

mi-4/mi-6

**Label and Tag
Printer**

**Programming
Manual**



Copyright Information:

CG Triumvirate is a trademark of Agfa Corporation.

CG Times based upon Times New Roman under license from the Monotype Corporation.

Macintosh is a trademark of the Apple Corporation.

PCL-4 and HP Laser JetII are trademarks of the Hewlett Packard Corporation.

Windows is a trademark of the Microsoft Corporation.

Information in this document is subject to change without notice and does not represent a commitment on the part of the vendor. No part of this manual may be reproduced or transmitted in any form or by any means, for any purpose other than the purchaser's personal use, without the expressed written permission of Checkpoint Systems International GmbH.

All rights reserved. Printed in the United States of America.

© Copyright 1999 by Checkpoint Systems International GmbH

Part Number: 425554

Revision: A

Ref Number: 88-2247-01

Contents

1.0 Introduction

Scope of this Manual	1
----------------------------	---

2.0 Control Codes

Attention Getter Functions	3
----------------------------------	---

3.0 Status & Control Commands <SOH>

Status & Control Command Functions	5
SOH # Reset	5
SOH A Send ASCII Status String	6
SOH B Toggle Pause	6
SOH C Stop/Cancel	7
SOH E Send Batch Quantity	7
SOH F Send Status Byte	8
SOH U Update System Database with Current Database	8

4.0 System-Level Commands <STX>

System-Level Command Functions	9
STX A Set Time and Date	9
STX a Enable Feedback Characters	10
STX B Get Printer Time and Date Information	10
STX c Set Continuous Paper Length	11
STX E Set Quantity For Stored Label	11
STX e Select Edge Sensor	11
STX F Form Feed	11

STX G	Print Last Label Format.....	12
STX I	Input Image Data.....	12
STX i	Downloading Scalable Fonts.....	13
STX J	Sets Pause for Each Label.....	13
STX K	Extended-System Commands.....	13
STX k	Test RS-232 Port.....	14
STX L	Enter Label-Formatting Command.....	14
STX M	Set Maximum Label Length.....	14
STX m	Set Printer To Metric.....	14
STX n	Set Printer to Imperial (Inches).....	15
STX o	Cycle Cutter.....	15
STX P	Character (HEX) Dump Mode.....	15
STX p	Controlled Pause.....	15
STX Q	Clear All Modules.....	15
STX q	Clear Module.....	16
STX R	Ribbon Saver On/Off.....	16
STX r	Select Reflective Sensor.....	16
STX S	Set Feed Rate.....	17
STX T	Printhead Dot Pattern Test Label.....	17
STX t	Test RAM Memory Module.....	17
STX U	Label Format Field Replacement.....	18
STX V	Software Settings.....	19
STX v	Printer's Firmware Version Information.....	19

STX W	Request Memory Module Information	20
STX w	Test Flash Memory Module	20
STX X	Set Default Module.....	21
STX x	Delete File from Module	21
STX Y	Output Sensor Values	22
STX y	Select Font Symbol Set.....	22
STX Z	Print Internal Information	23
STX z	Pack Module.....	23

5.0 Extended-System Commands <STX>K

Extended-System Command Functions		25
STX Kf	Set Present Distance.....	25
STX KQ	Query Memory Configuration	26
STX Kq	Query Memory Configuration (new format)	27
STX Kr	Resettable Counter Reset.....	28
STX KV	Verifier Enable/Disable.....	28

6.0 Label-Formatting Commands

Label-Formatting Command Functions		29
:	Set Cut By Amount.....	29
A	Set Format Attribute	30
C	Set Column Offset Amount	31
c	Set Cut By Amount.....	31
D	Set Dot Size Width and Height.....	32
E	Terminate Label Formatting Mode and Print Label	32

G	Place Data in Global Register	33
H	Enter Heat Setting	33
M	Select Mirror Mode	34
m	Set Metric Mode	34
n	Set Inch Mode (Imperial)	34
P	Set Print Speed	36
p	Set Label Backup Speed	36
Q	Set Quantity Of Labels To Print.....	37
R	Set Row Offset Amount.....	37
r	Recall Stored Label Format.....	38
S	Set Slew Rate.....	39
s	Store Label Format In Module.....	39
T	Set Field Data Line Terminator	40
U	Make Previous Field A String Replace Field	41
X	Terminate Label-Formatting Mode	42
y	Select Font Symbol Set.....	42
z	Zero (Ø) Conversion to "0"	42
+ (>)	Make Last Field Entered Increment Numeric (Alphanumeric).....	43
- (<)	Make Last Field Entered Decrement Numeric (Alphanumeric)	44
^	Set Count by Amount	45
	Special Label-Formatting Commands	46
STX S	Recall Global Data And Place In Field	46
STX T	Print Time and Date	47

7.0 Font-Loading Commands <ESC>

Font-Loading Command Functions	48
*c###D Assign Font ID Number	48
)s###W Font Descriptor	48
*c###E Character Code	49
(s#W Character Download Data	49

8.0 Generating Label Formats

Format Record Commands	50
Generating Records.....	51
The Structure of a Record	52
Record Structure Types.....	55
Internal Bit-Mapped Fonts	56
Smooth Font, Font Modules, and Downloaded Bit-Mapped Fonts	56
Scalable Fonts	57
Barcodes.....	57
Images	58
Graphics	58
Lines and Boxes	59
Polygons	60
Circles.....	60
Available Fill Patterns for Polygons and Circles:	61
Graphics Examples:.....	62

Appendix A

ASCII Control Chart	A1
---------------------------	----

Appendix B

Sample Programs	B1
-----------------------	----

Appendix C

Available Fonts – Sizes, Referencing, and Samples	C1
---	----

Appendix D

Reset Codes	D1
-------------------	----

Appendix E

Single Byte Symbol Sets	E1
-------------------------------	----

Appendix F

Barcode Summary Data	F1
----------------------------	----

Barcode Default Widths and Heights	F2
--	----

Appendix G

Barcode Details	G1
-----------------------	----

Appendix H

Font Mapping - Single and Double Byte Characters	H1
--	----

Appendix I

Symbol Sets and Character Maps	I1
--------------------------------------	----

Symbol Set Selection	I1
----------------------------	----

Double-Byte Symbols, Chinese, Kanji	I3
---	----

Appendix J

General Purpose I/O Signals	J1
-----------------------------------	----

Appendix K

Print Resolutions, Module Identifiers, Maximum Field, Column & Character Values	K1
--	----

Appendix L

Printer Speed Ranges	L1
----------------------------	----

Appendix M

Commands by Function	M1
----------------------------	----

Appendix N

<i>mi-4</i> and <i>mi-6</i> Constraint Cross-Reference	N1
--	----

Appendix O

Image Loading	O1
---------------------	----

Appendix P

UPC-A and EAN-13: Variable Price/Weight Barcodes	P1
--	----

Appendix Q

International Language Print Capability (ILPC) Programming Examples	Q1
---	----

Appendix R

Barcode Symbology Information Sources.....	R1
--	----

1.0 Introduction

Scope of this Manual

This manual is intended for programmers who wish to create their own label production software. Operators without programming experience may prefer to use a label-creation software package.

This manual explains in detail the printers programming language (DPL) and its related uses in the writing, loading and storing of programs for the control and production of label formats (designs) using the **mi-4** and **mi-6** Series printers.

Notes: This manual refers to IBM-PC based keyboard command characters for access to the ASCII character set. Systems based on different formats, (e.g., Apple's Macintosh™), should use the appropriate keyboard command to access the desired ASCII character. Appendix A contains the entire ASCII character set.

<CR> is used to identify the line termination character. Other strings placed between < > in this document represent the character of the same ASCII name.

References to 'menu settings' refer to the printer's menu driven display system. Please refer to the appropriate printer Operator's Manual for details.

The four main command types used to create labels and control the printer are:

1. Status & Control Commands
 2. System-Level Commands
 3. Label-Formatting Commands
 4. Font-Loading Commands
- **Status & Control Commands:** Perform status queries and printer control commands.
 - **System-Level Commands:** Performed in the sequence received, generally control printer hardware, allow scalable font and image downloading and include **Extended-System Commands**.
 - **Label-Formatting Commands:** Controls the position of text and images on the media. The label format termination commands can selectively store, print and end the formatting process.
 - **Font-Loading Commands:** Used to download font data in PCL-4 compatible bit-maps.

2.0 Control Codes

Attention Getter Functions

The printer requires a special “attention getter” character in order to receive a command sequence. This informs the printer that it is about to receive a command and the command type. Status and Control Commands, System-Level Commands, and Font-Loading Commands have their own unique attention getter, followed by a command character that directs printer action.

ASCII Character	Decimal Value	HEX Value	Attention Getter For:
SOH	1	01	Status and Control Commands
STX	2	02	System-Level Commands
ESC	27	1B	Font-Loading Commands

Table 2-1 Control Codes

The attention getters (e.g., “SOH”) are standard ASCII control labels that represent a one character control code, (i.e., ^A or Ctrl A). Appendix A contains the entire ASCII Control Code Chart. An alternate set is shown below. This set is enabled via the menu.

Control Character	Standard	Alternate
SOH	0x01	0x5E
STX	0x02	0x7E
CR	0x0D	0x0D
ESC	0x1B	0x1B
“Count By” ¹	0x5E	0x40

1. See Label-Formatting Commands, ^ set count by amount.

Table 2-2 Alternate Control Codes

☑Note: Throughout this manual <SOH>, <STX>, <CR>, <ESC>, ^, will be used to indicate the control codes. The actual values will depend on whether standard or alternate control codes are enabled for the particular application.

3.0 Status & Control Commands <SOH>

Status & Control Command Functions

When the printer receives a Status and Control Command, its current operation will be interrupted momentarily to respond to the command. Status and Control Commands may be issued before or after System-Level Commands; however, they cannot be issued among (a) Label-Formatting Commands, (b) during font downloading or (c) during image downloading. Status and Control Commands consist of:

1. Attention Getter, 0x01 or 0x5E, see Control Codes.
2. Command Character
3. Parameters (if any).

SOH # Reset

This command resets the printer. Resetting the printer returns all settings to default and clears both the communications and printing buffers. The command also clears the internal RAM memory. See Appendix K.

Syntax: <SOH>#

Sample: <SOH>#

Printer Response: The printer will reset.
 <XON> T
 (The T may come before the <XON>)

SOH A *Send ASCII Status String*

This command allows the host computer to check the current printer status. The printer returns a string of eight characters to the host, followed by a carriage return, see table below. Each character is either a 'Y' or 'N' indicating that the associated condition is true (Y) or false (N). Byte 1 is the first character transmitted by the printer. See <SOH>F.

Syntax: <SOH>A

Sample: <SOH>A

Printer Response: abcdefgh<CR>

Where:

Byte	Character Y/N			Condition	
1	a	=	Y/N	Y =	Interpreter busy (Imaging)
2	b	=	Y/N	Y =	Paper out or fault
3	c	=	Y/N	Y =	Ribbon out or fault
4	d	=	Y/N	Y =	Printing batch
5	e	=	Y/N	Y =	Busy printing
6	f	=	Y/N	Y =	Printer paused
7	g	=	Y/N	Y =	Label presented
8	h	=	N		Always No

Table 3-1 ASCII Status Bytes

SOH B *Toggle Pause*

This command toggles the printer's paused state on and off. (This is the same function achieved when pressing the PAUSE button on the printer.)

Syntax: <SOH>B

Sample: <SOH>B

Printer Response: This command will illuminate the pause/stop indicator, suspend printing and wait until one of the following happens:

- The <SOH>B command is sent to the printer.
- The <STX>p command is sent to the printer.
- The PAUSE button is pressed.

Upon which it will turn the pause/stop indicator off and resume operation from the point of interruption.

SOH C Stop/Cancel

This command performs the same function as pressing the STOP/CANCEL button on the printer's front panel. This function clears the current label format from the print buffer, pauses the printer and illuminates the pause/stop indicator.

Syntax: <SOH>C

Sample: <SOH>C

Printer Response: The pause condition is terminated as described under <SOH>B.

SOH E Send Batch Quantity

This command causes the printer to send back a 4-digit number indicating the quantity of labels left to print in the current batch, followed by a carriage return. Communications latency may cause this value to be higher than actual on some printers.

Syntax: <SOH>E

Sample: <SOH>E

Printer Response: *nnnn*<CR>

Where: *nnnn* - Is four decimal digits, 0-9999.

SOH F Send Status Byte

This command instructs the printer to send a single status byte where each bit (1 or 0) represents one of the printer's status flags, followed by a carriage return, see table below. If an option is unavailable for the printer, the single bit will always be 0. A satisfied condition equals 1. See <SOH>A.

Syntax: <SOH>F

Sample: <SOH>F

Printer Response: X<CR>

Where:

Bit	Value	Condition
8	0	Always zero
7	1 or 0	Label presented
6	1 or 0	Printer paused
5	1 or 0	Busy printing
4	1 or 0	Printing batch
3	1 or 0	Ribbon out or Fault
2	1 or 0	Paper out or Fault
1	1 or 0	Command interpreter busy (Imaging)

Table 3-2 Status Byte

Note: One is the least significant bit (LSB).

SOH U Update System Database with Current Database

This command will copy the current User database into the system database. There are various DPL commands that modify the configuration of the printer. See the <STX>K...command for specifics on how these commands function. The user should send the <SOH>U command after these configuration commands so that these commands can take affect. This command will cause the printer to reset. See the <SOH># command above for details on what events occur during a reset.

Syntax: <SOH>U

Sample: <SOH>U

Printer Response: The printer will reset.
 <XON>T
 (The T may come before the <XON>).

4.0 System-Level Commands <STX>

System-Level Command Functions

The most commonly used commands are the System-Level Commands. These are used to load and store graphic information, in addition to printer control. System-Level Commands are used to override default parameter values (fixed and selectable) and may be used before or after Status and Control Commands but cannot be issued among Label-Formatting Commands. Selectable parameter values may be assigned via menu selections. Menu selectable parameters are provided in the Operator's Manual of the printer.

System-Level Commands consist of:

1. Attention Getter, 0x02 or 0x7E, see Control Codes.
2. Command Character
3. Parameters (if any).

STX A Set Time and Date

This command sets the time and date. The initial setting of the date will be stored in the printer's internal memory. This date can be verified by printing a configuration label.

Syntax: <STX>AwmmddyyyhhMMjjj

Sample: <STX>A6070319990855000

Printed Response: SAT JUL 3, 1999, 8:55AM, 185 (185th day of the year).

Where:

w	1 digit for day of week; 1 = Monday
mm	2 digits for month
dd	2 digits for day
yyyy	4 digits for year
hh	2 digits for hour in 24 hour format
MM	2 digits for minutes
jjj	3 digits for Julian date / constant (see notes below)

Notes: Time can also be set via the menu.

Printers without the Time/Date Option lose the time/date when power is removed.

Sending the printer a non-zero Julian date will freeze the Julian date.

Response format is variable, see Special Label-Formatting Command<STX>T.

STX a Enable Feedback Characters

This command enables the feedback ASCII HEX characters 07, 1E, and 1F to be returned from the printer following specific events. For printers without menu driven operator panels, the default value is 'Off'.

Syntax: <STX>a

Sample: <STX>a

Printer response is event dependent:

Where:

Event	Printer Transmits
Invalid character	(BEL) 0x07
Label printed	(RS) 0x1E
End of batch	(US) 0x1F

Table 4-1 Printer Return Characters

STX B Get Printer Time and Date Information

This command retrieves the time and date information from the printer and transmits the response to a host device. See <STX>A for details and restrictions.

Syntax: <STX>B

Sample: <STX>B

Printer Response Syntax: wmmddyyyyhhMMjjj <CR>

Where:

w	1 digit for day of week; 1 = Monday
mm	2 digits for month
dd	2 digits for day
yyyy	4 digits for year
hh	2 digits for hour in 24 hour format
MM	2 digits for minutes
jjj	3 digits for Julian date / constant*

*See <STX>A

Printer Response Sample: 6070319990855185 <CR>

STX c Set Continuous Paper Length

This command is used to define the page (label) size when printing on continuous media. It disables the media sensor top-of-form function but continues to monitor paper-out conditions. See <STX>M.

Syntax: <STX>cnnnn

Where: nnnn - Is in inches/100 or millimeters/10 (see <STX>m), the length of the paper feed for each label format. It must be set to zero for edge or reflective top-of-form operation.

Sample: <STX>c0100

This sample sets a page length of 100, which equals 1.00 inch (assuming Imperial Mode is selected).

STX E Set Quantity For Stored Label

This command sets the number of labels to print for the label format currently stored in the print buffer. (The printer automatically stores the most recent label format in the buffer until the printer is reset or turned 'Off'.) To print, this command is used in conjunction with the <STX>G command.

Syntax: <STX>Ennnn

Where: nnnn - A four-digit quantity, including leading zeros.

Sample: <STX>E0025
 <STX>G

Printer Response: 25 labels of the current format in memory will be printed.

❑Note: This command may be issued prior to a label format without a specified quantity (Qnnn).

STX e Select Edge Sensor

This command enables "see-through" sensing for top-of-form detection of die-cut and notched media. The printer's transmissive media sensor will detect a minimum gap of 0.1 inches (2.54mm) between labels. Label stock must be at least 0.5 inches (12.7mm) between each top-of-form (start of print point on the next label).

Syntax: <STX>e

❑Notes: This command is ignored when <STX>c has been issued with a non-zero value for nnnn. See the <STX>r command for reflective sensing.
--

STX F Form Feed

This commands the printer to form feed one label.

Syntax: <STX>F

STX G *Print Last Label Format*

This command prints a previously formatted label and restarts a canceled batch job after the last processed label. This is used when there is a label format in the print buffer. The <STX>E command is used to enter the quantity.

Syntax: <STX>G

STX I *Input Image Data*

This command must precede image data downloading from a host computer to the printer. The data that immediately follows the command string will be image data. See Appendix O for more information. To print the image, see Generating Label Formats.

Syntax: <STX>labfnn...n<CR>data

Where: *a* - Memory Module Bank Select, A-D (see Appendix K)

b - Data value range (optional), A or omit (see Table 4-3).

<i>b</i> Value	Image Data Value Range
A	ASCII Characters 0-9, A-F, (7 bit)
omitted	00-FF, (8 bit)

Table 4-3 Image Data Values

f - Image format (see Table 4-4).

<i>f</i> Designator	Format
F	7-bit image load file
B	.BMP 8-bit format (image will be flipped), 256 color or B&W
b	.BMP 8-bit format (save image as received), 256 color or B&W
I	.IMG 8-bit format (image will be flipped), B&W
i	.IMG 8-bit format (save image as received), B&W
P	.PCX 8-bit format (image will be flipped), B&W
p	.PCX 8-bit format (save image as received), B&W

Table 4-4 Image Data Formats

nn...n - Up to 16 characters used as an image name;

<CR> - 0x0d terminates the name.

data - Image data

Sample: <STX>IDpTest <CR>
 data...data<CR>

The sample sets the printer to (1) receive an 8-bit PCX image sent by the host in an 8-bit data format, (2) name the image 'Test', and (3) store it in memory module D.

STX i Downloading Scalable Fonts

The command structure for downloading both IntelliFont (.CDI) and TrueType (.TTF) scalable fonts follows, (font files may be single or double byte character systems).

Syntax: <STX>*imtnnName*<CR>*xxxxxxxxdata...*

Where:

<i>m</i>	-	Module ID to save this font to ('A', 'B', 'C', 'D')
<i>t</i>	-	Type of scalable font being downloaded: <ul style="list-style-type: none">➤ 'T' - IntelliFont➤ 'T' - TrueType
<i>nn</i>	-	Two digit font reference ID. Valid range is 50-99, 9A-9Z, 9a-9z (base 62 numbers).
<i>Name</i>	-	The title for this font. Up to 16 characters.
<CR>	-	0x0d terminates the name.
<i>xx...x</i>	-	Eight-digit size of the font data, number of bytes, hexadecimal, padded with leading zeros.
<i>data</i>	-	The scalable font data.

Sample: <STX>iDT52Tree Frog<CR>000087C2data...

This sample downloads a TrueType font to module 'D' and assigns it the Font ID of 52 with the name "Tree Frog". The size of the font data is 0x87C2 bytes long.

STX J Sets Pause for Each Label

This command causes the printer to pause after printing each label and is intended for use with the optional Peel and Present Mechanism or with a Tear Bar when no Present Sensor is installed. After removing the printed label, the PAUSE button must be pushed in order to print the next label. (The printer must be reset to clear the <STX>J command.)

Syntax: <STX>J

STX K Extended-System Commands

This expands the System-Level Commands. See the Extended-System Commands for more information.

STX k Test RS-232 Port

This command instructs the printer to transmit the character Y from the printer's RS-232 port. (Failure to receive Y could indicate an interfacing problem.)

Syntax: <STX>k

Sample: <STX>k

Printer Response: Y

STX L Enter Label-Formatting Command

This command switches the printer to the Label-Formatting command input mode. Once in this mode, the printer expects to receive Record Structures and Label-Formatting commands. Status and Control, System-Level, and Font-Loading commands will be ignored until the label formatting mode is terminated with E, s, or X, (see Label-Formatting Commands for additional information.)

Syntax: <STX>L

STX M Set Maximum Label Length

This command instructs the printer to travel this distance in search of top-of-form (label edge, notch, black mark, etc.) before declaring a paper out fault. The fault condition can occur if this value is set too close to the physical length of the label (within 0.1inch/2.54mm). Therefore, it is good practice to set this value at 2.5 to 3 times the actual label length used. The minimum value should be at least 5" (127mm).

Syntax: <STX>Mnnnn

Where: nnnn - A 4-digit length, 0000-9999, in/100 or mm/10. Maximum setting is 9999 (99.99 inches or 2540mm).

Default: 12 inches/304.8mm

Sample: <STX>M0500

The sample sets a maximum travel distance of 5 inches (unless printer is in metric mode, see <STX>m).

STX m Set Printer To Metric

This command sets the printer to interpret measurements as metric values (e.g., <STX>c0100 will equal 10.0mm). See <STX>n.

Syntax: <STX>m

Default: Imperial (Inches)

STX n Set Printer to Imperial (Inches)

This command sets the printer to interpret measurements as Imperial values (e.g., <STX>c0100 will equal 1.00 inch). See <STX>m.

Syntax: <STX>n

Default: Imperial (Inches)

STX o Cycle Cutter

This command will immediately cycle the cutter blade on the option Cutter Mechanism. The Cutter must be installed, enabled and the interlocks closed for operation.

Syntax: <STX>o

STX P Character (HEX) Dump Mode

This command instructs the printer to enter the Character Hex Dump Mode (ASCII Dump / “monitor mode”). All data sent to the printer following this command will be printed in the raw ASCII format. Labels must be at least four inches (102mm) long and as wide as the maximum print width. This command has the same effect as turning the printer ‘On’ while pressing the FEED button; in this case however, a Configuration and Test label will be printed. To return to normal operation, the printer must be manually reset.

Syntax: <STX>P

STX p Controlled Pause

The controlled pause command will cause the printer to pause only after all previously received commands are executed. This is often useful between label batches. (This command will not clear the pause condition, see <SOH>B).

Syntax: <STX>p

STX Q Clear All Modules

This command instructs the printer to clear all of the Flash, RAM, and Internal Modules (see the corresponding printer Operator's Manual for applicable module options).

Syntax: <STX>Q

STX q ***Clear Module***

This command clears the selected memory module (Flash or RAM). If a module is corrupted during normal operations, it must be cleared. A corrupt module is identified when the printer responds with a 'No Modules Available' message to a <STX>W (request memory module information command).

Syntax: $\langle \text{STX} \rangle qa$

Where: a - Memory module bank select, A-D

Sample: $\langle \text{STX} \rangle_{\text{qA}}$

The sample clears memory module A.

❑Note: If a module directory intermittently returns the message ‘No Modules Available’ or if data continuously becomes corrupt with the write protect switch on, the module may be at the end of its service. However, before concluding that a module is defective, cycle the printer's power and test the module.

Turn 'Off' the Write Protect Switch on Flash Modules before using this command.

STX R Ribbon Saver On/Off

This is the only command used for the optional Ribbon Saver. This does not instruct the Ribbon Saver when to engage, rather it turns the operation ‘On’ or ‘Off’. The option will operate automatically, lifting when the minimum amount of label white space is exceeded. The operation is continuous when enabled. The printer must be set to thermal transfer (ribbon) printing mode.

Syntax: <STX>R_x

Where:

x	- Y - Enabled (Default = Menu selection.)
	N - Disabled

Sample: <STX>RN

The sample will turn the ribbon saver off.

✔Note: This command is ignored on units not equipped with the ribbon saver option.

STX_r ***Select Reflective Sensor***

This command selects the reflective media sensor for top-of-form detection of reflective stock (e.g., fan-folded tags with carbon-based black marks printed along the underside of the media). The end of the black mark determines label TOF. Labels must be at least 0.5" (13mm) between each TOF mark. (The <STX>e command switches from reflective back to edge sensing).

Syntax: $\langle \text{STX} \rangle_r$

Default setting: Edge sensing

STX S Set Feed Rate

This command determines the feed rate of the media when the printer's FEED button is pressed.

Syntax: <STX> S*n*

Where: *n* - Is a letter value from A to S (see Appendix L).

STX T Printhead Dot Pattern Test Label

This command instructs the printer to print a dot pattern test label. To view all dots, the labels must be at least four inches (102mm) long and as wide as the maximum print width.

Syntax: <STX>T

STX t Test RAM Memory Module

This command tests all RAM modules; however, the printer must be in feedback mode for the command to function. The printer returns a one-line message stating whether the module is good or bad, (no message is returned if a module is unavailable). Feedback mode can be enabled via the menu (see the printer's Operator Manual for additional information).

Syntax: <STX>t

Printer Response: Module A: xxxxK Ram Tested results<CR>
 Module B: xxxxK Ram Tested results<CR>
 Module D: xxxxK Ram Tested results<CR>

Where: xxxx - Module size in Kbytes.

Results: Module test results: 'Good' or 'Bad'.

STX U Label Format Field Replacement

This command places new label data into format fields. Format fields are used to build a label. The new data string must equal the original string length and contain valid data. To easily keep track of fields, place all of the fields to be updated with the command at the beginning of the label format. A maximum of 99 format fields can be updated. Fields are numbered consecutively 01 to 99, in the order received.

Syntax: <STX>Unnss..s<CR>

Where: nn - Is the format field number, 2 digits.

ss..s - Is the new string data followed by a <CR>

Sample: <STX>L
161100001000100data field 1<CR>
161100001100110data field 2<CR>
161100001200120data field 3<CR>
Q0001
E
<STX>U01New data F1<CR>
<STX>U02New data F2<CR>
<STX>E0002
<STX>G

The sample results in three printed labels. The first label is formatted with the commands between <STX>L and E. The next two labels print with the replacement data contained in the <STX>U commands (see <STX>E and <STX>G).

STX V Software Settings

Printer options are set by entering selections through the menu. The software setting command allows two of these option settings to be modified without returning to the menu. Choosing the appropriate values allow the option(s) to be turned 'On' or 'Off'. Each has a corresponding bit whose value is 1 when enabled. The bit value tables indicate the command value for the desired bit, see below.

Syntax: <STX>Vn

Where: n - Is a single digit ASCII numeric value from 0-F. The value of n is used to override the power up option settings. Reset or power-up returns the printer to the original settings.

Sample: <STX>V5

The sample corresponds to setting Bits 0 and 2. When applied, this sets the printer to the following: "Label Present 'On'; Cutter Enabled."

Bit Assignment	Options
0	Cutter
1	N/A
2	Label Present
3	N/A

Table 4-5 Bit Assignment Table

Use the bit assignment table above to determine the command value n in the binary table below. Example: a command value of 1 will set bit 0.

Command Values for Bits Assigned										
Value	Bit					Value	Bit			
n	3	2	1	0		n	3	2	1	0
0	0	0	0	0		8	1	0	0	0
1	0	0	0	1		9	1	0	0	1
2	0	0	1	0		A	1	0	1	0
3	0	0	1	1		B	1	0	1	1
4	0	1	0	0		C	1	1	0	0
5	0	1	0	1		D	1	1	0	1
6	0	1	1	0		E	1	1	1	0
7	0	1	1	1		F	1	1	1	1

Table 4-6 Value Assignments

STX v Printer's Firmware Version Information

This command causes the printer to respond by sending a version string. The version may be different from printer to printer (this data is the same as that printed on the configuration label).

Syntax: <STX>v

Printer Response: VER: 4206 – 01.01 06/15/1999<CR>

STX W Request Memory Module Information

This command requests a directory listing for memory module(s). Although a module can store font, graphics and format data together, it can display only one type of information at a time. If the module contains all three types of data, it will be necessary to check the directory three times, using each of the control parameters, F, G, and L, to determine the contents.

Syntax: <STX>W*a*

Where: *a* - F = Font data
 G = Graphic data
 L = Format data

Sample: <STX>WG

Printer Response*: MODULE: D<CR>
 LOGO<CR>
 CAM<CR>
 AVAILABLE BYTES IN MODULE: 00049083<CR>

*The response shows Module D contains two graphic image files, LOGO and CAM, with 49083 bytes remaining free.

STX w Test Flash Memory Module

This command performs a test sequence on flash memory modules; however, the time for each test will vary (from 20 seconds up to two minutes), depending on the size of the module. The module must have the write protect switch 'Off' for testing; all stored data will be destroyed. The printer responds with 'good' or 'bad' message results for each module tested. (No modules present will result in no printer response.)

Syntax: <STX>w*a*

Printer response format: Module *a*: xxxxK Flash test results.

Where: *a* - Module = A or B

 xxxx - Module size in Kilobytes

 results - Test results: 'Good' or 'Bad'

STX X Set Default Module

This command is used when downloading information to a module memory. It is designed to allow the user to select between modules. If 'C' is entered to select a memory bank, the data will go to the bank set by this command (Set Default Module). If the printer uses only one bank, this command is not required.

Syntax: <STX>X*a*

Where: *a* - A, B, C, D; memory module designator (See Appendix K)

Sample: <STX>XB

The sample sets the printer's default module to memory module B.

The default module is one of the following:

1. The first alpha designator of existing modules if items 2 or 3 have not occurred.
2. The most recent module to be inserted while the power is on.
3. The module selected by this command.

Note: This is typically used prior to loading of PCL-4 bit-mapped fonts, (see Font-Loading Commands).

STX x Delete File from Module

This command removes a specific file from the specified module. The file name is removed from the module directory and thus the file cannot be accessed. The actual storage space occupied by the file is not released. The pack command can be issued to reclaim all deleted file storage space, (see <STX>z).

Syntax: <STX>x*mtnn...n*<CR>

Where: *m* - The module identification character, A – D (see Appendix K).

t - The file type identification code:
 'G' = Graphic (Image) file type
 'L' = Label Format file type
 'F' = Bit-Mapped file type
 'S' = Smooth Scalable Font file type

nn...n - The file name to delete. Up to 16 characters for graphic (image) or label format files. Three digits for bit-mapped Font files and two digits for smooth scalable Font files.

STX Y ***Output Sensor Values***

This command causes the printer to respond with the sensor value status. The printer must be in feedback mode (menu setting). When <STX>Y is received, the printer will respond with the internal A-D sensor values, see table below. To repeat the display of values, send the printer a 'SPACE' (20 hexadecimal). Send <ESC> to terminate the function.

Syntax: <STX>Y

Sample: <STX>Y

Printer Response: THR-131; TRAN-141; RIBM-069; 24V-145; PS-007; HD-147; RANK-128<CR>

Where:

Abbreviation	Sensor Output
THR	Printhead Thermistor
TRAN (or REFL)	Transmissive / Reflective Media Sensor
RIBM	Ribbon Motion Detector
24V	24 Volt DC Power Supply
PS	Present Sensor
HD	Printhead Position Sensor (Head Down)
RANK	Printhead Ranking Resistance

Table 4-7 Printer Sensor Outputs

STX_y ***Select Font Symbol Set***

This command selects the scalable font symbol set. The selected symbol set remains active until another symbol set is selected. See Appendices E, I, and H for details. Option dependant. Not all symbol sets can be used with all fonts.

Syntax: $\langle \text{STX} \rangle_y S_{xx}$

Where:

<i>S</i>	-	Byte-size designation; see Appendix H: S = Single byte symbol sets. U = Double byte symbol sets.
----------	---	--

xx - Symbol set selection.

Sample: <STX>_ySPM

The sample selects the PC-850 multilingual set.

STX Z Print Internal Information

This command prints the configuration. To capture all printed information, use labels as wide as the print width and at least 4 inches (102mm) long.

Syntax: <STX>Z

STX z Pack Module

This command causes the printer to reclaim all storage space associated with all deleted files on the specified module. A Flash Module cannot be packed.

Syntax: <STX>zm

Where: *m* - The module identification character, A-D. (See Appendix K, <STX>X, and <STX>x for additional information.)

5.0 Extended-System Commands <STX>K

Extended-System Command Functions

Extended-System Commands expand certain System-Level Commands, providing extra printer control. Extended-System Commands are issued in the same context as System-Level Commands.

Command Character	Command
f	Set Present Distance
Q	Query Memory Configuration
q	Query Memory Configuration (new format)
r	Reset Internal Counters
V	Verifier Enable/Disable

Table 5-1 Command Characters

STX Kf Set Present Distance

This command specifies an additional amount to advance the label after printing.

Syntax: <STX>Kfnnnn

Where: nnnn - Is a four-digit present distance in inches/100 or mm/10.

Sample: <STX>Kf0100

The sample represents a one-inch label advance unless the printer is in metric mode (see <STX>m).

STX KQ Query Memory Configuration

This command causes the printer to transmit its internal DRAM memory configuration to the host device. The transmitted data provides information regarding the total amount of internal DRAM installed, the amount available for configuration and the amount currently assigned to specific functions or entities.

Syntax: <STX>KQ

Sample: <STX>KQ

Printer Response: Product: I4206 – 01.01 05/21/1999
 Installed RAM: 8 MB
 Label Width: 4.09 IN
 Print Buffer Size: 272 IN
 Allocation RAM: 6389 KB
 Internal Files: 512 KB
 Font Cache: 232 KB

Where:

Data	Description
Installed RAM	Total amount of RAM.
Label Width	Size in inches/millimeters of the printhead.
Print Buffer Size	Total number of inches/millimeters of Print Dot Buffers available. (This is not the maximum size of a label, which is limited to 99.99 inches.)
Allocation RAM	Amount of RAM that can be configured for the Internal Files, Font Cache and the remainder going to the Print Buffer Size.
Internal Files	Size of the Internal Module used to store downloaded fonts, graphics and label formats.
Font Cache	Size of the Font Buffer used to temporarily store characters. Increasing this buffer will increase performance if labels have a large variety of font sizes and characters.

STX Kq Query Memory Configuration (new format)

This command causes the printer to transmit its internal DRAM memory configuration to the host device. The transmitted data provides information regarding the total amount of internal DRAM installed, the amount available for configuration, and the amount currently assigned to specific functions or entities.

Syntax: <STX>Kq

Printer response format: Memory Configuration<CR>
Product: aaaacdd.ee mm/dd/yy<CR>
Installed Ram: iiiiiMB<CR>
Label Width: vvvvoo<CR>
Print Buffer Size: :xxxxoo<CR>
Allocation RAM: ssssKB<CR>
Internal Files LLLLKB<CR>
Font Cache wwwwww:KB<CR>

Where: <CR> - ASCII Carriage Return (0x0D) record delimiter.

aaaacdd.ee - ASCII string sequence that represents the firmware version
mm/dd/yy number string.

iiii - The number of Megabytes of installed internal DRAM memory.

vvvv - The length of the Label Width.

xxxx - The length of the Print Buffer.

ssss - The number of Kilobytes of internal memory assigned to the label Print Buffer

LLLL - The number of Kilobytes assigned to the internal memory module.

wwwwww - The number of Kilobytes assigned to the Scalable Cache.

oo - Current label dimension unit's designation. "IN" for inches and "MM" for millimeters.

Syntax: $\langle \text{STX} \rangle \text{Kr}$

STX KV ***Verifier Enable/Disable***

Syntax: $\langle \text{STX} \rangle \text{KV}_a$

mi-4 and mi-6 Series Programmer's Manual

6.0 Label-Formatting Commands

Label-Formatting Command Functions

An <STX>L command switches the printer from the System-Level to the Label-Formatting Command processor. All commands following the <STX>L are interpreted as label formatting commands. Label-Formatting Commands can be used to override default parameter values. Selectable parameter value defaults may be assigned via the menu selection on printers so equipped. Menu-selectable parameters are shown in the printer Operator's Manual. Label formats that contain no commands overriding printer default values will assume the defaults.

: *Set Cut By Amount*

This command allows a predetermined number of labels to be printed before a cut is initiated. This feature is useful when it is necessary to print an uncut strip of labels. Between 1 and 9999 labels may be printed before a cut is made. The amount must be smaller than the quantity of labels printed.

Syntax: :nnnn

Where: nnnn - Is a four digit decimal number indicating the number of labels to be printed before a cut is made.

Sample: <STX>L<CR>
 :0005
 141100001000100SAMPLE LABEL<CR>
 Q0021
 E

The sample instructs the printer to make cuts after labels 5, 10, and 20 have been printed. Label 21 will be cut at the start of a subsequent label format (batch) unless a default (cut by amount) greater than one (1) has been entered via the menu on equipped models.

Note: The cutter must be enabled and the interlocks closed for operation.
--

A *Set Format Attribute*

This command specifies the type of format operation and remains in effect until another format command is specified or another label format has begun (<STX>L). Each label format defaults to attribute 2 (Transparent Mode).

Syntax: *An*

Where: *n* - Is attribute mode 1, 2, 3 or 5. (See table below).





<i>n</i>	Attribute	Description	Example
1	XOR Mode:	In this mode, the region where text strings, images or barcodes intersect will not be printed. (An odd number of overlapping objects will print.)	
2	Transparent Mode	This is the default mode; the intersecting regions of text strings, images, and barcodes will print, allowing the user to print fields on top of one another.	
3	Opaque Mode	Interacting text is obliterated by the text formatted last. Each character cell is treated as opaque. This mode is effective only in rotation 1. See Record Structure Types.	
5	Inverse Mode	This mode allows inverse (white on black) printing; (e.g., a proportionally sized border and background are printed similar to photographic negative). If text or image fields overlap in this mode, the effect will be similar to the XOR mode.	

Table 6-1

Sample: <STX>L
A3
141100001000100METO<CR>
141100001100110METO<CR>
E

The sample sets the printer to opaque mode and produces one label.

C ***Set Column Offset Amount***

This command allows horizontal adjustment of the point where printing begins. This feature is useful when a single format must be printed on several different types of labels that contain pre-printed information. (If the pre-print does not appear in the same place on every label, the new data may overlap the pre-printed data.) The ‘C’ command instructs the printer to print label formats nnnn units to the right of the position that the format specifies.

Syntax: *Cnnnn*

Where: *nnnn* - Is a four-digit number for the column offset, inches/100 or mm/10. The printer default is 0 for offset.

Sample: <STX>L
 C0050
 141100001000100METO<CR>

The sample shifts all format data 0.5 inches to the right, unless the printer is in metric mode, (see Label-Formatting Command ‘m’).

c ***Set Cut By Amount***

This command is the same as the ‘:’ command except only a two-digit value can be entered. This command allows a predetermined number of labels to be printed before a cut is made. 1 to 99 labels may be printed before a cut is made.

Syntax: *cnn*

Where: *nn* - Is a two-digit number indicating the number of labels to be printed before a cut is made. The printer default is one.

Sample <STX>L<CR>
 c07<CR>
 141100001000100SAMPLE LABEL<CR>
 Q0021<CR>
 E

The sample instructs the printer to make cuts after labels 7, 14, and 21 have been printed. See Label-Formatting Command ‘:’.

☑Note: The cutter must be enabled and the cutter cover in place before the cutter will operate.

D Set Dot Size Width and Height

This command is used to change the size of a printed dot, hence the print resolution – dots per inch (DPI) of the printhead. By changing the height of a dot, the maximum length of a label can be increased or decreased. The table below lists the step sizes available.

Syntax: *Dwh*

Where: *w* - Is Dot Width multiplier 1 or 2.

h - Is Dot Height multiplier 1, 2, or 3.

Printhead Resolution (DPI)	Element (Dot) Size	
	Inches	Millimeters
203	.0049	.125
300	.0033	.085

Table 6-2 Printhead Dot Sizes

Note: D11 is the default value for all printers.

E Terminate Label Formatting Mode and Print Label

When the printer is processing Label-Formatting Commands and receives an ‘E’ command, it will immediately print a label based upon the data received at that point. Even if no printable data has been received, the printer will generate and feed a label (other termination commands are ‘X’ and ‘s’). Commands sent to the printer after a “terminate label” command must be Status and Control, System-Level or Font Download type.

Syntax: *E*

Sample: *<STX>L<CR>*
1211000000000000Testing<CR>
E<CR>

The sample will print one label.

G ***Place Data in Global Register***

The 'G' command saves the print data of a print format record in a global register (temporary storage). This data may be retrieved and copied to another record in the same label format using the special Label-Formatting Command, <STX>S. Global registers are named in the order received, beginning with register A, ending at register P, and incrementing with each instance of the G command use.

Syntax: G

Sample: <STX>L<CR>
1211000000000000Testing<CR>
G<CR>
1A2210001000000<STX>SA<CR>
E<CR>

The sample stores, retrieves and prints the data in global register A. One label is printed with "Testing" in two locations.

H ***Enter Heat Setting***

This command changes the "on time" of elements of the printhead. The default setting is 10 (except in the case of printers with a menu, where the default setting can be changed through the keypad). An increase or decrease in this value results in a change of heat applied by the printhead to the media, lightening or darkening the print contrast accordingly. This is helpful when using different media types, each requiring a different amount of heat to properly image the media. The host device can send this command value to correct the heat setting per the application.

Syntax: Hnn

Where: nn - Is a two-digit heat value (00-30)

Sample: <STX>L<CR>
H15<CR>
141100001000100SAMPLE LABEL<CR>
E

The sample sets the printer for a heat value of 15 and prints one label.

M ***Select Mirror Mode***

This command instructs the printer to “mirror” all subsequent print field records. This command toggles the mirroring mode. Mirrored fields are transposed visually, as if the object is viewed in a mirror.

Syntax: M

Sample: <STX>L
 161100003200010 NOT MIRRORED<CR>
 M<CR>
 161100003000400 MIRRORED<CR>
 E

Printed Result: NOT MIRRORED
 MIRRORED

m ***Set Metric Mode***

This command sets the printer to measure in metric. When this command is sent, all measurements will be interpreted as metric values, (e.g., a column offset of 0010 will be interpreted as 1.0mm). All printers default to Imperial (inch) mode. Menu selectable.

Syntax: m

Sample: <STX>L<CR>
 m
 141100001000100SAMPLE LABEL<CR>
 E

The sample will result in the text “SAMPLE LABEL” printed at starting location coordinates 10.0mm, 10.0mm.

n ***Set Inch Mode (Imperial)***

This command sets the printer to measure in inches. When this command is sent, all measurements will change to inches. All printers default to Imperial units. Menu selectable.

Syntax: n

Sample: <STX>L<CR>
 n
 141100001000100SAMPLE LABEL<CR>
 E

The sample will result in the text, “SAMPLE LABEL”, printed at starting location coordinates 1.00 inch, 1.00 inch.

P ***Set Print Speed***

This command sets a print speed for a label or batch of labels. See Appendix L for valid print speed ranges.

Syntax: *Pa*

Where: *a* - Is a single character (A to P) representing a speed.

Sample: <STX>L
 PC
 141100001000100LABEL1<CR>
 E
 <STX>L
 141100001000100LABEL2<CR>
 E

The sample prints two labels, the first at a speed of 3.0 inches per second (76 mm per second) and the second at the printer default.

p ***Set Label Backup Speed***

This command is used for the “Peel and Present” and “Cut Label Operations”. It allows the user to specify the rate at which the printer will reverse the label direction to align the next start of print position. The rate set by this command remains in effect until another backup speed command is received, see Appendix L for valid ranges. The speed is modified via the menu or when the printer is reset.

Syntax: *pa*

Where: *a* - Is a single character (C to I) setting a particular backup speed.

Sample: <STX>L
 pF

The sample sets the printer to a backup speed of 3.5 IPS.

Q Set Quantity Of Labels To Print

This command sets the number of the label copies to be printed. All printers default to 1.

Syntax: *Qnnnn*

Where: *nnnn* - Is a four-digit value setting for the number of labels to be printed.

Sample:
 <STX>L
 1211000000000000Testing<CR>
 Q0020<CR>
 E<CR>

The sample will print a batch of 20 identical labels.

R Set Row Offset Amount

This command allows vertical adjustment of the point where printing begins. This is useful when a single format is printed on several different types of labels that contain pre-printed information. However, if the pre-printing does not appear in the same place on every label, data may overprint the pre-printed areas. The 'R' command instructs the printer to print label formats *nnnn* increments above the position the format specifies. Valid input values are numbers between 0000 and 9999, (refer to the C, Set Column Offset Amount Command.)

Syntax: *Rnnnn*

Where: *nnnn* - Is a four-digit offset 0000-9999, inches/100 or millimeters/10.

Sample:
 <STX>L
 R0037<CR>
 141100001000100SAMPLE LABEL<CR>
 E

The sample sets the printer's offset row amount to 37 hundredths of an inch, (unless the printer is in metric mode).

r ***Recall Stored Label Format***

This command is used to retrieve entire label formats that have been stored on a memory module.

Syntax: *rmn...n*

Where: *nn...n* - Is a label name, up to 16 characters long.

The samples below show how to recall label formats. (To view a list of available label formats, use the memory module directory, <STX>W command.)

String Sent to Printer:	Printer Interpretation:
Sample 1: <STX>L<CR> rTEST<CR> Q0002<CR> E<CR>	Begin label format Retrieve format named TEST Quantity requested = 2 Terminate formatting and print
Sample 2: <STX>L<CR> rTEST<CR> X<CR> <STX>G<CR>	Begin label format Retrieve format named test Terminate formatting Print
Sample 3: <STX>L<CR> D11<CR> PO<CR> SO<CR> rTEST<CR> E<CR>	Begin label format Dot size = 1x1 Print speed 0 Slew speed 0 Retrieve format named test Terminate formatting and print

S ***Set Slew Rate***

This command sets the rate for the printer to feed non-printed areas of the label through the printer. The slew rate remains unchanged unless another slew rate command is sent, the printer is reset or a new rate is set via the menu. See Appendix L for values.

Syntax: *Sa*

Where: *a* - Is a one character value (C to S) which sets a maximum speed for label feeding.

Sample: <STX>L
 SE
 141100001000100LABEL1<CR>
 E
 <STX>L
 1411000010001000LABEL2<CR>
 E

The sample sets the slew rate to 3.0 IPS (76mms) and prints two labels. The slew rate for the second label is the same as the first.

s ***Store Label Format In Module***

This command stores a label format in a specified module. Supplying memory module name A, B, or D will store the label to that module. (Using C will cause the label format to be stored in whichever module has been set as the default module, refer to the Set Default Module Command, <STX>X.) The store label-format command will terminate the Label-Formatting Command.

Syntax: *sann...n*

Where: *a* - Is the module designator, A-D, representing a single character module name (Appendix K).

nn...n - Represents the label name (a maximum of 16 characters).

Sample: <STX>L<CR>
 D11<CR>
 191100501000000123456789012<CR>
 1911005020000001234567<CR>
 191100500000000Sample<CR>
 1X1100000000000B250250002002<CR>
 Q0001<CR>
 sATEST<CR>

The sample stores a label called 'Test' in memory module A. To recall the label format from the module use the Label-Formatting Command 'r'.

T ***Set Field Data Line Terminator***

This command is valid only for the next format record, after which the terminator defaults to a carriage return. This allows the user to embed special binary control codes (e.g., carriage returns) into the data to be printed. It is intended to be used with record types (e.g., PDF417), that will accept binary data.

Syntax: *Tnn*

Where: *nn* - Is an ASCII two-character representation of a HEX code to be used for the end of data terminator.

Sample: <STX>L<CR>
 T00<CR>
 1911002000000000TEST<NULL>
 141100001000100TERMIATOR<CR>
 Q0001<CR>
 E<CR>

The sample sets the terminator code to use a NULL terminator (ASCII NULL, DEC 0, HEX 00) for the end of data line. The terminator is immediately restored to a carriage return <CR> as seen in the format record containing the text 'TERMINATOR'.

U Make Previous Field A String Replace Field

This command does not require the use of an updated field (register loading); the command only controls the way the bit-mapped data is formatted. The data used when it is created must be valid for the font type being used. If the command is used, only the changing data will be reformatted; if this command is not used the entire label will be formatted. The U command is used in conjunction with the <STX>U command to reformat only portions of a label, resulting in faster label generation.

Syntax: U

Sample: <STX>L
 D11
 121100001000000123456789012<CR>
 U<CR>
 1211000020000001234567<CR>
 U<CR>
 161100000000000Sample<CR>
 1X1100000000000B250250002002<CR>
 Q0001
 E
 <STX>U01ABCDEFGHJKLM<CR>
 <STX>U028901234<CR>
 <STX>G

The sample sets up the label format for register loading and prints two labels. The first two of the four format records have been designated replacement (or update) fields. The second label is generated with System-Level Commands for field replacement and prints the last label.

☑Note: The data string length of any register is set by the length of the string when it was created and the new string must be the same length as the old string. The data being used when it is created must be valid for the font type being used.

X Terminate Label-Formatting Mode

When the printer is in Label-Formatting mode and receives this command, it will immediately switch to the System-Level Command mode and generate a label format based on whatever data it has already received. However, unlike the 'E' command, it will not print a label. (Other termination commands are the 'E' and 's' Label-Formatting Commands.)

Syntax: X

Sample: <STX>L<CR>
141100001000100SAMPLE<CR>
X<CR>

The sample will result in label formatting, but no label will be printed. The System-Level Command <STX>G will cause the label to print.

y Select Font Symbol Set

Same as a System-Level Command <STX>y. Symbol sets are used only with scalable fonts (see Generating Label Formats; also see <STX>y, <STX>I and Appendices E, I and H).

Syntax: ySxx

Where: S - Byte-size designation; see Appendix H:
S = Single byte symbol sets.
U = Double byte symbol sets.

xx - Symbol set selection.

Sample: ySSW<CR>
:

The sample selects the Swedish symbol set for use with all succeeding format records that use scalable fonts.

z Zero (Ø) Conversion to "0"

This command removes the slashes from zeros in fonts 0 to 8 and barcodes. The command is effective only for the label format in which it appears, and applies to all format records containing fonts 0 through 8 or barcodes A through Z. None of the smooth fonts (font 9) have a slash in the zero. The command has no effect on scalable fonts.

Syntax: z

Sample: <STX>L
z
1211000000000000Test0000<CR>
E

+ (>) *Make Last Field Entered Increment Numeric (Alphanumeric)*

The printer is capable of automatically incrementing fields on each label of a batch. This command is useful to print labels numbered in sequence. The data in the field will increment by the value after the + sign, each time a label is printed. The + character may be replaced by a > character to make the field increment alphabetically rather than numerically. This command is effective only on the label format record it follows. It is intended for use with the label batch quantity Q or the System-Level Commands for quantities and reprint, <STX>E and <STX>G.

Syntax: **pii*

Where: * - Is + for numeric increment, or > for alphanumeric increment.

p - Is the fill character for the left-hand character of the field.

ii - Is the amount by which to increment the field.

Sample: <STX>L<CR>
 13220000000000012345<CR>
 +01<CR>
 Q0003<CR>
 E<CR>

The sample will generate a single field label format that prints the initial label with a value of 12345 and then increments by one for the next two labels.

- (<) *Make Last Field Entered Decrement Numeric (Alphanumeric)*

The printer is capable of counting down fields on labels in a batch. This command is useful when printing labels need to be numbered in reverse sequence. The data in the field will decrement by the value after the minus sign with every label printed. The minus character may be replaced by a < character in order to make the field decrement alphabetically rather than numerically. This command has effect on only the label format record that it follows. It is intended for use in conjunction with the label batch quantity Q, or the System-Level Commands for quantity and reprint, <STX>E and <STX>G.

Syntax: **pii*

Where: * - Is - for numeric decrement, or < for alphanumeric decrement.

p - Is the fill character for the left-hand character of the field.

ii - Is the amount by which to decrement the field.

Sample: <STX>L<CR>
 132200000000000123AB<CR>
 <01<CR>
 Q0003<CR>
 E<CR>

The sample will generate a single field label format that prints the initial label with a value of 123AB and then decrements by one for the next two labels.

^ *Set Count by Amount*

An application using incrementing or decrementing fields (+, -, >, <) will occasionally require that more than one label be printed with the same values before the field data is updated. This situation is handled with the *^nn* command. All printers default to 1.

Syntax: *^nn*

Where:

- ^* - May be 0x55 or 0x40, see Control Codes.
- nn* - Is a two-digit value that specifies the number of labels to be generated before the incrementing or decrementing fields on the label.

Sample:

```
<STX>L<CR>
13220000000000012345<CR>
-01<CR>
^02<CR>
Q0006<CR>
E<CR>
```

The sample prints two labels with the same field value before decrementing the field. Six labels are printed.

☑Notes: This command can only be issued once per label format.

When alternate Control Codes are enabled, the '^' character must be replaced by '@' (hexadecimal 0x40). See Control Codes.

Special Label-Formatting Commands

There are two special commands used, the <STX>S (recall global data) and the <STX>T (print date and time) commands. Unlike the other Label-Formatting Commands, these special commands are entered directly into the data field of label format records. These should not be confused with System-Level Commands, although the same control character is used. When alternate Control Codes are enabled, the <STX> becomes '~' (hexadecimal 0x7E). See Control Codes.

Command	Description
<STX>S	Recall global data and place in field
<STX>T	Print time and date

Table 6-3 Special Label-Formatting Commands

STX S Recall Global Data And Place In Field

Once a global register has been defined, its contents can be used as data in other fields. When the printer receives the command <STX>S_n in a format record data field, it will place data from the specified global register into the data field, (see Label-Formatting Command G). Global registers contain the data in the first A through P format record data fields.

Syntax: <STX>S_n

Where: _n - Specifies the global register containing the data to copy into the data field, A – P.

Sample: <STX>L<CR>
 12110000000000METO<CR>
 G<CR>
 1A2210001000000<STX>SA<CR>
 E<CR>

In the sample above, the label-formatting command 'G' (line 3) places the string "METO" into the next available global register (in this case register A). The <STX>SA (line 4) is effectively replaced by the data from global register A.

STX T Print Time and Date

Time and date string data is selected by choosing the string character A through Z and a through h to retrieve data from the printer's internal clock. The date string characters are not printed, instead the printed label will show the corresponding print values, given in the table below.

New features (not available on all printers): The <STX>T may now be preceded by data to be printed/encoded and/or the string may now be terminated by a <STX> command followed by data terminated by a <CR>.

❑Note: When using substitution you must ensure the converted string produces valid characters for the selected bar code/font.

Syntax: <STX>Tstring<CR>

Where: *string* - String is any set of characters A through Z or a through h, see the table below.

String Characters	Print Values	String Markers	Print Values
A	Day of the week; (1 = Mon.)	VW	Hour in 24 hour format
BCD	Day of week name	XY	Hour in 12 hour format
EF	Month number	Za	Minutes
GH...O	Month name	gh	seconds
PQ	Day	bc	AM or PM
RSTU	Year	def	Julian date

Table 6-4 Time and Date String Characters

Sample Listing: (Assuming the current printer date is December 21, 1998.)

Sample 1: <STX>L<CR>
 121100001000100<STX>TBCD GHI PQ, TU<CR>
 E<CR>

The printed label will show: SUN DEC 21, 98

Sample 2: <STX>L<CR>
 191100100100010<STX>TEF/PQ<CR>
 E<CR>

The printed label will show: 12/21

Sample 3: <STX>L<CR>
 191100100100010ABC <STX>TEF/PQ<STX> DEF<CR>
 E<CR>

The printed label will show: ABC 12/21 DEF (This example illustrates a method of embedding the time string. The time string must be terminated by an <STX>).

7.0 Font-Loading Commands <ESC>

Font-Loading Command Functions

The commands used for font loading are usually generated by font creation software; however, the assigned font ID number command must be sent to the printer before the font file. Font-Loading Commands are listed in Table 7-1. All Font-Loading Commands begin with <ESC>. <ESC> represents the ASCII control character 27 (decimal).

The downloaded Font will be stored in the "default" Module, (refer to the <STX>X command). The commands in the table below are listed in their order of appearance, top to bottom, during font downloading.

Command	Description
*c###D	Assign Font ID Number
)s#Wnn...n	Font Descriptor
*c#E	Character Code
(s#W	Character Download Data

Table 7-1 Font-Loading Commands

***c###D** *Assign Font ID Number*

This command is the first command required for downloading a font to either RAM or Flash Memory modules. Esc represents the ASCII control character 27.

Syntax: <ESC>*c###D

Where: ### - Is the font ID numbers 100-999, (000-099) are reserved for resident fonts.

)s###W *Font Descriptor*

This command (typically first data in a font file), contains all of the information about the font contained in the file. Different font generation software will create different length header information, but the initial 64 bytes will remain consistent with the PCL-4 (HP LaserJet II) format.

Syntax: <ESC>)s###Wddd...d

Where: ### - Is the number of bytes of font descriptor data from 1 to 3 ASCII decimal digits.

ddd...d - Is the descriptor.

c###E *Character Code

This code is the ASCII decimal value corresponding to the next downloaded character.

Syntax: <ESC>*c###E

Where: ### - Is the ASCII value of the character, three digits maximum, 0 to 999.

(s#W *Character Download Data*

This command contains all of the information for one downloaded character.

Syntax: <ESC>(s###Wnn...n

Where: ### - Is the number of bytes of bit-mapped data, three digits maximum, 1 to 999.

 nn...n - Is the bit-mapped data.

8.0 Generating Label Formats

Format Record Commands

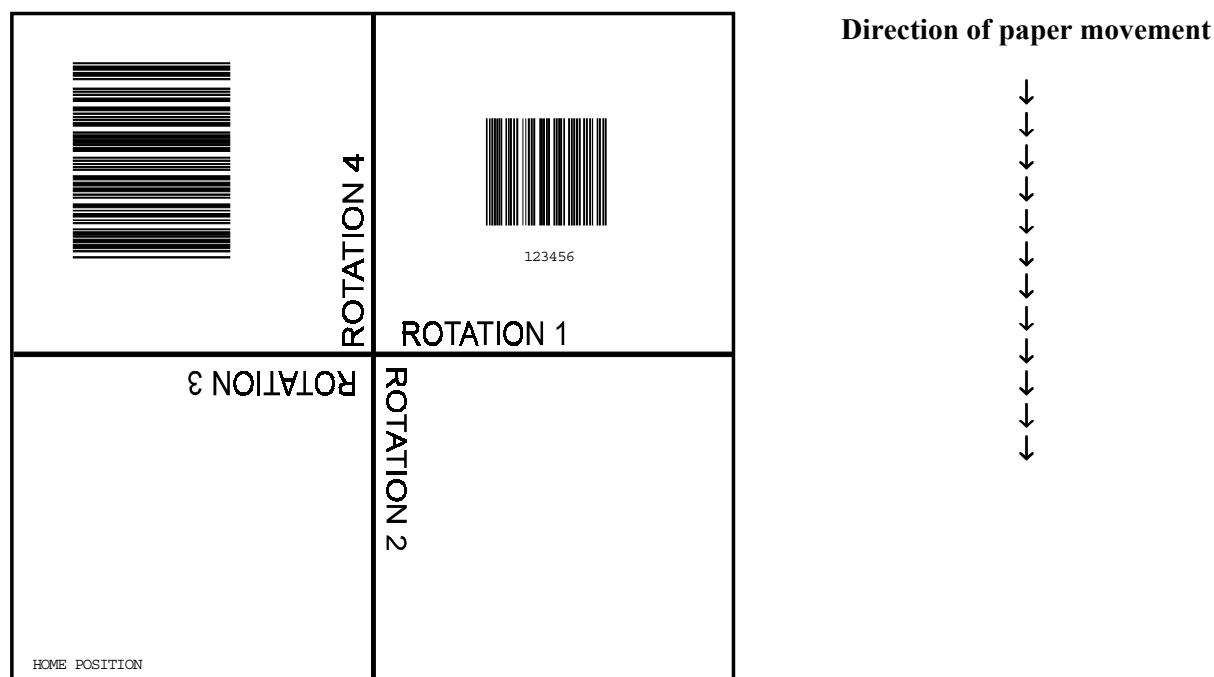
This section explains how to use the fields in a print format record. Table 8-1 is an example of a label format as seen by the printer. Figure 8-1 is a label generated from that format. The printer receives the data sequentially, left to right, top to bottom.

String Sent to Printer	Printer Interpretation
<STX>L<CR>	Begin label format
D11<CR>	Set dot size
121100000050005HOME POSITION<CR>	Format text
191100602000200ROTATION 1<CR>	Format text
291100602000200ROTATION 2<CR>	Format text
391100602000200ROTATION 3<CR>	Format text
491100602000200ROTATION 4<CR>	Format text
1A3104003000260123456<CR>	Format barcode with text
4a6210002500140123456<CR>	Format barcode
1X1100000000000B400400003003<CR>	Format box
1X1100002000000L400001<CR>	Format line
1X1100000000200L001400<CR>	Format line
121100004100010The Printer is here<CR>	Format text
Q0002<CR>	Number of labels
E<CR>	End formatting, begin print

Table 8-1 Sample Label Format

☑Note: This example assumes that the printer was in ‘inch’ mode (see <STX>m and <STX>n), when generating the label on the following page.

THE PRINTER IS HERE

**Figure 8-1 Rotations Example Label**

The first line in the sample format (Table 8-1) is the System-Level Command directing the printer to begin label formatting. (Other System-Level Commands may precede the <STX>L for printer setup.) Lines 2, 14, and 15 are Label-Formatting Commands. Line 15 is the exit and print command. The remaining lines (3-13) are print format records, explained in this chapter.

A record is a data string that contains the information to be printed on the labels. Records are the building blocks of label formats. Every record must end with a termination character (usually a carriage return, <CR>). Omitting termination characters will result in the concatenation of records. Omitting the carriage return that precedes the termination character E, which is necessary to complete the label formatting and begin printing, will cause the printer to continue interpreting all subsequent data as label print format records.

Generating Records

Every record is made of three parts: (1) A header that is 15 bytes in length, (2) the data to be printed, and (3) a termination character (e.g., <CR>) marking the end of the field. The header is used to select the appearance of the data when printed by choosing rotation, font type, size, and position options. Every header contains similar information, but different types of records may use this information in different ways. The six record types are:

1. Internal Bit-Mapped Font
2. Smooth Font (Simulated)
3. Scalable Font
4. Barcode
5. Images
6. Graphics

The Structure of a Record

The basic structure of the record is described below. For details regarding the various interpretations of the six types see Record Structure Types.

The third line of the label format example in Table 8-1 consists of the following:

121100000050005HOME POSITION<CR>

This string comprises a complete record, shown below, divided into its three basic component parts.

Header	Data String	Termination Character
121100000050005	HOME POSITION	<CR>

Table 8-2 Record Structure Components

The record conforms to the following fixed field format (spaces added for visual clarity). Identifying lower case letters have been placed below field values for reference in the following sections:

1	2	1	1	000	0005	0005	HOME POSITION	<CR>
a	b	c	d	eee	ffff	gggg	[hhhh iiiii] jj...j	Termination character

Location Within Record	Record Type					
	Internal Bit-Mapped Font	Smooth Font	Scalable Font	Barcode	Images	Graphics
a	Rotation	Rotation	Rotation	Rotation	Rotation	1
b	Font ID	9	9	Barcode	Y	X
c	Width Multiplier	Width Multiplier	Width Multiplier	Wide Bar	Width Multiplier	1
d	Height Multiplier	Height Multiplier	Height Multiplier	Narrow Bar	Height Multiplier	1
eee	000	Font Size/ID	ID	Barcode Height	000	000
ffff	Row Position	Row Position	Row Position	Row Position	Row Position	Row Position
gggg	Column Position	Column Position	Column Position	Column Position	Column Position	Column Position
hhhh	N/A	N/A	Font Height	N/A	N/A	N/A
iiii	N/A	N/A	Font Width	N/A	N/A	N/A
jj...j	Data String	Data String	Data String	Data String	Image name	Graphic Specifiers

Table 8-3 Record Type Structure

In Table 8-3, the record structure is shown for each of the record types. The left-most column shows the locations of all characters in the record, and corresponds to the example above the table. Each record structure interprets the characters of the record in its own way, though some of the interpretations of the characters are identical across all record types. For example, the characters ffff are interpreted as Row Position in all record types. While c is a Width Multiplier for Internal Bit-Mapped Font, Smooth Font, Scalable Font, and Image record types, it has other interpretations for Barcode and Graphics record types.

The Header Fields

Each of the fields in the record header is generally described below. Please reference the detailed descriptions under Record Structure Types for variations. The field name titles of the following paragraphs are preceded with a reference letter from Table 8-3. All characters sent to the printer within the header fields are ASCII, alphanumeric.

a: Rotation

The first field of a header is a single ASCII character that selects the degree of rotation for the data to be printed on a label. Valid rotation values are 1 (0°); 2 (90°); 3 (180°); and 4 (270°) clockwise. Figure 8-1 shows the direction and amount of rotation clockwise, relative to the label feed direction. The bottom left corner of the object is the pivot point.

b: Fonts, Barcodes, Graphics and Images

The second field (b) determines how the rest of the fields are interpreted, as shown in the table below. Values 0 through 9 select human-readable fonts. 0 through 8 will select standard fonts, value 9 selects the CG Triumvirate smooth scalable font (internal), scalable fonts, or a cartridge (module) font (external). When 9 is used to select a cartridge font or scalable font, the font size (font I.D. number) is chosen by entering a value in the height field eee.

Values A through z select barcodes. Values A through T (uppercase) will print barcodes with human-readable interpretations. Values a through z (lowercase), will print barcodes only.

Value W requires two additional characters to specify the Barcode/Font ID.

A font field value X selects a drawing object (line, box, circle or polygon), and field value Y is used to print an image stored in a module.

Font Field Value (b)	Interpretation
0-9	Font
A-T	Barcode with human readable text.
a-z	Barcode without human readable text.
Wxx	Barcode/Font Expansion
X	Line, box, polygon, circle
Y	Image

Table 8-4 Font Field Interpretations

c: Width Multiplier

Values 1-9 and A-O represent multiplication factors (base 25 numbers). For human-readable fonts, the width multiplier represents the number of times the selected font dot tables are multiplied and has no effect on the character height. For barcodes, this character specifies the wide bar width or ratio. Values 1 through 9 and A through O will give a wide bar width of from 0.0033" (0.085mm) to 0.792" (2.011mm) at a resolution dependent upon the printer model. See Appendix F for default values.

d: Height Multiplier

The height multiplier has the same range and function as the width multiplier, but vertical. When used in the context of barcodes, this field is the ratio denominator, or the small bar (module) width. Values 1 through 9 and A through O will give a narrow bar width of one dot (dot size = 1/printhead resolution) to 24 dots. The narrow bar width resolution and range are dependent upon the printhead resolution, see table below. A “dot multiplier” command can also be used to change the printed dot size (see Label-Formatting Command D and Appendix F for default values).

Printhead Resolution	
Dots Per Inch	Dots Per Millimeter
203	8.0
300	11.8

Table 8-5 Printhead Resolution**eee: Barcode Height (Font Size/Selection)**

This field has interpretations dependent upon the value of the font b field, as shown below.

b Font Field Value	eee Field Range	eee Field Interpretation
0-8	000	Not used –Internal bitmapped font
9	000-999, A04-A72, S00-S9z, U00-U9z, u00-u9z	Font height; Font selection
A-T	000-999	Barcode height (with human readable)
a-z	000-999	Barcode height
Wxx	000-999	Barcode height (with human readable)
X,Y	000	Not used

Table 8-6 Barcode Height Field Interpretations**ffff: Row Position**

The lower left corner of a label is considered the “home position”, see Figure 8-1. The row position field is a vertical coordinate that determines how far above the home position the data is to be printed. Field data is interpreted in hundredths of an inch or tenths of millimeters.

gggg: Column Position

This field is a horizontal coordinate that determines how far to the right of “home position” the data will be printed. The range of the gggg field is dependent upon the printer used. See Appendix G for a listing of the maximum values.

hhhh: Optional Scalable Font Height

The height of a scalable font can be specified in two ways, points or dots. To specify the height in points the first character of the field is a 'P' followed by the number of points, 004 to 999 points. To specify the size in dots, all four characters must be numeric. This field must be specified for scalable fonts. See note below.

iiii: Optional Scalable Font Width

The width of a scalable font can be specified in two ways, points or dots. To specify the width in points, the first character of the field is a 'P' followed by the number of points, 004 to 999 points. To specify the size in dots, all four characters must be numeric. This field must be specified for scalable fonts. See note below.

☑Note: To ensure that the data stream is portable to different model printers, specify the font size in points. If the font is specified in dots, it will output differently on printers with different DPI/MMPI resolutions. There are 72.307 points per 1 inch (2.847 mm).

jj...j: Data Field

The final field contains the data that will actually be printed on the label. A string of data can be up to 255 characters in length, (except when using the PDF 417 barcode, which may be up to 3000 characters long) ending with a carriage return. Characters placed in the data field will be printed as long as they fall within the physical range of the printhead. Consult Appendix K for a listing by printer.

Record Structure Types

Each of the six record types has its own field structure and is described in the following section. The record types allow quick reference to the field types and their valid data inputs for the field. There are similar, but unique, record structures for each: internal, bit-mapped fonts, internal smooth fonts, font modules, downloaded bit-mapped fonts, scalable fonts, barcodes, images, and graphics. The field location identifiers in the tables that follow are the same as those in Table 8-3.

(Continued next page)

1. Internal Bit-Mapped Fonts

This record type is used for internal bitmapped fonts (see Tables C-1 and C-2).

When a 0 through 8 is entered in field b, then the height field eee is not used. The bitmapped fonts include 8 different fonts (see Appendix C). The character mapping for these fonts is shown in Appendix A, or a subset thereof. The character mapping for these fonts is shown in Appendix A, or a subset thereof.

Field	Valid Inputs	Meaning
a	1, 2, 3 and 4	Rotation
b	0 to 8, see Appendix C.	Font
c	1 to 9 and A to O	Width Multiplier
d	1 to 9 and A to O	Height Multiplier
eee	000	N/A
ffff	0000 to 9999	Row
gggg	0000 to 9999 Dependent upon printer. See Appendix K.	Column
jj...j	Valid ASCII character string up to 255 characters followed by a termination character.	Data

Table 8-7 Internal Bit-mapped Font Record Structure

2. Smooth Font, Font Modules, and Downloaded Bit-Mapped Fonts

This record type is used for internal smooth fonts (CG Triumvirate – see Table C-4), an external font module (cartridge), or a bit-mapped font downloaded to a memory module (see 7.0 Font-Loading Commands <ESC>).

When a 9 is entered in field b, then the height field eee determines the font. The internal smooth font has up to 13 font sizes (see Appendix C). Values 100 through 999 select individual fonts stored on RAM, Flash, or ROM Font Modules. These include downloaded bit-mapped fonts, and cartridge fonts. See Table 8-6. Use eee values of 096 – 099 for Kanji fonts, if equipped. The character mapping for these fonts is shown in Appendix A, or a subset thereof.

Field	Valid Inputs	Meaning
a	1, 2, 3 and 4	Rotation
b	9	Fixed Value
c	1 to 9 and A to O	Width Multiplier
d	1 to 9 and A to O	Height Multiplier
eee	000 to 999 (000 to 099 Reserved), A04 to A72, x04 – x72*	Font/size
ffff	0000 to 9999	Row
gggg	0000 to 9999 Dependent upon printer. See Appendix K.	Column
jj...j	Valid ASCII character string up to 255 characters followed by a termination character.	Data

* Where x is an upper case letter, see Appendix H.

Table 8-8 Smooth Font Record Structure

3. Scalable Fonts

The Smooth Scalable Font Technology has been licensed from AGFA. Both IntelliFont (.CDI) and TrueType (.TTF) Scalable Font file formats are supported. The *eee* field identifies the scalable font, and data type – normal (binary) or hex ASCII. Uppercase S or U – binary, lowercase u – Hex ASCII. See Appendix H for additional information. Values S00 to S9z, and U00 to U9z (u00 to u9z), select a scalable font, either internal or downloaded. S00 and S01 are used for the standard internal (resident) fonts.

Field	Valid Inputs	Meaning
a	1, 2, 3 and 4	Rotation
b	9	Fixed Value
c	1 to 9 and A to O	Width Multiplier
d	1 to 9 and A to O	Height Multiplier
eee	S00 to Szz, U00-Uzz, u00-uzz	Font data type
ffff	0000 to 9999	Row
gggg	Dependent upon printer. See Appendix K.	Column
hhhh	P004-P999, 0016-4163*	Character height; points, dots
iiii	P004-P999, 0014-4163*	Character width; points, dots
jj...j	Valid ASCII character string up to 255 characters followed by a termination character.	Data

* Character size specifications are printhead resolution dependent as shown in the following table.

Table 8-9 Scalable Font Record Structure

Printhead Resolution (DPI)	Character size (dots)	
	Width	Height
203	16-2817	16-2817
289.56	16-4018	16-4018
300	14-4163	16-4163

Table 8-10 Scalable Character Size Ranges

Note: A scalable font cache must be allocated to print. Minimum cache size is 15. The double byte fonts require five units of additional cache.

4. Barcodes

Valid inputs for the barcode field *b* are letters. Uppercase letters will print a human-readable text below the barcode. Lowercase letters will print the barcode only. For example, entering a ‘p’ in the *b* field selects the Postnet barcode. Because the Postnet font does not provide human-readable data, the uppercase P is not valid. Other barcodes without a human-readable counterpart include u-MaxiCode, z-PDF417, etc. See Appendix F.

For module-based barcodes, field *d* is the narrow bar width in dots (barcode module size). For consistent results in all rotations for barcodes of this type, field *d* and field *c* must have the same value. For ratio-based barcodes field *c* is the wide bar width in dots (the numerator); field *d* is the narrow bar width in dots (the denominator). See Appendix G for specific barcode information and

variations in record format field usage. The eee height field represents the barcode (symbol) height. The valid range translates to bar heights ranging from 0.01in. (0.254mm) to 9.99in. (253.7mm). Barcodes that require additional parameters specified use the jj...j data field as the location for these parameters. See the specific barcode for details in Appendix G.

Field	Valid Inputs	Meaning
a	1, 2, 3 and 4	Rotation
b [bb]	A to Z and a to z (except P, u, v, z), or Wna where n is 1 to 9 and a is A to S and a to s. No n is an implied 1.	Barcode
c	1 to 9 and A to O	Wide Bar
d	1 to 9 and A to O	Narrow Bar
eee	001 to 999	Symbol height
ffff	0000 to 9999	Row
gggg	See Appendix K.	Column
jj...j	Valid ASCII character string up to 255 characters followed by a termination character.	Data

Table 8-11 Barcode Record Structure

Placing a 0 (zero) in both c and d will cause the printer to use the default barcode ratio or module size. Placing a 000 (zero) in the symbol height field causes the printer to use the default barcode height.

5. Images

An image record is used to print an image that is stored in a memory module. Images can be printed only in rotation 1 (see Input Image Data <STX>I).

Field	Valid Inputs	Meaning
a	1	Fixed Value
b	Y	Image
c	1 to 9 and A to O	Width Multiplier
d	1 to 9 and A to O	Height Multiplier
eee	000	Fixed Value
ffff	0000 to 9999	Row
gggg	See Appendix K.	Column
jj...j	ASCII string, up to 16 characters followed by a termination character.	Image name

Table 8-12 Image Fields

6. Graphics

Using graphics, the printer can produce lines, boxes, polygons, and circles. This function is selected by entering an X in field b. The values entered in the data field determine the sizes and shapes of the objects to be drawn. Forms can be created using shaded boxes, complex logos, or even a simple diagonal line without the need to download a graphics file to the printer. The following sub-sections describe how to generate each kind of graphic.

Lines and Boxes

Lines and boxes are drawn by values that determine column and row starting position, length, width, and wall thickness of the line or box (see Appendix K). All measurements are interpreted as inches/100 or millimeters/10, depending on the printer's mode, (see <STX>m). The data field jj...j is used to describe the line or box dimensions.

Segment	Valid Inputs	Meaning
a	1	Fixed value
b	X	Line box
c	1	Fixed value
d	1	Fixed value
eee	000	Fixed value
ffff	0000 to 9999	Row
gggg	0000-9999, see Appendix K.	Column
jj..j	Lhhhvvv - Line Drawing lhhhhvvvv - Line Drawing Bhhhvvtss - Box Drawing bhhhhvvvtss - Box Drawing	Line * Line ** Box *** Box ****

Table 8-13 Line and Box Parameters

- * LINES:** Lhhhvvv
Where: L = "L" and specifies line drawing,
 hhh = horizontal width of line,
 vvv = vertical height of line.
- ** LINES:** lhhhhvvvv
Where: l = "l" and specifies line drawing,
 hhhh = horizontal width of line,
 vvvv = vertical height of line.
- *** BOXES:** Bhhhvvtss
Where: B = "B" and specifies box drawing,
 hhh = horizontal width of box,
 vvv = vertical height of box,
 bbb = thickness of bottom and top,
 sss = thickness of sides.
- **** BOXES:** bhhhhvvvtss
Where: b = "b" specifies box drawing,
 hhhh = horizontal width of box,
 vvvv = vertical height of box,
 bbbb = thickness of bottom and top box edges,
 ssss = thickness of sides of box.

Note: Lines are sometimes better understood as filled in boxes, while boxes are hollow.

Polygons

Polygons are created by defining the positions of the corners. Specify a number of data points that represent the vertices of the object, which can range from a simple line (two points), or a triangle (three points), to any free-form outline. Polygons may be filled with a variety of different patterns. All row/column specifiers are interpreted as inches/100 or millimeters/10 depending on the printer mode (see <STX>m).

Record structure for a polygon (spaces added for visual clarity):

1 X 11 ppp rrrr cccc P ppp bbbb rrrr cccc rrrr cccc ... <CR>

Where:

1	Rotation (must be 1)	001	Fixed Value
X	Graphic field ID	0001	Fixed Value
1	Multiplier (must be 1)	rrrr	Row of point 2
1	Multiplier (must be 1)	cccc	Column of point 2
ppp	Fill pattern #	rrrr	Row of point 3
rrrr	Row of point 1	cccc	Column of point 3
cccc	Column of point 1	Additional points
P	Polygon ID (Fixed Value)	<CR>	Termination character

Table 8-14 Polygon Record Structure

☑Note: The points must be specified in the order to be drawn. The last point specified is automatically connected to the first, thereby closing the polygon. If only two points are specified, a single line will be drawn. See Label-Formatting Command A.

Circles

A circle is created by defining by its center point and radius. Circles may be filled with a variety of different patterns. Row, column, and radius are interpreted as inches/100 or millimeters/10 depending on printer mode.

Record structure for a circle (spaces added for visual clarity):

1 X 11 fff rrrr cccc C ppp bbbb rrrr <CR>

Where:

1	Rotation (must be 1)	cccc	Column of the center point
X	Graphic field	C	Circle ID (Fixed Value)
1	Multiplier (must be 1)	001	Fixed Value
1	Multiplier (must be 1)	0001	Fixed Value
fff	Fill pattern #	rrrr	Radius of the circle
rrrr	Row of the center point	<CR>	Termination character

Table 8-15 Circle Record Structure

Available Fill Patterns for Polygons and Circles:

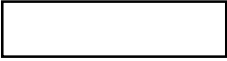











0		no pattern
1		solid black
2		6% black
3		12% black
4		25% black
5		38% black
6		50% black
7		little diamonds
8		little circles
9		right diagonal lines
10		left diagonal lines
11		small grid

Figure 8-2 Fill Patterns

Graphics Examples:

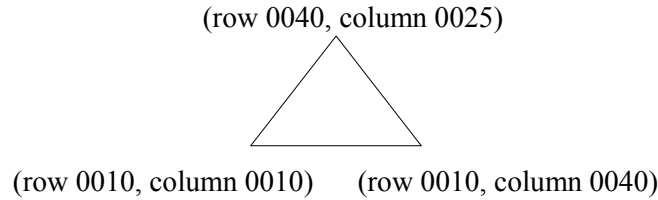
(Spaces are shown in the record for visual clarity only.)

1. Triangle

The record:

```
1 X 11 000 0010 0010 P 001 0001 0040 0025 0010 0040<CR>
```

Produces a triangle with no fill pattern:

**2. Rectangle with Fill**

The record:

```
1 X 11 004 0010 0010 P 001 0001 0050 0010 0050 0200 0010 0200 <CR>
```

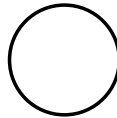
Produces a rectangle filled with pattern 4 (25% black):

**3. Circle**

The record:

```
1 X 11 000 0100 0100 C 001 0001 0025 <CR>
```

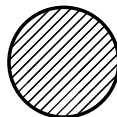
Produces a circle centered at row 0100, column 0100 with a radius of 0025 and no fill pattern:

**4. Circle with Fill**

The record:

```
1 X 11 009 0100 0100 C 001 0001 0025 <CR>
```

Produces a circle centered at row 0100, column 0100 with a radius of 0025 and filled with pattern 9 (right diagonal lines):



Appendix A

ASCII Control Chart

	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex
Ctrl @	NUL	0	00		32	20	@	64	40	`	96	60
Ctrl A	SOH	1	01	!	33	21	A	65	41	a	97	61
Ctrl B	STX	2	02	“	34	22	B	66	42	b	98	62
Ctrl C	EXT	3	03	#	35	23	C	67	43	c	99	63
Ctrl D	EOT	4	04	\$	36	24	D	68	44	d	100	64
Ctrl E	ENQ	5	05	%	37	25	E	69	45	e	101	65
Ctrl F	ACK	6	06	&	38	26	F	70	46	f	102	66
Ctrl G	BEL	7	07	Ö	39	27	G	71	47	g	103	67
Ctrl H	BS	8	08	(40	28	H	72	48	h	104	68
Ctrl I	HT	9	09)	41	29	I	73	49	i	105	69
Ctrl J	LF	10	0A	*	42	2A	J	74	4A	j	106	6A
Ctrl K	VT	11	0B	+	43	2B	K	75	4B	k	107	6B
Ctrl L	FF	12	0C	,	44	2C	L	76	4C	l	108	6C
Ctrl M	CR	13	0D	-	45	2D	M	77	4D	m	109	6D
Ctrl N	SO	14	0E	.	46	2E	N	78	4E	n	110	6E
Ctrl O	SI	15	0F	/	47	2F	O	79	4F	o	111	6F
Ctrl P	DLE	16	10	0	48	30	P	80	50	p	112	70
Ctrl Q	DC1	17	11	1	49	31	Q	81	51	q	113	71
Ctrl R	DC2	18	12	2	50	32	R	82	52	r	114	72
Ctrl S	DC3	19	13	3	51	33	S	83	53	s	115	73
Ctrl T	DC4	20	14	4	52	34	T	84	54	t	116	74
Ctrl U	NAK	21	15	5	53	35	U	85	55	u	117	75
Ctrl V	SYN	22	16	6	54	36	V	86	56	v	118	76
Ctrl W	ETB	23	17	7	55	37	W	87	57	w	119	77
Ctrl X	CAN	24	18	8	56	38	X	88	58	x	120	78
Ctrl Y	EM	25	19	9	57	39	Y	89	59	y	121	79
Ctrl Z	SUB	26	1A	:	58	3A	Z	90	5A	z	122	7A
Ctrl [ESC	27	1B	;	59	3B	[91	5B	{	123	7B
Ctrl \	FS	28	1C	<	60	3C	\	92	5C		124	7C
Ctrl]	GS	29	1D	=	61	3D]	93	5D	}	125	7D
Ctrl ^	RS	30	1E	>	62	3E	^	94	5E	~	126	7E
Ctrl _	US	31	1F	?	63	3F	_	95	5F		127	7F

(Continued next page)

ASCII Control Chart (concluded)

Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex
Ç	128	80	á	160	A0		192	C0	Ó	224	E0
ü	129	81	í	161	A1		193	C1	ß	225	E1
é	130	82	ó	162	A2		194	C2	Ô	226	E2
â	131	83	ú	163	A3		195	C3	Õ	227	E3
ä	132	84	ñ	164	A4		196	C4	ö	228	E4
à	133	85	Ñ	165	A5		197	C5	Ö	229	E5
å	134	86	ª	166	A6	ã	198	C6	µ	230	E6
ç	135	87	º	167	A7	Ã	199	C7	þ	231	E7
ê	136	88	¿	168	A8		200	C8	þ	232	E8
ë	137	89	®	169	A9		201	C9	Û	233	E9
è	138	8A		170	AA		202	CA	Ü	234	EA
ï	139	8B	1/2	171	AB		203	CB	Û	235	EB
î	140	8C	1/4	172	AC		204	CC	ý	236	EC
ì	141	8D	ì	173	AD		205	CD	Ý	237	ED
Ä	142	8E		174	AE		206	CE		238	EE
Å	143	8F	—	175	AF		207	CF		239	EF
É	144	90		176	B0	ð	208	D0		240	F0
æ	145	91		177	B1	Ð	209	D1	±	241	F1
Æ	146	92	²	178	B2	Ê	210	D2		242	F2
ô	147	93	³	179	B3	Ë	211	D3	3/4	243	F3
ö	148	94	´	180	B4	È	212	D4		244	F4
ò	149	95	Á	181	B5		213	D5		245	F5
û	150	96	Â	182	B6	Í	214	D6	÷	246	F6
ù	151	97	Ã	183	B7	Î	215	D7	,	247	F7
ÿ	152	98	©	184	B8	Ï	216	D8	°	248	F8
Ö	153	99	¹	185	B9		217	D9	ˆ	249	F9
Ü	154	9A		186	BA		218	DA	·	250	FA
ø	155	9B	»	187	BB		219	DB		251	FB
£	156	9C		188	BC		220	DC		252	FC
Ø	157	9D	¢	189	BD		221	DD		253	FD
x	158	9E	¥	190	BE	İ	222	DE		254	FE
f	159	9F		191	BF		223	DF	€	255	FF

Notes: For the hardware handshake XON/XOFF commands:

XON = Ctrl Q (DC1)

XOFF = Ctrl S (DC3)

The Euro € character has been added to the table above at 255 (FF) as a standard for resident bit-mapped fonts 0,1,2,3,4,5,6, and 9 (CG Triumvirate).

Appendix B

Sample Programs

'C' Language Program

The following sample 'C' program is included for reference. Figure B-1 shows the output generated by this program.

```
/* METO SERIES Sample C program */

#include <stdio.h>

main ()
{
    char *pcs = "590";
    char *desc = "10K OHM 1/4 WATT";

    fputs ("Meto Printer Test Program\n", stdout);

    fputs ("\x02L\n", stdaux);           /* STX L – Enter Label Formatting */
    fputs ("H07\n", stdaux);             /* Enter Heat Setting of 7 */
    fputs ("D11\n", stdaux);             /* Set Width and Height Dot Size */

    fprintf (stdaux, "191108010000025%s\n", desc); /* Select smooth Font */

    fprintf (stdaux, "1a6210000000050%sPCS\n", pcs); /* Select Barcode type 'a' */
    fputs ("E\n", stdaux);                /* End Label format mode and print */
}
```

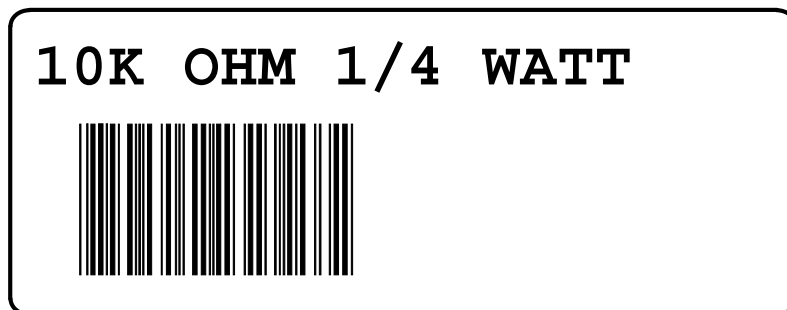


Figure B-1 Sample Label

ASCII text file

This ASCII text file will also generate the label shown in Figure B-1.

```
^BL
H07
D11
19110080100002510K OHM 1/4 WATT<CR>
1a6210000000050590PCS<CR>
E<CR>
```

Appendix C

Available Fonts – Sizes, Referencing, and Samples

All character bit-mapped fonts available on the printers are described in this section. Each font has a name (Font ID) associated with it for use in programming. Use the Font Number (in the left column of Table C-1) in field b of the Format Record header to cause the printer to use the corresponding font.

Fonts 0 through 8 use the slash zero (Ø) conventions for distinguishing between the zero and the alphabetic O. The slash can be removed with the label-formatting command Z. These fonts are non-proportional (monospaced). Therefore, all of the characters take up the same amount of space when printed. This is helpful when using variable data in a fixed area. The sizes of these fonts are shown on the following pages.

The CG Triumvirate font number 9 is a proportional font. Each character will take up a different amount of space when printed. For example, the letter W will be wider than the letter I.

Human-Readable Fonts (Internal)

Font Number	Valid ASCII Characters (decimal)	Use with Record Structure Type
0	32-127	Internal Bit-Mapped Fonts
1	32-168, 171, 172, 225	
2	32-168, 171, 172, 225	
3	32, 35-38, 40-58, 65-90, 128, 142-144, 146, 153, 154, 156, 157, 165, 168, 225	
4	32, 35-38, 40-58, 65-90, 128, 142-144, 146, 153, 154, 156, 157, 165, 168, 225	
5	32, 35-38, 40-58, 65-90, 128, 142-144, 146, 153, 154, 156, 157, 165, 168, 225	
6	32, 35-38, 40-58, 65-90, 128, 142-144, 146, 153, 154, 156, 157, 165, 168, 225	
7	32-126	
8	32, 48-57, 60, 62, 67, 69, 78, 83, 84, 88, 90	
9	32-126, 128-169, 171-173, 181-184, 189, 190, 198, 199, 208-216, 222, 224-237, 241, 243, 246-250	Smooth Font
9	Dependent upon selected symbol set, see Appendix H.	Scalable Font

Table C-1 Valid Human-Readable Font ASCII Characters

Font sizes are dependent upon the printer model used. Tables C-2 and C-3 list the font sizes by resolution with dimensions given in dots.

FONT	HEIGHT	WIDTH	SPACING	Point Size (203dpi)
Font 0	7	5	1	2.5
Font 1	13	7	2	4.6
Font 2	18	10	2	6.4
Font 3	27	14	2	9.6
Font 4	36	18	3	12.8
Font 5	52	18	3	18.4
Font 6	64	32	4	22.7
Font 7	32	15	5	11.3
Font 8	28	15	5	9.9

Table C-2 Font Sizes - 203 DPI Resolution

FONT	HEIGHT	WIDTH	SPACING	Point Size (300 dpi)
Font 0	10	7	1	2.4
Font 1	19	10	3	4.6
Font 2	27	15	3	6.5
Font 3	40	21	3	9.6
Font 4	53	27	4	12.7
Font 5	77	27	4	18.5
Font 6	95	47	6	22.8
Font 7	47	22	7	11.3
Font 8	41	22	7	9.8

Table C-3 Font Sizes - 300 DPI Resolution

Internal Smooth Font 9 (Smooth Font) Point Size Specifiers

Label format records with font code 9 (in Format Record header field b) can specify any of the font sizes in the left-most column of the table below. The corresponding specification, in either of the columns – labeled Ann or nnn, is used in the font size/selection (see height) field to select the desired font size. In the example below, a 300 DPI printer will produce the printed string “four point font” using a 4pt smooth font. The character mapping for this font is the selected scalable symbol set, see Appendix E.

E.g., 1911A0400100010four point font<CR>

Point Size	Font Size Specification Syntax	
	Ann	nnn
		All Others
4	A04 ¹	-
5	A05 ¹	000 ¹
6	A06	001
8	A08	002
10	A10	003
12	A12	004
14	A14	005
18	A18	006
24	A24	007
30	A30	008
36	A36	009
48	A48	010
72	A72	-

1. 300 DPI print resolution only.

Table C-4 Internal Bit-Mapped (Smooth Font) 9 Size Chart

Internal Bit-Mapped and Smooth Font Samples

The identifying number is used in the Format Record header field b to cause the printer to use the corresponding font.

0: Identifies a 96-character alphanumeric font, uppercase and lowercase.

```
Font 0:
!"#$%&'()*+,-./0123456789:;<=>?@
ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`
abcdefghijklmnopqrstuvwxyz{|}~
À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï
Ð Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ
```

1: Identifies a 145-character uppercase and lowercase alphanumeric font that includes desenders and ascenders.

```
Font 1:
!"#$%&'()*+,-./0123456789:;<=>?@
ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`
abcdefghijklmnopqrstuvwxyz{|}~
À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï
Ð Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ
```

2: Identifies a 138-character alphanumeric upper and lowercase font.

```
Font 2:
!"#$%&'()*+,-./0123456789:;<=>?@
ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`
abcdefghijklmnopqrstuvwxyz{|}~
À Á Â Ã Ä Å Æ Ç È É Ê Ë Ì Í Î Ï
Ð Ñ Ò Ó Ô Õ Ö × Ø Ù Ú Û Ü Ý Þ
```

3: Identifies a 62-character alphanumeric uppercase font.

```
FONT 3:
!"#$%&'()*+,-./0123456789:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
ÇÀÁÊËÜÝÞ
```

4: Identifies a 62-character alphanumeric uppercase font.

```
FONT 4:
!"#$%&'()*+,-./0123456789:
ABCDEFGHIJKLMNOPQRSTUVWXYZ
ÇÀÁÊËÜÝÞ
```

5: Identifies a 62-character alphanumeric upper case font.

FONT 5:
 # \$ % & () * + , - . / 0 1 2 3 4 5 6 7 8 9 :
 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
 Ç Ä Å Æ Ö Ü £ Ø Ñ Ò Ó

6: Identifies a 62-character alphanumeric uppercase font.

FONT 6:
 # \$ % & () * + , - . /
 0 1 2 3 4 5 6 7 8 9 :
 A B C D E F G H I J K L
 M N O P Q R S T U V W X Y Z
 Ç Ä Å Æ Ö Ü £ Ø Ñ Ò Ó

7: Identifies a font that prints OCR-A, size I.

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

8: Identifies a font that prints OCR-B, size III.

Font 8:
 0 1 2 3 4 5 6 7 8 9
 < > C E N S T X Z I

- 9: Identifies the Internal CG Triumvirate font. Point sizes are selected by the number in the Format Record header eee height field, see Table C-4.

4 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
5 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
6 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
8 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
10 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
12 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
14 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
18 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
24 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
30 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
36 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
48 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789
72 pt ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789

Appendix D

Reset Codes

The most common codes that could be transmitted by the printer to the host are described below.

Uppercase "R"

This code is sent every time the printer is turned 'On', signaling a hardware reset.

Uppercase "T"

This code signals a software reset. A software reset is made by sending the command sequence to the printer or by performing a reset using the front panel buttons.

Appendix E

Single Byte Symbol Sets

The following tables list the standard symbol sets. Not all of these symbol sets can be used with every font. Symbol sets containing the Euro currency character are WE, W1, WR (optional), and WT; see <STX>y.

Note: The following sets were produced using a Windows®-based PC-compatible with English (United States) keyboard properties layout. Results may vary if printing this document using a different input locale.

(DN) ISO 60: Danish / Norwegian Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	Æ	Ø	Å	^	_
60	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	æ	ø	å	—	⌘

(DT) DeskTop Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	⌘
80																
90																
A0		¶	§	†	‡	©	®	™	¢	£	—	—	...	fi	fl	
B0	“	”	μ	‰	•	●	°	○	þ	■	□	□	’	¬	¡	=
C0	—	±	×	÷	°	’	”	¼	½	¾	¹	²	³	/		
D0	()	«	»	,	„	’	ı	ı	Pt	ℓ	£	¥	¤	f	ß
E0	ª	º	æ	Æ	ð	Ð	ıı	IJ	t	t	œ	Œ	ø	Ø	þ	Þ
F0	‘	’	^	”	˘	˘	˘	˘	˘	˘	—	’	’	’	ı	

(E1) ISO 8859/1 Latin 1 Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	-
60	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	☒
80																
90																
A0		ı	ç	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
B0	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C0	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D0	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E0	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F0	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

(E2) ISO 8859/2 Latin 2 Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	-
60	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	☒
80																
90																
A0			˘	Ł	◻			§	¨	°	Š			·	Ž	
B0	°			ł	´			·	¸	¹	š			”	ž	
C0		Á	Â		Ä			Ç		É		Ê		Í	Î	
D0	Ð			Ó	Ô		Ö	×	Ø		Ú		Ü	Ý		ß
E0		á	â		ä			ç		é		ê		í	î	
F0				ó	ô		ö	÷		ù		û		ý		

(E5) ISO 8859/5 Latin 5 Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	-
60	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	☒
80																
90																
A0		ı	ç	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
B0	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C0	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D0		Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü			ß
E0	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F0		ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ı		ÿ

(FR) ISO 69: French Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	£	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	à	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	°	ç	§	^	—
60	μ	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	é	ù	è	™	☒

(GR) ISO 21: German Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	§	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	Ä	Ö	Ü	^	—
60	‘	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	ä	ö	ü	ß	☒

(IT) ISO 15: Italian Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	£	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	§	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	°	ç	é	^	—
60	ù	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	à	ò	è	ì	☒

(LG) Legal Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	=	=	¢	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[®]	©	—
60	°	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	§	¶	†	™	☒

(MC) Macintosh Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80	À	Á	Â	Ã	Ä	Å	Ö	Û	à	á	â	ä	å	ç	é	è
90	ê	ë	í	ì	ï	î	ñ	ó	ô	õ	ö	ø	ù	ú	û	ü
A0	†	°	¢	£	§	•	¶	ß	®	©	™	‘	’	≠	Æ	Ø
B0	∞	±	≤	≥	¥	μ	∂	Σ	Π	π	∫	ˆ	˜	Ω	æ	ø
C0	˘	ı	˚	√	ƒ	≈	Δ	«	»	…		À	Á	Ö	Œ	œ
D0	—	—	“	”	‘	’	÷	◊	ÿ	/	/	◻	<	>	fi	fl
E0	‡	˙	˘	˘	‰	Å	Ê	Ä	Ë	È	Í	Ï	Î	Ó	Ô	
F0		Ò	Ú	Û	Ü	Í	ˆ	˜	˘	˙	˚	˛	˜	˘	˙	˚

(PC) PC-8 Code Page 437 Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00		☺		♥	♦	♣	♠	●	◻	○			♂	♀	♫	⚙
10	▶	◀	↕	!!	¶	§	-	↓	↑	↓	→	←	↶	↷	▲	▼
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	△
80	Ç	ü	é	â	ä	à	â	ç	ê	ë	è	ï	î	ì	Å	Ä
90	É	æ	Æ	ô	ö	ò	û	ü	ÿ	Ö	Ü	ø	£	¥	Pt	f
A0	á	í	ó	ú	ñ	Ñ	ª	¿	¬	¬	½	¼	½	¼	«	»
B0	⌠	⌡	⌢	⌣	⌤	⌥	⌦	⌧	⌨	〈	〉	⌫	⌬	⌭	⌮	⌯
C0	⌰	⌱	⌲	⌳	⌴	⌵	⌶	⌷	⌸	⌹	⌺	⌻	⌼	⌽	⌾	⌿
D0	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿
E0	α	β	γ	π	Σ	σ	μ	τ	φ	θ	Ω	δ	∞	φ	ε	∩
F0	≡	±	≥	≤			÷	≈	○	•	•	√	∞	²	■	

(PD) PC-8 D/N, Code Page 437N Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00		☺		♥	♦	♣	♠	●	◻	○			♂	♀	♫	⚙
10	▶	◀	↕	!!	¶	§	—	↓	↑	↓	→	←	↶	↷	▲	▼
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	△
80	Ç	ü	é	â	ä	à	â	ç	ê	ë	è	ï	î	ì	Å	Ä
90	É	æ	Æ	ô	ö	ò	û	ü	ÿ	Ö	Ü	ø	£	Ø	Ł	ł
A0	á	í	ó	ú	ñ	Ñ	ø	Ö	¿	ª	À	ℓ	h	j	ª	◻
B0	⌠	⌡	⌢	⌣	⌤	⌥	⌦	⌧	⌨	〈	〉	⌫	⌬	⌭	⌮	⌯
C0	⌰	⌱	⌲	⌳	⌴	⌵	⌶	⌷	⌸	⌹	⌺	⌻	⌼	⌽	⌾	⌿
D0	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿
E0	α	β	γ	π	Σ	σ	μ	τ	φ	θ	Ω	δ	∞	φ	ε	∩
F0	≡	±	≥	≤			÷	≈	○	•	•	√	∞	²	■	

(PE) PC-852 Latin 2 Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00		☺		♥	♦	♣	♠	●	◻	○			♂	♀	♫	⚙
10	►	◄	↕	!!	¶	§	—	‡	↑	↓	→	←	↵	↔	▲	▼
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	△
80	Ç	ü	é	â	ä			ç	ł	ë			î		Ä	
90	É			ô	ö					Ö	Ü			Ł	×	
A0	á	í	ó	ú			Ž	ž							«	»
B0	⌠	⌡	⌢	⌣	⌤	⌥	⌦			⌧	⌨	〈	〉			⌫
C0	⌬	⌭	⌮	⌯	⌰	⌱			⌲	⌳	⌴	⌵	⌶	⌷	=	⌸
D0	ø	Ð	⌿	⌿			⌿	⌿		⌿	⌿	⌿	⌿	⌿		⌿
E0	Ó	ß	Ô				Š	š		Ù			Ý	Ý		´
F0	-	"	„	˘	˙	§	÷	˚	°	ˆ	˙				■	

(PI) PI Font Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20			”	”	“	”	‘	’	<	>	™	SM	®	©	®	
30	—	˘	˙	˙	↗	↘	↙	↘	Δ	>	∇	<	«	§	»	¶
40	::	Δ					f		h				ℒ	ℓ		
50	ø	ø	℞	Σ									ℓ	ℓ	<	>
60	⌈	⌊	⌈	⌊	+	+	—		U	I	⌈	⌊	□	◇		
70	⌋	⌋	⌋	⌋	⌋	⌋	⌋		⌋	⌋	⌋	⌋	■	◆		⌋

(PM) PC-850 Multilingual Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00		☺		♥	♦	♣	♠	●	◻	○			♂	♀	♫	⚙
10	►	◄	↕	!!	¶	§	—	‡	↑	↓	→	←	↵	↔	▲	▼
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	△
80	Ç	ü	é	â	ä	à	â	ç	ê	ë	è	ï	î	ì	Ä	Å
90	É	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	Ø	×	f
A0	á	í	ó	ú	ñ	Ñ	ª	º	¿	®	⌈	½	¼	¿	«	»
B0	⌠	⌡	⌢	⌣	⌤	⌥	⌦	⌧	⌨	〈	〉	⌫	⌬	⌭	⌮	⌯
C0	⌱	⌲	⌳	⌴	⌵	⌶	⌷	⌸	⌹	⌺	⌻	⌼	⌽	⌾	⌿	⌿
D0	ø	Ð	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿	⌿
E0	Ó	ß	Ô	Õ	ö	Ö	μ	þ	Þ	Ù	Ú	Û	ý	Ý	—	´
F0	-	±	=	¾	¶	§	÷	˚	°	ˆ	˙	˙	˙	˙	˙	˙

(PT) PC-8 TK, Code Page 437T Symbol Set*																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00		☺		♥	♦	♣	♠	●	■	○		♂	♀		♪	⊙
10	►	◄	↕	!!	¶	\$	—	±	↑	↓	→	←	⌞	↔	▲	▼
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	△
80	Ç	ü	é	â	ä	à	â	ç	ê	ë	è	ï	î	ì	Ä	Å
90	È	æ	Æ	ô	ö	ò	û	ù	ÿ	Ö	Ü	ø	£	¥	Pt	f
A0	á	í	ó	ú	ñ	Ñ	ª	º	¸	¸	½	¼	¿	¡	«	»
B0	⋮	⋮	⋮													
C0	L	L	T	T	—	+	F		L	F		T	F		+	
D0		T	T	L	L	F		+		F		T		+		
E0	ª	ß	L	π	Σ	σ	μ	τ	φ	Θ	Ω	δ	∞	φ	ε	∩
F0	≡	±	≥	≤			÷	≈	◊	▪	▪	√	°	?	■	

* Default Symbol Set

(R8) Roman-8 Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	-
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	⋮
80																
90																
A0		À	Á	È	É	Ê	Ë	Ì	Í	Î	°	²	³	¼	½	£
B0	-	Ý	ÿ	º	Ç	ç	Ñ	ñ	¿	¿	¤	£	¥	§	f	¢
C0	â	ê	ô	û	á	é	ó	ú	à	è	ò	ù	ä	ë	ö	ü
D0	Ä	î	Ø	Æ	â	í	ø	æ	Å	ì	Ö	Ü	É	Ï	ß	Ö
E0	Á	Ã	ä	Ð	ð	Í	Ì	Ó	Ö	Õ	ø	Š	š	Ú	Ý	ÿ
F0	Þ	þ	·	μ	¶	¼	—	¼	½	ª	º	«	■	»	±	

(SP) ISO 17: Spanish Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	£	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	§	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	¿	Ñ	¿	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	°	ñ	ç	~	⋮

(SW) ISO 11: Swedish Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	□	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	É	À	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	Ä	Ö	Å	Ü	—
60	é	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	ä	ö	å	ü	☒

(TS) PS Text Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	—
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80																
90																
A0		ı	¢	£	/	¥	ƒ	§	□	'	“	«	‹	›	fi	fl
B0		-	†	‡	·		¶	•	„	”	»	»	...	‰		¿
C0		`	´	^	~	—	˘	.	..		°	„		”	ˆ	
D0	—															
E0		Æ		ª					Ł	Ø	Œ	°				
F0		æ							ł	ø	œ	ß				

(UK) ISO 4: United Kingdom Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	£	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	—
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	-	☒

(US) ISO 6: ASCII Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	.	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	-
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	☒

(VI) Ventura International Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80																
90												↵	→	□	▪	□
A0	„	À	Á	Ê	Ë	Ē	Î	Ï	©	®	™	<	>	Ú	Û	
B0	‰	“	”	°	Ç	ç	Ñ	ñ	ı	İ	£	¥	₹	₪	ƒ	¢
C0	â	ê	ô	û	á	é	ó	ú	à	è	ò	ù	ä	ë	ö	ü
D0	Ä	Î	Ø	Æ	ä	í	ø	æ	Å	ì	Ö	Ü	É	Ï	ß	Ö
E0	Á	Ã	ä			í	ı	Ö	Ö	Ö	Š	š	Ÿ	Ÿ	Ÿ	Ÿ
F0	Œ	œ	ſ	†	‡	—	—			ª	º	«	•	»		...

(VU) Ventura US Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	
80																
90												↵	→	□	▪	□
A0	„								©	®	™					
B0	‰	“	”	°										§		¢
C0																
D0																
E0																
F0			ſ	†	‡	—	—						•			...

(W1) Windows 3.1 Latin 1 Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	⌘
80	□			ƒ	„	...	†	‡	^	‰	Š	<	Œ			
90	‘	’	“	”	•	—	—	~	™	š	>	œ				ÿ
A0		ı	€	£	¤	¥	¦	§	¨	©	ª	«	¬	-	®	¯
B0	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
C0	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D0	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E0	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F0	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

(WE) Windows 3.1 Latin 2 Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	⌘
80	□		„		„	...	†	‡		‰	Š	‹			Ž	
90		‘	’	“	”	•	—			™	š	›			ž	
A0		˘	˘	Ł	□		ı	§	ˆ	©		«	¬	-	®	
B0	°	±	²	³	´	μ	¶	·	¸			»		”		
C0		À	Á		Ä			Ç		É		Ê		Ë	Ì	Í
D0	Ð			Ó	Ô		Ö	×			Ù		Û	Ý		ß
E0		á	â		ä			ç		é		ê		ë	ì	í
F0				ó	ô		ö	÷			ú		û	ý		

(WO) Windows 3.0 Latin 1 Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	⌘
80																
90		˘	˘													
A0		ı	¢	£	□	¥	ı	§	ˆ	©	ª	«	¬	-	®	—
B0	°	±	²	³	´	μ	¶	·	¸	¹	º	»	¼	½	¾	¿
C0	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D0	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
E0	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F0	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

(WT) Windows 3.1 Latin 5 Symbol Set																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00																
10																
20		!	~	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	⌘
80	□		„		„	...	†	‡	ˆ	‰	Š	‹	Œ			
90		‘	’	“	”	•	—		˜	™	š	›	œ			Ÿ
A0		ı	¢	£	□	¥	ı	§	ˆ	©	ª	«	¬	-	®	—
B0	°	±	²	³	´	μ	¶	·	¸	¹	º	»	¼	½	¾	¿
C0	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
D0		Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü			ß
E0	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
F0		ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ı		ÿ

Appendix F

Barcode Summary Data

Barcode fonts have alpha names left column in the table below. Uppercase alpha names will print barcodes with human-readable interpretations. Lowercase alpha names will print barcodes only. Place the ID in field b of the Format Record header to cause the printer to encode the data field, using the associated barcode symbology, see Appendix G for details. See Table F-2 for default values.

Barcode ID	Type	Length	Checksum	Valid ASCII Characters, decimal value representation
A	Code 3 of 9	Varies	No	32, 36, 37, 42, 43, 45-57, 65-90
B	UPC-A	11	Yes	48-57 Numeric only. Option V used in the 6th & 7th position
C	UPC-E	6	Yes	48-57 Numeric only
D	Interleaved 2 of 5 (I 2 of 5)	Varies	No	48-57 Numeric only
E	Code 128	Varies	M-103	32-127
F	EAN-13	12	Yes	48-57 Numeric only. Option V used in the 7th & 8th position
G	EAN-8	7	Yes	48-57 Numeric only
H	HBIC	Varies	M-43	32, 36-39, 42, 43, 45-57, 65-90
I	Codabar	Varies	No	36, 43, 45-58, 65-68
J	Interleaved 2 of 5 with a modulo 10 checksum	Varies	M-10	48-57 Numeric only
K	Plessey	Up to 14	M-10	48-57 Numeric only. Option + is Last Character for Second M-11 Checksum
L	Interleaved 2 of 5 with a modulo 10 checksum & shipping bearer bars	13	M-10	48-57 Numeric only
M	2 digit UPC addendum	2	Yes	48-57 Numeric only
N	5 digit UPC addendum	5	Yes	48-57 Numeric only
O	Code 93	Varies	No	35-38, 42-58, 65-90, 97-122
p	Postnet	Varies	Yes	48-57 Numeric only
Q	UCC/EAN Code 128	19	Yes	48-57 Numeric only
R	UCC/EAN Code 128 K-Mart NON EDI barcode	18	Yes	48-57 Numeric only
S	UCC/EAN Code 128 Random Weight	34 +	Yes	48-57 Numeric only
T	Telepen	Varies	Yes	Alphanumeric
U	UPS MaxiCode	84	Yes	Alphanumeric
v	FIM	1	No	A, B, C, D
z	PDF-417	Varies	Yes	All
W1c	DataMatrix	Varies	Yes	All 8-bit values

Table F-1 Barcode Summary Data

Barcode Default Widths and Heights

Font	203 DPI Resolution		300 DPI Resolution	
	Height (inches)	Ratio/Module Size	Height (inches)	Ratio/Module Size
A	.40	9:4	.40	6:2
B	.80	4	.80	3
C	.80	4	.80	3
D	.40	9:4	.40	6:2
E	.40	4	.40	2
F	.80	4	.80	3
G	.80	4	.80	3
H	.40	9:4	.40	6:2
I	.40	9:4	.40	6:3
J	.40	9:4	.40	5:2
K	.40	9:4	.40	6:3
L	1.30	9:4	1.30	5:2
M	.90	4	.90	3
N	.80	4	.80	3
O	.40	4	.40	3
p	.08	N/A	.08	N/A
Q	1.40	4	1.40	2
R	1.40	4	1.40	2
S	1.40	3	1.40	2
T	.80	1	.80	1
U	1.00	N/A	1.00	N/A
v	.5	1	.5	1
z	N/A	N/A	N/A	N/A
Wlc	N/A	N/A	N/A	N/A

Table F-2 Barcode Default Data

☑Note: Some barcodes will be sensitive to the label command D (Set Width and Height Dot Size), see Label-Formatting Commands.

Appendix G

Barcode Details

All barcode symbols shown here were printed using the ratio/module values of 00 causing the printer to produce symbols using default bar widths. The barcode height fields are also 000 causing default heights to be used. See Appendix F for the default values.

A: Code 3 of 9

Valid Characters: 0-9, A-Z, -, *\$/% and the space character.

Variable Length.

Valid bar widths: The expected ratio of wide to narrow bars can range from 2:1 to 3:1.

The following example prints a code 3 of 9 barcode with a wide to narrow bar ratio of 3:1:

```
<STX>L
D11
1A00000001501000123456789<CR>
121100000000100Barcode A<CR>
E
```



B: UPC-A

Valid Characters: 0-9

Length: 12 digits. If the user provides 11 digits, the printer will compute the checksum. If the user provides the checksum, the printer will check that it matches the expected checksum. If it does not match, the printer will print out all zeros and the expected checksum. See Appendix P.

Valid bar widths: The fourth character of record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width).

The following example prints a UPC-A barcode:

```
<STX>L
D11
1B000000015010001234567890<CR>
121100000000100Barcode B<CR>
E
```



C: UPC-E

Valid Characters: 0-9

Length: Seven digits. If the user provides six digits, the printer will compute the checksum. If the user provides the checksum, the printer will check that it matches the expected checksum. If it does not match, the printer will print out all zeros and the expected checksum.

Valid bar widths: The fourth character of record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width).

The following example prints a UPC-E barcode:

```
<STX>L
D11
1C0000000150100012345<CR>
121100000000100Barcode
C<CR>
E
```

**D: Interleaved 2 of 5 (I 2 of 5)**

Valid Characters: 0-9

Variable Length.

Valid bar widths: The expected ratio of wide to narrow bars can range from 2:1 to 3:1.

The following example prints an Interleaved 2 of 5 barcode with a wide to narrow bar ratio of 3:1:

```
<STX>L
D11
1D000000015010001234567890<CR>
121100000000100Barcode D<CR>
E
```

**E: Code 128**

Valid Characters: The entire 128 ASCII character set.

Variable Length

Valid bar widths: The fourth character of record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times the narrow bar width, 3 times the narrow bar width, and 4 times the narrow bar width).

This printer supports the Code 128 subsets A, B, and C. The printer can be selected to start on any code subset and switch to another within the data stream. The default code subset is B, otherwise the first character (A, B, C) of the data field determines the subset. Subset switching is only performed in response to code switch command. These commands are placed in the data to be encoded at appropriate locations, see Table G-1.

(Continued next page)

E: Code 128 (Continued)

Code 128 Subset A: Includes all of the standard uppercase alphanumeric keyboard characters plus the control and special characters. To select Code 128 Subset A, place an ASCII A (DEC 65, HEX 41) before the data to be encoded.

Code 128 Subset B: Includes all of the standard uppercase alphanumeric keyboard characters plus the lowercase alphabetic and special characters. To select Code 128 Subset B, place an ASCII B (DEC 66, HEX 42) before the data to be encoded. If no start character is sent for the 128 font, Code 128 Subset B will be selected by default.

Code 128 Subset C: Includes the set of 100 digit pairs from 00 through 99 inclusive, as well as special characters. Code 128 Subset C is used for double density encoding of numeric data. To select Code 128 Subset C, place an ASCII C (DEC 67, HEX 43) before the data to be encoded. Subset C can only encode an even number of numeric characters. When the data to be encoded includes an odd number of numeric characters, the last character causes the printer to automatically generate a 'switch to subset B' and encode the last character appropriately in subset B.

Note: It is recommended to use a B as the first character to prevent an A or C from changing the subset.

Special Character Handling: Characters with an ASCII value greater than 95 are considered special characters. To access these values, a two-character reference table is built into the printer, see table below. As an example, to encode FNC2 into a Code 128 Subset A barcode, send the ASCII & (DEC 38, HEX 26) followed by an ASCII B (DEC 66, HEX 41). Code FNC2 will be encoded.

Example: ATEST&B123 Data Encoded: TEST<FNC2>123

ASCII	2 CHAR	CODE A	CODE B	CODE C
96	&A	FNC3	FNC3	-NA-
97	&B	FNC2	FNC2	-NA-
98	&C	SHIFT	SHIFT	-NA-
99	&D	CODEC	CODEC	-NA-
100	&E	CODEB	FNC4	CODEB
101	&F	FNC4	CODEA	CODEA
102	&G	FNC1	FNC1	FNC1

Table G-1 Special Character Handling

Control Codes: Control character encoding into Code 128 Subset A by sending these control codes:

`	=	NUL
a through z	=	1 - 26
{	=	ESC
	=	FS
}	=	GS
~	=	RS
ASCII 127	=	US

The following example prints a Code 128 barcode:

```
<STX>L
D11
1E000000015010001234567890<CR>
121100000000100Barcode E<CR>
E
```



Barcode E

F: EAN-13

Valid Characters: 0-9

Length: 13 digits. If the user provides 12 digits, the printer will compute the checksum. If the user provides the checksum, the printer will check that it matches the expected checksum. If it does not match, the printer will print all zeros and the expected checksum. See Appendix P.

Valid bar widths: The fourth character of record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width).

The following example prints an EAN-13 barcode:

```
<STX>L
D11
1F0000000150100012345678901<CR>
121100000000100Barcode F<CR>
E
```

**G: EAN-8**

Valid Characters: 0-9

Length: 8 digits. If the user provides 7 digits, the printer will compute the checksum. If the user provides the checksum, the printer will check that it matches the expected checksum. If it does not match, the printer will print all zeros and the expected checksum.

Valid bar widths: The fourth character of record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width).

The following example prints an EAN-8 barcode:

```
<STX>L
D11
1G00000001501000123456<CR>
121100000000100Barcode G<CR>
E
```

**H: Health Industry Bar Code (HBIC) (Code 39 barcode with a modulo 43 checksum).**

Valid Characters: 0-9, A-Z, -\$ /. %

Variable Length.

Valid bar widths: The expected ratio of wide to narrow bars can range from 2:1 to 3:1.

The host must supply leading "+"s

The following example prints a HBIC barcode with a wide to narrow bar ratio of 3:1:

```
<STX>L
D11
1H0000000150050+0123456789<CR>
121100000000100Barcode H<CR>
E
```



I: Codabar

Valid Characters: 0-9, A-D, -, ., \$, :, /, +, (comma is not valid).

Variable Length: Requires at least three characters.

Valid bar widths: The expected ratio of wide to narrow bars can range from 2:1 to 3:1.

Valid Codabar symbols require start and stop characters, (characters A-D). These characters should be placed in the data field along with other data to be included in the symbol.

The following example prints a Codabar barcode with a wide to narrow bar ratio of 3:1:

```
<STX>L
D11
1I63040001501000A1234567890D<CR>
121100000000100Barcode I<CR>
E
```

**J: Interleaved 2 of 5 with a modulo 10 checksum.**

Valid Characters: 0-9

Variable Length.

Valid bar widths: The expected ratio of wide to narrow bars can range from 2:1 to 3:1.

The following example prints an Interleaved 2 of 5 barcode with a modulo 10 checksum added and with a wide to narrow bar ratio of 3:1:

```
<STX>L
D11
1J000000015010001234567890<CR>
121100000000100Barcode J<CR>
E
```

**K: Plessey**

Valid Characters: 0-9

Length: 1 to 14 digits

Valid bar widths: The expected ratio of wide to narrow bars can range from 2:1 to 3:1.

If a + character is the last data character, an additional MSI checksum will be added to the barcode in place of the + character.

The following example prints a Plessey barcode with a wide to narrow bar ratio of 3:1:

```
<STX>L
D11
1K000000015010001234567890<CR>
121100000000100Barcode K<CR>
E
```



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

Valid Characters: 0-9

Variable Length: For the bearer bars to be printed, 14 characters are required.

Valid bar widths: The expected ratio of wide to narrow bars can range from 2:1 to 3:1.

The following example prints an Interleaved 2 of 5 barcode with a modulo 10 checksum with a wide to narrow bar ratio of 3:1 and bearer bars:

```
<STX>L
D11
1L00000001501000123456789012<CR>
121100000000100Barcode L<CR>
E
```

**M: 2-digit UPC addendum**

Valid Characters: 0-9

Length: 2 digits.

Valid bar widths: The fourth character of record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width). Human readable characters for this barcode symbology are printed above the symbol.

The following example prints a 2 digit UPC barcode addendum:

```
<STX>L
D11
1M000000015010042<CR>
121100000000100Barcode M<CR>
E
```

**N: 5-digit UPC addendum**

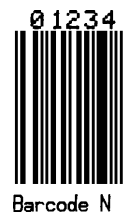
Valid Characters: 0-9

Length: 5 digits.

Valid bar widths: The width multiplier is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width). Human readable characters for this barcode symbology are printed above the symbol.

The following example prints a 5 digit UPC barcode addendum:

```
<STX>L
D11
1N000000015010001234<CR>
121100000000100Barcode N<CR>
E
```



O: Code 93

Valid Characters: 0-9, A-Z, -.\$/+% and the space character.

Variable Length.

Valid bar widths: The width multiplier is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width).

The following example prints a Code 93 barcode:

```
<STX>L
D11
1O000000015010042<CR>
121100000000100Barcode O<CR>
E
```



Barcode O

p: Postnet

Valid Characters: 0-9

Length: 5, 9 or 11 digits

Valid bar widths: The width and height multiplier values of 00 will produce a valid Postnet symbol.

Usage: The barcode height field is ignored since the symbol height is U.S.P.S specific. This barcode is to display the zip code on a letter or package for the US Postal Service.

The following example prints a Postnet barcode:

```
<STX>L
D11
1p000000015010032569<CR>
121100000000100Barcode p<CR>
E
```



Barcode P

Q: UCC/EAN Code 128

Valid Characters: 0-9

Length: 19 digits.

Valid bar widths: The fourth character of record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width). Human readable characters for this barcode symbology are printed above the symbol.

The printer spreads a weighted module 103 check sum.

The following example prints a UCC/EAN Code 128 barcode:

```
<STX>L
D11
1Q00000001501000123456789012345678<CR>
121100000000100Barcode Q<CR>
E
```



Barcode Q

R: UCC/EAN Code128 K-MART NON EDI barcode.

Valid Characters: 0-9

Length: 18 digits

Valid bar widths: The fourth character of record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width). Human readable characters for this barcode symbology are printed above the symbol.

This barcode is set up according to K-MART specifications.

The following example prints a KMART barcode.

```
<STX>L
D11
1R0000000150100012345678901234567<CR>
121100000000100Barcode R<CR>
E
```

**S: UCC/EAN Code 128 Random Weight**

Valid Characters: 0-9

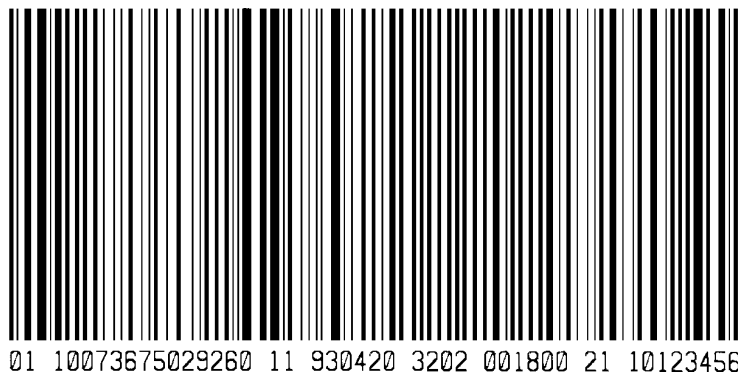
Length: At least 34 digits.

Valid bar widths: The fourth character of record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width).

This barcode is commonly used by the food and grocery industry.

The following example prints a UCC/EAN Code 128 Random Weight barcode:

```
<STX>L
D11
1S000000015005001100736750292601193042032020018002110123456<CR>
121100000000100Barcode S<CR>
E
```



T: Telepen

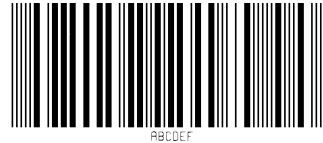
Valid Characters: All 128 ASCII characters.

Variable Length

Valid bar widths: The fourth character of the record is the width of the narrow bar in dots. All other bars are a ratio of the narrow bar (2 times, 3 times, and 4 times the narrow bar width).

The following example prints a Telepen barcode:

```
<STX>L
D11
1T0000000150100ABCDEF<CR>
12110000000100Barcode T<CR>
E
```

**u: UPS MaxiCode (Modes 2&3)**

The printer supports MaxiCode as defined in the AIM Technical Specification. The following examples illustrate various label format record message syntaxes for encoding data as MaxiCode.

In the following examples, special formatting is used to denote special ASCII characters as shown in the table:

Symbol	Hexadecimal Value
--------	-------------------

^R _S	1E
^G _S	1D
^E _{O_T}	04

Printer message syntax allows for ^E_{O_T} to be substituted with <CR> or the use of both ^E_{O_T}<CR>.

MaxiCode fixed field format:

Example:

```
1u0000001000100327895555840666THIS PACKAGE IS GOING TO METO.<CR>
```

Where:

```
32789  5 digit ASCII, Postal code
5555   4 digit ASCII, +4 Postal code
840    3 digit ASCII, country code
666    3 digit ASCII, class of service
THIS... 84 maximum ASCII characters, data string
```

In the UPS 3.0 protocol examples that follow, Primary Message control characters ^G_S will not be encoded in the MaxiCode symbol. All characters, the Secondary Message, with the exception of the leading ^G_S, in are encoded.

The UPS 3.0 zip + 4 with Message data format and message header:

Example:

```
1u0000001200120[]>RS01GS96841706672GS840GS001GS1Z12345675GSUPSNGS12345EGS089GSGS1/GS10.1GSYGS
GSGSUTRSEOT
```

(Continued next page)

u: UPS MaxiCode (Continued)

Where:

[> ^R _s 01 ^G _s 9	message header		
6			
841706672	Maximum 9 alphanumeric ASCII, Postal code		Primary Message
840	country code		
001	Class		
^G _s 1Z1...			Secondary Message
...T ^R _s ^E _O T			

The UPS 3.0 international postal “V6C3E2” with Message data format and message header:

Example:

1u0000001200120[>^R_s01^G_s96**V6C3E2**^G_s**068**^G_s**001**^G_s1Z12345675^G_sUPSN^G_s12345E^G_s089^G_s^G_s1/1^G_s10.1^G_sY^G_s^G_s^G_s^G_sUT^R_s^E_OT

Where:

[> ^R _s 01 ^G _s 9	message header		
6			
V6C3E2	Maximum 6 alphanumeric ASCII, International Zip code		Primary Message
068	country code		
001	Class		
^G _s 1Z1...			Secondary Message
...T ^R _s ^E _O T			

The UPS 3.0 international zip “V6C3E2” without Message data format and message header:

Example:

1u0000001200120**V6C3E2**^G_s**068**^G_s**001**^G_s1Z12345675^G_sUPSN^G_s12345E^G_s089^G_s^G_s1/1^G_s10.1^G_sY^G_s^G_s^G_s^G_sUT^R_s^E_OT

Where:

V6C3E2	maximum 6 alphanumeric ASCII, International Zip code		Primary message
068	country code		
001	Class		
^G _s 1Z1...			Secondary Message
...T ^R _s ^E _O T			

The UPS 3.0 zip + 4 “32707-3270” without Message data format and message header:

Example:

1u0000001200120**327073270**^G_s**068**^G_s**001**^G_s1Z12345675^G_sUPSN^G_s12345E^G_s089^G_s^G_s1/1^G_s10.1^G_sY^G_s^G_s^G_s^G_sUT^R_s^E_OT

Where:

32707	5 digit ASCII, Zip code		Primary Message
3270	4 digit ASCII, +4 Zip code (not required)		
068	country code		
001	class		
^G _s 1Z1...			Secondary Message
...T ^R _s ^E _O T			

v: FIM

Valid Characters: A, B, C, or D

Length: 1 character

Valid bar widths: The width and height multiplier works the same as for fonts on this barcode.

Usage: This barcode is to display the Facing Identification Mark (FIM) that is used on certain types of letter mail for the U S Postal Service:

FIM A: Courtesy reply mail with Postnet.

FIM B: Business reply, penalty or franked mail without Postnet.

FIM C: Business reply, penalty or franked mail with Postnet.

FIM D: OCR readable mail without Postnet (usually used on courtesy reply window envelopes).

The following example prints an FIM A barcode:

```
<STX>L
D11
1v0000000150100A<CR>
12110000000100Barcode v<CR>
E
```

**z: PDF-417**

Valid Characters: All ASCII characters. This is a 2 dimensional barcode capable of holding large amounts of data in a small area. It provides a high level of redundancy and error checking if requested.

Variable Length

The barcode's data stream consists of six different sections:

1. 1 character specifying a normal or truncated barcode, (T to truncate, F for normal).
2. 1-digit security level ranging from 0 to 8.
3. 2-digit aspect ratio. This is specified as a fraction with the first digit being the numerator and the second being the denominator. Use "00" for the default of 1:2. Valid range is from "00" to "99."
4. 2-digit number specifying the number of rows requested. Use "00" to let the printer find the best fit. Valid range is from "03" to "90". Values less than 3 are set to 3, and values greater than 90 are set to 90.
5. 2-digit number specifying the number of columns requested. Use "00" to let the printer find the best fit. Valid range is from "01" to "30". Values greater than 30 are set to 30.
6. A data stream to be encoded.

The following example prints a PDF-417 barcode, not truncated, security level of 1, default aspect ratio, and the best fit rows and columns:

```
<STX>L
D11
1z0000000150100F1000000PDF417<CR>
12110000000100Barcode z<CR>
E
```



☒ **Note:** Format Record header fields c and d should both be zero.

W1c: DataMatrix

Valid Characters: Any 8-bit byte data

Variable Length

DataMatrix is a two-dimensional matrix symbology, which is comprised of square modules arranged within a perimeter finder pattern. There are two basic types of DataMatrix symbols, ECC 000-140 and ECC 200.

ECC 000 - 140 symbols:

These square symbols can be any odd sizes from 9x9 to 49x49, which may be specified in fields iii, and jjj. If an ECC 000-140 symbol is specified with even numbers of rows or columns, the next largest odd value will be used. Input values greater than 49 or less than 9 will cause the symbol to be automatically sized for the input character stream. The record format is shown here, expanded with spaces.

a W b[b] c d eee ffff gggg hhh i jjj kkk ll..l

Where:

Field	Valid Inputs	Meaning
a	1,2,3, and 4	Rotation
W	W	Fixed value, extended barcode set
b[b]	c, 1c	Selects DataMatrix Barcode - the two differing values have no other significance.
c	1 to 9 and A to O	module size horizontal multiplier
d	1 to 9 and A to O	module size vertical multiplier
eee	000 to 999	No Effect; Must be numeric
ffff	0000 to 9999	label position, row
gggg	0000 to 9999	label position, column
hhh	000, 050, 080, 100, 140	A 3-digit convolutional error correction level. If any number other than one of these options is entered then the nearest lesser value from the valid entries is used. Example: selecting an ECC value of 099 will cause the actual ECC value of 080 to be used.
i	0 - 6	1 digit format identification: 0 - Automatically choose the encodation scheme based on the characters to be encoded. 1 - Numeric data. 2 - Upper-case alphabetic. 3 - Upper-case alphanumeric and punctuation characters (period, comma, hyphen, and slash). 4 - Upper-case alphanumeric. 5 - ASCII, the full 128 ASCII character set. 6 - any 8-bit byte. If a format identifier is selected which will not encode the input character stream then the barcode symbol will not be printed. It is recommended to use the auto-encodation format identification since it will select the best possible encodation scheme for the input stream.

Table G-2 DataMatrix ECC 000 – 140 Record Structure

(Continued next page)

W1c: DataMatrix (Continued)**ECC 200 symbols:**

There are 24 square symbol sizes available, with both row and column dimensions, which may be specified in fields iii, and jjj, measured in modules as indicated in the following list - 10, 12, 14, 16, 18, 20, 22, 24, 26, 32, 36, 40, 44, 48, 52, 64, 72, 80, 88, 96, 104, 120, 132, and 144. If an ECC 200 symbol is specified with odd numbers of rows or columns, the next largest even value will be used. Input values greater than 144 or less than 10 will cause the symbol to be automatically sized for the input character stream. The record format is shown here, expanded with spaces.

a W b[b] c d eee ffff gggg hhh i jjj kkk ll..l

Where:

Field	Valid Inputs	Meaning
a	1,2,3, and 4	Rotation
W	W	Fixed value, extended barcode set
b[b]	c, 1c	Selects DataMatrix Barcode - the two differing values have no other significance.
c	1 to 9 and A to O	module size horizontal multiplier
d	1 to 9 and A to O	module size vertical multiplier
eee	000 to 999	No Effect; Must be numeric
ffff	0000 to 9999	label position, row
gggg	0000 to 9999	label position, column
hhh	200	ECC 200 uses Reed-Solomon error correction.
i	0	Fixed value, not used
jjj	10, 12, 14, 16, 18, 20, 22, 24, 26, 32, 36, 40, 44, 48, 52, 64, 72, 80, 88, 96, 104, 120, 132, 144	A 3 digit even number (or 000) of rows requested. 000 causes rows to be automatically determined. The symbol will be sized to a square if the rows and columns do not match by taking the larger of the two values.
kkk	10, 12, 14, 16, 18, 20, 22, 24, 26, 32, 36, 40, 44, 48, 52, 64, 72, 80, 88, 96, 104, 120, 132, 144	A 3 digit even number (or 000) of columns requested. 000 causes columns to be automatically determined. The symbol will be sized to a square if the rows and columns do not match by taking the larger of the two values.
ll..l	8-bit data	Data to be encoded in the symbol

Table G-3 DataMatrix ECC 200 Record Structure

Appendix H

Font Mapping - Single and Double Byte Characters

Label format records with font code 9 in the b field of the Format Record header can specify any of the following bit-mapped or scalable fonts with the associated specification in the font size/selection (eee height) field, as shown in the tables on the following pages.

Example: 1911u4000100010A0215134<CR>

The example above will produce a printed string consisting of the two Kanji characters referenced by the two HEX ASCII pairs A0, 21, and 51, 34, on appropriately equipped printers.

Example: 1911U4001000100P012P012<0x38><0x77><0x00>

The above example will produce a printed string consisting of the one 12 point Kanji character referenced by the byte pair with hex values 38 and 77 on appropriately equipped printers.

❑Note: Double byte hex representation character strings terminate with two null bytes and a <CR>, i.e., 0x 00 00 0D. The Hex-ASCII representation is terminated with <CR>.

The alphanumeric portion (nn) of the scalable font specifiers, Snn, Unn, unn, numbering system is a base 62 numbering system, 0, 1, 2...8, 9, A, B, C...X, Y, Z, a, b, c...x, y, z. For scalable fonts the S designation signifies single byte characters and U designates double byte. The lower case U counterpart signifies that print data in the label format record is in a hex-ASCII format. Fonts that have been downloaded with designators of the form nn, where nn are alphanumeric, as seen in the font size specifier (eee height) column below, may be referenced in label format records by their upper or lower case specifiers as available. However, fonts created for double-byte access cannot be accessed using Snn as the font designator, and vice versa, single-byte fonts cannot be accessed using Unn or unn.

❑Note: Downloading scalable fonts requires specifying the font ID, a two character alphanumeric. The S, or U, u used in referencing the font within label format records is not used in the download specification. Attempting to utilize a scalable font with an inappropriate byte-size designation, (e.g. S on double byte or U, u on single byte) will have unpredictable results.

(Continued next page)

Font 9, Fonts Specifications (eee Height) and Associated Characteristics				
Font Name	Character Mapping	Font Size Specifier (eee Height)	Point Size	
Font 9 Bit-Mapped Resident Fonts Specifications				
CG Triumvirate ¹	Single Byte	000 - 010	5, 6, 8, 10, 12, 14, 18, 24, 30, 36, 48, respectively	standard
CG Triumvirate ¹	Single Byte	A04, A05, A06, A08, A10, A12, A14, A18, A24, A30, A36, A48, A72	4, 5, 6, 8, 10, 12, 14, 18, 24, 30, 36, 48, 72, respectively	
Font 9 Bit-Mapped Downloaded Fonts Specifications				
User downloaded bit-mapped typeface	Single Byte	100 - 999	user defined	
Font 9 Scalable Resident Fonts Specifications (optional except as noted)				
CG Triumvirate Bold Condensed Scalable	Single Byte	S00	scalable (standard)	
CG Triumvirate ¹ Scalable	Single Byte	S01	scalable (standard)	
Gothic B Kanji Scalable	Double Byte (Binary)	U40	scalable	
Gothic B Kanji Scalable	Double Byte (HEX ASCII)	u40	scalable	
GB Simplified Chinese Scalable	Double Byte (Binary)	UC0	scalable	
GB Simplified Chinese Scalable	Double Byte (HEX ASCII)	uC0	scalable	
Gothic E Kanji Scalable	Double Byte (Binary)	UK1	scalable	
Gothic E Kanji Scalable	Double Byte (HEX ASCII)	uK1	scalable	
CG Times Scalable	Single Byte	SA0	scalable	
CG Times Italic Scalable	Single Byte	SA1	scalable	
CG Times Bold Scalable	Single Byte	SA2	scalable	
CG Times Bold Italic Scalable	Single Byte	SA3	scalable	
Font 9 Scalable Downloaded Fonts Specifications (see note above)				
User downloaded Scalable typeface	Single Byte (Binary)	S50 - S5z... S90 - S9z	scalable	
User downloaded Scalable typeface	Double Byte (Binary)	U50...,U5z...,...U90...,U9z	scalable	
User downloaded Scalable typeface	Double Byte (HEX ASCII)	u50...,u5z...,...u90...,u9z	scalable	

1. Standard Internal Fonts

Table H-1 Font 9 Specifications

Appendix I

Symbol Sets and Character Maps

Symbol Set Selection

Scalable fonts are mapped through a symbol set sometimes referred to as a ‘code page’. This mapping allows the host application to select a variety of characters to match the application. For example in the code page (CP), character code 0xE4 causes character Φ to be printed. In CP E7, the character code 0xE4 causes δ to be printed. Each of the CPs allows the host application to “emulate” a character set for their application. Printers that support scalable fonts contain either a standard or an enhanced group of CPs as defined below. The CP (symbol set) is selected using a DPL Command, <STX>ySxx, where xx is the two letter CP Identifier.

Note: Not all fonts have an entire compliment of character codes for a given code page (symbol set).

Single Byte Code Pages						
Code Page Identifier		Code Page Family		Font Format		Description
DPL	HP ¹	Std ²	Enhanced ²	IntelliFont	TrueType	
AR			√	√		Arabic-8
CP			√	√		Cyrillic
DN		√	√	√	√	ISO 60 Danish / Norwegian
DT	7J	√	√	√	√	DeskTop
D1 ⁴	11L		√	√		ITC Zapf Dingbats/100
D2 ⁴	12L		√	√		ITC Zapf Dingbats/200
D3 ⁴	13L		√	√		ITC Zapf Dingbats/300
DS ⁴	10L		√	√		PS ITC Zapf Dingbats
E1	0N	√	√	√	√	ISO 8859/1 Latin 1
E2	2N	√	√	√	√	ISO 8859/2 Latin 2
E5	5N	√	√	√	√	ISO 8859/9 Latin 5
E6	6N		√	√	√	ISO 8859/10 Latin 6
E7			√	√	√	ISO 8859/7 Latin/Greek
EG			√		√	ISO 8859/7 Latin/Greek
EH			√	√		ISO 8859/8 Latin/Hebrew
ER			√	√		ISO 8859/5 Latin/Cyrillic
FR		√	√	√	√	ISO 69: French Symbol Set
G8			√	√		Greek-8
GK			√	√		PC-8 Greek
GR		√	√	√	√	ISO 21: German
H0			√	√		Hebrew-7
H8			√	√		Hebrew-8
IT		√	√	√	√	ISO 15: Italian
LG	1U	√	√	√	√	Legal
M8	8M		√	√	√	Math-8
MC	12J	√	√	√	√	Macintosh
MS	5M		√	√	√	PS Math

Table I-1 Single Byte Code Pages (continued next page)

Single Byte Code Pages						
Code page Identifier		Code Page Family		Font Format		Description
DPL	HP ¹	Std ²	Enhanced ²	IntelliFont	TrueType	
PB	6J		√	√	√	Microsoft Publishing
PC	10U	√	√	√	√	PC-8, Code Page 437
PD	11U	√	√	√	√	PC-8 D/N, Code Page 437N
PE	17U	√	√	√	√	PC-852 Latin 2
PG			√	√		PC-851 Latin/Greek
PH			√	√		PC-862 Latin/Hebrew
PI	15U	√	√	√	√	Pi Font
PM	12U	√	√	√	√	PC-850 Multilingual
PR			√	√		PC-864 Latin/Arabic
PT	9T	√	√	√	√	PC-8 TK, Code Page 437T
PU	9J		√	√	√	PC-1004
PV	26U		√	√	√	PC-775 Baltic
PX ⁴		√	√	√		PTXT3000
R8	8U	√	√	√	√	Roman-8
SP		√	√	√	√	ISO 17: Spanish
SW		√	√	√	√	ISO 11: Swedish
SY ⁴			√	√	√	Symbol
TS	10J	√	√	√	√	PS Text
TK			√	√		Turkish-8
UK		√	√	√	√	ISO 4: United Kingdom
US		√	√	√	√	ISO 6: ASCII
VI	13J	√	√	√	√	Ventura International
VU	14J	√	√	√	√	Ventura US
VM	6M		√	√		Ventura Math
W1 ³	19U	√	√	√	√	Windows 3.1 Latin 1
WA			√	√		Windows Latin/Arabic
WD ⁴			√	√	√	AgfaTidbits
WE ³	9E	√	√	√	√	Windows 3.1 Latin 2
WG			√	√		Windows Latin/Greek
WL	19L		√	√	√	Windows 3.1 Latin 5
WN			√	√		Windows
WO	9U	√	√	√	√	Windows 3.0 Latin 1
WR ³			√	√		Windows Latin/Cyrillic
WT ³	5T	√	√	√	√	Windows 3.1 Latin 5

1. HP - Hewlett Packard PCL-5 Comparison Guide, Edition 1, Internal Symbol Set Charts, Chart B, for comparison.
2. Standard and Enhanced Code Page Families are printer configuration respective.
3. As of this writing, the following symbol sets contain references to the Euro currency symbol (€), with the associated single byte decimal values:
 - W1 - Windows 3.1 Latin 1- 128
 - WE - Windows 3.1 Latin 2 - 128
 - WT - Windows 3.1 Latin 5 - 128
 - WR - Windows Latin/Cyrillic – 136 (optional)
4. Not supported by ILPC CG Times – characters appearing in Code Page identifiers not supported are unpredictable.

Table I-1 Single Byte Code Pages

Double-Byte Symbols, Chinese, Kanji

Character Map Selection

Double byte scalable fonts are mapped through a ‘character map’. This mapping allows the host application to select a variety of characters to match the application. Each of the code pages allows the host application to emulate a character set for their application.

The double-byte symbol set is selected using the <STX>yUxx command.

Double Byte Character Map			
Character Map Identifier	Code Page Family ¹	Font Format	Description
	Enhanced	TrueType	
B5	√	√	BIG 5 (Taiwan) encoded
EU	√	√	EUC (Extended UNIX Code)
GB	√	√	Government Bureau Industry Standard; Chinese (PRC) Default.
JS	√	√	JIS (Japanese Industry Standard) Default.
SJ	√	√	Shift JIS
UC	√	√	Unicode

1. Enhanced Code Page Families are printer configuration respective.

Table I-2 Double Byte Character Map

The double-byte symbol set is selected using <STX>yUxx command. The single-byte symbol set is selected using the same command, <STX>ySxx. Each affects an independent database selection and has no impact on the other.

Appendix J

General Purpose I/O Signals

Pin Number	Input	Output	Description
1		✓	+ 5 Volts DC
2		✓	Printer Fault
3		✓	End of Print
4		✓	Ribbon Out
5		✓	Label Out
6		✓	Ribbon Low
7	✓		Input
8	✓		Input
9		✓	Signal Ground

Table J-1 Optional GPIO Signal Descriptions

Appendix K

Print Resolutions, Module Identifiers, Maximum Field, Column & Character Values

Module ID (Memory Bank)	Description (type)
A	plug-in
B	plug-in
C	Default
D	Internal DRAM (default 512KB)

Table K-1 Printer Module Identifiers

Printer	Fields	Total Characters for all Fields
mi-4206	500	32768
mi-4308	500	32768
mi-6208	500	32768
mi-6308	500	32768

Table K-2 Maximum Label Format Fields & Characters

Printer	Maximum “gggg” Value	
	Inch	Metric
mi-4206	0410	1041
mi-4308	0410	1041
mi-6208	0660	1680
mi-6308	0640	1625

Table K-3 Column Field (“gggg”) Values, Format Record header

Printer	Print Resolution		Maximum Print Width	
	DPI	DPM	Inches	Millimeters
mi-4206	203	8.0	4.094	104.0
mi-4308	300	11.8	4.161	105.7
mi-6208	203	8.0	6.610	168.0
mi-6308	300	11.8	6.401	162.6

Table K-4 Printer Widths and Resolutions

Appendix L

Printer Speed Ranges

Speed Command Value	Speed:	
	Inches per Second	Millimeters per Second
C	2.0	51
D	2.5	63
E	3.0	76
F	3.5	89
G	4.0	102
H	4.5	114
I	5.0	127
J	5.5	140
K	6.0	152
L	6.5	165
M	7.0	178
N	7.5	191
O	8.0	203
P	8.5	216
Q	9.0	227
R	9.5	241
S	10.0	254

Table L-1 Speed Command Values

Printer	Print		Slew		Backup	
	Range	Default	Range	Default	Range	Default
mi-4206	C – K	K	C – K	K	C – I	E
mi-4308	C – O	K	C – O	K	C – I	E
mi-6208	C – O	K	C – O	K	C – I	E
mi-6308	C – O	K	C – O	K	C – I	E

Table L-2 Speed Ranges and Defaults by Model

Appendix M

Commands by Function

Commands by Function	
Function	Command
Backup speed	pa
Batch quantity request	<SOH>E
Cancel	<SOH>C
Character bit-mapped data	<ESC>(snnnWdata
Character code	<ESC>*cnnnE
Character dump mode	<STX>P
Column offset amount	Cnnnn
Configuration label and dot pattern print	<STX>Z
Continuous paper length	<STX>cnnnn
Count by	^nn
Cut	<STX>o
Cut by	:nnnn
Cut by	cnn
Decrement alphanumerically	<fii
Decrement numerically	- fii
DIP switch, host controlled settings	<STX>Vn
Dot size height and width	Dwh
Edge sensor enable	<STX>e
Feed rate	<STX>Sa
Feedback characters enable	<STX>a
Field data line terminator	Tnn
File delete from module	<STX>xmfname
Firmware version request	<STX>v
Font descriptor	<ESC>)snnnW
Font ID number	<ESC>*cnnnD
Form feed	<STX>F
Set Present Distance, printhead burnline relative	<STX>Kfnnnn
Format attribute	An
Graphics image download	<STX>lafbname ^c _r
Heat setting	Hnn
Inches	<STX>n
Increment alphanumerically	>fii
Increment numerically	+fii
Label format field replacement	<STX>Unnstring
Label formatting start	<STX>L
Label length maximum	<STX>Mnnnn

Table M-1 (continued next page)

Commands by Function	
Function	Command
Memory query	<STX>KQ
Memory query (new format)	<STX>Kq
Metric	<STX>m
Metric	m
Mirror	M
Module clear	<STX>qm
Module, compress	<STX>zm
Module, directory request	<STX>Wa
Module, set default	<STX>Xm
Module, FLASH memory Test	<STX>w
Module, RAM memory Test	<STX>t
Modules, clear all	<STX>Q
Pause for each label	<STX>J
Pause toggle	<SOH>B
Pause, controlled	<STX>p
Place data in global register	G
Print last label format	<STX>G
Print speed	Pa
Print time and date	<STX>Tstring
Printhead dot pattern test label	<STX>T
Quantity labels printed	<STX>Ennnn
Quantity of labels	Qnnnn
Recall global data and place in field	<STX>Sa
Recall stored label	rname
Reflective sensor select	<STX>r
Replacement field tag	U
Reset	<SOH>#
Resettable counters reset	<STX>Kr
Ribbon saver	<STX>Rx
Row offset amount	Rnnnn
RS-232 port test	<STX>k
Scalable font download	<STX>imtaabbb...b ^c ,xxxxxxxxfff...f
Sensor values request	<STX>Y
Slew rate	Sa
Status ASCII string request	<SOH>A
Status byte request	<SOH>F
Store label in module & terminate formatting	smname
Symbol set select	<STX>ySaa
Symbol set select	ySaa
Terminate formatting - print label format	E
Terminate label formatting, do not print label	X
Time and date request	<STX>B
Time and date set	<STX>AwMMddyearhhmmjjj
Update system database with current database	<SOH>U
Zero (Ø) conversion to "0"	z

Table M-1

Appendix N

mi-4 and *mi-6* Constraint Cross-Reference

Immediate Command	Description
D	The internal firmware architecture no longer requires the execution of an “SOH” shutdown for binary downloading.
All others	The firmware architecture no longer will “go busy” due to SOH commands filling the receive buffer while statusing the printer during batch printing.

Table N-1 Immediate Command Constraints

System-Level Command	Description
A & B	The printer calculates the Julian date. A non-zero date will not be stored and returned.
C	Copy Module – Not Supported.
D	Memory Dump – Not Supported.
H	Set Cutter Signal Time – Not Supported.
N	Enter Internal Batch Mode – Not Supported.
O	Set Start of Print Position – Not Supported – Use Operator Panel or Label command Row Offset.
S	Test Module Memory – Not Supported.
V	Software Settings – Previously different for each printer – Verify format.
Y	Output Sensor Values – Verify response format.
b	Set Cutter Signal Time – Not Supported.
d	Double Buffer Mode – Not Supported.
f	Set Form Stop Position – Not Supported – Use Operator Panel or <STX>Kf command.
g	Internal Batch Software Mode – Not Supported.
s	Single Buffer Mode – Not Supported.
t	Test RAM Memory Module – Verify response format.
v	Firmware Version – Verify response format ¹ .
w	Test Flash Memory – Verify response format ¹ .

Table N-2 System-Level Command Constraints

Extended System Command	Description
KD	Configuration – Not Supported.
Kf	Set Present Distance – Replaces System Command Set Form Stop Position.
KI & KO	GPIO Configuration – Verify Format.
KM, KR, KS, & KW	Set Memory Configuration - Not Supported. Default Configuration will handle most applications. Use Operator Panel.
KQ	Memory Configuration – Verify response format ¹ .

Table N-3 Extended-System Command Constraints

Label Command	Description
W	Wait Mode – Not Supported.
Z	Zip Mode – Not Supported.

Table N-4 Label Command Constraints

Internal Font	Description
9	(Ann) Smooth Internal CG Triumvirate Bitmap Font – 9. Referencing the Font 9 to Smooth CG Triumvirate Fonts are automatically converted to the CG Triumvirate Scalable Font. The printed scalable font is slightly smaller than previous printers, measuring less than 1/16 inch over three inches of print. The Euro Currency symbol is referenced as determined by the selected scalable font symbol set. See Appendix E.
9	(S00) CG Triumvirate Bold Condensed Scalable Font.
9	(S01) CG Triumvirate Scalable Font – Standard.

Table N-5 Internal Font Constraints

Modules	Description
Default	When specifying a module that is not available, the printer will automatically select the default module.

Table N-6 Module Constraints

¹ Printer Response message format may deviate from older standard DPL Printers; we suggest the program developer examine the printer responses in detail.

Appendix O

Image Loading

The printer will accept four types of image files: .BMP, .IMG, .PCX and a special 7-bit ASCII file (as defined in this section). Using the 7-bit ASCII format will require at least twice as much data transmission time as the other formats, (see <STX>I). The ASCII image file format is made up of a set of records with identical formats, each representing a dot row of the image; a terminator follows the last of these records.

Dot-row record
.
.
.
.
Dot-row record
Terminator

Each dot-row record has the following format:

Syntax: 80nndd...d<CR>

Where: nn - Is the number of character pairs in dd...d, represented in ASCII hex.
 dd...d - Is dot data, character pairs, ASCII hex, 00-FF.

Duplicate records may be encoded using a repeat data record, following the data record that needs duplicating. The repeat data record format is:

Syntax: 0000FFnn<CR>

Where: nn - Is the number of duplicates, ASCII hex, 00-FF.

The terminator, last record, at the image download is: FFFF<CR>

[illegible]

Figure O-1 Sample 7-bit ASCII File Image



Figure O-2 Sample Label

Appendix P

UPC-A and EAN-13: Variable Price/Weight Barcodes

The EAN/UPC standard allows for an additional checksum to be generated in the middle of the barcode based on the data. This is used when the price or weight of an item is embedded into the barcode data (commonly used in the food industry).

For the printer to generate this checksum, a 'V' must be placed in the data stream in the position the checksum is requested. If the 'V' is placed in the 6th position for UPC-A or the 7th position for EAN-13, a checksum will be generated using the next five digits in the data stream. If the 'V' is placed in the 7th position for UPC-A or the 8th position for EAN-13, a checksum will be generated using the next four digits in the data stream. The checksum is generated per the EAN/UPC barcode standard.

Examples:

1B110000200020012345V01199

Prints the UPC-A barcode with the variable price checksum in the sixth position.

1B1100002000200123456V0150

Prints the UPC-A barcode with the variable price checksum in the seventh position.

1F1100002000200123456V01199

Prints the EAN-13 barcode with the variable price checksum in the seventh position.

1F11000020002001234567V0150

Prints the EAN-13 barcode with the variable price checksum in the eighth position.

Appendix Q

International Language Print Capability (ILPC) Programming Examples

ILPC, offered as a field upgrade or a factory installable option, allows the printing of non-English character sets, available with Western European language support (CG TIMES), KANJI language support (GOTHIC B / GOTHIC E), and Chinese language support (SIMPLIFIED GB). All of the features are embedded in the printer resident firmware and accessible through DPL thus eliminating excessive download time of bitmapped characters. Using scalable technology licensed from AGFA, this firmware allows users to print smooth characters in sizes from 4pt (1.4mm) to 999pt (350mm) in over 40 languages. Consult Appendix I for code page selections. Specific details regarding which characters are supported in each option can be obtained through Technical Support.

ILPC - CG® TIMES Option

The CG Times Option is a single byte scalable font consisting of four typefaces in 38 Western European languages. This option contains over 900 unique characters in each of the four typefaces from the CG Times typeface family, Normal, Italic, Bold, and Bold Italic. Single byte scalable fonts are selected using a print format record (see Chapter 8 and Appendix H for details).

Scalable CG TIMES Font Code ('eee' field):

SA0 - CG® TIMES
SA1 - CG® TIMES *ITALIC*
SA2 - CG® TIMES **BOLD**
SA3 - CG® TIMES **BOLD ITALIC**

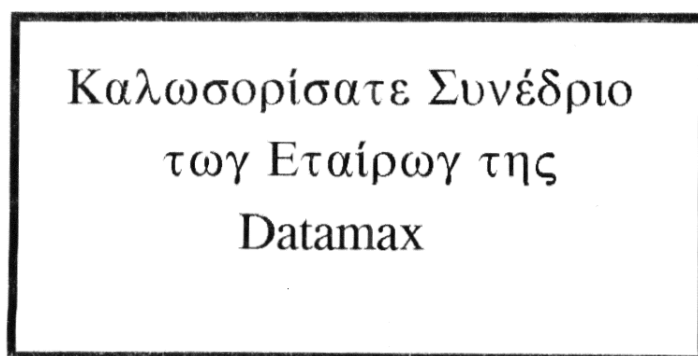
Sample Greek DPL file and resulting label:

```
<02>L<CR>
D11<CR>
ySWG<CR>
1911SA003600020P020P020(WG) Greek Characters from<CR>
1911SA003000085P020P020the internal Symbol Set,<CR>
1911SA002400085P020P020font code SA0<CR>
1911SA001500050P020P020<ca><e1><eb><f9><f3><ef><f1><df><f3><e1><f4><e5><20><d3><f5><ed><dd><e4><f1>
<e9><ef><20><CR>
1911SA001100100P020P020<f4><f9><e3><20><c5><f4><e1><df><f1><f9><e3><20><f4><e7><f2><CR>
1X1100000100020B365190005005<CR>
Q0002<CR>
E<CR>
```

Note: The notation “<xx>” in this DPL file should be interpreted by the reader as representing the hexadecimal value of the character sent to the printer.

(Continued next page)

(WG) Greek Characters from
the internal Symbol Set,
font code SA0



ILPC-KANJI Option

The Kanji Option is a double byte scalable font supporting Kanji Gothic B and Gothic E. In the double byte format the printer recalls one character printed from every two 8 bit bytes sent from the host. Double byte scalable fonts are selected using a print format record (see Chapter 8 and Appendix H for details).

Scalable Double Byte Font Map - KANJI					
eee (Font Code)	Scalable Font Type	Font Name	Binary Addressing	Hex ASCII Addressing	Code Pages
U40	Scalable Resident	HG-Gothic-B Kanji Scalable	√		EUC, JIS, SJIS, UC
u40	Scalable Resident	HG-Gothic-B Kanji Scalable		√	EUC, JIS, SJIS, UC
UK1	Scalable Resident	HG-Gothic-E Kanji Scalable	√		EUC, JIS, SJIS
uK1	Scalable Resident	HG-Gothic-E Kanji Scalable		√	EUC, JIS, SJIS
u50 - u5z u90 - u9z	Scalable Non-Resident (Download)	User defined		√	
U50 - U5z... U90 - U9z	Scalable Non-Resident (Download)	User defined	√		

Note: Not all fonts contain an entire compliment of character codes for a given character map.

Sample Kanji Gothic B DPL file (binary addressing) and the resulting label:

```

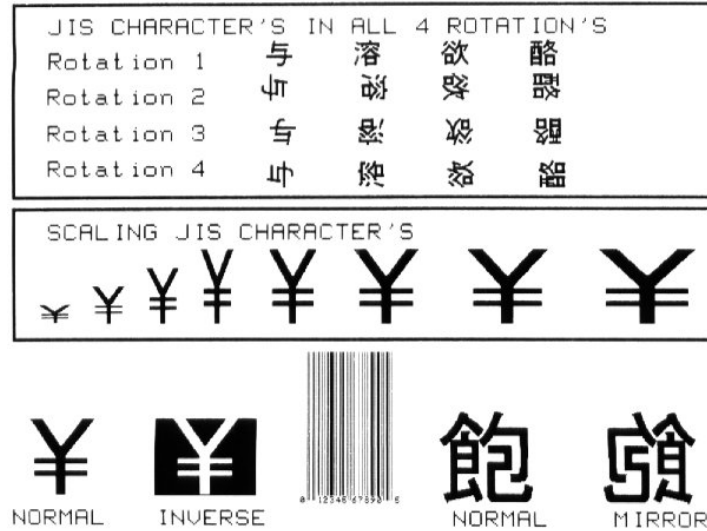
<02>L<CR>
D11<CR>
ySPM<CR>
1911S0003100010P020P015Scalable Kanji Gothic B Available<CR>
1B110000020017001234567890<CR>
yUJS<CR>
1X1100001900010b0392011000020002<CR>
112200002800030JIS CHARACTER'S IN ALL 4 ROTATION'S<CR>
112200002600030Rotation 1<CR>
1911U4002650150P012P012<4D><3F><21><21><21><21><4D><4F><21><21><21><21><4D><5F><21><21><21><21>
<4D><6F><00><00><CR>
112200002400030Rotation 2<CR>
2911U4002600150P012P012<4D><3F><00><00><CR>
2911U4002600205P012P012<4D><4F><00><00><CR>
2911U4002600250P012P012<4D><5F><00><00><CR>
2911U4002600300P012P012<4D><6F><00><00><CR>
112200002200030Rotation 3<CR>
3911U4002330315P012P012<4D><6F><21><21><21><21><4D><5F><21><21><21><21><4D><4F><21><21><21><21>
<4D><3F><00><00><CR>
112200002000030Rotation 4<CR>
4911U4001950165P012P012<4D><3F><00><00><CR>
4911U4001950215P012P012<4D><4F><00><00><CR>
4911U4001950265P012P012<4D><5F><00><00><CR>
4911U4001950315P012P012<4D><6F><00><00><CR>
1X1100001100010b0392007500020002<CR>
112200001650030SCALING JIS CHARACTER'S<CR>
1911U4001200020P010P020<21><6F><00><00><CR>
1911U4001200050P020P020<21><6F><00><00><CR>
1911U4001200080P030P020<21><6F><00><00><CR>
1911U4001200110P040P020<21><6F><00><00><CR>
1911U4001200145P040P030<21><6F><00><00><CR>
1911U4001200190P040P040<21><6F><00><00><CR>
1911U4001200250P040P050<21><6F><00><00><CR>
1911U4001200320P040P060<21><6F><00><00><CR>
112200000050010NORMAL INVERSE<CR>
112200000050245 NORMAL MIRROR<CR>
1911U4000250010P040P040<21><6F><00><00><CR>
1911U4000250245P040P040<4B><30><00><00><CR>
A5<CR>
1911U4000250090P040P040<21><6F><00><00><CR>
A1<CR>
M<CR>
1911U4000250390P040P040<4B><30><00><00><CR>
M<CR>
E<CR>

```

Note: The notation “<xx>” in this DPL file should be interpreted by the reader as representing the hexadecimal value of the byte sent to the printer.

(Continued next page)

Scalable Kanji Gothic B Available



Sample Kanji Gothic E DPL file (Hex-ASCII addressing) and resulting label:

```

<02>L<CR>
D11<CR>
ySPM<CR>
1911S0003100010P020P015Scalable Kanji Gothic E Available<CR>
1B110000020017001234567890<CR>
yUJS<CR>
1X1100001900010b0392011000020002<CR>
112200002800030JIS CHARACTER'S IN ALL 4 ROTATION'S<CR>
112200002600030Rotation 1<CR>
1911uK102650150P012P0124D3F212121214D4F212121214D5F212121214D6F<CR>
112200002400030Rotation 2<CR>
2911uK102600150P012P0124D3F<CR>
2911uK102600205P012P0124D4F<CR>
2911uK102600250P012P0124D5F<CR>
2911uK102600300P012P0124D6F<CR>
112200002200030Rotation 3<CR>
3911uK102330315P012P0124D6F212121214D5F212121214D4F212121214D3F<CR>
112200002000030Rotation 4<CR>
4911uK101950165P012P0124D3F<CR>
4911uK101950215P012P0124D4F<CR>
4911uK101950265P012P0124D5F<CR>
4911uK101950315P012P0124D6F<CR>
1X1100001100010b0392007500020002<CR>
112200001650030SCALING JIS CHARACTER'S<CR>

```

(Continued next page)

1911uK101200020P010P020216F<CR>
 1911uK101200050P020P020216F<CR>
 1911uK101200080P030P020216F<CR>
 1911uK101200110P040P020216F<CR>
 1911uK101200145P040P030216F<CR>
 1911uK101200190P040P040216F<CR>
 1911uK101200250P040P050216F<CR>
 1911uK101200320P040P060216F<CR>
 112200000050010NORMAL INVERSE<CR>
 112200000050245 NORMAL MIRROR<CR>
 1911uK100250010P040P040216F<CR>
 1911uK100250245P040P0404B30<CR>
 A5<CR>
 1911uK100250090P040P040216F<CR>
 A1<CR>
 M<CR>
 1911uK100250390P040P0404B30<CR>
 M<CR>
 E<CR>

Scalable Kanji Gothic E Available

JIS CHARACTER 'S IN ALL 4 ROTATION 'S				
Rotation 1	与	溶	欲	酪
Rotation 2	𠂔	𠂔	𠂔	𠂔
Rotation 3	𠂔	𠂔	𠂔	𠂔
Rotation 4	𠂔	𠂔	𠂔	𠂔

SCALING JIS CHARACTER 'S							
¥	¥	¥	¥	¥	¥	¥	¥

¥	¥		飽	𠂔
NORMAL	INVERSE		NORMAL	MIRROR

ILPC-CHINESE Option

The Chinese Option is a double byte scalable font supporting Simplified GB Chinese. In the double byte format the printer recalls one character printed from every two 8 bit bytes sent from the host. Double byte scalable fonts are selected using a print format record (see Chapter 8 and Appendix H for details).

DPL Big 5 Encoding Support: With the ILPC Chinese option, the printer firmware supports font files that are encoded for the GB Character Map and the Big 5 Character Map. The resident Asian font in the printer is encoded in the GB Character Map. To utilize the Big 5 Character Map, the user must download a font file that is Big 5 encoded. The font file downloaded must be of a size compatible with the internal module size available or of a size compatible with an external (plug in) module where applicable. Printing characters from the Big 5 encoded font file is accomplished by:

1. Setting the character mapping with a System Command or Label Format Command (<STX>yUB5 or yUB5, respectively).
2. Setting the 'b' field = '9' and 'eee' field = 'Unn', where 'nn' is equal to the Font ID number selected for the Big 5 encoded font file downloaded.
3. Selecting string data corresponding to the Big 5 Character Map.

Scalable Double Byte Font Map - CHINESE					
eee (Font Code)	Scalable Font Type	Font Name	Binary Addressing	Hex ASCII Addressing	Code Pages
UC0	Scalable Resident	Simplified GB Chinese	√		GB
uc0	Scalable Resident	Simplified GB Chinese		√	GB
U50 - U5z... U90 - U9z	Scalable Non-Resident (Download)	Big 5	√		B5
u50 - u5z u90 - u9z	Scalable Non-Resident (Download)	Big 5		√	B5
U50 - U5z... U90 - U9z	Scalable Non-Resident (Download)	User defined	√		
u50 - u5z u90 - u9z	Scalable Non-Resident (Download)	User defined		√	

Sample Simplified GB Chinese DPL file (binary addressing) and resulting label:

```
<02>L<CR>
D11<CR>
ySPM<CR>
1911S0003100010P020P015Scalable Chinese Available in GB Character Set<CR>
1B110000020017001234567890<CR>
yUGB<CR>
1X1100001900010b0392011000020002<CR>
112200002800030GB CHARACTER'S IN ALL 4 ROTATION'S<CR>
112200002600030Rotation 1<CR>
1911UC002650150P012P012<BD><D0>A1<A1><A1><A1><BD><D1><A1><A1><A1><A1><BD><D2><A1><A1><A1>
<A1><BD><D3><00><00><CR>
112200002400030Rotation 2<CR>
```

(Continued next page)

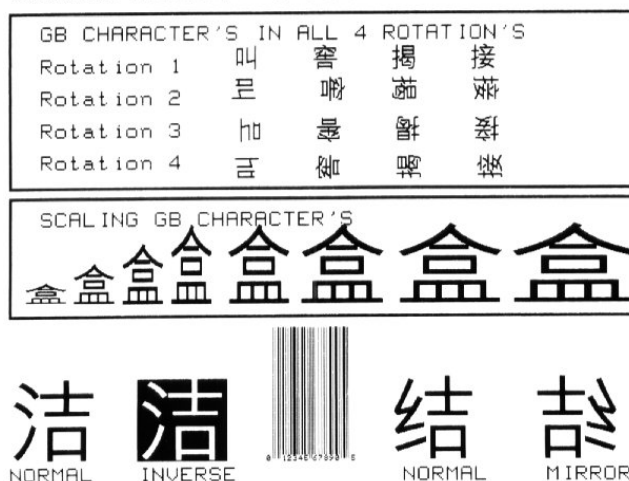
```

2911UC002600150P012P012<BD><D0><00><00><CR>
2911UC002600205P012P012<BD><D1><00><00><CR>
2911UC002600250P012P012<BD><D2><00><00><CR>
2911UC002600300P012P012<BD><D3><00><00><CR>
112200002200030Rotation 3<CR>
3911UC002330315P012P012<BD><D3><A1><A1><A1><A1><BD><D2><A1><A1><A1><A1><BD><D1><A1><A1>
<A1><A1><BD><D0><00><00><CR>
112200002200030Rotation 4<CR>
4911UC001950165P012P012<BD><D0><00><00><CR>
4911UC001950215P012P012<BD><D1><00><00><CR>
4911UC001950265P012P012<BD><D2><00><00><CR>
4911UC001950315P012P012<BD><D3><00><00><CR>
1X1100001100010b0392007500020002<CR>
112200001650030SCALING GB CHARACTER'S<CR>
1911UC001200020P010P020<BA><D0><00><00><CR>
1911UC001200050P020P020<BA><D0><00><00><CR>
1911UC001200080P030P020<BA><D0><00><00><CR>
1911UC001200110P040P020<BA><D0><00><00><CR>
1911UC001200145P040P030<BA><D0><00><00><CR>
1911UC001200190P040P040<BA><D0><00><00><CR>
1911UC001200250P040P050<BA><D0><00><00><CR>
1911UC001200320P040P060<BA><D0><00><00><CR>
112200000050010NORMAL INVERSE<CR>
112200000050245 NORMAL MIRROR<CR>
1911UC000250010P040P040<BD><E0><00><00><CR>
1911UC000250245P040P040<BD><E1><00><00><CR>
A5<CR>
1911UC000250090P040P040<BD><E0><00><00><CR>
A1<CR>
M<CR>
1911UC000250390P040P040<BD><E1><00><00><CR>
M<CR>
E<CR>

```

Note: The notation “<xx>” in this DPL file should be interpreted by the reader as representing the hexadecimal value of the byte sent to the printer.

Scalable Chinese Available in GB Character Set



Appendix R

Barcode Symbology Information Sources

AIM International, Inc.
11860 Sunrise Valley Drive, Suite 101
Reston, VA 22091 USA
Tel: 703-391-7621 Fax: 703-391-7624

Automotive Industry Action Group
26200 Lahser Road
Suite 200
Southfield, MI 48034 USA
Tel: 313-358-3570 Fax: 313-358-3253

AIM JAPAN
Aios Gotanda Bldg. 6F
1-10-7 Higashigotanda
Shinagawa-ku Tokyo 141 Japan
Tel: 03-3440-9085 Fax: 03-3440-9086

Computing Technology Industry Association
450 E. 22 Street Suite 230
Lombard, IL 60148-6158 USA
Tel: 630 268-1818 Fax: 630 278-1384

AIM EUROPE
The Old Vicarage
Haley Hill, Halifax HX3 6DR
West Yorkshire, England
Tel: 44-1422-359161 Fax: 44-1422-3556904

Health Industry Business Communications Council
PO Box 53528
Phoenix, AZ 85018 USA
Tel 602-318-1091

AIM UK
The Old Vicarage
Haley Hill, Halifax HX3 6DR
United Kingdom
Tel: 44-1422-359161 Fax: 44-1422-355604

International Article Numbering Association
(EAN)
Rue Royal 29
B-1000 Bruxelles
Belgium
Tel: 32-22-187674 Fax: 32-22-187585

AIM USA
634 Alpha Drive
Pittsburgh, PA 15238-2802 USA
Tel: 412-963-8588 Fax: 412-963-8753

Uniform Code Council, Inc. (UCC)
8163 Old Yankee Rd. Suite J
Dayton, OH 45458 USA
Tel: 513-435-3870 Fax: 513-435-4749

American National Standards Institute (ANSI)
11 West 42nd Street
New York, New York 10036 USA
Tel: 212-642-4900 Fax: 212-398-0023

U.S. Government Printing Office
732 North Capitol Street NW
Washington, DC 20401 USA
Tel: 202-512-1991 Fax: 202-512-1293

