>

<http://aubethermostatmanual.com>

Golf course search by state

<http://golfingnear.com>

Email search by domain

<http://emailbydomain.com>

Auto manuals search

<http://auto.somanuals.com>

TV manuals search

<http://tv.somanuals.com>