PACKET REFERENCE MANUAL

TABLETOP PRINTER

1

Monarch® Products

Trademarks

TABLE OF CONTENTS

GETTING STARTED

This manual provides the necessary information to design, write and print a Monarch® Printer Control Language II (MPCLII) format on the Avery Dennison® Monarch® Tabletop Printer 1 (ADTP1). Review the printer information in the *Operator's Handbook* and *System Administrator's Guide*.

About This Manual

You do not need to be a programmer to use this manual, but you must be familiar with creating text files and using basic commands. This chapter describes how to

♦ create and download a sample MPCLII packet.

♦ design an MPCL II packet with or without an RFID field.

See "Defining Text Fields" in Chapter 3 for a list of available fonts for the printer. See Chapter 4, "Defining Field Options," for a list of available options for the printer.

Before You Begin

1. Connect the printer to the host. Refer to the *System Administrator's Guide* for more information.

2. Load supplies in the printer. Refer to the *Operator's Handbook* for more information.

3. Turn on the printer.

4. Set the communication parameters and configure the printer. The communication parameters at the printer must match those at the host. See Chapter 2, "Configuring the Printer," for more information.

5. Design the format. See "Designing a Format" for more information.

6. Download the format to the printer. See Chapter 6, "Printing," for more information.

Creating an MPCLII Format Packet

A format defines which fields appear and where the fields are printed on the label. The printer requires this information in a special form, using Monarch® Printer Control Language II (MPCL). This section describes how to create a sample MPCLII format packet.

Note: MPCLII formats designed for another MPCLII printer such as the 9855® or 9825® printer can be printed on this printer.

For detailed information about the format header, text, constant text, and bar code fields, see Chapter 3, "Defining Fields." For information about batch packets, see Chapter 6, "Printing."

1. Type the following format header in any text editor:

   {F,25,A,R,E,200,200,"FMT-25" |

2. Type the following constant text field:

   C,140,40,0,1,2,1,W,C,0,0,"SAMPLE FORMAT",0 |

3. Type the following bar code field:

   B,1,12,F,85,40,1,2,40,5,L,0 |

4. Type the following text field:

   T,2,18,V,50,50,1,1,1,1,B,L,0,0,1 | }

This is an MPCLII format packet. Next, create a batch packet before printing the format.

5. Type the following batch header, after the text field line:

   {B,25,N,1 |

6. Type the following bar code data:

   1,"02802811111" |

7. Type the following text field data:

   2,"TEXT FIELD" | }

8. Save the file as SAMPLE.FMT.

9. Type MODE COM1:9600,N,8,1 at the Command prompt when using serial communications. This sets the communication parameters at the host. These communication parameters must match those at the printer. See "Setting Serial Communication Parameters," in Chapter 2, or the host's documentation for more information.

10. Type COPY SAMPLE.FMT COM1. The following 2 inch by 2 inch label prints:

    {F,25,A,R,E,200,200,"Fmt 25" |
    C,140,40,0,1,2,1,W,C,0,0,"SAMPLE FORMAT",0 |
    B,1,12,F,85,40,1,2,40,5,L,0 |
    T,2,18,V,50,50,1,3,1,1,B,L,0,0,0 | }
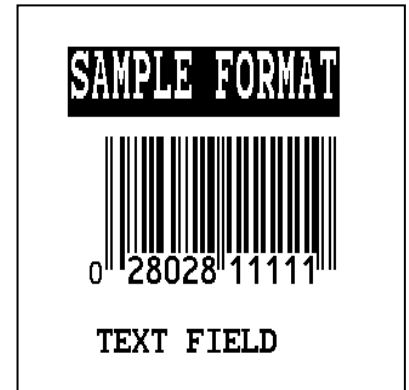
Sample Batch Packet

> **{B,25,N,1 |**
> **1,"02802811111" |**
> **2,"TEXT FIELD" | }**

Designing a Format

Before designing a format, answer these questions. How large is the supply, which fonts should be used, should a bar code be included, what about a graphic? There are several steps to designing a custom format:

Determine the label size. Labels are available from Avery Dennison in a wide variety of sizes. The application and the amount of printed data determine the supply size. Contact your Account Manager or Technical Support for more information.

1. Draw a rough sketch of the label. Note any areas that are preprinted on the label, such as a logo.

2. Identify the field types that appear on the label. See "About Field Types" for more information.

3. Decide which fonts to use. When working with fonts, there are three considerations: font appearance, font size (scalable or bitmapped), and font spacing (monospaced or proportional). The TrueType® scalable font, EFF Swiss Bold™ (font 50) is standard on the printers. See Appendix B, "Fonts," for samples of each font.

At this point, send the design to the printer. To do this:

4. Create a format packet. See Chapter 3, "Defining Fields," for more information.

5. Download the format packet to the printer. See Chapter 6, "Printing," for more information.

Keep backup copies of the format, batch data, check digit, and graphic packets.

RFID Format Design Tips

This section provides some tips about designing formats on large-gap RFID supplies.

Printing over the RFID inlay may cause printing irregularity.

Do not use skip index with RFID supplies. Unexpected results may occur.

1. Determine the label length. Measure the supply according to the supply type:

  ♦ Die cut supplies: ♦ Measure from the leading edge of one label to the leading edge of the next label accounting for the die cut gap in the format header.

  ♦ Black mark supplies: ♦ Measure from the top of one black mark to the top of the next black mark.

  ♦ Aperture supplies: ♦ Measure from the top of one aperture hole to the top of the next aperture hole.

Note: The leading edge is the edge of the label that exits the printer first; regardless of how the format is designed on the label.

Use the Monarch® RFID Printer Setup Utility to configure the printer (read and write power, position settings, etc.) for the RFID supplies. Find this Web-based utility at http://printers.averydennison.com/en/home/resources/service-and-support/rfid-printer-setup-utility.html
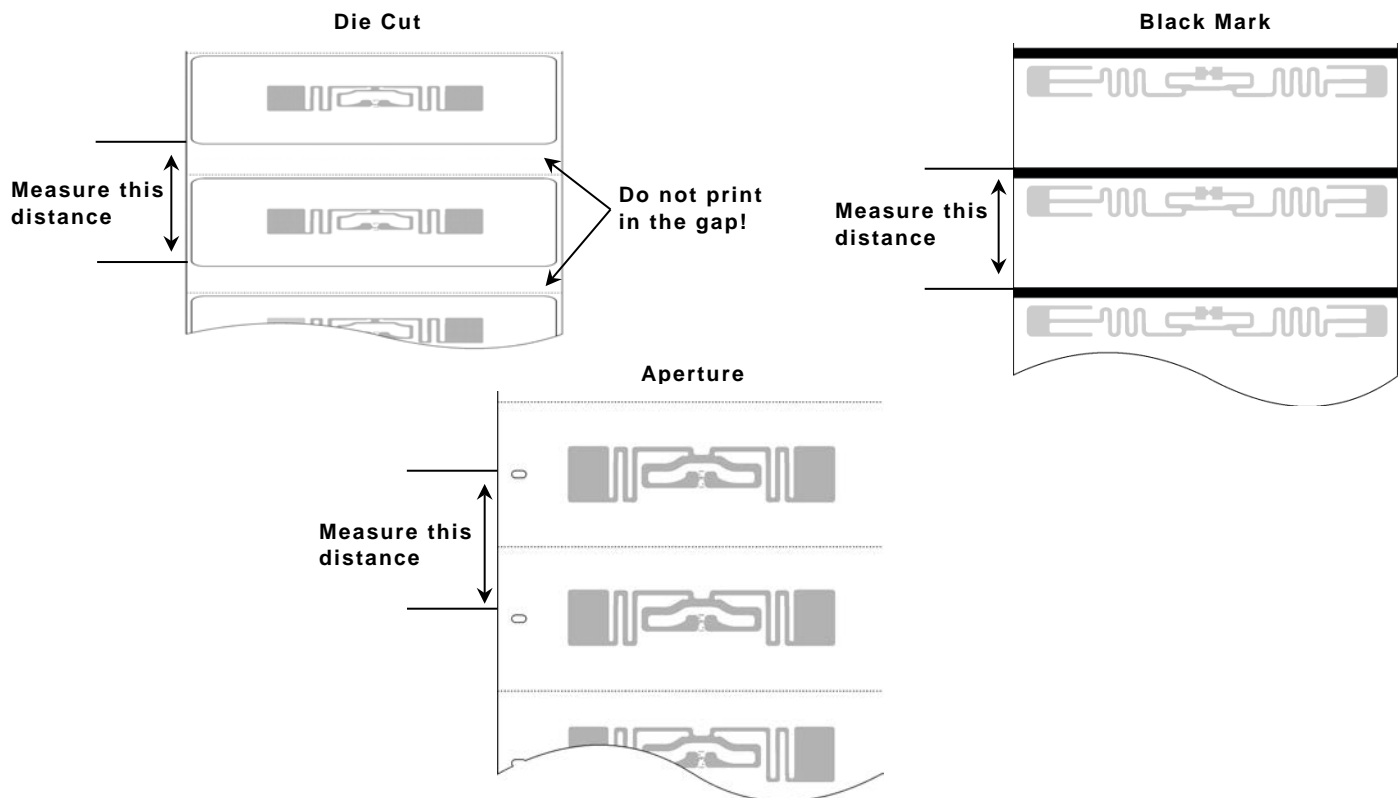
2. Enter the label length for parameter *f6. length* in the format header. See "Defining the Format Header" for more information.

3. Determine the label width. Measure the width of the printable area of the supply. **Do not** include the liner (backing paper) in this dimension.

4. Enter the label width for parameter *f7. width* in the format header. See "Defining the Format Header" for more information.

Determining the Print Area

The "bottom" (or leading edge) is the edge that exits the printer first. The 0,0 point is at the bottom left corner of the label. The print area is center justified.

Unit of Measure	Max. Supply (Wid x Len)	Max. Print Area (Wid x Len)	Min. Supply (Wid x Len)	Min. Print Area (Wid x Len)
English	432 x 1750	409 x 1600	75 x 32	75 x 32
Metric	1080 x 4445	1038 x 4064	191 x 81	191 x 81
Dots (203)	877 x 3553	832 x 3248	152 x 65	152 x 65
Dots (300)	1296 x 3750	1248 x 3600	225 x 96	225 x 96

Note: Longer images can be created by reallocating memory to increase the image buffer. See "Defining the Memory Configuration Packet" in Chapter 2 for more information.

Non-Print Zones

The following non-print zones are recommended

Left and Right   0.025 inches (0.6mm or 5 dots) on either edge of the label.

Top and Bottom   0.025 inches (0.6mm or 5 dots) on the leading and trailing edge of the label.

The leading edge is the edge of the label that exits the printer first; the trailing edge is the edge of the label that exits the printer last; regardless of how the format is designed on the label. If using the optional bar code verifier, allow a minimum no-scan zone on the trailing edge of the label of 0.5 inch (13 mm). .

Use the following formulas to convert inches to dots and metric:

> **Dots = inches x 203 (or 300 dots per inch)**
> **Metric (1/10mm) = inches x 254**
> **English (1/100 inch) = 100 x (dots/203) or (dots/300)**
> **Dots = Metric (1/10 mm) x 799/1000 (or 1181/1000)**
> **300 dpi depends on the printer.**

Using Supply Layout Grids

A supply layout grid contains measurement markers. These markers help accurately position information on the label.

Decide whether to design formats using English, Metric, or Dot measurements. Choose from the following grids:

English   The English grid is measured in 1/100 inches.

Metric   The Metric grid is measured in 1/10 millimeters (mm).

Graphic   The printer uses dots to print images on a label. The printhead has 203 dots per inch (dpi) or an optional 300 dots per inch printhead.

Choose English or Metric units when designing formats to use with different printers. English or Metric units allow more direct use of formats on printers with different density printheads.

Supply layout grids are in Appendix D, "Format Design Tools."
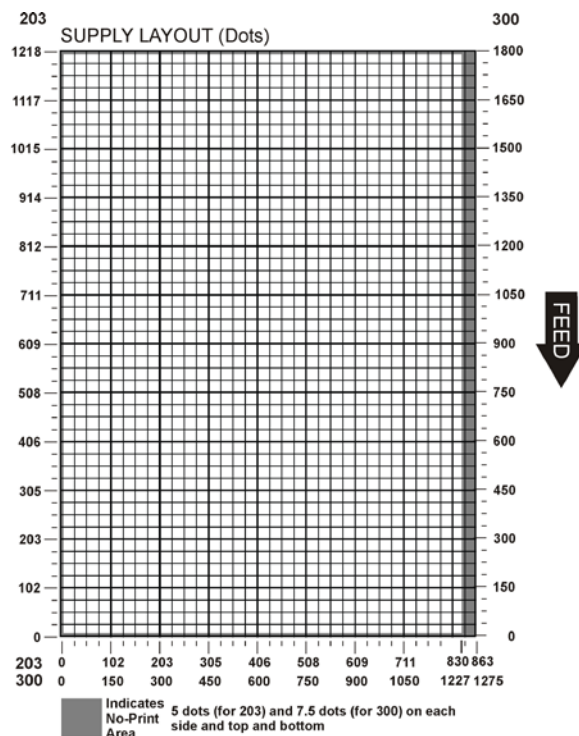
About Field Types

Decide what information to print on the format from the following categories.

Field Type	Description	Examples
Text	Contains letters, numbers, or symbols.	item number, item description, department number, price, date
Non-Printable Text	Holds data for use later, such as for merging into another field. The printer does not print non-printable text fields.	city, state, and zip code to be included in a bar code
Bar Code	Used for printing bar codes that can be scanned.	item or serial numbers, zip codes, information not visible to customers
Constant Text	Prints fixed characters that do not change.	company name, company address
Line or Box	Highlights or separates items.	line marking out the regular price, border around the supply
Graphic	Contains a bitmap image or a compliance label overlay.	logos
Verifier	Specifies a bar code verifier configuration to use for each format.	Mode 2, mode 19
RFID	Contains the RFID data to program into the RFID inlay.	EPC Class 1 Gen2 data

All of the above field types except graphics are discussed in Chapter 3. See Chapter 5, "Creating Graphics" for information on including graphics in the format.

CONFIGURING THE PRINTER

This chapter discusses how to

♦ set communication parameters.

♦ upload the printers configuration or font information.

♦ configure the printer using online configuration packets.

♦ use immediate commands to control the printers operation at any time.

Setting Serial Communication Parameters

For serial communications, the communication settings at the printer must match those at the host. Use the Communication Settings Packet to set these parameters.

Use the MODE command (from the Command prompt) to set communication values on the PC.

For example

MODE COM1:9600,N,8,1

This command sets the host to these communication values: 9600 baud, no parity, an 8 bit word length, 1 stop bit.

About the ACK/NAK Protocol

Using version 1.10 or greater firmware, the printer supports the ACK/NAK protocol. See "Sending Immediate Commands" for more information.

Using MPCLII Conventions

Here are some guidelines to follow when using MPCLII.

MPCLII Punctuation

Use the following symbols when creating MPCLII packets:

Character	Decimal Value	Description
{ (left bracket)	123	start of header
} (right bracket)	125	end of header
| (vertical bar)	124	field separator*
, (comma)	044	parameter separator
"ABC" (quotation marks)	034	Quotation marks enclose character strings. Empty quotes (" ") identify null strings or unused fields.
'comment' (single quotation marks)	039	Grave accents enclose comments. Any data enclosed in grave accents is ignored. Do not embed comments within a quoted string. Grave accents are also used to reject mainframe data.

* The field separator is the split vertical bar (|). The decimal value is 124. To enter this character, use the Shift key plus the Split Vertical Bar key on the computer's keyboard. Depending on the text editor, it may appear as a solid vertical bar or as a split vertical bar.

Note: These MPCL characters are the default.

Standard Syntax Guidelines

When creating MPCLII packets:

♦ Begin each packet with a **start of header** (**{**).

♦ End each packet with an **end of header** (**}**).

♦ Define no more than **1000** fields in a format. Each | indicates one field. However, options are not counted as fields. The actual number of fields a format can have may be less, because the number of fields is limited by the available memory.

♦ The field number (**0** to **999**) must be unique. Start at 1, instead of 0.

♦ Do not use a field number more than once per format.

♦ Define all fields in the order to image/print them. The printer does not print in field number order.

♦ Separate all parameters with a **Parameter Separator** (**,**).

♦ End each field with a **Field Separator** (**|**).

♦ Enter all information in CAPITAL letters, except words or phrases within quotation marks.

♦ Include all parameters for a field unless documented as optional.

♦ Define non-printable text fields before the field to which they apply.

♦ Define options immediately after the field to which they apply.

♦ Multiple options can be used with most fields. Options can be used in any combination except as noted with each definition. Options are processed in the order they are received.

♦ Keep in mind that proportionally spaced fonts need wider fields than monospaced fonts. For variable field data, use a letter W to determine the maximum field size.

♦ Do not place a new line (return) or any other non-printing character within a field definition. However, a carriage return or line break after each | makes formats easier to read.

   T,1,20,V,30,30,1,1,1,1,B,C,0,0,0 |
   T,2,10,V,50,30,1,1,1,1,B,C,0,0,0 |

♦ Spaces are ignored, except within character strings.

♦ Indenting options improves readability of formats.

   T,1,18,V,30,30,1,1,1,1,B,C,0,0,0 |
   R,42,1 |

♦ Use a tilde (~) followed by a 3-digit ASCII code in a quoted string to send function codes or extended characters or send the 8-bit ASCII code.

Modify formats and fields with the optional entry method. See "Optional Entry Method" in Chapter 6 for more information.

Using Online Configuration Packets

Use online configuration packets to change the printer's settings. Send an individual configuration packet or a single packet containing all the configuration packets. Supply all parameters for each packet. Leave the parameters blank that do not need to change. For example,

   {I,A,,,,,1 | }

prints a slashed zero and uses the last sent online System Setup parameters.

Make a copy of the online configuration worksheet in Appendix D, "Format Design Tools," and save the original. All online configuration packets are listed on the worksheet.

When turning off the printer, all the information in the online configuration packets is saved and used when the printer is turned back on. After changing the printer's configuration, resend the format, batch, or graphic to the printer before the changes take effect.

Configuration Packet Header

Always include an **I**, immediately after the left bracket **{** and before the packet identifier (A, B, C, etc.). The **I** parameter identifies the data stream as a configuration packet.

Note: Include the **I** parameter with each packet if sending them individually. Include it only at the beginning of a data stream if sending multiple packets.

Use this syntax to create online configuration packets:

Syntax

{	Start of Header
I,	Configuration Header
1 - 10 optional records	
A, parameter 1...parameter 5 |	System Setup
B, parameter 1...parameter 6 |	Supply Setup
C, parameter 1...parameter 7 |	Print Control
D, parameter 1...parameter 3 |	Monetary Formatting
E, parameter 1...parameter 9 |	Control Characters
F, parameter 1...parameter 5 |	Communication Settings
G, parameter 1...parameter 3 |	Backfeed Control
M, parameter 1...parameter 3 |	Memory Configuration
T, parameter 1...parameter 5 |	Time/Date
Y, parameter 1 parameter 5 |	RFID Setup
}	End of Header

Syntax for single packet

{	Start of Header
I,	Configuration Header
A, parameter 1...parameter 5 |	System Setup
}	End of Header

Add a configuration to RAM or specify units for supply, print, margin, and cut positions. If using the optional parameters with the **I** packet, any online configuration packets following the split vertical bar (|) must specify distances using the selected units. However, the test labels display the units in dots, even if entered in English or Metrics units.

Syntax   {**header,ID#,action,device | }**

1. header	Constant **I**.
2. ID#	ID. Use **0**.
3. action	Action. Options:
	A Add configuration.
	U Upload User Configuration.
4. device	Storage Device. Use **R** (Volatile RAM).
5. units	Units. (Optional parameter.) Options:
	E English
	M Metric
	G Dots

Example   **{I,0,A,R,E |**
           C,0,25,0,0,0 | }

Adds a configuration to volatile RAM and specifies English units. It also uses the default contrast, moves print 0.25 inches closer to the bottom of the supply and does not change the margin adjustment, prints at the default print speed, and uses the default printhead width.

If optional parameters are not used, the syntax for the online configuration packets does not change. For example,

{I,C,0,50,0,0,0 | }

uses the default contrast, moves print 50 dots (0.25) inches closer to the bottom of the supply and does not change the margin adjustment, prints at the default print speed, and uses the default printhead width.

Example   **{I,0,U,R | }**

Uploads the printer configuration from volatile RAM and returns the following to the host.

A,0,0,0,0,0 |
B,1,1,0,0,0,0,0 |
C,0,0,0,0,0,0 |
D,1,0,2 |
E,"~123~044~034~124~125~126","","~013~010" |
F,3,1,0,0,1 |
G,0,65,65 |
M,R,30715,23995,N,0,0|
M,R,R,0|
M,T,R,0|
M,I,R,3300|
M,D,R,640|
M,F,R,1500|
M,V,R,0|
T,0,0,0,0,0 |
Y,0,1,18,12,1,0,120,220,0,1,0,0 |

The parameters for each packet (**A-Y**) are displayed. In the first line that begins with **M**, 30715 is the total volatile memory available, 23995 is the memory available in volatile RAM. There is no non-volatile RAM available. The remaining lines beginning with **M** list the buffer sizes in 1/10K for the Image, Downloadable Fonts, and Formats Buffers.

Configuration Syntax Guidelines

When creating a printer configuration packet:

♦ Follow the Standard Syntax Guidelines listed at the beginning of this manual.

♦ The first character after the **start of header** (**{**) is the **configuration header** (**I**).

♦ Download multiple configuration packets within one packet or download a single configuration packet.

♦ If changing any of the online configuration packets, resend the format packet to the printer, so the configuration changes take effect.

♦ Include the first five ANSI codes, at a minimum, in the control characters packet.

♦ Send configuration packets once per session (each time the printer is turned off and then back on), not with every format or batch packet.

♦ The printer uses the settings from the printer configuration packets until a new configuration packet is sent; or a setting is adjusted through the printer's menu.

Making Print Adjustments

Horizontal and vertical adjustments can be made by adjusting the supply, print, or margin positions. However, keep in mind the following:

♦ Supply adjustments across the width of the supply, such as the margin position, are based in dots- either 203 dpi or 300 dpi, depending on the printhead density.

♦ Supply adjustments for the length of the supply, such as supply position or print adjustment, are always measured in 1/203 of an inch, regardless of the printhead density.

Defining the System Setup Packet

Use the system setup packet (A) to select the power up mode, display language, print separators between batches, print a slashed zero, and select the symbol set.

Syntax   {I,A,powup_mode,language,sep_on,slash_zero,symbol_set | }

A1. A	System Setup Packet	
A2. powup_mode	Printer's powerup mode. **0** is the default. Options:	
	0 online mode. Printer is ready to receive data and print.	
	1 offline mode. Operator can select a format and enter data to print.	
A3. language	Display Language. **0** (English) is the default. Options	

0	English		
1	Français	**8**	Español-MX
2	Deutsch	**9**	Dansk
3	Español-ES	**10**	Nederlands
4	日本語	**13**	Polski
5	Português	**14**	Türkçe
6	Italiano	**15**	简体中文
7	Svenska	**16**	Français-CA

A4. sep_on	Batch Separators. **0** is the default. Options:
	0 Does not print a separator
	1 Prints a separator
	2 Double-length separator – print 2 tags

Note: Do not use batch separators with continuous (non-indexed) supply. If using a stacker, the batch separator is 3.66 inches long.

A5. *slash_zero*	Slash Zero. **0** is the default. Options:
	0 Print a standard zero
	1 Print a zero with a slash through it
A6. *symbol_set*	Symbol Set. **0** is the default. Options:

0   Internal
1   ANSI
2   Code Page 437 (Latin U.S.)
3   Code Page 850 (Latin 1)
4   Code Page 1250 (Latin 2)
5   Code Page 1251 (Cyrillic)
6   Code Page 1252 (Latin 1)
7   Code Page 1253 (Greek)
8   Code Page 1254 (Turkish)
9   Code Page 1255 (Hebrew)
10   Code Page 1256 (Arabic)
11   Code Page 1257 (Baltic)
12   Code Page 1258 (Vietnamese)
13   DOS Code Page 852 (Latin 2)
14   DOS Code Page 855 (Russian)
15   DOS Code Page 857 (IBM Turkish)
16   DOS Code Page 860 (Portuguese)
17   Wingdings
18   Macintosh
19   Unicode
20   BIG5
21   GB2312
22   SJIS to SJIS (CP 932, Japanese)
23   GB2312 to GB2312 (CP 936, Simplified Chinese)
24   BIG5 to BIG5 (CP 950, Traditional Chinese)
25   KSC5601
26   Unicode UTF-8

Note: The Standard, Reduced, Bold, OCRA and HR fonts only support the Internal Symbol Set (0). The CG Triumvirate™ typefaces only support the ANSI (1) and DOS Code Page 437 (2) and 850 (3) Symbol Sets. The scalable font (font#50) does not support Code Page 1256 Arabic (10). Code pages 13-16 and 4-12 are for downloaded TrueType fonts or the scalable font. Code pages 19-26 require a downloaded International TrueType font. TrueType fonts are designed to be regionally specific; therefore, all code pages may not be supported in a given font. See Appendix C, "Symbol Sets/ Code Pages" for more information.

Example {I,A,0,0,1,1,0 | }

Powers up the printer in the online mode, displays prompts in English, prints a separator after each batch, prints zeros with slashes through them, and uses the internal symbol set.

Defining the Supply Setup Packet

Use the supply setup packet (B) to select supply type, ribbon, feed mode, supply position, and cut position.

Syntax   **{I,B,supply_type,ribbon,feed_mode,supply_posn,cut_posn, skip_index,cut_mode | }**

B1. B		Supply Setup Packet
B2. supply_type		Supply Type. **1** is the default. Options:

0   Center Black mark
1   Edge Die Cut
2   Continuous (non-indexed)
3   Center Aperture
4   Reserved
5   Tag Edge Aperture (commonly with RFID)
6   Center Tag Edge
7   Edge Black Mark
8   Center Die Cut

Use continuous supply in continuous mode.
If the supply has black marks, always use black mark sensing whether or not the supply has attacher holes.

0 - Center Black Mark   **1** - Edge Die   **2** - Continuous   **3** - Center   **5** - Tag Edge Aperture   **6** - Center Tag Edge   **8** - Center Die Cut
7 - Edge Black Mark   Cut   Aperture

B3. ribbon	Ribbon. **1** is the default. Options:
	0 Ribbon not installed
	1 Ribbon installed
	2 High Energy Ribbon

B3. ribbon Ribbon. **1** is the default. Options:

0   Ribbon not installed
1   Ribbon installed
2   High Energy Ribbon
Use a print speed of 2.5 IPS with the high energy ribbon. Serial bar codes cannot be printed with the high energy ribbon.
Settings for ribbon and supply type must match the supplies loaded in the printer; otherwise, an error occurs. To clear the error, change the configuration packet and resend it.

B4. feed_mode Feed Mode. **0** is the default. Options:

0   Continuous operation
1   On-demand mode
2   Liner take-up

B5. supply_posn Supply Position. Adjusts the position of the supply relative to the index marks in the feed direction. Range: **-300** to **300** in 1/203 inch. **0** is the default. Increase the supply position to move print up, decrease to move print down on the label. This adjustment accounts for mechanical tolerances from machine to machine. The supply position adjustment only needs to be made on the initial machine setup.
The supply position cannot be changed while the printer is active. Changing the supply position effects the print position, dispense position, and backfeed distance. Once the supply position is set, use the print control packet and backfeed control packet to adjust printing and the dispense position.

B6. cut_posn Cut position. Range: **-300** to **300** in 1/203 inch. **0** is the default. Adjusts where the tag is cut. The printer adjusts the cut position according to the index marks on the supply. Aperture supplies may need adjustments. Increase to move the cut up, decrease to move the cut down.

B7. skip_index Skip index mode. **0** is the default. Options:

0   Disable skip index mode
1   Enable skip index mode. Allows the printer to skip (ignore) a sense mark and print an image over multiple labels.

Note: Do not use skip index with RFID supplies. Unexpected results may occur!

B8. cut_mode   Enables or disables the optional cutter. Since the cutter is 3.4 inches away from the printhead, printed tags may be left between the printhead and cutter. **0** (does not cut) is the default.

Printed tags left between the printhead and cutter - Options:
1   Cuts each tag.
2   Cuts in strips, not each tag.
4   Does not cut before first tag, cuts each tag.

None or ***one*** *printed tag left between the printhead and cutter - Options:*
3   Cuts each tag.
5   Cuts in strips, not each tag.

Description of cut mode operation:

Cut Mode	Cut Before first tag in Batch?	Cut Between Tags?	Cut After Batch?	Printed tags left between printhead and cutter?
0	No	No	No	No
1	Yes	Yes	Yes	Yes*
2	Yes	No (Feeds strips)	Yes	Yes*
3	Yes	Yes	Yes	No*
4	No	Yes	Yes	Yes*
5	Yes	No (Feeds strips)	Yes	No*

No*   Cut Modes 3 and 5 minimize the number of tags left between the printhead and the cutter. Depending on the length of supply being cut, there may be one tag left between the printhead and the cutter.

Yes*   The last tag in the batch is queued to be cut once it reaches the cutter; however, there may be printed tags left between the printhead and the cutter. The user must send another batch to feed the last tag out far enough to be cut.

Note: Using "overstrike/continue" as the Error Action minimizes the number of RFID tags left between the printhead and cutter. To use the least amount of RFID tags, set overstrike to five.

Recommended Cutter Operation:

Printhead Density	With RFID Supply	With Stacker Attached	Recommended Cut Modes
203	No	No	1, 2, 4 – **Note**: Cut Modes 3 and 5 may cause ribbon tearing
203	No	Yes	1, 2, 4 – **Note:** Cut Modes 3 and 5 may cause ribbon tearing
203	RFID is not supported with 203 dpi		
203	RFID is not supported with 203 dpi		
300*	No	No	1, 2, 3, 4, 5
300*	No	Yes	1, 2, 3, 4, 5
300*	Yes	No	1, 2, 4
300*	Yes	Yes	3, 5 – **Note:** Some RFID supplies may curl and cause a jam (with cut mode 3). If this happens, use cut mode 1.

*   Contact Service to order replacement 300 dpi printheads (KST) for use with the cutter and RFID supplies.

Example   **{I,B,0,0,1,10,50,1,5 | }**

Indicates black mark and thermal direct stock has been loaded, causes the printer to operate in on-demand mode, feeds the supply approximately .05 inches up before printing the format on each label (10/203 inches), feeds the supply .25 inches (50/203 inches) before cutting, enables skip index mode and cuts in strips.

Defining the Print Control Packet

Use the print control packet (C) to set the contrast, print, and margin adjustment, print speed, and image rotation.

Syntax   **{I,C,contrast,print_adj,margin_adj,speed_adj,ph_width,bat_volt, rotate_img | }**

C1. C		Print Control Packet
C2. contrast		Print Contrast. Range: **-699** to **699**. **0** is the default. Using 399 or higher displays a warning message and tracks high energy inches. Solid black print cannot exceed 30% of any given square inch of the supply.
C3. print_adj		Print adjustment (position). Adjusts the image's position on the supply in the feed direction. Range: **-450** to **450** in 1/203 inch for 203 dpi or 1/300 inch for 300 dpi. **0** is the default. Increase the print position to move print up, decrease to move print down.
C4. margin_adj		Margin adjustment (position). Adjusts where the format prints side to side on the supply. Range: **-99** to **99** in 1/203 inch or 1/300 inch for 300 dpi printers). **0** is the default. Increase the margin position to move print to the right, decrease to move print to the left. Margin and print position are format adjustments. They do not affect the supply position, dispense position, or backfeed distance.
C5. speed_adj		Print Speed in inches per second (ips). **0** is the default. Options:

0   print at 6.0 ips, except formats with serial bar codes automatically print at 2.5 ips

25   2.5 ips (use for high energy ribbon)

40   4.0 ips (recommended for supply lengths less than 1.2")

60   6.0 ips

80   8.0 ips

100 10.0 ips

120 12.0 ips

The RFID settings in the Easy Setup Menu and the RFID Printer Setup Utility are based on encoding 96-bits of EPC data with a lock code. If your RFID data is more complex than 96-bits of EPC data with a lock code, you may need to decrease the printer's speed in non-stop encode mode. See "Creating Gen2 RFID Data" for more information about using a lock code.

Serial bar code printing information:

Print Speed	Printhead Density	Minimum Bar Code Narrow Element
2.5 ips	203/300 dpi	Less than 3 dots
4.0 ips	203 dpi	3 dots
6.0 ips or greater	203 dpi	4 dots or more

Serial bar codes with an 8-dot narrow element do not automatically print at 2.5 ips. Serial bar codes printed at speeds greater than 2.5 ips may not scan properly.

C6. ph_width   Width of the printhead in dots. Use **0**.

C7. bat_volt   Battery voltage. Use **0**.

C8. rotate_img   Image rotation. Rotates the printed image 180°. Design formats as needed, so image rotation is not required. **0** is the default. Options:
0   No rotation
1   180° rotation

Note: The image *length* and *width* must exactly match the loaded supply or the rotated image does not print correctly.

Example   **{I,C,0,-20,-10,0,0,0,1 | }**

Uses the default contrast, moves print 0.1 inch closer to the bottom of the supply (20/203 inches) and .05 inch to the left on the supply (10/203 inches), prints at six inches per second, uses the default printhead width, the battery voltage is ignored, and rotates the image 180°.

Defining the Monetary Formatting Packet

The monetary formatting packet (D) selects the monetary symbols to print for a price field. Use the monetary formatting packet to select primary and secondary monetary symbols, and designate the number of digits to appear at the right of a decimal.

Syntax   {I,D,cur_sym,secondary,decimals | }

D1. D	Monetary Formatting Packet	
D2. cur_sym	Currency Symbol. **1** is the default. Options:	

0	No symbol	**9**	Finland (₥, Markka)	
1	USA ($, Dollar)	**10**	Austria (₰, Shilling)	
2	UK (£, Pound)	**11**	India (Rs, Rupee)	
3	Japan (¥, Yen)	**12**	Russian (₽, Ruble)	
4	Germany (₧, Deutsche Mark)			
5	France (F, Franc)	**13**	Korean (₩, Won)	
6	Spain (P, Peseta)	**14**	Thai (฿, Baht)	
7	Italy (L., Lira)	**15**	Chinese (¥, Yuan)	
8	Sweden (Kr, Krona)	**16**	Euro (€)	

Note:	To use these symbols, select the internal symbol set.
D3. secondary	Secondary Sign. Secondary symbols only print if at least one decimal place is used. **0** is the default. Options:

0	No secondary sign
1	Print secondary sign

D4. decimals	Number of digits to the right of the decimal. **2** is the default. Options:

0	No digits
1	One digit
2	Two digits
3	Three digits

Example   {I,D,1,1,2 | }

Prints the dollar sign, uses a secondary symbol, and places two digits to the right of the decimal.

Defining the Control Characters Packet

Use the control characters packet (E) to change the MPCLII control characters, enable and disable the immediate commands, and change the default terminator character for job requests and ENQs.

Changes take effect with the first character following the end of header character of the configuration packet. Each control character must be unique and cannot appear anywhere else in a packet, except within quotation marks. Customize the trailer characters to work with the host.

Note: Wait two seconds for the new characters to take effect before sending packets using the new characters.

If using symbol set 110 (Unicode UTF-8) to print International characters, set the MPCL control characters (start of header, etc.) to decimal values between 0 and 128; otherwise, errors may occur with the Unicode data entered.

Use the following syntax for the control characters packet. Notice all but the first parameter is within quotation marks.

Syntax	{I,E,"ANSI_cd","string1","string2"	}			

E1. E	Control Characters Packet			
E2. "ANSI_cd"	**~123**	Start of header	{	(left bracket)
	~044	Parameter separator	,	(comma)
	~034	Quoted strings	"	(quotes)
	~124	Field separator	|	(vertical bar)

The field separator is the split vertical bar (|). The decimal value is 124. To enter this character, use the Shift key plus the Split Vertical Bar key on the computers keyboard. Depending on the text editor, it may appear as a solid vertical bar or as a split vertical bar.

	~125	End of header	}	(right bracket)
	~126	Data escape character (optional)	~~	(double tilde)
	def. ch.	Immediate command character (optional). Up to any 3 characters in the **0** to **255** decimal range. The character must be defined before this command can be used. The caret (~094) is normally used.		

Note: "ANSI_cd" includes seven separate parameters. The first five parameters are required. The other parameters are optional.

E3. "string 1" Terminator for status requests and ENQ requests. Up to any 3 characters in the **0** to **255** decimal range. The default is "**013**". Sending "" disables this sequence.

E4. "string 2" Terminator for job requests and data uploads. Up to any 3 characters in the **0** to **255** decimal range. The default is **none**. Sending "" disables this sequence.

After changing these parameters, all packets, including any future configuration packets, must use the new control characters. Use the tilde and ASCII character code sequence when sending this packet multiple times. Also, set the packet delimiters to characters within the 21 hex to 7E hex range.

Send the control characters packet to enable the immediate commands. An immediate command executes immediately, even if it is embedded within quotation marks, and all data following the command in the string is ignored.

Example   **{I,E,"~123~063~034~124~125~126~094" | }**

Changes the parameter separator character from , to ?. The other control characters remain unchanged. It also enables the immediate commands by defining the ^ symbol as the command identifier.

Resetting Control Characters

Change the characters in the previous example back to their original settings by downloading this packet:

{I?E?"~123~044~034~124~125~126~094" | }

Notice that the parameter separator is ? in this packet. This is the parameter separator that was set before this packet. Once the packet is received by the printer, the new parameter separator (a comma, in this case) is valid.

Be careful when using this feature. If you forget what the control characters were changed to, print a test label. (The test label lists the current control characters.) See "Printing a Test Label," in Chapter 8 for more information.

Using Immediate Commands

Immediate commands effect printer operation as soon as the printer receives them, even if they are included within a packet or used inside quotation marks.

Use immediate commands to change immediate command or status polling control characters, reset the printer, or cancel and repeat batches.

Enabling Immediate Commands

When the printer is first turned on, these commands are not available. To use these commands, send the control characters packet and define the immediate command control character. The immediate command control character is saved in non-volatile RAM so it is not lost after turning off the printer. Once the immediate command control character is defined, the immediate commands are enabled.

Sending Immediate Commands

Immediate commands consist of a three- or four-character sequence sent in a packet or embedded in an application. Each command must be sent separately.

Syntax   **control character_immediate command**

The printer can accept only one immediate command at a time. Sending a command before the previous one is completed results in an error.

Example   **^CB**

Immediately cancels the batch currently printing unless an error exists in the printer. This example assumes that the defined immediate command control character is the caret (^).

The table represents the defined immediate command control character as **^** and the defined status polling control character as **d**. These characters can be redefined if necessary.

Note: To use the immediate command control character or the status polling character within data, use the tilde sequence.

Command	Parameter
^CA	Cancels all the batches in the printer's queue unless an error exists on the printer. **Note:** This does not cancel batches in the printer's receive buffer.
^CB	Cancels only the current batch being printed unless an error exists.
^CT	Cut command.
^DD or ^DCd	Disables the MPCL data escape character (the tilde) and inhibits MPCL from acting on ANY data escape sequence from the host. Sets the MPCL data escape character to the ASCII value given by the **d** parameter. The value can be any ASCII character.
^EA	Aborts an error condition. This command is the same as pressing **ENTER** to clear an error. May need to be sent multiple times. Use ^RB to reprint batch. **CAUTION:** This command causes the current batch to stop and the condition that caused the error to remain uncorrected.
^ER	Resets the error. This command is the same as pressing **ENTER** to acknowledge an error. Normal operation resumes.
^FD	Feeds a label when printer is idle. Simulates the operation of pressing **FEED** and dispenses the next label if printer is in the on-demand mode. **Note:** Printer ignores this command if printing.
^FF1	Formats flash memory
^FF2	Returns the amount (in bytes) of the available flash memory.
^GD	Disables the conversion of formats designed in 203 dpi dot units to 300 dpi.
^GE	Enables the conversion of formats designed in 203 dpi dot units (not English or Metric) to 300 dpi.
^ID or ^ICd	Disables the Immediate Command feature by turning off the Immediate Command escape character. Sets the Immediate Command escape character to the ASCII value given by the d parameter. The value can be any ASCII character. Use ^IE to enable immediate commands.
^LCx	Adjusts the LCD contrast. X range is 1 to 9; the default is 3. Must use Version 1.9 or greater firmware.
^LE	Uploads the last successful EPC encoded data in ASCII Hex format to the last-used communication port/channel. **Note:** If the command is sent before printing is complete, the previous EPC data (if any) is returned
^MC	Returns the customer ID or RPQ version to the host. (00 to 99)
^MD	Returns the printhead dot density to the host. 00 = 203 dpi 01 = 300 dpi
^MI	Returns the customer ID or RPQ revision level to the host. (00 to 99)
^MM	Returns the model number to the host. M46 = ADTP1
^MP	Returns the prototype number to the host. (00 to 99)
^MR	Returns the revision number to the host. (00 to 99)
^MV	Returns the version number to the host. (00 to 99)
^NR	Performs a network factory reset.

Command	Parameter
^PR	Resets the printer. This command takes five seconds to complete and then the printer is ready to receive data. It has the same effect as turning off and then turning on the printer. **Note:** Command should be used only when the printer is not printing.
^RB	Repeats the last printed batch, printing the same number of labels as specified in the original batch. This command does not work if using batch separators. **Note:** Printer ignores this command if printing.
^RM	Checks the RFID hardware version.
^RS	Resynchronizes supply when supply roll is changed. **Note:** Printer ignores this command if printing.
^SD or ^SCd	Disables the status polling feature by turning off the status polling control character. Sets the status polling control character to the ASCII value given by the **d** parameter. The value of **d** can be any ASCII character.
^SFa	Loads script with host response.
^SFb	Loads script without host response.
^SFc	Enables script.
^SFd	Disables script.
^SFe	Uploads script version information.
^SFf	Deletes script.
^SFg	Turns on ENQ status polling before it reaches the script.
^SFh	Turns off ENQ status polling before it reaches the script.
^SFi	Turns on immediate commands before it reaches the script.
^SFj	Turns off immediate commands before it reaches the script.
^TC	Prints a series of three test labels: printer, MNet2, and MPCL.
^TP	Prints a test label. **Note:** Printer ignores this command if printing.
^UPD	Disables ACK/NAK protocol.
^UPE	Enables ACK/NAK protocol.
^VLC	Only clears the data that has been uploaded, not all the logged data. Data is cleared with this command after a successful upload. Clear the log file after uploading.
^VLD	Disables bar code verifier data logging (default when the printer is reset).
^VLE	Enables bar code verifier data logging.
^VLU	Uploads the log file to the last port that received host data (serial, USB, or Ethernet). Repeat this command until data is successfully uploaded. Data can be cleared after it has been uploaded.

Using the ACK/NAK Protocol

Enable ACK/NAK with the ^UPE command.

The printer sends the log message (saved in RAM) to the host and waits for an ACK character (hex 06) before the message is removed. Upon receipt of the ACK, the printer returns the next pending log message, with the above sequence repeating until no more messages exist.

ACK/NAK Error Handling

If five consecutive messages have been kept in RAM without receiving an ACK response, the printer stops with the error message "**Waiting for Host**". The operator cannot clear this error and no printer buttons are active.

ACK/NAK Error Recovery ("Waiting for Host" Message)

Depending on the host's condition, the host must send either an ACK or a NAK (hex 15) to clear the printer's "**Waiting for Host**" message and restart the ACK/NAK process. Once the printer receives an ACK or a NAK, it automatically processes any pending print jobs. If no unacknowledged log messages are saved, the printer responds with a NAK to any host ACK or NAK message.

Uploading RFID Data

When the printer encodes an RFID inlay, the encoded data is temporarily stored in the printer until the next RFID inlay is encoded.

To upload the RFID EPC encoded data, send immediate command **^LE**.

Retrieving EPC Encoded Data

After the printer receives the immediate command, **^LE**, the last successful EPC encoded data is returned in ASCII Hex format to the last-used communication port/channel.

Note: The **^LE** command must be sent after a successful encode and print. If the command is sent before printing is complete, the previous EPC data (if any) is returned.

Example

{F,1,A,R,E,300,150,"RFID"|
X,1,24|
 R,60|}
{B,1,N,999|
1,"30345678901234567890AB01"|}
^LE

The printer returns:

30345678901234567890AB01

Note: If the printer receives the ^LE command before encoding and printing are completed, the previous (if any) EPC data is returned.

Defining the Communication Settings Packet

Use the communication settings packet (F) to set the baud rate, word length, stop bits, parity, and flow control for serial communications. Changing the communication settings takes approximately two seconds. Communications sent during this interval will be lost. The host communication values must match the values on the printer. The host must be capable of communicating at the selected printer speed.

Do not add any characters, such as a carriage return/line feed, in the communication settings packet or communications errors may occur.

Syntax   {I,F,baud,word_length,stop_bits,parity,flow_control | }

F1. F	Communication Settings Packet	
F2. baud	Baud Rate. Options:	

0	1200	**2**	4800	**4**	19200	**6**	57600
1	2400	**3**	9600 (default)	**5**	38400	**7**	115200

F3. word_length   Word Length. Options:

0	7-bit word length
1	8-bit word length (default)

F4. stop_bits   Stop Bits. Options:

0	1-stop bit (default)
1	2-stop bits

F5. parity   Parity. Options:

0	None (default)
1	ODD parity
2	EVEN parity

F6. flow_control   Flow Control. Options:

0	None	**2**	(CTS)
1	DTR (default)	**3**	XON/XOFF

Note: If using the COPY command to download formats, set Flow Control to DTR (not XON/XOFF).

Example   {I,F,3,1,0,0,1 | }

Uses 9600 baud, an 8-bit word length, one stop bit, no parity, and the DTR mode.

Defining the Backfeed Control Packet

Use the backfeed control packet (G) to enable or disable the backfeed option, set the dispense position and the backfeed distance. Backfeed works by advancing each printed label to the desired dispense position. Once that label is removed, the next label to be printed is backed up underneath the printhead.

In continuous mode, only the last label in the batch is advanced to the dispense position. Adjust the dispense position to allow labels to be removed, die cut labels to be removed easily, or to prevent them from falling off. While the printer is active, the backfeed distance cannot be changed.

Do not use backfeed with supplies less than 0.75 inches. Use 0.5-inch gap supplies in peel mode when backfeed is disabled.

The dispense position and backfeed distance are optional parameters that do not have to be specified. However, they allow for greater precision when positioning the supply.

Syntax   {I,G,action,dis_pos,bkfd_dis | }

G1. G		Backfeed Control Packet
G2. action		Action. If using one-inch RFID supplies, do not use backfeed. Options:
	0	disable backfeed (default)
	1	enable backfeed
G3. dis_pos		Dispense Position. Adjusts the stopping point of the label. Range: **50** to **200** dots (default **65** dots).
G4. bkfd_dis		Backfeed Distance. Amount to move label backwards. **10** to **200** dots (default **65** dots). This distance cannot be greater than the dispense position.
		The backfeed distance should equal the dispense position.
		When tearing supplies (instead of peeling), the backfeed distance must be 30 dots (.150 inches) less than the dispense position. The 30-dot difference accounts for improper tearing of butt cut supplies to prevent exposed adhesive under the printhead. However, that adds a 30 dot non-print zone on the supply.

Example   {I,G,1,50,20 | }

Enables backfeed and sets the dispense position to 0.25 inches (50/203) and the backfeed distance to 0.10 inches (20/203).

Special Considerations When Using Backfeed

Make a note of the following items:

♦ Be careful when tearing supplies, because the adhesive can adhere to the printhead or platen roller.

♦ Backfeed affects each label in the on-demand mode or the first and last label of the batch in continuous mode.

♦ When backfeed is enabled and multiple batches are sent, the printer may not backfeed between each batch.

♦ Backfeed should only be used to advance labels to the desired dispense point.

♦ Backfeed does not interfere with the set supply, print, or margin positions.

♦ If the supply inter-label gap is not between .07 inch to .15 inch (14 to 30 dots), adjust the dispense position and backfeed distance accordingly.

See the following graphic for a representation of the following adjustments: dispense position, backfeed distance, supply position, print position, and margin position.
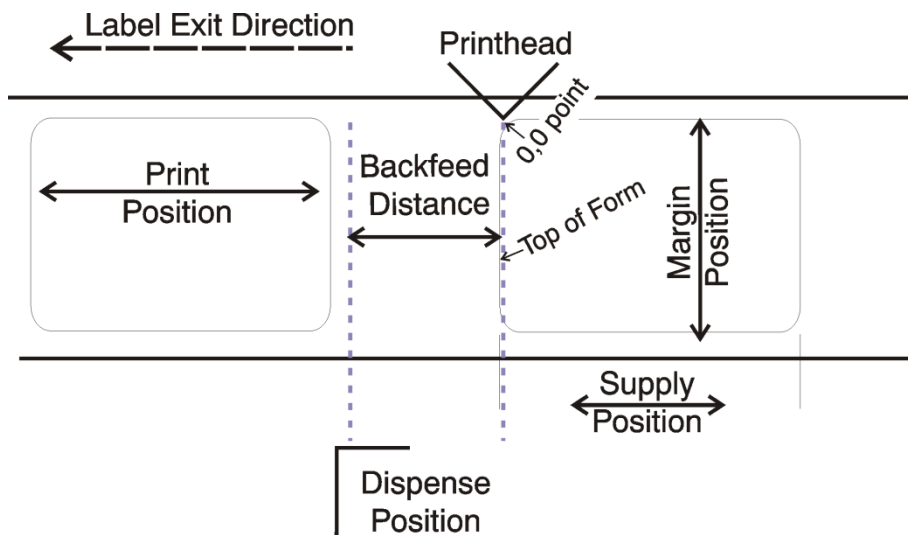
Defining the Memory Configuration Packet

Use the memory configuration packet (M) to customize the size of the printer's buffers.

Memory must be allocated in 1/2K increments. The memory configuration packet does not accept decimals, so enter whole numbers. Multiply the amount to reallocate (in K) by 10. For example,

To reallocate (in K)	Enter this amount
2.5	25
153	1530
229.5	2295

Each buffer's allocated memory remains in effect until that buffer is reallocated. For this reason, reallocate all the buffers when reallocating any buffer. Reallocating more memory than what is available causes an error.

Syntax   {I,M,buffer,device,buffer_size | }

M1. M	Memory Configuration Packet	
M2. buffer	Buffer type. Options:	

D   Downloadable Fonts
F   Format
I   Image

M3. device   Storage type. Use **R** for volatile RAM. Packets stored in volatile RAM are deleted when the printer is turned off.

M4. buffer size   Buffer size in 1/10K ranges. The following table lists the configured buffer sizes and min-max values for the printer. The printer's configuration is stored in non-volatile RAM and saved when the printer is turned off.

The maximum value for each buffer is listed; however, the sum of all the buffers cannot exceed the maximum available memory of the printer.

Buffer	Type	Default	Range
Image	Volatile	330K	41.5K – 3047K
Downloadable Fonts	Volatile	64K	8K- 3014K
Formats	Volatile	155K	16K- 1126.4K

Note: The printer has a transmit, receive, and scalable fonts buffer that are set to the maximum amount. These buffers cannot be changed.

Example   {I,M,I,R,1530 | }

Stores the image buffer in volatile RAM and allocates 153K for it.

Checking Current Buffer Sizes

Send a configuration upload packet to check the sizes of the buffers. See "Configuration Packet Header" for more information. After checking the current buffer sizes, begin reallocating memory.

To increase the image buffer and not use downloaded fonts, add that memory into the image buffer.

Example

{I,D,R,80 |	**Downloadable fonts buffer**	**8K**
M,I,R,4400 |}	**Image buffer**	**440K**

Make sure memory is available before adding memory to a buffer. In the above example, if the image buffer (M,I,R,4400) was defined before the downloadable fonts (M,D,R,80) buffer, an error would have occurred.

About Memory Buffers

Image Buffer Used to image the current format. The printer supports two 4-inch by 8-inch buffers. Images up to 16 inches long can be defined without reallocating memory. The image buffer is cleared after a printer reset.

Use the formula below to calculate the approximate image buffer size.

21K x Length (for 203 dpi) OR
46K x Length (for 300 dpi)

Length is the length of the label in inches.

203 DPI Example
21K x 25 = 525

(Multiply by 10 and round up to the next whole number if necessary)

525K x 10 = 5250

Enter 5250 for the Image buffer.

300 DPI Example
46K x 26.75 = 1230.5

(Multiply by 10 and round up to the next whole number if necessary)

1230.5 x 10 = 12305

Enter 12305 for the Image buffer.

Format Buffer Used to store formats, batch data, and graphics. Use the following formula to calculate the required format buffer size:

Linecount x 50/1024

Linecount is the number of lines in the format packet including the format header and all the options.

The result of the above calculation is in kilobytes. Formats remain in memory when the printer is turned off.

Downloadable Fonts Buffer Used to store downloaded (bitmapped or TrueType) fonts. To determine the size of the downloadable fonts, send a font packet. For TrueType fonts, the file size, in bytes, is the minimum amount needed in this buffer. See "Using the Font Packet" for more information. Fonts stored in this buffer are deleted when the printer is turned off. Store downloaded fonts into flash memory.

Buffer Allocation Considerations

Keep these items in mind when allocating memory.

♦ Do not allocate more memory than what is available.

♦ Free memory from one buffer before adding it to another buffer.

♦ Reallocate all the buffers if reallocating any buffer.

♦ Send all buffer (re)allocations in one packet. The printer evaluates each individual buffer allocation separately. If one buffer allocation is invalid, the entire packet is invalid. If a defined buffer size exceeds the maximum value, an error occurs. However, no information is lost.

♦ Whenever the printer accepts a memory configuration packet, it takes effect immediately, causing a printer reset. Any information contained in the buffers is lost. Resend the formats, batches, graphics, or fonts to the printer.

♦ When remapping the image buffer, make sure the length and width specified in the format header are not too large for the current image buffer. For example, remapping for a 4 x 4 inch label and then printing a 4 x 6 inch label causes an error, until the format header is changed or the image buffer is increased.

Memory Considerations with Downloaded TrueType Fonts

Save the fonts to flash memory for optimal performance with downloaded fonts. No memory configuration packets are necessary since flash memory cannot be reallocated.

About Flash Memory

Items stored in flash memory are saved when the printer is turned off. **Flash memory cannot be reallocated.**

Flash memory can contain formats, fonts, graphics, ADK scripts, and check digits. Store formats in flash to use the printer offline (not connected to a host). When formats, graphics, and check digits are saved in flash (a copy is automatically placed in RAM). Any of the packets saved in flash cannot exceed the memory available in RAM. For this reason, s**tore fonts or ADK scripts in flash memory**. (Fonts and ADK scripts saved in flash are not copied into RAM.)

Before sending packets to flash memory, it must be formatted.

Formatting Flash Memory

Formatting flash memory **is required once during initial printer setup**. From the Main Menu, select Setup, Flash Memory, then Format Flash. The process takes a few minutes.

Note: When formatting flash memory, all packets (formats, fonts, graphics, etc.) stored in flash memory are deleted and must be resent to the printer.

Checking Available Flash Memory

To check the amount of available flash memory in the printer, from the Main Menu, Select Setup, Flash Memory, then Unused Flash. The number (in bytes) of available flash memory is shown. Divide this number by 1024 to get the number of available kilobytes and make sure the font or ADK script files do not exceed this amount. For example, if the printer displays "Unused Flash 10223616," there is 9984K (9.75 Meg) of available flash memory for fonts or ADK script files. Flash memory cannot be reallocated.

Flash Memory Guidelines

Use these basic guidelines for storing files in the printer's flash memory.

♦ The printer has volatile RAM (packets deleted when the printer is turned off) and flash memory (packets saved when the printer is turned off). **Flash memory cannot be reallocated**.

♦ Flash memory must be formatted before any packets can be saved. See "Formatting Flash Memory" for more information.

♦ Formats, graphics, and check digits can be saved in flash (a copy is automatically placed in RAM). Any of these packets saved in flash cannot exceed the memory available in RAM.

♦ Packets with the same number should not be added to flash memory. If two packets with the same number are stored in flash, only the last packet sent can be used. For example, if you send the following packets to flash memory, only the Textiles format can be used.

{F,10,A,F,E,300,200,"UPC" | }
{F,10,A,F,E,300,200,"Textiles" | }

♦ There is not a 1-to-1 ratio between the RAM memory (formats, graphics, and check digits) in the printer and the memory in the PC. For example, a file that is 5K in Windows may require 15-20K to store in the printer.

♦ When using graphics, use the lowest-resolution graphic possible. If the printer uses a 203 dpi printhead, the graphics resolution should be 150 to 200 dpi. 96 dpi may work. Create the smallest file size possible.

♦ Use temporary storage for graphics that are only used once or twice. However, for a logo that is used on multiple formats, save the graphic in flash. Create the smallest file size possible.

♦ Each line in a packet requires the same amount of memory. The smaller the format, the less memory required to save it.

Clearing Packets from Memory

Remove packets from the printer to increase memory storage capacity or if the formats/fonts are no longer needed. In some cases, turning the printer off may clear the packets from memory. If not, send a format clear packet.

Syntax   **{header,packet#,action,device | }**

1. header	Identifies the packet. Options:

        A   Check Digit Scheme
        F   Format
        G   Graphic
        W   Font

2. packet#   Identification number of the packet to clear (**1** to **999**) or font number (**0** to **9999**). **0** is for all fonts.

3. action   Action. Enter **C** to clear the packet.

4. device   Storage device. Options:

        F   Flash
        R   Volatile RAM

Example   **{F,1,C,R | }**

Clears Format #1 from volatile RAM.

Defining the Time/Date Packet

Use the time/date packet (T) to set the printer's time and date. The printer must contain the optional clock. The printer keeps track of the date and time even when the printer is turned off.

The clock operates on 24-hour time. For example, 4:00 pm is 16:00 hours.

See "Using Option 7 Print Time/Date" in Chapter 4 to print the current date and time in a format.

Syntax   {I,T,hour,min,day,month,year | }

T1. T	Time/Date Packet	
T2. hour	Hour in 24-hour time. Range: **0** to **23**.	
T3. min	Minutes. Range: **0** to **59**.	
T4. day	Day of the month. Range: **1** to **31**.	
T5. month	Month of the year. Options:	

1	January	**7**	July
2	February	**8**	August
3	March	**9**	September
4	April	**10**	October
5	May	**11**	November
6	June	**12**	December

T6. year   Year (as two-digit). Range: **0** to **99**. 15 is 2015.

Example   {I,T,18,15,12,06,15 | }

Sets the printer's clock to 6:15 pm June 12, 2015.

Using the Font Packet

Use a font packet to add or clear downloaded fonts from memory, upload the font buffer, or upload the cell size information for a particular font. The font packet is useful when downloading fonts. If using downloaded fonts, the font number and the number of bytes each downloaded font uses is listed.

This packet does not list the number of bytes the standard printer fonts use.

Use the MONARCH® MPCL Toolbox Font Utility (available on our Web site) to create the font header and data. Refer to the online help for more information.

Syntax {W,font#,action,device,data_length,data_record | }

W1. W	Writable Font Header.	
W2. font#	The font identifier from **0** to **32000**. 0 is for all fonts. **1** - **5** digits is the font number. Example: 3 is the standard printer font, Bold.	
W3. action	Action. Options:	

    A   Adds the specified font.
    C   Clears all or specified fonts, except ones in flash.
    H   Uploads font size information.
    M   Uploads font memory usage information.

Note: C does not clear (erase) fonts saved in the printer's flash memory. To erase ALL fonts from the printer's flash memory, format flash.

W4. device   Device. Options:

    F   Flash
    R   Volatile RAM.

W5. data_length   The length of the font data. The range is 68 to 16384. This is optional. When creating fonts, include the font data with this packet.

W6. data_record   Multiple data records define the font. The first character is either an **H** (hex) or an **R** (run-length), referring to the algorithm. The rest of the record is up to 2710 characters of font data in double quotes. Separate the algorithm and the data with a comma, and end the record with **|**. This is optional.

Example {W,0,M,R | }

Selects all fonts and checks the memory usage in RAM. The printer returns the following to the host:

{W,0,M,R |
Number of bytes free, Number of bytes used | }

Example **{W,0,H,Z | }**

Selects all fonts and uploads the font size information for any downloaded fonts.

The printer returns the following to the host:

```
{W,0,H,Z|
0,1,0,"Standard",0,0,0,14,22,14,22,3|                    Font Style/Number
0,1,1,"CGTriBd9",1,0,7,25,31,10,15,0|
0,1,437,"CGTriBd9",1,0,7,25,31,10,15,0|                  Symbol Set
0,1,850,"CGTriBd9",1,0,7,25,31,10,15,0|
0,10,0,"CGTriBd9",1,0,7,25,31,10,15,0|                   Font Name
0,10,1,"CGTriBd9",1,0,7,25,31,10,15,0|
0,10,437,"CGTriBd9",1,0,7,25,31,10,15,0|                 Spacing
0,10,850,"CGTriBd9",1,0,7,25,31,10,15,0|
0,2,0,"Reduced",0,0,0,7,14,7,14,1|                       Type
0,2,1,"CGTriumv6",1,0,5,17,21,5,10,0|
0,2,437,"CGTriumv6",1,0,5,17,21,5,10,0|                  Baseline
0,2,850,"CGTriumv6",1,0,5,17,21,5,10,0|
0,11,0,"CGTriumv6",1,0,5,17,21,5,10,0|                   Cell Width
0,11,1,"CGTriumv6",1,0,5,17,21,5,10,0|
0,11,437,"CGTriumv6",1,0,5,17,21,5,10,0|                 Cell Height
0,11,850,"CGTriumv6",1,0,5,17,21,5,10,0|
0,3,0,"Bold",0,0,0,24,34,24,34,3|
0,4,0,"OCRA",0,0,0,13,24,13,24,3|
0,5,0,"HR1",0,0,0,12,20,12,20,2|                         Nominal Width
0,6,0,"HR2",0,0,0,10,16,10,16,1|
0,15,0,"CGTriumv7",1,0,7,21,28,9,14,0|                   Nominal Height
0,15,1,"CGTriumv7",1,0,7,21,28,9,14,0|
0,15,437,"CGTriumv7",1,0,7,21,28,9,14,0|                 Inter-Character Gap
0,15,850,"CGTriumv7",1,0,7,21,28,9,14,0|
0,510,0,"CGTriumv7",1,0,7,21,28,9,14,0|
0,510,1,"CGTriumv7",1,0,7,21,28,9,14,0|
0,510,437,"CGTriumv7",1,0,7,22,28,9,14,0|
0,510,850,"CGTriumv7",1,0,7,22,28,9,14,0|
0,16,0,"CGTriumv9",1,0,8,28,35,12,18,0|
0,16,1,"CGTriumv9",1,0,8,28,35,12,18,0|
0,16,437,"CGTriumv9",1,0,8,28,35,12,18,0|
0,16,850,"CGTriumv9",1,0,8,28,35,12,18,0|
0,511,0,"CGTriumv9",1,0,8,28,35,12,18,0|
0,511,1,"CGTriumv9",1,0,8,28,35,12,18,0|
0,511,437,"CGTriumv9",1,0,8,29,35,12,18,0|
0,511,850,"CGTriumv9",1,0,8,29,35,12,18,0|
0,17,0,"CGTriumv11",1,0,9,31,40,13,22,0|
0,17,1,"CGTriumv11",1,0,9,31,40,13,22,0|
0,17,437,"CGTriumv11",1,0,9,31,40,13,22,0|
0,17,850,"CGTriumv11",1,0,9,31,40,13,22,0|
0,512,0,"CGTriumv11",1,0,9,31,40,13,22,0|
0,512,1,"CGTriumv11",1,0,9,31,40,13,22,0|
0,512,437,"CGTriumv11",1,0,9,33,40,13,22,0|
```

```
0,512,850,"CGTriumv11",1,0,9,33,40,13,22,0|
0,18,0,"CGTriumv15",1,0,13,47,59,20,31,0|
0,18,1,"CGTriumv15",1,0,13,47,59,20,31,0|
0,18,437,"CGTriumv15",1,0,13,47,59,20,31,0|
0,18,850,"CGTriumv15",1,0,13,47,59,20,31,0|
0,513,0,"CGTriumv15",1,0,13,47,59,20,31,0|
0,513,1,"CGTriumv15",1,0,13,47,59,20,31,0|
0,513,437,"CGTriumv15",1,0,13,49,59,20,31,0|
0,513,850,"CGTriumv15",1,0,13,49,59,20,31,0|
0,70,0,"Paxar15",1,0,7,32,42,22,42,0|
0,71,0,"Paxar18",1,0,9,34,51,27,51,0|
0,72,0,"Nafta15",1,0,6,32,42,32,41,0|
0,73,0,"Nafta18",1,0,8,48,51,46,50,0|
0,21,0,"ZIAFont",0,0,0,5,9,5,9,1 |
0,22,0,"ZIBFont",0,0,0,7,11,7,11,2 |
0,23,0,"ZICFont",0,0,0,10,18,10,18,2 |
0,24,0,"ZIDFont",0,0,0,10,18,10,18,2 |
0,25,0,"ZIEFont",0,0,0,15,28,15,28,5 |
0,26,0,"ZIFFont",0,0,0,13,26,13,26,3 |
0,27,0,"ZIGFont",0,0,0,40,60,40,60,8 |
0,50,0,"EffSwissBold",1,1,92248 |
0,56,0,"PaxSymbols",1,1,12260 | }
```

Note: The CG Triumvirate™ typefaces are trademarks of Monotype Imaging, Inc.
Fonts 21-27 are only for the MLI™ printer.
Fonts 510, 511, 512, and 513 are the same as fonts 15, 16, 17, and 18.

Spacing	Monospaced (0) or proportional (1).
Type	Bitmapped (0) or scalable (1).
Baseline	Bottom of the font.
Cell Width	Horizontal number of dots to contain the widest character.
Cell Height	Vertical number of dots to contain the tallest character.
Nominal Width	Average width for lower-case letters.
Nominal Height	Average height for lower-case letters.
Inter-Character Gap	Default spacing between characters in monospaced fonts.
Printhead Density	Displays whether a 203 (0) dpi or 300 (1) dpi printhead is used. The scalable font (font 50) does not display which printhead (203 dpi or 300 dpi) is used.

Uploading Format Header Information

Upload format header information from the formats in memory to check the supply length and width for each format. Formats stored in flash memory are loaded into RAM when the printer boots. However, the formats remain in flash memory when the printer is turned off.

Syntax   {header,format#,action,device | }

F1. header	Format Header	
F2. format#	Format number from **0** to **999**. 0 is for all formats in memory.	
F3. action	Action. Options:	
	A	Adds the specified format
	C	Clears the specified format
	H	Uploads format header information
F4. device	Device. Options:	
	F	Flash
	R	Volatile RAM
	Z	All devices (use for upload)

Example   {F,0,H,Z | }

Selects all formats in memory and returns the following:

Example   {F,0,H,Z |
        Fmt_1,406,406 |
        Fmt_10,324,406 |
        Fmt_15,812,812 |
        Fmt_20,305,609 |
        Fmt_25,1218,406 | }

Displays the format number, supply length and supply width (in dots) for each format in memory.

Example   {F,1,H,Z | }

Selects format1 and returns the following to the host:

{F,1,H,Z |
Fmt_1,406,406 | }

Displays the supply length and supply width (in dots) for format1.

Defining a Network Console Packet

Use the network console packet to send console commands directly to the wireless module via the printer's serial port. If the wireless module does not appear to be communicating with the printer, use the network console packet to change the printer's SSID, IP Address, etc.

Syntax   **{N,number,action,device,"name" |**
            C,"con_comds" | }

N1. N	Network Console Packet.
N2. number	Number from **0** to **999** to identify the network console packet. **1** is the default.
N3. action	Action. Enter **A** to add a network console packet.
N4. device	Storage device. Enter **T** to pass the packet through the printer and store the packet in the wireless module.
N5. "name"	Packet name, **0** to **8** characters, enclose within quotation marks. "" is the default.
C1. C	Command field.
C2. "con_comds"	Console commands. Must be enclosed within quotation marks. "" is the default. Each command must be on a separate line. The maximum number of characters per command is 100. Refer to the list of console commands in the *ADTP1 MonarchNet2 Operating Instructions* for more information.
Note:	The maximum number of commands per packet is twenty five (25).

Example   **{N,1,A,T,"mystore" |**
            C,"set ip me static" |
            C,"init" |
            C,"exit" | }

Sends the network console packet mystore directly to the wireless module (**T**) and sets the IP method to static for determining the IP Addresses. The Init and Exit commands must be used to save changes and initialize the wireless module.

Defining a Bar Code Verifier Configuration Packet

Use the bar code verifier configuration packet to customize the bar code verifier for each format. Select which bar codes to scan, the acceptable ANSI grade levels, quiet zone, X-dimension range, etc.

Syntax **V,format#,action,device,"name" |**
D,"vfr_comds" |

V1. *V*	Bar Code Verifier Configuration Packet.
V2. *format#*	Number from **0** to **999** to identify the format number that corresponds to this particular bar code verifier configuration packet. Assign a bar code verifier configuration packet to any format even if the format number is not changed. See "Defining Bar Code Verifier Fields" in Chapter 3 for more information.
Note:	Specify up to 50 different bar code verifier configurations for the formats.
V3. *action*	Action. Enter **A** to add a bar code verifier configuration.
V4. *device*	Storage device. Options:

F   Flash. Stores the bar code verifier configuration in the printer. Packets stored in flash memory are saved when the power is turned off.

T   Temporary. Passes the packet through the printer and stores the packet in the bar code verifier. Packets stored in temporary memory are lost when the power is turned off.

V5. *"name"*	Packet name, **0** to **8** characters, enclose within quotation marks. The name entered here is shown as a custom configuration on the menu for MODE. Refer to the bar code verifier's *Operating Instructions* (available on our Web site) for more information.
D1. *D*	Non-Printable Text/Data Field.
D2. *"vfr_comds"*	Bar code verifier commands. Must be enclosed within quotation marks. Refer to the *SV Series Operators Guide* for the complete list of verifier commands.

Example **V,1,A,T,"BEAMON" | D,"~LV02" |**

The bar code verifier's configuration packet ID is "1" and named "BEAMON." The bar code verifier packet is added to temporary memory. It uses Mode 2.

Defining the RFID Setup Packet

The printer supports EPC Class-1 Generation-2 UHF (C1Gen2) protocol encoding following the *EPC™ Radio-Frequency Identity Protocols Class 1 Generation 2 UHF RFID Protocol for Communications at 860-960MHz Standards* (GS1 EPCglobal™ Specification for RFID Air Interface).

Use the RFID setup packet (Y) to set the encode mode, singulate mode, write power, read power, tag saver, first TID position, start encode position, encode zone, encode position, tag verify, tag verify position, and retry encode. Refer to the *System Administrator's Guide* for more detail about each of these parameters.

We recommend using the printer's Easy Setup Menu to configure RFID settings or the Monarch® RFID Printer Setup Utility. Find this Web-based utility at http://printers.averydennison.com/en/home/resources/service-and-support/rfid-printer-setup-utility.html

Note: Printing over the RFID inlay may cause printing irregularity.
Do not use skip index with RFID supplies. Unexpected results may occur.

Syntax {I,Y,stop_to_encode,singulate,write_power,read_power,tag_saver, first_TID_pos, start_enc_pos,encode_zone,encode_pos,tag_verify,tag_verify_pos,retry_encode, verify_power | }

Y1. Y	RFID Setup Packet.
Y2. stop_to_encode	Stop-to-Encode mode. The printer operates in Stop-to-Encode mode or Non-stop encode mode. Options:

0 No. The printer operates in non-stop encode mode (default). The printer does not pause (or stop) while encoding the RFID inlay.
1 Yes. The printer operates in stop-to-encode mode. The printer pauses (or stops) while encoding the RFID inlay.

Note: Stop-to-encode mode must be used for serialized EPC data. See "Using Option 65 (Serialized EPC Data)" for more information.

Y3. singulate Singulate mode. Read and use the TID field to only program that specific inlay. Enabling singulate mode decreases throughput but may prevent adjacency issues. If enabled, set *first_TID_pos*.

Singulate mode must be enabled to upload TID data. See "Using Option 6 (Upload Field Data)" for more information.

Options:
0 Disabled (default)
1 Enabled

Y4. write_power The amount of power to write (program) an RFID inlay. Use the RFID Printer Setup Utility to determine this value. The range is -15 to 23 and the default is -15.

Y5. read_power The amount of power to read an RFID inlay. Use the RFID Printer Setup Utility to determine this value. The range is -15 to 23 and the default is -15.

Y6. tag_saver Tag saver (use only in non-stop encode mode). Enabling tag saver moves the supply backwards before encoding the first inlay in a batch, which prevents waste. Options:
0 disable tag saver (default). The printer does not move the first tag in the batch backwards before encoding.
1 enable tag saver. The printer moves the first tag in the batch backwards to the proper position before encoding.

Y7. first_TID_pos	First TID Position (use only in non-stop encode mode with singulate mode enabled). The first TID position moves the 'hotspot' of the first inlay over the antenna so the TID order can be calculated. Use the RFID Printer Setup Utility to determine this value. The range is **0** to **2000** in 1/203 inch. **0** is the default.
Y8.start_enc_pos	Start encoding position (use only in non-stop encode mode). The printer uses this initial starting position to look for the RFID inlay to program while moving the supply to print. Measure from the leading edge of the supply to the beginning of the RFID inlay encoding area in 0.01 inches. Use the RFID Printer Setup Utility to determine this value. The range is **10** to **2000** in 1/203 inch. **100** is the default.
Y9. encode_zone	Encode zone (use only in non-stop encode mode). The encode zone is the area where the antenna and inlay communicate with each other during programming as the inlay moves over the antenna. It is the length in dots of the programming zone (area). Use the RFID Printer Setup Utility to determine this value. The range is **50** to **2000** in 1/203 inch. **200** is the default.
Y10. encode_pos	Encode Position (use only in stop-to-encode mode). Measure from the leading edge of the supply to the best programming position within the inlay. The Encode Position is different for each type of inlay. Use the RFID Printer Setup Utility to determine this value. The range is **0** to **2000** in 1/203 inch. **0** is the default.
Note:	*When using a 0 encode position*, the printer encodes when the label is at the defined top-of-form position – no extra supply movement is required. When using a **non-zero** encode position, the printer automatically backfeeds between labels, thereby ***decreasing throughput***.
Y11. tag_verify	Tag verify. The printer confirms the read value matches the programmed value after writing the RFID data. Options:

0 Disabled (default). The printer does not verify the programmed value.

1 Internal. Confirms the read EPC data matches the programmed EPC data using the internal antenna.

2 External. Confirms the read EPC data matches the programmed EPC data using the optional external verifier. Requires setting the tag verify position tag verify position.

3 Both. Confirms the read EPC data matches the programmed EPC data using the internal antenna and optional external verifier. Requires setting the tag verify position.

Note:	Enabling internal or both may decrease throughput.
Y12. tag_verify_pos	Tag verify position (use only when *verify* is 2 - external or 3 - both). Sets the position for when the label's leading edge reaches the external verifier. Use the RFID Printer Setup Utility to determine this value. The range is **10** to **2000** in 1/203 inch. **100** is the default.

Y13.retry_encode	The printer tries to reprint and program a label where a supply error occurred. The printer uses the same EPC data from the errored label on the next label. Options:

0   Disabled. The printer does not reprint or re-program the EPC data. Once EPC data is already programmed into an inlay, the printer will not program the same EPC data.

1   Enabled (default). The printer reprints and re-programs the EPC data from a label when a supply error occurs.

Retry_enc must be used with incrementing batches. If disabled, the printer does not reprint a label using the EPC data from the label with a supply error. For example, printing a batch of 100 labels prints 99 labels (or less) if a supply error occurs.

Y14. verify_power	The amount of power to read an RFID inlay. Use the RFID Printer Setup Utility to determine this value. The range is **-15** to **23** and the default is **-15**.

Example   **{I,Y,0,1,18,12,1,0,120,220,0,1,0,1,12 | }**

Uses the default non-stop encode mode, enables singulate mode (1), sets the write power to 18, the read power to 12, enables tag saver (1), does not set a TID position (0), sets the start encode position to 120, sets the encode zone to 220, does not set an encode position (0), uses the internal reader (1) to verify the RFID data, does not set a tag verify position (0), enables retry encode (1), and sets the verify power to 12.

Performance Considerations with RFID

Using the printer's internal or both RFID verification setting may decrease throughput.

Improving data accuracy may also decrease throughput. To improve data accuracy:

♦ enable TID *singulation*.

♦ use a *Lock Code* for the EPC data. See "Creating Gen2 RFID Data" for more information.

♦ select external for the RFID *tag verify.*

The printer has two RFID modes of operation: stop-to-encode and non-stop encode:

Stop-to-encode mode	Non-stop encode mode
♦ The printer pauses (or stops) while encoding the RFID inlay.	♦ The printer does **not** pause (or stop) while encoding the RFID inlay.
♦ The amount of programmable EPC data, user memory, access password, and lock code data varies by RFID chip.	♦ Some RFID chips encode faster than others.
♦ The printer errors if it cannot program all the RFID data within the allotted time.	♦ The faster the print speed, the less time the printer has to encode data.
♦ Stop-to-encode mode must be used for serialized EPC data. See "Using Option 65 (Serialized EPC Data)" for more information.	♦ The amount of programmable EPC data, user memory, access password, and lock code data varies by *print speed* and RFID chip.
	♦ The printer errors if it cannot program all the RFID data while the inlay is within the encode zone.

DEFINING FIELDS

This chapter provides a reference for defining

- ♦ the format header
- ♦ text and constant text fields
- ♦ non-printable text fields
- ♦ bar code fields
- ♦ line and box fields
- ♦ RFID fields
- ♦ bar code verifier fields

Defining the Format Header

A Format Header begins a format file.

Syntax **{F,format#,action,device,measure,length,width,"name" |**

F1. F	Format Header.	
F2. format#	Unique number from **1** to **999** to identify the format. **1** is the default.	
F3. action	Action. Enter **A** to add the format to the printer.	
F4. device	Format storage device. **R** is the default. Options:	

F   Flash (saved when the printer is turned off.) Format flash before use. See "Formatting Flash Memory" in Chapter 2 for more information.

R   Volatile RAM (deleted when the printer is turned off).

F5. measure   Unit of measure. **G** (dots) is the default.

E   (English - in 1/100 inches)
M   (Metric -in 1/10 mm)
G   (Graphic - in dots)

F6. length   Printable length of the supply in selected units. **600** is the default.

English	**32 – 1600**
Metric	**81 – 4064**
203 Dots	**65 – 3248**
300 Dots	**96 – 3600**

Note: The maximum length with optional 300 dpi is 12.0" (305 mm). The length may require adjustment for RFID supplies. See "RFID Format Design Tips" for more information.
Reallocate memory (increase image buffer) to print longer labels. See "Defining the Memory Configuration Packet" in Chapter 2 for more information.

F7. width   Print width, from left to right, in selected units. **Do not** include the liner (backing paper) in this dimension. **400** is the default.

English	**75 – 409**
Metric	**191 – 1038**
203 Dots	**152 – 832**
300 Dots	**225 – 1248**

F8. "name"   Format name (optional), **0** to **8** characters, enclose within quotation marks. **""** is the default. The range is **1** to **256** characters for the XML-enabled printer.

Example **{F,1,A,R,E,300,200,"TEXTILES" |**

Adds Format 1 ("TEXTILES") to the printer. It uses a three inch long by two inch wide label.

Defining Text Fields

Create a separate definition for each text field. If text falls on two lines, each line of text requires a separate definition.

Syntax T,field#,# of char,fix/var,row,column,gap,font,hgt mag,wid mag,color,alignment,char rot,field rot,sym set |

T1. T	Text Field.
T2. field#	Unique number from **1** to **999** to identify this field. **1** is the default.
T3. *# of char*	Maximum number of printed characters (**0** to **2710**) in the field. **30** is the default.
T4. *fix/var*	Fixed or variable length field. **V** is the default. Options:

F Fixed length
V Variable length

T5. row — Row location – distance from the bottom of print area to the pivot point.
10 is the default. For monospaced fonts, distance from bottom of print area to the pivot point. The pivot point varies depending on how text is justified..

For proportionally spaced fonts, distance from bottom of print area to baseline of characters in field.

| Left/Center/Right-Justified | Balanced | End-Justified |

English	**0 – 1599**
Metric	**0 – 4063**
203 Dots	**0 – 3247**
300 Dots	**0 – 3599**

T6. column — Column location - distance from the left edge of the print area to the pivot point to find the column location. **10** is the default.

English	**0 – 409**
Metric	**0 – 1039**
203 Dots	**0 – 831**
300 Dots	**0 – 1247**

T7. gap	Number of dots between characters in 203 dpi (or 300 dpi). Range: **0** to **99**. **0** is the default.
Note:	For monospaced fonts, the additional spacing is added to the existing inter-character gap. This is also true for proportionally spaced fonts; however, the inter-character gap varies with character combinations.

Any number other than 0 affects the field width. Default spacing:

Standard	**3 dots**
Reduced	**1 dot**
Bold	**3 dots**
OCRA-like	**3 dots**
HR1	**3 dots**
HR2	**3 dots**
CG Triumvirate™ Typeface Bold	**varies with each letter**
CG Triumvirate™ Typeface	**varies with each letter**
EFF Swiss Bold	**varies with each letter**

Note:	HR1 and HR2 are only used with the UPC bar code family and must be numeric.
T8. font	Style of font. **1** is the default. Options:

1	Standard	**10**	CG Triumvirate™ Typeface Bold
2	Reduced	**11**	CG Triumvirate™ Typeface
3	Bold	**15**	7 pt. CG Triumvirate™ Typeface
4	OCRA-like	**16**	9 pt. CG Triumvirate™ Typeface
5	HR1	**17**	11 pt. CG Triumvirate™ Typeface
6	HR2	**18**	15 pt. CG Triumvirate™ Typeface
		50	EFF Swiss Bold (TrueType® Scalable)
		56	PaxarSymbols

Or a valid downloaded font selector number.
Fonts 5 and 6 are for numeric data only. The CG Triumvirate™ typefaces support only the ANSI and DOS Code Page 437 and 850 Symbol Sets. The scalable font does not support Code Page 1256 (Arabic). See Appendix C for more information.

T9. hgt mag	Height magnifier, **1** to **7** times (**4** to **255** points for scalable/downloaded TrueType fonts). **1** is the default. Use a magnifier of 1 with proportionally spaced fonts, because characters lose smoothness at higher magnifications. See Appendix B, "Fonts," for more information about fonts.
T10. wid mag	Width magnifier, **1** to **7** times (**4** to **255** points for scalable/downloaded TrueType fonts). **1** is the default.Proportionally spaced fonts do not have a set width. To estimate the size of the field, use the letter "W" for the widest field or an "L" for an average width field. Find the selected font and the desired width in Appendix B, "Fonts."

T11. color	Field color overlay attributes. **B** is the default.

Transparent The overlay field (text or constant text) does not block out (or "erase") existing fields.

Opaque The overlay field blocks out (or "erases") existing fields.

Options for standard printer fonts:

B	Opaque, Normal, Black, Normal
D/R/W	Opaque, Normal, White, Normal
O	Transparent, Normal, Black, Normal

Options for scalable fonts:

A	Opaque, Normal, Black, Bold
B	Opaque, Normal, Black, Normal
E	Opaque, Italics, Black, Bold
F	Opaque, Italics, Black, Normal
N	Transparent, Normal, Black, Bold
O	Transparent, Normal, Black, Normal
S	Transparent, Italics, Black, Bold
T	Transparent, Italics, Black, Normal

Note:	Solid black print should not exceed 30% on a given square inch of the label, or the printhead life may be decreased.

Field placement in the packet is an important consideration when using field color attributes. If a line field is defined before the overlay (text or constant text) field, the line field is blocked out by the overlay field, depending on the overlay field's color attribute. If a line field is defined after the overlay field, the line field is not blocked out by the overlay field, regardless of the overlay field's color attribute.

T12. alignment	Alignment of text in the field. **L** is the default. Options:

L	Align on left side of field
C	Center text within field (monospaced fonts only)
R	Align on right side of field (monospaced fonts only)
B	Align at midpoint of field
E	Align at endpoint of the field

Use **L**, **B**, or **E** for any font.

The red dot indicates the field origin and the line indicates the column position for each field in the following graphic.

T13. char rot   Character rotation. **0** is the default. The field or supply does not rotate, only the characters do. Options:
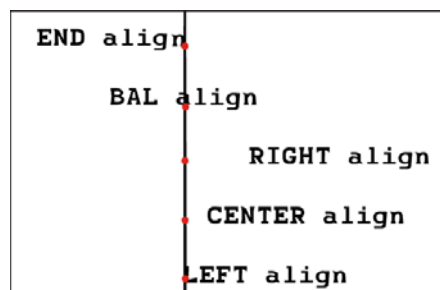
0 Top of character points to top of field
1 Top of character points to left of field
2 Top of character points to bottom of field
3 Top of character points to right of field

MONARCH	MONARCH	MONARCH	MONARCH
ABCD	⊏⊞⊂⊐	∀BCD	⊐⊂⊞⊏
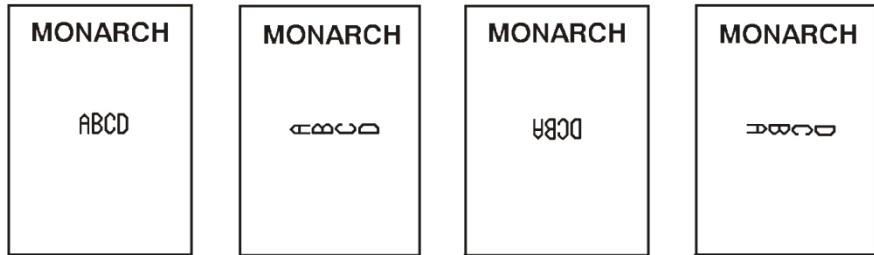
Note: Font 50 and downloaded TrueType fonts do not support character rotation.

T14. field rot   Field rotation. **0** is the default. Field rotation rotates the whole field, not just the characters. Rotation is affected by the pivot point, which varies depending on how text is justified. Lower left corner of field is the pivot point. Options:

0 Top of field points to top of supply
1 Top of field points to left of supply
2 Top of field points to bottom of supply
3 Top of field points to right of supply

MONARCH	MONARCH	MONARCH	MONARCH
Field	Field	Field	Field
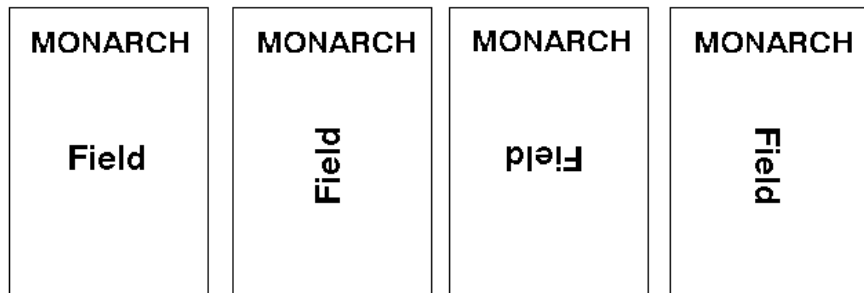
| *T15. sym set* | Symbol set. **0** is the default (Internal Symbol Set). |
| | For scalable or TrueType® fonts, use: |

1	ANSI Symbol Set
100	Macintosh
101	Wingdings
102	Unicode (user input) for particular mapping
103	BIG5 (user input) for Unicode mapping
104	GB2312 (user input) for Unicode mapping
105	SJIS (user input) for Unicode mapping (CP 932, Japanese)
106	GB2312 (user input) for GB2312 mapping (CP 936, Simplified Chinese)
107	BIG5 (user input) for BIG5 mapping (CP 950, Traditional Chinese)
108	KSC (user input) for KSC mapping (CP 5601 Korean)
110	Unicode UTF-8
437	DOS Code Page 437 (Domestic)
850	DOS Code Page 850 (International)
852	DOS Code Page 852 (Latin 2)
855	DOS Code Page 855 (Russian)
857	DOS Code Page 857 (IBM Turkish)
860	DOS Code Page 860 (MS-DOS Portuguese)
1250	Code Page 1250 (Latin 2)
1251	Code Page 1251 (Cyrillic)
1252	Code Page 1252 (Latin 1)
1253	Code Page 1253 (Greek)
1254	Code Page 1254 (Turkish)
1255	Code Page 1255 (Hebrew)
1256	Code Page 1256 (Arabic)
1257	Code Page 1257 (Baltic)
1258	Code Page 1258 (Vietnam)

Note: The Standard, Reduced, Bold, OCRA and HR fonts only support the Internal Symbol Set (0). The CG Triumvirate™ typefaces only support the ANSI (1) and DOS Code Page 437 (2) and 850 (3) Symbol Sets. The scalable font (font#50) does not support Code Page 1256 Arabic (10).

Code pages 852-860 and 1250-1258 are for downloaded TrueType fonts or the scalable font. Code pages 102-110 require a downloaded International TrueType font. TrueType fonts are designed to be regionally specific; therefore, all code pages may not be supported in a given font. See Appendix C, "Symbol Sets/ Code Pages" for more information.

If using symbol set 110 (Unicode UTF-8), set the MPCL control characters (start of header, etc.) to decimal values between **0** and **128**; otherwise, errors may occur with the Unicode data entered. See Chapter 2, "Configuring the Printer" for more information about the MPCL control characters.

Example **T,2,10,V,250,80,0,1,1,1,B,C,0,0,0 |**

Defines a text field (field #2) with a variable length of up to 10 characters. The field begins at row 250, column 80. There is no additional gap between characters, and the Standard font is used without any additional magnification. The printing is black on white and centered. No field or character rotation is used. The internal symbol set is used.

Defining Bar Code Fields

Each bar code field requires a separate definition.

The printer can shift some bar code fields to avoid bad dots on the printhead.

Shiftable bar codes include: Codabar (NW7), Code 128, Code 39 (w/ or w/o CD) or MOD43, Code 93, EAN13, EAN13+Price CD, Interleaved 2 of 5/Interleaved 2 of 5 with Barrier Bar, MSI, UPCA, UPCA+Price CD, and UPCE.

The printer **cannot** shift the following bar codes: Aztec, Code 16K, Data Matrix, EAN13+2, EAN13+5, EAN8+2, EAN8+5, GS1 DataBar, Intelligent Mail, MaxiCode, PDF417, POSTNET, Quick Response (QR Code), UPCA+2, UPCA+5, UPCE+2, or UPC+5.

Syntax **B,field#,# of char,fix/var,row,column,font,density,height, text,alignment,field rot,type,sep_height,segment |**

B1. B	Bar Code Field.
B2. field#	Unique number from **1** to **999** to identify this field. **1** is the default.
B3. # of char	Maximum number of characters. If the bar code uses a check digit, allow an extra character for the check digit. The actual maximum number of characters is limited by the size of the label and bar code density. Range: **0** to **2710**. **30** is the default.

Bar Code	Number of Characters
Aztec	0 - 2710
Codabar (NW7)	0 – 2710
Code 16K	0 – 2710
Code 39 (w/ or w/o CD) or MOD43	0 – 2710
Code 93	0 – 2710
Code 128	0 – 2710
Data Matrix	0 – 2710 numeric; 0 – 2335 alphanumeric
EAN8	8
EAN8+2	10
EAN8+5	13
EAN13	13
EAN13+2	15
EAN13+5	18
EAN13+Price CD	13

Bar Code	Number of Characters
GS1 DataBar Bar Code Type (B13)	
1 - GS1 DataBar 14	13 - no check digit input
2 - GS1 DataBar 14 Truncated	13 - no check digit input
3 - GS1 DataBar 14 Stacked	13 - no check digit input
4 - GS1 DataBar 14 Stacked Omni directional	13 - no check digit input
5 - GS1 DataBar Limited	13 - no check digit input
6 - GS1 DataBar Expanded	Refer to the *GS1 General Specification* for more information.
7 - UPCA	11 - no check digit input
8 - UPCE	10 - no check digit input
9 - EAN13	12 - no check digit input
10 - EAN8	7 - no check digit input
11 - UCC/EAN128 and CC A/B	If not enough characters are entered, the bar code pads to the left with zeros. If too many characters are entered, unpredictable results may occur.
12 - UCC/EAN128 and CC C	
If FNC1 (function 1) is supported, use the pound sign (#) in the batch data to invoke it.	
Intelligent Mail	31
Refer to the *USPS® Intelligent Mail® Specification* for more information.	
Interleaved 2 of 5 or Interleaved I 2 of 5 with Barrier Bar	0 - 2710
MaxiCode*	0 to 93 (alphanumeric) 0 to 128 (numeric)
MSI	0 – 14
PDF417	0 – 2710
POSTNET™	9 or 11

Quick Response (QR) Code Data Type	Model 1	Model 2
Numeric Data	1167	2710
Alphanumeric data	707	2710
8-byte data	486	2710
Kanji data	299	1817

Note: The maximum number of characters depends on the selected level of error correction. When increasing the error correction level, the maximum number of characters decreases. See "Entering Batch Data for QR Code" in Chapter 6 to enter data for the QR Code.

Bar Code	Number of Characters
UPCA	12
UPCA+2	14
UPCA+5	17
UPCA+Price CD	12
UPCE	7
UPCE+2	9
UPCE+5	12

Note: For more information about two-dimensional bar codes, see Appendix A, "Samples."

B4. fix/var   Fixed (F) or variable (V) length field.

Bar Code	Fixed or Variable	Bar Code	Fixed or Variable
Aztec*	Fixed or Variable	Intelligent Mail®*	Fixed
Codabar (NW7)	Fixed or Variable	Interleaved 2 of 5 or Interleaved I 2 of 5 with Barrier Bar	Fixed or Variable
Code 16K	Variable	MaxiCode*	Fixed or Variable
Code 39 (w/ or w/o CD) or MOD43	Fixed or Variable	MSI	Fixed or Variable
Code 93	Variable	PDF417	Fixed or Variable
Code 128	Fixed or Variable	POSTNET™	Fixed
Data Matrix*	Variable	Quick Response*	Variable
EAN8	Fixed	UPCA	Fixed
EAN8+2	Fixed	UPCA+2	Fixed
EAN8+5	Fixed	UPCA+5	Fixed
EAN13	Fixed	UPCA+Price CD	Fixed
EAN13+2	Fixed	UPCE	Fixed
EAN13+5	Fixed	UPCE+2	Fixed
EAN13+Price CD	Fixed	UPCE+5	Fixed
GS1 DataBar	Fixed or Variable		

*   For more information about the two-dimensional bar codes, see Appendix A, "Samples."

| B5. row | Row location -distance from bottom of the print area to the pivot point of the field. The pivot point varies, depending on how the field is justified. **10** is the default. Remember to include text or numbers that may appear with the bar code for the row measurement. Pivot points |

Left/Center/Right-Justified Fields

Balanced Fields   **End-Justified Fields**

English	**0 – 1599**	
Metric	**0 – 4063**	
203 Dots	**0 – 3247**	
300 Dots	**0 – 3599**	

| B6. column | Column location - distance from the lower left edge of the print area to the pivot point. **10** is the default. |

English	**0 – 409**
Metric	**0 – 1039**
203 Dots	**0 – 831**
300 Dots	**0 – 1247**

Allow a minimum of 1/10 inch between the scan edge of bar code and label edges or other data. If using the optional bar code verifier, allow a minimum of 1.3 inches (33 mm) between the bar code and the top of the label.

| B7. font | Bar code. **4** is the default. Options: |

1	UPCA	**11**	UPCA +5	**31**	Code 16K
2	UPCE	**12**	UPCE +2	**32**	PDF417
3	Interleaved 2 of 5	**13**	UPCE +5	**33**	MaxiCode
4	Code 39 (no check digit)	**14**	EAN8 +2	**35**	Data Matrix (ECC-200)
5	Codabar	**15**	EAN8 +5	**36**	Quick Response
6	EAN8	**16**	EAN13 +2	**37**	Aztec
7	EAN13	**17**	EAN13 +5	**38**	GS1 DataBar
8	Code 128	**22**	POSTNET	**40**	Code 39 (MOD 43 check digit)
9	MSI	**23**	Code 93	**41**	UPCA & Price CD
10	UPCA +2	**24**	Intelligent Mail	**44**	EAN13 & Price CD
				50	Interleaved 2 of 5 with Barrier Bar

| B8. density | Bar code density. The default varies by bar code type. Use the following tables. |

203 DPI Bar Code Densities

Bar Code Type	Density Selector	Density (% or cpi)	Narrow Element (dots/mils)	Narrow to Wide Ratio	Data Length	Appearance Codes	Char Set
UPCA +2/+5 Price CD	**2** 4	76% 114%	2/9.9 3/14.8	N/A	11 or 12 14/17	1, 5, 6, 7 or 8	0 to 9
UPCE+2/+5	**2** 4	76% 114%	2/9.9 3/14.8	N/A	6 or 7 9/12	1, 5, 6, 7 or 8	0 to 9
EAN8+2/+5	**2** 4	76% 114%	2/9.9 3/14.8	N/A	7 or 8 10/13	1, 5, 6, 7 or 8	0 to 9
EAN13+2/+5 Price CD	**2** 4	76% 114%	2/9.9 3/14.8	N/A	12 or 13 15/18	1, 5, 6, 7 or 8	0 to 9
Interleaved2 of 5or I2of5with Barrier Bar	1 2 3 4 5 6 7 8 9 10 11 **12** 13	1.1 2.1 3.2 4.2 5.6 6.3 7.5 8.8 9.6 11.2 11.0 12.7 14.5	21/103.4 12/59.1 7/34.5 6/29.6 4/19.7 4/19.7 3/14.8 3/14.8 3/14.8 2/9.9 2/9.9 2/9.9 2/9.9	1:3.0 1:2.5 1:3.0 1:2.5 1:3.0 1:2.5 1:3.0 1:2.3 1:2.0 1:3.0 1:3.0 1:2.5 1:2.0	0 to 2710	8	0 to 9
(Code 39 or MOD43 (Extended Code 39)	1 2 3 4 6 **7** 11 12 20	1.4 1.7 3.5 4.2 6.3 7.0 3.9 12.7 3.0	10/49.3 8/39.4 4/19.7 3/14.8 2/9.9 2/9.9 4/19.7 1/4.9 5/24.6	1:2.5 1:2.5 1:2.5 1:3.0 1:3.0 1:2.5 1:2.0 1:3.0 1:2.2	0 to 2710	8	SPACE $%*+-./ 0 to 9 A to Z
Codabar (NW7)	2 3 4 5 7 **8** 9	2.1 3.0 4.6 5.1 8.4 9.2 10.1	8/39.4 6/29.6 4/19.7 4/19.7 2/9.9 2/9.9 2/9.9	1:3.0 1:2.5 1:2.5 1:2.0 1:3.0 1:2.5 1:2.0	0 to 26	8	$+-./ 0 to 9 a to d
Code 128 or Code 16K	20 4 6 **8**	3.5/7.0 4.4/8.7 5.8/11.7 8.7/17.5	5/24.6 4/19.7 3/14.8 2/9.9	N/A	0 to 2710	8	00H to 7FH

Note: The start (*) and stop (+) characters are automatically added for Code 39. Code 93, density 12, produces a one-dot narrow bar. This density is intended for special U.S.P.S. ACT-tag applications only. Synthetic supplies are recommended to produce scannable bar codes.

Bar Code Type	Density Selector	Density (% or cpi)	Narrow Element (dots/mils)	Narrow to Wide Ratio	Data Length	Appearance Codes	Char Set
CODE 93	3 4 5 **7** 10	3.7 4.5 5.6 7.5 11.2	6/29.6 5/24.6 4/19.7 3/14.8 2/9.9	N/A	0 to 2710	8	00H to 7FH
MSI	4 5 **7**	4.2 5.6 7.2	4/19.7 3/14.8 2/9.9	1:2.0 1:2.0 1:2.5	0 to 14	8	0 to 9
POSTNET	0 (fixed at 4.3 cpi)	24/118.2	10/49.3	4/19.7 (5 dot gap)	0,5,6,9 or 11	8	0 to 9
Intelligent Mail	0 (fixed at 4.3 cpi)	24/118.2	10/49.3	4/19.7 (5 dot gap)	0,5,6,9 or 11	8	0 to 9
MaxiCode	**7**	N/A	N/A	N/A	99	8	00H to FFH

Bar Code Type	Density Selector	Element Width (dot/mils)	Row Height (dots/mils)	Aspect Ratio	Data Length	Appearance Codes	Char Set
PDF417	1 2 3 4 5 **6** 7 8 9	2/9.8 2/9.8 2/9.8 3/14.8 3/14.8 3/14.8 4/19.7 4/19.7 4/19.7	2/9.8 4/19.7 6/29.6 3/14.8 6/29.6 9/44.3 4/19.7 8/39.4 12/59.1	1:1 1:2 1:3 1:1 1:2 1:3 1:1 1:2 1:3	0 to 2709	8	00H toFFH

203/300 DPI Bar Code Densities

Bar Code Type	Density Selector	Data Length
Quick Response (QR Code) Models 1 and 2	**0**	Model 1: 0 - 1167 Numeric;0 - 707 Alphanumeric; 0 - 486 (8-bit); 0 - 299 (Kanji) Model 2: 0 - 2710 Numeric, Alphanumeric and 8-bit; 0 - 1817 Kanji

Note: Values in bold indicate the default.

203/300 DPI Bar Code Densities

Bar Code	Size Row x Col.	Density Selector	Max. Data Length Num. X Alphanum.	Appearance Codes	Char Set
Data Matrix Square symbols	10 x 10	1	6 x 3	8	00H to FFH
	12 x 12	2	10 x 6		
	14 x 14	3	16 x 10		
	16 x 16	4	24 x 16		
	18 x 18	5	36 x 25		
	20 x 20	6	44 x 31		
	22 x 22	7	60 x 43		
	24 x 24	8	72 x 52		
	26 x 26	9	88 x 64		
	32 x 32	10	124 x 91		
	36 x 36	11	172 x 127		
	40 x 40	12	228 x 169		
	44 x 44	13	288 x 214		
	48 x 48	14	348 x 259		
	52 x 52	15	408 x 304		
	64 x 64	16	560 x 418		
	72 x 72	17	736 x 550		
	80 x 80	18	912 x 682		
	88 x 88	19	1152 x 862		
	96 x 96	20	1392 x 1042		
	104 x 104	21	1632 x 1222		
	120 x 120	22	2100 x 1573		
	132 x 132	23	2608 x 1954		
	144 x 144	24	2710 x 2335		
		0 default (bar code size automatically determined by data)			

Bar Code	Size Row x Col.	Density Selector	Max. Data Length Num. X Alphanum.	Appearance Codes	Char Set
Data Matrix Rectangular symbols	8 x 18	25	10 x 6	**8**	00H to FFH
	8 x 32	26	20 x 13		
	12 x 26	27	32 x 22		
	12 x 36	28	44 x 31		
	16 x 36	29	64 x 46		
	16 x 48	30	98 x 72		
		0 default (bar code size automatically determined by data)			

Note: The printers support printing a Data Matrix symbol with an X-dimension of 13 mils or greater (3 dots @ 203 dpi). If using a denser bar code, make sure the bar code scans in that particular application. Avery Dennison "premium" supplies and increasing the print contrast are recommended for denser bar codes. Depending on the application, additional densities are available.

203 DPI Bar Code Densities

Bar Code Type	Density Selector	Narrow Element (dots/mils)	Data Length	Char Set
GS1 DataBar	**2** 3 4 5 6 7 8	2/9.9 3/14.8 4/19.7 5/24.6 6/29.6 7/34.5 8/39.4	0 to 2710	00H to FFH
Aztec	**2** 3 4 5 6 7 8 9 10 11 12 13 14 15	0.0099 0.0148 0.0197 0.0247 0.0296 0.0345 0.0394 0.0444 0.0493 0.0542 0.0592 0.0641 0.0690 0.0740	0 to 2710	00H to FFH

Note: Values in bold indicate the default.

300 DPI Bar Code Densities

Bar Code Type	Density Selector	Density (% or cpi)	Narrow Element (dots/mils)	Narrow to Wide Ratio	Data Length	Appearance Codes	Char Set
UPCA +2/+5 Price CD	**2** 4	77% 103%	3/10 4/13.3	N/A	11 or 12 14/17	1, 5, 6, 7 or 8	0 to 9
UPCE+2/+5	**2** 4	77% 103%	3/10 4/13.3	N/A	6 or 7 9/12	1, 5, 6, 7 or 8	0 to 9
EAN8+2/+5	**2** 4	77% 103%	3/10 4/13.3	N/A	7 or 8 10/13	1, 5, 6, 7 or 8	0 to 9
EAN13+2/+5 Price CD	**2** 4	77% 103%	3/10 4/13.3	N/A	12 or 13 15/18	1, 5, 6, 7 or 8	0 to 9
Interleaved2 of 5 or I2of5 with Barrier Bar	1 2 3 4 5 6 7 8 9 10 11 **12** 13	1.1 2.1 3.3 4.2 5.6 6.2 8.3 9.4 9.9 11.1 11.1 13.0 14.3	31/103.4 18/60.1 10/33.4 9/30.0 6/20.0 6/20.0 4/13.3 4/13.3 4/13.3 3/10 3/10 3/10 3/10	1:3.0 1:2.5 1:3.0 1:2.4 1:3.0 1:2.5 1:3.0 1:2.5 1:2.3 1:3.0 1:3.0 1:2.3 1:2.0	0 to 2710	8	0 to 9
(Code 39 or MOD43 (Extended Code 39)	1 2 3 4 6 **7** 11 12 20	1.4 1.7 3.4 4.7 6.2 7.1 3.8 11.5 3.1	15/50.0 12/40.0 6/20.0 4/13.3 3/10.0 3/10.0 6/20.0 2/6.7 7/23.4	1:2.5 1:2.5 1:2.5 1:3.0 1:3.0 1:2.5 1:2.0 1:3.0 1:2.2	0 to 2710	8	SPACE $%*+-./ 0 to 9 A to Z
Codabar (NW7)	2 3 4 5 7 **8** 9	2.3 3.4 5.0 5.6 9.1 10.4 11.1	12/40.0 9/30.0 6/20.0 6/20.0 3/10.0 3/10.0 3/10.0	1:3.0 1:2.5 1:2.5 1:2.0 1:3.0 1:2.5 1:2.0	0 to 26	8	$+-./ 0 to 9 a to d
Code 128 or Code 16K	4 6 **8** 20	4.5 6.8 9.1 3.9	6/20.0 4/13.3 3/10.0 7/23.4	N/A	0 to 2710	8	00H to 7FH

Note: The start (*) and stop (+) characters are automatically added for Code 39. Code 93, density 12, produces a one-dot narrow bar. This density is intended for special U.S.P.S. ACT-tag applications only. Synthetic supplies are recommended to produce scannable bar codes.

300 DPI Bar Code Densities

Bar Code Type	Density Selector	Density (% or cpi)	Narrow Element (dots/mils)	Narrow to Wide Ratio	Data Length	Appearance Codes	Char Set
CODE 93	3 4 5 **7** 10	3.7 4.8 5.6 8.3 11.1	9/30.0 7/23.4 6/20.0 4/13.3 3/10.0	N/A	0 to 2710	8	00H to 7FH
MSI	4 5 **7**	4.2 6.2 7.5	6/20.0 4/13.3 3/10.0	1:2.0 1:2.0 1:2.3	0 to 2710	8	0 to 9
POSTNET	0 (fixed at 4.3 cpi)	24/118.2	10/49.3	6/20.0 (5 dot gap)	0, 5, 6, 9 or 11	8	0 to 9
Intelligent Mail	0 (fixed at 4.3 cpi)	24/118.2	10/49.3	4/19.7 (5 dot gap)	0,5,6,9 or 11	8	0 to 9
MaxiCode	**7**	N/A	N/A	N/A	99	8	00H to FFH

Bar Code Type	Density Selector	Element Width (dot/mils)	Row Height (dots/mils)	Aspect Ratio	Data Length	Appearance Codes	Char Set
PDF417	1 2 3 4 5 **6** 7 8 9	3/10.0 3/10.0 3/10.0 4/13.3 4/13.3 4/13.3 6/20.0 6/20.0 6/20.0	3/10.0 6/20.0 9/30.0 4/13.3 9/30.0 12/40.0 6/20.0 12/40.0 18/60.0	1:1 1:2 1:3 1:1 1:2 1:3 1:1 1:2 1:3	0 to 2709	8	00H to FFH

300 DPI Bar Code Densities

Bar Code Type	Density Selector	Narrow Element (dots/mils)	Data Length	Char Set
GS1 DataBar	**2**	3/9.8	0 to 2710	00H to FFH
	3	4/13.1		
	4	5/16.4		
	5	7/23		
	6	8/26.2		
	7	10/32.8		
	8	11/36.1		
Aztec	**2**	0.0100	0 to 2710	00H to FFH
	3	0.0133		
	4	0.0167		
	5	0.0233		
	6	0.0266		
	7	0.0333		
	8	0.0366		
	9	0.0433		
	10	0.0466		
	11	0.0533		
	12	0.0566		
	13	0.0633		
	14	0.0666		
	15	0.0733		

Note: Values in bold indicate the default.

B9. height   Bar code height, in 1/100 inches, 1/10 mm, or dots. Minimum values:

English	**19**
Metric	**48**
203 Dots	**38**
300 Dots	**57**

Height	Bar Code Type	Description
0	Aztec Code 16K Intelligent Mail PDF417 PostNet MaxiCode	These bar codes have a fixed height. Use **0**.
0	Data Matrix QR Code	The printer determines the size of the symbol, but the user's scanner determines the functional size (minimum height) of the symbol. Small Data Matrix and QR Codes may not be scannable. The Data Matrix and QR Code's height depends on the number entered for this parameter. For example, 80 means the symbol could be smaller than 0.80," but it will not be greater than the amount specified in this parameter. The symbol arranges the data according to rows and columns within the specified height.
0	GS1 DataBar	The printer determines the height of the symbol based on the density. Use **0** for all GS1 DataBar *types*.
20	Code 93	**20** is the default for Code 93.
40	EAN/UPCA/UPCE I 2of5 Codabar Code 39 Code 128 MSI	**40** is the default for UPCA/UPCE/EAN, I 2of5, Codabar, Code 39, Code 128, and MSI.

B10. text   Appearance of text with bar code. Options:

Text	Bar Code Type	Description	Sample
0	MaxiCode QR Code EAN/UPC	MaxiCode Mode 0 (obsolete) QR Code Model 2 default	0 28028 11111 9
1	Code 39 EAN/UPC QR Code	No check digit or number system No check digit or number system QR Code Model 1	28028 11111 *02802811*
2	MaxiCode QR Code	MaxiCode Mode 2 (Numeric Postal Code) QR Code Model 2	
3	MaxiCode	MaxiCode Mode 3 (Alphanumeric Postal Code)	
5	EAN/UPC	Number system at bottom, no check digit	0 28028 11111
6	EAN/UPC	Check digit at bottom, no number system	28028 11111 9
7	EAN/UPC	Check digit and number system at bottom (default)	0 28028 11111 9
8	MaxiCode All other bar codes	MaxiCode (auto detect modes 0, 2, 3, or for compressed data) default No text, bar code only (default)	

B11. alignment   Choose **L, R, C, B** or **E** to align the bar code data correctly in the field. **L** is the default.

Alignment	Bar Code Type	Description
L	All Aztec, Data Matrix, GS1 DataBar, Intelligent Mail, MaxiCode, QR Code (Quick Response)	Align on left side of field. Must use **L** for these bar codes.
C	Code 39, Codabar, I 2of5, MSI	Center within field.
R	Code 39, Codabar, I 2of5, MSI	Align on right side of field.
B	All except where noted.	Align at midpoint of field; centers variable width bar codes, which may not allow pad-character centering.
E	All except where noted.	Align at endpoint of the field; right justifies variable-width bar codes.

The red dot indicates the field origin and the line indicates the column position for each field in the following graphic.
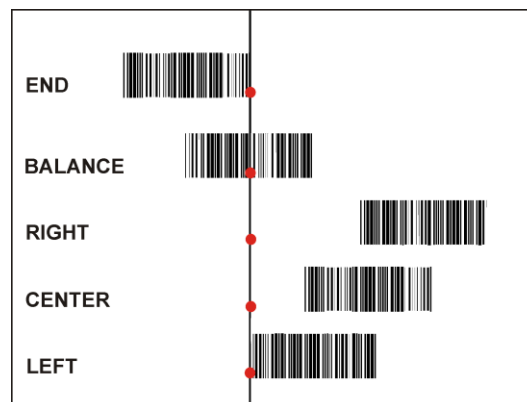
B12. field rot	Field rotation. Field rotation rotates the whole field, not just the characters. Rotation is affected by the pivot point, which varies depending on how text is justified. Lower left corner of field is the pivot point. **0** is the default. Options:

0   Top of field points to top of supply (use for MaxiCode)
1   Top of field points to left of supply
2   Top of field points to bottom of supply
3   Top of field points to right of supply

Note: Serial bar codes with an 8-dot narrow element do not automatically print at 2.5 IPS. Serial bar codes printed at speeds greater than 2.5 IPS may not scan properly.

Left/Center/Right-Justified   **Balanced**   **End-Justified**
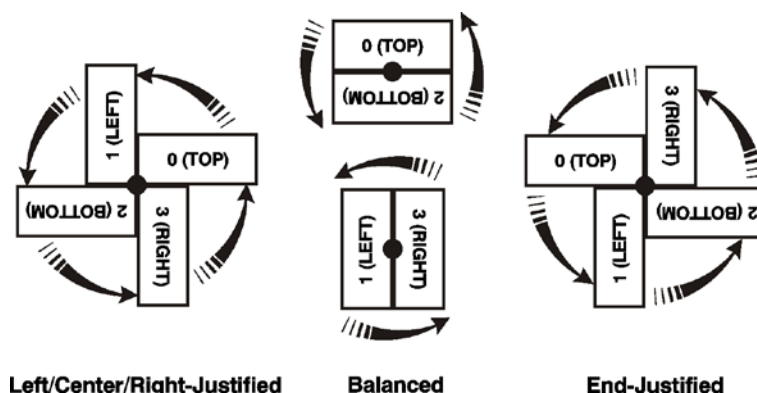
B13. type   Select from the bar code family. *This parameter only applies to the GS1 DataBar bar code.* **1** is the default. For other bar codes, do not include this parameter. Options:

1   GS1 DataBar 14
2   GS1 DataBar 14 Truncated
3   GS1 DataBar 14 Stacked
4   GS1 DataBar 14 Stacked Omni directional
5   GS1 DataBar Limited
6   GS1 DataBar Expanded
7   UPCA
8   UPCE
9   EAN13
10   EAN8
11   UCC/EAN128 and CC A/B
12   UCC/EAN128 and CC C

B14. sep_height   Height of the separator between the linear bar code and 2D bar code. *This parameter only applies to the GS1 DataBar bar code.* For other bar codes, do not include this parameter. The value is **1** or **2**. **1** is the default.

B15. segment   Width of the segment. *This parameter only applies to the GS1 DataBar bar code.* For other bar codes, do not include this parameter. The range is even numbers from **2** to **22**. **22** is the default.

Example   **B,3,12,V,50,40,1,2,80,7,L,0 |**

Defines a bar code field (field #3) with 12 characters of variable length starting at row 150, column 40. A UPCA bar code with a density of 2 and a height of 80 is used. The check digit and number system are shown at the bottom. The bar code is left aligned without any field rotation.

Example   **B,1,30,V,5,5,38,4,0,0,L,0,1,2,22 |**

Defines a bar code field (field #3) with up to 30 characters of variable length starting at row 5, column 5. The GS1 DataBar uses a density of 4. No text is shown with the bar code. The bar code is left-aligned with no field rotation.

Defining Non-Printable Text Fields

Non-printable text fields hold entered data without printing it in its entered form. Typically, non-printable fields hold data that later combines with other fields to form a merged field. Define non-printable text fields before defining the field where the information prints.

When copying this field into another field, the maximum number of characters for the final field is **2710**.   Allow only as many characters as needed, because extra characters use up space. Also, if copying into a bar code field, the maximum number of characters in the destination bar code is determined by the bar code specification (UPCA-12, EAN-13, etc.). Apply field options to manipulate the text entered in this field. Copying data from this field into another field is an example. See "Using Option 4 Copy Data" in Chapter 4 for more information.

In the following example, data is entered into four non-printable fields and merged to form field 5, and is then printed as a bar code. See "Merging Fields" in Chapter 4 for more information.

Field	Data	Field Type
1	20374	Non-printable
2	339	Non-printable
3	8	Non-printable
4	15	Non-printable
5	20374339815	Bar Code

Each non-printable text field requires a separate definition.

Syntax   **D,field#,# of char |**

D1. D	Non-Printable Text Field.
D2. field#	Unique number from **0** to **999** assigned to this field. **1** is the default.
D3. # of char	Maximum number of characters in this field: **0** to **2710**. **30** is the default.

Example   **D,4,20 |**

Defines a non-printable text field (field #4) with a maximum of 20 characters.

Defining Constant Text Fields

A constant text field is a set of fixed characters that prints on all labels. Define each constant text field separately. This field is not assigned a field number, but is counted as a field (keep this in mind, as the printer allows a maximum of **1000** fields per format). The characters in this field cannot be changed by batch data. Field options do not apply to constant text fields. Mark the pivot point of the field, which varies depending on the fields' justification.

Syntax   **C,row,column,gap,font,hgt mag,wid mag,color,alignment, char rot,field rot,"fixed char",sym set |**

C1. C	Constant Text Field.	
C2. row	Row location – distance from the bottom of print area to the pivot point. **10** is the default. For monospaced fonts, distance from bottom of print area to the pivot point. For proportionally spaced fonts, distance from bottom of print area to baseline of characters in the field. (Bottom exits the printer first.)	

English	**0 – 1599**
Metric	**0 – 4063**
203 Dots	**0 – 3247**
300 Dots	**0 – 3599**

C3. column   Column location - distance from the left edge of the print area to the pivot point to find the column location. **10** is the default.

English	**0 – 409**
Metric	**0 – 1039**
203 Dots	**0 – 831**
300 Dots	**0 – 1247**

C4. gap   Number of dots between characters (203 dots per inch). Range: **0** to **99**. **0** is the default.

Any number other than 0 affects the field width.
Default spacing:

Standard	**3 dots**
Reduced	**1 dot**
Bold	**3 dots**
OCRA-like	**3 dots**
HR1	**3 dots**
HR2	**3 dots**
CG Triumvirate™ Typeface Bold	**varies with each letter**
CG Triumvirate™ Typeface	**varies with each letter**
EFF Swiss Bold	**varies with each letter**

Note: HR1 and HR2 are only used with the UPC bar code family and must be numeric.

C5. font   Style of font. **1** is the default. Options:

1	Standard	**10**	CG Triumvirate™ Typeface Bold
2	Reduced	**11**	CG Triumvirate™ Typeface
3	Bold	**15**	7 pt. CG Triumvirate™ Typeface
4	OCRA-like	**16**	9 pt. CG Triumvirate™ Typeface
5	HR1	**17**	11 pt. CG Triumvirate™ Typeface
6	HR2	**18**	15 pt. CG Triumvirate™ Typeface
		50	EFF Swiss Bold (TrueType® Scalable)
		56	PaxarSymbols

	Or a valid downloaded font selector number.
	Fonts 5 and 6 are for numeric data only. The CG Triumvirate™ typefaces support only the ANSI and DOS Code Page 437 and 850 Symbol Sets. The scalable font does not support Code Page 1256 (Arabic). See Appendix C for more information.
C6. hgt mag	Height magnifier, **1** to **7** times (**4** to **255** points for scalable/downloaded TrueType fonts). **1** is the default. Use a magnifier of 1 with proportionally spaced fonts, because characters lose smoothness at higher magnifications. See Appendix B, "Fonts," for more information about fonts.
C7. wid mag	Width magnifier, **1** to **7** times (**4** to **255** points for scalable/downloaded TrueType fonts). **1** is the default. Proportionally spaced fonts do not have a set width. To estimate the size of the field, use the letter "W" for the widest field or an "L" for an average width field. Find the selected font and the desired width in Appendix B, "Fonts."
C8. color	There are two types of field color overlay attributes. **B** is the default.

Transparent	The overlay field (text or constant text) does not block out (or "erase") existing fields.
Opaque	The overlay field blocks out (or "erases") existing fields.

Options for standard printer fonts:

B	Opaque, Normal, Black, Normal
D/R/W	Opaque, Normal, White, Norma
O	Transparent, Normal, Black, Normal

Options for scalable fonts:

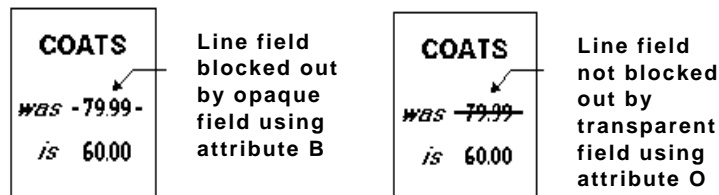| **A**	Opaque, Normal, Black, Bold
B	Opaque, Normal, Black, Normal
E	Opaque, Italics, Black, Bold
F	Opaque, Italics, Black, Normal
N	Transparent, Normal, Black, Bold
O	Transparent, Normal, Black, Normal
S	Transparent, Italics, Black, Bold
T	Transparent, Italics, Black, Normal

Note: Solid black print should not exceed 30% on a given square inch of the label, or the printhead life may be decreased.

Field placement in the packet is an important consideration when using field color attributes. If a line field is defined before the overlay (text or constant text) field, the line field is blocked out by the overlay field, depending on the overlay field's color attribute. If a line field is defined after the overlay field, the line field is not blocked out by the overlay field, regardless of the overlay field's color attribute.
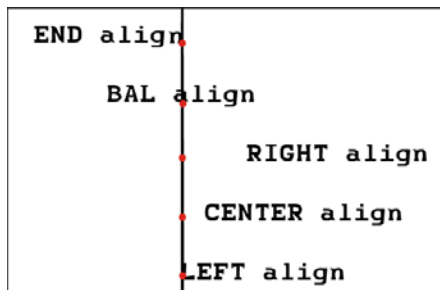
| *C9. alignment* | Alignment of constant text in the field. **L** is the default. Options: |

L   Align on left side of field (default)
C   Center text within field (monospaced fonts only)
R   Align on right side of field (monospaced fonts only)
B   Align at midpoint of field
E   Align at endpoint of the field
Use **L**, **B**, or **E** for any font.

The red dot indicates the field origin and the line indicates the column position for each field in the following graphic.

```
END align•
     BAL •lign
               • RIGHT align
          • CENTER align
          •EFT align
```

| *C10. char rot* | Character rotation. **0** is the default. The field or supply does not rotate, only the characters do. Options: |

0   Top of character points to top of field
1   Top of character points to left of field
2   Top of character points to bottom of field
**3**   Top of character points to right of field

MONARCH	MONARCH	MONARCH	MONARCH
ABCD	⊏⊞⊂⊃	�892∀	⊃⊞⊂⊏
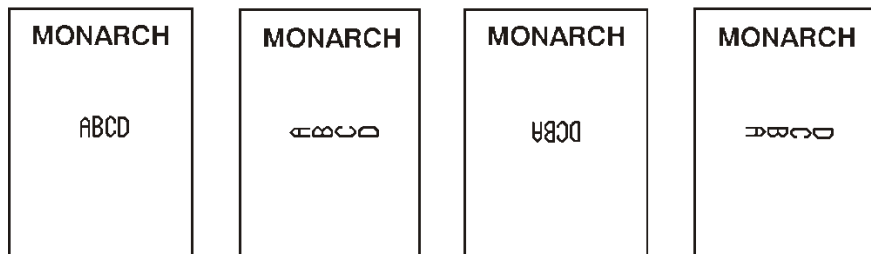
| **Note:** | Font 50 and downloaded TrueType fonts do not support character rotation. |
| *C11. field rot* | Field rotation. **0** is the default. Field rotation rotates the whole field, not just the characters. Rotation is affected by the pivot point, which varies depending on how text is justified. Lower left corner of field is the pivot point. Options: |

0   Top of field points to top of supply
1   Top of field points to left of supply
2   Top of field points to bottom of supply
3   Top of field points to right of supply

MONARCH	MONARCH	MONARCH	MONARCH
Field	Field	pləiℲ	Field
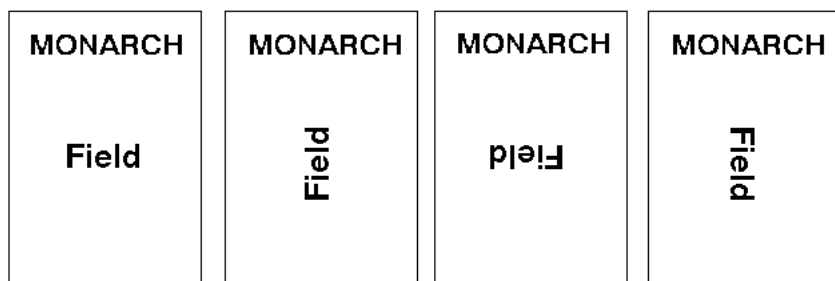
C12. "fixed char"		Fixed characters to appear in the field. Maximum **2710** characters. "**"** is the default. Enclose in quotation marks.
C13. sym set		Symbol set. Use **0** for the Internal Symbol Set. **0** is the default. For scalable or TrueType® fonts, use:

1	ANSI Symbol Set
100	Macintosh
101	Wingdings
102	Unicode (user input) for particular mapping
103	BIG5 (user input) for Unicode mapping
104	GB2312 (user input) for Unicode mapping
105	SJIS (user input) for Unicode mapping (CP 932, Japanese)
106	GB2312 (user input) for GB2312 mapping (CP 936, Simplified Chinese)
107	BIG5 (user input) for BIG5 mapping (CP 950, Traditional Chinese)
108	KSC (user input) for KSC mapping (CP 5601 Korean)
110	Unicode UTF-8
437	DOS Code Page 437 (Domestic)
850	DOS Code Page 850 (International)
852	DOS Code Page 852 (Latin 2)
855	DOS Code Page 855 (Russian)
857	DOS Code Page 857 (IBM Turkish)
860	DOS Code Page 860 (MS-DOS Portuguese)
1250	Code Page 1250 (Latin 2)
1251	Code Page 1251 (Cyrillic)
1252	Code Page 1252 (Latin 1)
1253	Code Page 1253 (Greek)
1254	Code Page 1254 (Turkish)
1255	Code Page 1255 (Hebrew)
1256	Code Page 1256 (Arabic)
1257	Code Page 1257 (Baltic)
1258	Code Page 1258 (Vietnam)

Note: The Standard, Reduced, Bold, OCRA and HR fonts only support the Internal Symbol Set (0). The CG Triumvirate™ typefaces only support the ANSI (1) and DOS Code Page 437 (2) and 850 (3) Symbol Sets. The scalable font (font#50) does not support Code Page 1256 Arabic (10). Code pages 852-860 and 1250-1258 are for downloaded TrueType fonts or the scalable font. Code pages 102-110 require a downloaded International TrueType font. TrueType fonts are designed to be regionally specific; therefore, all code pages may not be supported in a given font. See Appendix C, "Symbol Sets/ Code Pages" for more information.

If using symbol set 110 (Unicode UTF-8), set the MPCL control characters (start of header, etc.) to decimal values between 0 and 128; otherwise, errors may occur with the Unicode data entered. See Chapter 2, "Configuring the Printer" for more information about the MPCL control characters.

Example   **C,30,10,0,1,1,1,B,L,0,0,"MADE IN USA",0 |**

Defines a constant text field starting at row 30, column 10. It does not have any additional inter-character gap. The Standard font is used without any additional magnification. The printing is black on white and left justified. No field or character rotation is used. "MADE IN USA" is printed in this field. The internal symbol set is used.

Defining Line Fields

Use lines to form borders and mark out original prices. Define each line separately. This field is not assigned a field number, but is counted as a field (keep this in mind, as the printer allows a maximum of **1000** fields per format). Define any line length and a thickness up to 99 dots, as long as the solid black print does not exceed 30 percent of any given square inch of the label.

## Line Types

Create horizontal and vertical lines. There are two ways to define lines.

Segments   Select the starting point and ending point.

Vectors   Select the starting point, the angle, and the length of the line.

Syntax   **L,type,row,column,angle/end row,length/end col,thickness,"pattern" |**

L1. L	Line Field.	
L2. type	Type of line. Only vertical and horizontal lines are supported. **S** is the default. Options:	

S   Segment. Select the starting point and ending point.
V   Vector. Select the starting point, angle, and length.

L3. row   Row location - distance from bottom of print area to the starting point. **10** is the default.

|	
English	**0 – 1599**
Metric	**0 – 4063**
203 Dots	**0 – 3247**
300 Dots	**0 – 3599**
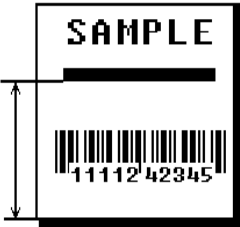
L4. column   Column location - distance from left edge of the print area to line origin. **10** is the default.

English	**0 – 409**
Metric	**0 – 1039**
203 Dots	**0 – 831**
300 Dots	**0 – 1247**

L5. angle
/end row

If Using <u>*Segments*</u>:
Row location of ending point. Measure from bottom of print area. On horizontal lines, this value must match item L3. **100** is the default.

English	**0 – 1599**
Metric	**0 – 4063**
203 Dots	**0 – 3247**
300 Dots	**0 – 3599**

If Using <u>*Vectors*</u>:
Angle of line. Options: **0**, **90**, **180**, or **270**. **0** is the default.
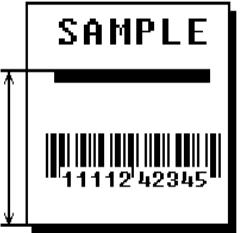
| *L6. length/ end col* | If Using *Segments*:
Column location of end point. Measure from left edge of print area. On vertical lines, this value must match parameter **L4**. **100** is the default. |

L6. length/ end col

If Using <u>Segments</u>:
Column location of end point. Measure from left edge of print area. On vertical lines, this value must match parameter **L4**. **100** is the default.

English	**0 – 409**
Metric	**0 – 1039**
203 Dots	**0 – 831**
300 Dots	**0 – 1247**

If Using <u>Vectors</u>:
Length of the line in selected units. **10** is the default.

L7. thickness

Using the chart below for reference, write the line thickness (**1** to **99**) in box **L7**. **2** is the default. Line thickness fills upward on horizontal lines, or to the right on vertical lines. Measured in dots.

Dots	Thickness
1	
10	
24	
48	
96	

L8. "pattern"

Line pattern. Enter **""**.

Example   **L,S,110,30,110,150,10,"" |**

Defines a horizontal line field as a segment starting at row 110, column 30 and ending at row 110, column 150. The line thickness is 10 dots.

Defining Box Fields

Use boxes to form borders or highlight items of interest. Define each box field separately. This field is not assigned a field number, but is counted as a field (keep this in mind, as the printer allows a maximum of **1000** fields per format). Define any line length and a thickness up to 99 dots, as long as the solid black print does not exceed 30 percent of any given square inch of the label.

Syntax   **Q,row,column,end row,end col,thickness,"pattern" |**

Q1. Q	Box (Quadrilateral) Field.	
Q2. row	Row location - distance from bottom of print area to lower left corner of box. **10** is the default.	

English	**0 − 1599**
Metric	**0 − 4063**
203 Dots	**0 − 3247**
300 Dots	**0 − 3599**
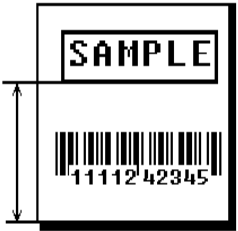
Q3. column   Column location - distance from left edge of print area to lower left corner of box. **10** is the default.

English	**0 − 409**
Metric	**0 − 1039**
203 Dots	**0 − 831**
300 Dots	**0 − 1247**

Q4. end row   Distance from bottom of print area to upper right corner of box. **100** is the default.

English	**0 − 1599**
Metric	**0 − 4063**
203 Dots	**0 − 3247**
300 Dots	**0 − 3599**
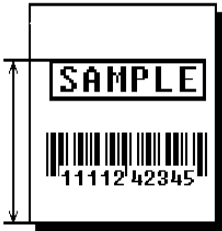
Q5. end col   Distance from left edge of print area to upper right corner of box. **100** is the default.

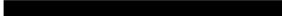|	
English	**0 − 409**
Metric	**0 − 1039**
203 Dots	**0 − 831**
300 Dots	**0 − 1247**

Q6. thickness   Using the chart below for reference, write the desired line thickness (**1** to **99**) in box Q6. **2** is the default. Boxes fill inward, so make sure boxes do not overwrite other fields. Measured in dots.

Dots	Thickness
1	
10	
24	
48	
96	

Q7. "pattern"   Line pattern. Enter **""**.

Example   **Q,240,30,270,150,3,"" |**

Defines a box field starting at row 240, column 30. It ends at row 270, column 150. It has a thickness of 3 dots.

Defining Bar Code Verifier Fields

The bar code verifier field in a format references the bar code verifier configuration packet ID to use for this particular format. Every format can have a specific bar code verifier configuration packet. See "Defining a Bar Code Verifier Configuration Packet," in Chapter 2 for more information.

If a particular bar code verifier configuration packet is not specified, the last sent bar code verifier configuration packet is used.

Note: Include the bar code verifier field after the format header in the format packet.

Syntax   **V,vfrID |**

  V1. *V*   Bar Code Verifier Field.
  V2. *vfrID*   Unique number from 1 to 999 to identify the bar code verifier configuration packet to use with this format.

Example   **{F,25,A,R,M,508,508,"FMT-25" |**
      V,3 |
      B,1,12,F,110,115,1,2,120,5,L,0 |}

Specifies bar code verifier configuration packet #3 starting with format 25 until another bar code verifier configuration packet is specified.

Defining the RFID Data Field

The RFID Data Field contains the information programmed into the RFID inlay. The syntax of the RFID Data Field is similar to the standard non-printable text field format.

Note: Printing over the RFID inlay (or transponder) causes printing irregularity.
    Do not use skip index with RFID supplies. Unexpected results may occur.

Syntax   **X,field#,# of char,data_type |**

X1. *X*		RFID Data Field.
X2. *field#*		Unique number from **0** to **999** assigned to this field. **1** is the default.
X3. *# of char*		This number must be equal to or greater than the total number of characters in the RFID Data Field. Range: **0** to **2710**. **30** is the default. The amount varies according to the RFID data, protocol, and module type. For example, C1Gen2 can be up to 496 bits, **plus** the access password, lock code, etc.
	Note:	An error 715 occurs if the printer does not receive the correct amount of data. Pad data to ensure the correct amount of bits. See "Using Option 30," in Chapter 4 for more information.
		Using Option 65 (Serialized EPC Data), the number of characters specified here should match the number of characters entered for the batch data (including the 38-bit serial number). Pad data with zeros to ensure the correct amount of bits.
X4. *data_type*		Data type. Use **0** (ASCII Hex - ASCII representation of Hex), which uses characters **A** to **F** and **0** to **9**. For example, the letter A in ASCII Hex has a value of 41 and is represented in MPCL batch data as "4141414141414141414141".
	Note:	**Do not** use the ASCII file separator character (1C) in RFID batch data.

Example   **X,5,24,0 |**

Defines an RFID Data Field (field #5) with exactly 24 ASCII Hex characters for a 96-bit RFID inlay.

DEFINING FIELD OPTIONS

This chapter provides a reference for defining

♦ field options in formats

♦ check digit packets.

Note: When using multiple options in a format, options are processed in the order they are received.

Applying Field Options

Field options further define text, bar code, and non-printable text fields. The text, constant text, or bar code field must be previously defined *before* applying any field option to it.

Note: Define field options immediately after the field to which they apply.

Combining Field Options

Multiple options can be used with most fields. Options can be used in any combination except as noted with each definition. Options are processed in the order they are received.

For example, use Option 4 to copy data from another field, and then use Option 30 to pad the field. When using multiple options for the same field, place the options in the order to apply them in the format.

Restrictions

Some options cannot be used together. For example, incrementing (Option 60) and price field (Option 42) options cannot be applied to the same field. See each Option's section for specific combinations to avoid.

Option 4 (copy a field) is the only option that can be repeated for a single field.

Syntax   **R,option#,parameter...parameter |**

R1. R	Indicates field option header.
R2. option#	Option number:

1 Define fixed characters
2 Data type restrictions
3 Data entry template
4 Copy data from previous field
5 Define data entry sources
6 Upload field data
7 Print Time/Date
20 Define data entry prompts
21 Define extended field names (XML-enabled printer)
30 Pad data to left or right with specified character
31 Generate check digit
42 Format as a price field
50 Define bar code densities
51 Define security and truncation of PDF417 bar codes
52 Define width or length of PDF417 bar codes
53 Define optional settings for Aztec bar codes
60 Define incrementing or decrementing field
61 Reimage fields
62 Do not scan/verify a particular bar code
65 Serialized EPC data

R3. parameter(s)   Varies per option. See the following option descriptions.

Using Option 1 (Fixed Data)

Fixed data is information (a company name or store number) to print on all labels. Define fixed characters for an entire field or for part of a field.

Syntax   **R,1,"fixed char" |**

R1. R	Option Header.
R2. 1	Option **1**.
R3. "fixed char"	Characters to insert. Enclose in quotation marks. If defining fixed characters for part of a field, place underscores(_) in non-fixed positions. Any spaces in the phrase are fixed characters. Range: **0** to **2710**.
Note:	Underscore characters are stripped out and the data is compressed if no data is supplied by the batch and the field length is variable.

Example   **R,1,"_ _ _%$_ _ _ _" |**

Uses fixed characters (%$) in positions 4 and 5. The other positions are variable.

Example   **R,1,"MONARCH" |**

"MONARCH" appears as a fixed field in this example.

To fill in the non-fixed portion of the field, see "<u>Defining Batch Data Fields</u>" in Chapter 6. As an alternative, apply Option 4 to copy data into the non-fixed character positions.

Using Option 2 (Data Type Restrictions)

This option restricts the data type for a particular field. Use Options 2 or 3 only once per field. Do not use with Option 3 (Data Entry Templates). If batch data is entered in the batch packet, **do not** apply Options 2 and 3. Only use these options for *offline* batch entry.

Note: Use a keyboard or scanner for offline data entry.

Syntax   **R,2,char_code |**

R1. R	Option Header.
R2. 2	Option **2**.
R3. char code	Character type for the field. **1** is the default. Options:

    1   Numeric only (0..9)
    2   Letters only (A..Z,a..z)
    3   Symbols only (printable characters other than letters or numbers)
    4   Letters and numbers only
    5   Numbers and symbols only
    6   Letters and symbols only

	Spaces are permitted in all categories. Use a combination of any two (letter, numbers, or symbols) character types.
Note:	A use for this option is a quantity field, where the operator could enter only numeric data.

Example   **R,2,2 |**

Restricts the field data to letters only (A-Z or a-z).

Using Option 3 (Data Entry Templates)

This option provides more specific restrictions than Option 2. This option can be used to select certain letters (such as A through F or the numbers 1 through 4) from a character set. Use this option to create a template of allowable characters for a field. Do not use with Option 2 (Data Type Restrictions).

Only use this option for *offline* batch entry. Define a template up to 30 characters long, but the printer only displays 16 characters at a time. The character template must contain the same number of characters as the field.

Note: Use a keyboard or scanner for offline data entry.

Syntax **R,3,code,chars |**

R1. R	Option Header.
R2. 3	Option **3**.
R3. code	Data types. **T** is the default. Options

S Defines a specific set of characters for the entire field. The string length does not have to match the field length. Maximum is **30** characters.

T Creates a template of allowable data types by placing character indicators in each character position. The number of characters in the string must match the length of the field.

Note: A sample use for this option is a field on a patient record containing blood type. Acceptable characters would be A, B, O, +, or -.

R4. chars Characters to include in a field or a specific template. **""** is the default. Must be enclosed within quotation marks. Indicators can be

* any printable character
# any digit 0-9
@ any letter a-z, A-Z
- no user input for this position (for fixed data or copied data)

Example **R,3,S,"ABC1234567890" |**

Restricts the field data to letters A, B, and C, and all digits.

Example **R,3,T,"***#_ _ _ _" |**

Creates a template that allows any printable character in positions 1, 2, and 3; digits in position 4; and reserves positions 5 through 8 for fixed or copied data.

Using Option 4 (Copy Data)

Use option 4 to create a field that uses data from another field. This creates merged fields or sub-fields. Copy the information from multiple fields into one field by applying the copy procedure more than once. Copy data is the only option that can be applied to a field more than once.

The maximum number of characters defined in **T3** (# of char) or **B3** (# of char) must allow for the total number of characters that will be placed in the field, including price, check digit, or fixed characters inserted by the printer. The maximum number of characters in the field cannot exceed **2710** or the maximum number of characters permitted by the **bar code**.

Option 4 can be applied to an RFID Data Field. With Version 1.8 or greater firmware, the printer supports copying ASCII Hex data and converting it to ASCII Decimal data.

Note: When copying from more than one field, copy into the destination field from left to right.

Syntax **R,4,src fld,src start,# to copy,dest start,copy code |**

R1. R	Option Header.
R2. 4	Option **4**.
R3. src fld	Field number from which data is copied. Range: **0** to **999**. **1** is the default.
R4. src start	Position number in the source field of the first character to be copied. Character positions are numbered **1** to **2710**, starting from the left. **1** is the default.
R5. # to copy	Number of characters to copy. Range: **1** to **2710**. **1** is the default. For *copy codes* 5 and 6 (ASCII hex source), this value is in **bits**.
R6. dest start	Position number where copied characters are to begin printing in the destination field. Range: **1** to **2710**.
R7. copy code	Copy Method. **1** is the default.

Value	Description
1	Copy field with formatting (including price symbols, pad characters, check digits, etc.).
2	Copy unformatted data (without price characters, pad characters, etc.).

Special methods for RFID Data:

Value	Source Field	Destination Field	Formatting
5	ASCII hex	ASCII decimal	with formatting
6	ASCII hex	ASCII decimal	no formatting

Note: Make sure the # *to copy* is in **bits**.

Example **R,4,3,1,3,1,1|**

Copies data from field #3, starting at the first position and copying three characters. In the destination field, the information is placed in position 1 and copied as formatted data.

RFID ASCII Hex Data Convert to ASCII Decimal Example

```
{F,20,A,R,E,200,270,"EPC-DEC"|
X,1,24|
R,60|
T,2,40,V,20,10,0,50,8,8,B,L,0,0|
R,1,"SGTIN96 EPC : _____"|
R,4,1,1,25,15,1|
T,3,30,V,40,10,0,50,8,8,B,L,0,0|
R,1,"MFG ID # : _____"|
R,4,1,15,24,12,5| `Convert MFG ID to ASCII Decimal`
T,4,30,V,60,10,0,50,8,8,B,L,0,0|
R,1,"ITEM ID # : _____"|
R,4,1,39,20,13,5| `Convert ITEM ID to ASCII Decimal`
T,5,30,V,80,10,0,50,8,8,B,L,0,0|
R,1,"SERIAL # : _____"|
R,4,1,59,38,12,5| `Convert SER # to ASCII Decimal`}
{B,20,N,5|
1,"30340FFFFF7777C98DF3EA6D"|}
```

This example produces the following 5 labels.

The **MFG ID#, ITEM ID#**, and **SERIAL#** are converted from ASCII Hex data (**30340FFFFF7777C98DF3EA6D**) to ASCII Decimal data then printed.

```
SERIAL #: 41036278385
ITEM ID#: 908767
MFG ID #: 262143
SGTIN96EPC : 30340FFFFF7777C98DF3EA71
```
← **Fifth printed label**

```
SERIAL #: 41036278384
ITEM ID#: 908767
MFG ID #: 262143
SGTIN96EPC : 30340FFFFF7777C98DF3EA70
```

```
SERIAL #: 41036278383
ITEM ID#: 908767
MFG ID #: 262143
SGTIN96EPC : 30340FFFFF7777C98DF3EA6F
```

```
SERIAL #: 41036278382
ITEM ID#: 908767
MFG ID #: 262143
SGTIN96EPC : 30340FFFFF7777C98DF3EA6E
```

```
SERIAL #: 41036278381
ITEM ID#: 908767
MFG ID #: 262143
SGTIN96EPC : 30340FFFFF7777C98DF3EA6D
```
← **First printed label**

Merging Fields

Copy data to merge the contents of fields. Use the copy data option as many times as necessary to copy all the appropriate fields into the merged field. RFID Data Fields can be merged.

In the following example, two text and two non-printable fields are shown. Data from these fields is merged to form field 5, and is then printed as a bar code.

Field	Data	Field Type
1	203	Non-printable
2	339	Non-printable
3	8	Text
4	BLUE	Text
5	2033398BLUE	Bar Code

To create this sequence:

1. Define fields 1, 2, 3, and 4.

2. Define field 5 as a bar code. Allow enough characters in the bar code field to hold all the copied characters.

3. Apply Option 4 to field 5 once for every source field.

Sub-Fields

Copy a segment of data from one field into a new location, called a sub-field. For example, extract part of the data in a bar code and display it in text form in a sub-field. Then, use the copy data option.

Using Option 5 (Define Data Entry Sources)

Defines how data is entered into a field. Option 5 is required for offline data entry. If the field holds all fixed characters or copied data only, eliminate the operator prompt. Use Option 5 only once per field. Use a keyboard or scanner for offline data entry.

Use this option to read pre-programmed data in the RFID chip embedded within the supply. Using Option 5 stops the printer while reading each label; regardless of the print speed. If using a batch quantity greater than one, the data is read from each label.

When reading data, make sure the maximum number of characters in the field is equal to or greater than the number of characters being read. If not, the data may be incomplete.

Use option 6 (Upload Field Data) with Option 5 to upload the data from the RFID chip to a host.

Note: Option 5 re-images each label in the batch.

Syntax   **R,5,code |**

R1. R	Option Header.	
R2. 5	Option **5**.	
R3. code	Input code for the data in the field. Options:	
	K Keypad	
	R RFID (read data from the RFID chip). This is ignored on non-RFID printers.	

Example   **R,5,K |**

Allows data to be entered from a keyboard.

Example   **T,2,10,V,250,50,0,1,1,1,B,C,0,0,0 |**
           R,5,R |

Reads the pre-programmed data from the RFID chip and saves that data into the text field.

Using Option 6 (Upload Field Data)

Upload data from any field using Option 6. When uploading multiple fields of data, the data is comma separated.

Apply Option 6 to an RFID field to upload EPC data and the TID (tag identifier) data. To upload TID data, enable singulate mode; otherwise, an error 737 occurs. See "Defining the RFID Setup Packet" for more information.

Note: When uploading both EPC and TID data, the TID data is always first, followed by a comma, then the EPC data; regardless of Option 6 ordering within the format.

Data is uploaded to the last port that received host data (serial, USB, or Ethernet) at the end of the batch or label, as follows:

Uploading Data

Batch Qty	Option 6	Option 60	Result
1	Yes **R,6,H|** **R,6,H,T|**	No	EPC data is uploaded at the end of the batch. TID data is uploaded at the end of the batch.
>1	Yes	No	Uploads the last data or uploads the data preceded by an error message.
>1	Yes **R,6,H|** **R,6,H,T|**	Yes (**R,60,I,1|**)	EPC data is uploaded after each label. TID data is uploaded after each label.
>1	Yes **R,6,H|** **R,6,H,T|**	Yes (**R,60,I,0|**)	EPC data is uploaded after each label. TID data is uploaded after each label. Note: **NO** incrementing is performed.

Note: Option 60 (**R,6,I,0\|**) can be used anywhere in the format.

If a printing or encoding error occurs, the uploaded string begins with "ERROR xxx" before the data. This uploaded string, **ERROR 750 3034257BF400B78000A00111** indicates the printhead is overheated and includes the EPC data.

In singulate mode, if an errors occurs while reading the TID, no TID data is uploaded.

Uploading the label or batch's data occurs when a label is successfully printed (or there is an error).

Syntax   **R,6,device,type |**

R1. R	Option Header.
R2. 6	Option **6**.
R3. device	Last port that received host data. Use **H** (host). To upload EPC data, apply this option to an RFID field.
R4. type	Type of data to upload. Use **T** to upload the TID (tag identifier). Configure the RFID Setup Packet to enable singulate mode first.

Example   **T,2,10,V,250,50,0,1,1,1,B,C,0,0,0 |**
            R,6,H |

Uploads the text field's data to a file.

Example **B,3,12,F,50,50,1,2,60,7,L,0 |**
    R,60,I,0 |
    R,6,H |

Uploads the UPCA bar code field's data to a file and uploads data for each label in the batch.

Example **{F,2,A,R,E,300,200,""|**
    X,2,24,0|
    R,60,I,1|
    R,6,H|
    R,6,H,T|

Increments the RFID EPC data for each label, uploads the EPC data for each label, and uploads the TID for each label.

Example **{F,2,A,R,E,300,200,""|**
    X,2,24,0|
    R,60,I,0|
    R,6,H|
    R,6,H,T|

Does not increment the RFID EPC data, but uploads the EPC data and the TID for each label.

Example **T,150,V,230,130,0,1,1,1,B,L,0,0 |**
    R,5,R |
    R,6,H |

Reads the RFID data from the RFID chip embedded in the supply. Uploads the data to the last-used port.

Example **{F,1,A,R,E,600,400,"RDCI" |**
    X,5,24,0 |
    T,6,20,V,415,270,0,50,15,15,B,L,0,2 |
    R,4,5,1,16,1,0 |
    R,6,H | }

Copies data from field 5 (RFID Data Field) to field 6 (Text field). Uploads the data to the last-used port.

Sample Upload Packet

Example **X,2,24,0|**
    R,60,I,1|◄———————— **increments EPC data by 1**
    R,6,H|◄———————— **upload EPC data**
    R,6,H,T|◄———————— **upload TID data**

Returns the following TID and EPC data separated by a comma. TID data is always first, then the EPC data; regardless of Option 6 ordering within the format:

E2801160200062FEB79F08A7,3034257BF400B78000A00111
E2801160200062FEB678DAE6,3034257BF400B78000A00112
E2801160200062FEB70E67FA,3034257BF400B78000A00113

Example **R,5,R |**
    R,6,H |

Returns the following pre-programmed EPC data in the RFID chip in the upload packet:

3132333435363738393030132

Example **B,3,12,F,50,50,1,2,60,7,L,0 |**
    R,6,H |

Returns the following UPCA bar code data entered from the batch in the upload packet:
123456789012

Using Option 7 Print Time/Date

The printer must contain the optional real time clock. Use this option to print the printer's current time and date in a format. To set the printer's current time and date, see "Defining the Time/Date Packet" in Chapter 2.

Use with text, constant text or bar code fields.

Note: To use Option 7 with a bar code field, the bar code must support the full ASCII set of characters.

Syntax   **R,7,param,format |**

R1. R	Option Header.
R2. 7	Option 7.
R3. param	Parameter to print. Use **1** for real time clock.
R4. format	Time and Date formats. **1** is the default. Options:

1   Date as 06/09/2014
2   Date and time as 06/09/2014 18:15
3   Date and time as 06/09/2014
4   Time as 18:15
5   Time as 18:15:05

Example   **R,7,1,2 |**

Prints the date and time in the format as 06/09/2014 18:15.

Using Option 20 (Define Data Entry Prompts)

This option defines the operator prompt and it is not recommended on fields filled entirely by fixed characters or copied data. This option must be defined before Option 5, or the prompt does not display during data entry. Use a keyboard or scanner for offline data entry.

Syntax   **R,20,"prompt" |**

R1. R	Option Header.
R2. 20	Option **20**.
R3. "prompt"	Displays the exact phrase to prompt the operator. Must be enclosed within quotations. The prompt must be **15** characters or less.

Example   **R,20,"Order Number" |**

Displays the prompt Order Number for the operator when this field is imaged.

Using Option 21 (Define Extended Field Names)

This option defines names for each field in a format, which can be longer than eight characters. This option is only available on the XML-enabled printer.

Syntax   **R,21,"field_name" |**

R1. R	Option Header.
R2. 21	Option **21**.
R3. "field_name"	Specifies the field name. Must be enclosed within quotations. The name can be **1** to **256** characters, excluding non-printable control characters and the quotation mark ("). **""** is the default. This name must match the one specified in the variable name attribute.

Example   **T,2,10,v,250,50,0,1,1,1,B,C,0,0,0 |**
            R,21,"SALEPRICE_FIELD" |

Defines the name for text field 2 as "SALEPRICE_FIELD."

Using Option 30 (Pad Data)

Add characters to one side of a field to "pad" the field. Padding fills in the remaining spaces when entered data does not fill an entire field. If a variable length field is not completely filled with batch data, this option fills the remaining positions in the field with the character designated by Option 30.

Note: Option 30 can be applied to an RFID Data Field.

Syntax   **R,30,L/R,"character" |**

R1. R	Option Header.
R2. 30	Option **30**.
R3. L/R	Indicates type of padding. **R** is the default. **L** Pad field on left side **R** Pad field on right side
R4. "character"	Pad character must be within the **0** to **255** decimal range and enclosed inside quotation marks. **"0"** is the default. The pad character must be in ASCII Hex format. Do **not** use on fixed length fields.

Example   **R,30,L,"X" |**

Pads data with an "X" on the left side of the field.

Example   **X,2,24,0 |**
            R,30,R,"0" |

Pads the data in the RFID Data Field with a "0" on the right side of the data. This example uses a 96-bit RFID inlay.

Sample Use for Padding

Use pad characters to make a variable length bar code occupy a fixed amount of space on the supply. If the maximum number of characters in the bar code is 15, but the batch packet only has 10 characters, the padding option fills the remainder of the field with pad characters.

Using Option 31 (Calculate Check Digit)

The printer generates a check digit when Option 31 is applied to the field. Do not use this option if the field contains a UPC, EAN, or Code 39 (with the MOD43 check digit) bar code.

Syntax **R,31,gen/ver,check digit # |**

R1. R	Option Header.
R2. 31	Option **31**.
R3. gen/ver	Enter **G** to generate a check digit.
R4. check digit #	Specifies a check digit scheme. Enter a number that identifies a check digit scheme that has been defined. For more information, see "<u>Using Check Digits</u>." Range: **1** to **10**. **1** is the default.

Example **R,31,G,5 |**

Generates a check digit using the previously defined check digit scheme 5.

Using Option 42 (Price Field)

Apply options to automatically insert monetary symbols. Do not use this option with Option 31 (define a check digit) or Option 60 (increment or decrement a field). This option is not recommended for bar codes. When determining the maximum number of characters, add the maximum number of digits and the monetary symbols.

Syntax **R,42,appear_code |**

R1. R	Option Header.
R2. 42	Option **42**.
R3. appear_code	Enter **1** to print the price field in standard notation, as defined by the country setting.
	Use the monetary formatting packet to select monetary notations and symbols by country setting. See "<u>Defining the Monetary Formatting Packet</u>" for more information. See Appendix C, "<u>Symbol Sets/Code Pages</u>," to make sure the monetary symbol is printable in the font selected for this field. For monetary symbols other than the dollar sign, use the internal symbol set.

Example **R,42,1 |**

Uses a price field that prints the monetary symbol and notations as defined in the monetary formatting packet.

Using Option 50 (Bar Code Density)

Apply this option to bar code fields to create custom densities. This option overrides the density value in the bar code field. ***To use this option, set the density parameter in the bar code field to the default value.*** Only use this option once for each bar code field. Bar codes produced using Option 50 may not be scannable. Synthetic supplies are recommended to produce scannable bar codes.

The additional character gap, narrow space, and wide space parameters (*R5, R6, R7*) only need to be specified with

- ◆ Code 39
- ◆ I 2of5

- ◆ Codabar
- ◆ MSI

If these parameters are specified for any other bar codes, they will be ignored by the printer. Do not use Option 50 with fixed density bar codes, such as POSTNET or Intelligent Mail.

Option 50 can also be used with Option 52 to further customize the dimensions of a PDF417 bar code. Option 52 allows customization of the number of columns or number of rows for the bar code.

Syntax   **R,50,narrow,wide,gap,nar_space,wide_space |**

R1. R	Field Option Header.
R2. 50	Option **50**.
R3. narrow	Dot width of the narrow element. Range: **1** to **99**. **1** is the default.
R4. wide	Dot width of the wide element. Range: **1** to **99**. **1** is the default.
R5. gap	Additional dot space between characters. Enter a value of **1** to **99**. (Code 39, Codabar, I 2of5, MSI only.) **1** is the default.
R6. nar_space	Additional dot width of the narrow bar code/bar width space. (Code 39, Codabar, I 2of5, PDF417, MSI only). Range: **1** to **99**. **1** is the default.
R7. wide_space	Additional dot width of the wide bar code space/height of each individual stacked bar code. (Code 39, Codabar, I 2of5, PDF417, MSI only). Range: **1** to **99**. **1** is the default.

Example   **B,3,12,V,25,40,3,12,80,8,L,0 |**
               R,50,4,8,8,8,8 |

Creates a custom bar code density for an I 2of5 with a narrow element of 4 dots, a wide element of 8 dots, a gap of 8 dots, 8 additional dot widths for the narrow bar code space and the wide bar code space.

The same I 2of5 bar code is shown with the ***default*** density for 300 dpi. This sample does **not** use Option 50.
B,4,12,V,125,40,3,12,80,8,L,0 |

Custom Density

I 2of5

Default Density

Example **B,5,12,V,25,40,4,7,80,1,L,0 |**
          R,50,4,8,8,8,8 |

Creates a custom bar code density for a Code 39 with a narrow element of 4 dots, a wide element of 8 dots, a gap of 8 dots, 8 additional dot widths for the narrow bar code space and the wide bar code space.

Custom Density

028028123456

CODE 39

The same Code 39 bar code is shown with the ***default*** density for 300 dpi. This sample does **not** use Option 50.
B,6,12,V,125,40,4,7,80,1,L,0 |

Default Density

028028123456

Example **B,1,40,V,100,100,32,1,0,8,L,0 |**
          R,50,2,10 |

Creates a custom PDF417 bar code density for 203 dpi printers. The narrow element width is 2 dots and the height is 10 dots.

Example **B,1,40,V,100,100,32,1,0,8,L,0 |**
          50,3,15 |

Creates a custom PDF417 bar code density for 300 dpi printers. The narrow element width is 3 dots and the height is 15 dots.

Using Option 51 (PDF417 Security/Truncation)

Define a security level and choose whether or not to truncate a PDF417 bar code. Higher security levels add data to a bar code, improving scan reliability. Some damaged bar codes may still be scannable if the security level is high enough. Use this option to create standard PDF417 bar codes or use the truncated option to create a narrower bar code. This option can appear only once per PDF417 field, in any order, following the bar code field.

As the security level is increased, so is the size of the PDF417 bar code. For each level increased, the bar code doubles in size.

Syntax	**R,51,security,stand/default |**	
	R1. R	Option Header.
	R2. 51	Indicates Option **51**.
	R3. security	Security level ranges from **0** to **8**. **0** is the default.
		Higher security levels add data to a bar code, improving scan reliability. Some damaged bar codes may still be scannable if the security level is high enough.
	R4. stand/def	Truncation selector. S is the default. Valid values:
		S a standard PDF417 bar code
		T truncated

Example   **R,51,2,S |**

Defines a security level of 2 for a standard PDF417 bar code.

Using Option 52 (PDF417 Width/Length)

This option defines the image width or length of a PDF417 bar code. If defining a fixed number of columns (width), the bar code expands in length. If defining a fixed number of rows (length), the bar code expands in width. Column value does not include start/stop or left/right indicator columns. If this option does not immediately follow the PDF417 bar code field, the default settings are used. Use this option once per PDF417 bar code field.

Syntax	**R,52,row/column,dimension |**	
	R1. R	Option Header.
	R2. 52	Indicates Option **52**.
	R3. row/column	Indicates the number of rows or columns. If rows are specified, the bar code expands in columns, or vice versa.
		R Row
		C Column
	R4. dimension	The number of rows or columns defined for the bar code. Valid values:
		3-90 for rows. 3 is the default for rows.
		1-30 for columns. 1 is the default for columns.

Example   **R,52,C,10 |**

Defines the column width of 10, which expands the PDF417 bar code length by 10.

Using Option 53 (Optional Settings for Aztec)

Use Option 53 to set the error control level, enable ECI data, enable a menu symbol, and add appended data to an Aztec bar code. Typically, these settings are not used. When using Option 53, the printer may take longer to image the bar code and require more time to print the format.

Syntax   **R,53,error_ctrl,ECI,menu_sym,str_append,"string" |**

R1. R	Option Header.	
R2. 53	Option **53**.	
R3. error_ctrl	Error control level. Some damaged bar codes may still be scannable if the error control level is high enough. **0** is the default. Options:	
	0 Default level	
	1-99	
	101-104	
	201-232	
	300	
R4. ECI	Sets the ECI Data flag. **0** is the default. Options:	
	0 Disable	
	1 Enable	
R5. menu_sym	Sets the Menu Symbol flag. **0** is the default. Options:	
	0 Disable	
	1 Enable	
R6. str_append	Structured append information. Range: **1** to **26**. **1** is the default.	
R7. "string"	String to append. Range: **0** to **24**. **""** is the default. Must be enclosed in quotation marks.	

Example   **R,53,0,0,0,1,"" |**

Uses option 53 to set the error control to 0, disables the ECI data and menu symbol flags, and does not append any data to the bar code.

Using Option 60 (Incrementing/Decrementing Fields)

Applications such as serial numbers require a numeric field to increment (increase in value) or decrement (decrease in value) on successive labels within a single batch. Incrementing or decrementing can be applied to **numeric** data only. If a field includes letters and digits, apply incrementing or decrementing to only the portion of the field that contains digits. Do not use with Option 42 (price field).

When using Option 60 with RFID Data Fields or EPC data:

♦ Use caution when incrementing an RFID field if data is coming directly from a host because the field must be incremented in ASCII Hex.

♦ The data increments by **one** as follows: 0 to F (0123456789ABCDEF), then back to 0.

♦ Parameters *R5 and R6* are ignored for RFID and EPC data fields.

♦ Option 60 allows data to be uploaded using Option 6. See "Using Option 6 (Upload Field Data" for more information.

Syntax　　**R,60,I/D,amount,l pos,r pos |**

R1. R		Option Header.
R2. 60		Option **60**.
R3. I/D		Increment or decrement. **I** is the default.
	I	incrementing field
	D	decrementing field (not valid for RFID)
R4. amount		Amount to increase or decrease. Range: **0** to **999**. **1** is the default.
R5. l pos		Leftmost position in inc/dec portion of field. If this value is not entered, the default value is used. Range: **0** to **2710**. **1** is the default.
R6. r pos		Rightmost position in inc/dec portion of field. If this value is not used, the entire field length is used as the default. Range: **0** to **2710**. **2710** is the default.

Example　　**R,60,I,5,1,6 |**

Increments a field by 5 each time the field is printed. The field increments beginning with the first left position and ending with the sixth position.

Incrementing EPC Data Example

```
{F,1,A,R,E,400,400,""|
T,1,30,V,50,70,0,1,1,1,B,L,0,0|
R,60,I,1|
B,2,12,F,10,130,1,4,50,8,L,0|
R,60,I,1|
X,3,100,0|          ⟵          RFID Data Field
R,60| }                        Incrementing Option

{B,1,N,3|                                RFID Data Field Batch
1,"001"|                                 Data in ASCII Hex with
2,"123456789012"|                        EPC Data included
3,"12345678123456781234561B~028~028~02812345678~02812345678~02800000~028"| }
```

This example uses option 60 to increment the EPC RFID data by one.

Fixing the First Number in the Incrementing Sequence

There are two ways to enter the first number in the incrementing sequence. Use batch data or Option 1 to define the first number as a fixed character. The first number in the sequence must contain the same amount of digits as the highest number to be counted. For example, to count the numbers **1** to **999**, the first number in the sequence must be entered as **001**.

Using Option 61 (Re-image Field)

This option redraws (reimages) a constant field when a constant field is next to a variable field. It can be used on text, constant text, bar code, line, or box fields.

These printers do not redraw an area if the field data does not change. When a field changes, that area is cleared and the new field data is imaged. However, the new field data may require a larger area than the previous field did. In some cases, neighboring fields that do not change (constant fields) may be covered with white space from the changing field's (variable fields) area. Use this option to reimage the constant field, or it may appear broken.

Note: The most common use for this option is with incrementing fields on a label, because they may cover a constant field.

Syntax   **R,61 |**

R1. R		Option Header.
R2. 61		Option **61**.

Example **R,61 |**

Reimages the constant field that appears next to a variable field.

In this example, Option 61 is applied to the bar code field to keep the incrementing field (Box #) from blocking out the bar code field.
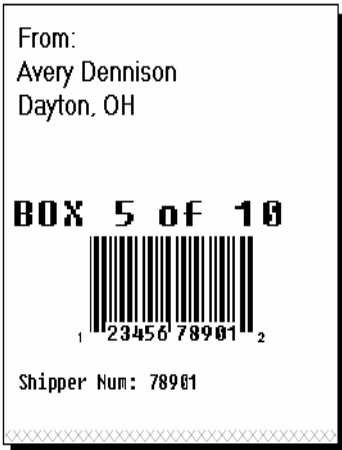
Re-Image ON
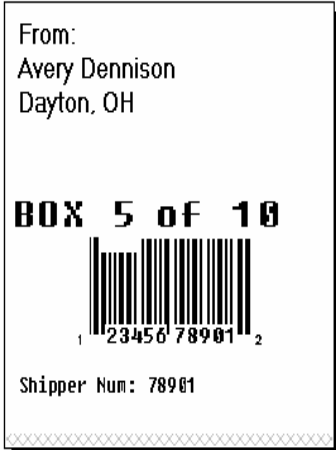
Re-Image OFF

Using Option 62 (Bypass Bar Code)

This option allows the optional bar code verifier to bypass (skip) bar code(s) on a format. The bar code verifier does not scan any bar code with Option 62 applied to it.

Syntax   **R,62 |**

R1. R	Option Header.	
R2. 62	Option **62**. The bar code verifier does not scan the bar code associated with this option.	
Note:	Adjacent bar codes on a format need to start and end on the same row (and be the same height). Both adjacent bar codes (or neither bar code) must be scanned. However, do not apply Option 62 to all bar codes on a format or 764 errors appear.	
	The bar code verifier's beam scans the entire width of the label. It cannot scan half the labels width. The bar code verifier errors if Option 62 is applied to only ONE of the adjacent bar codes.	
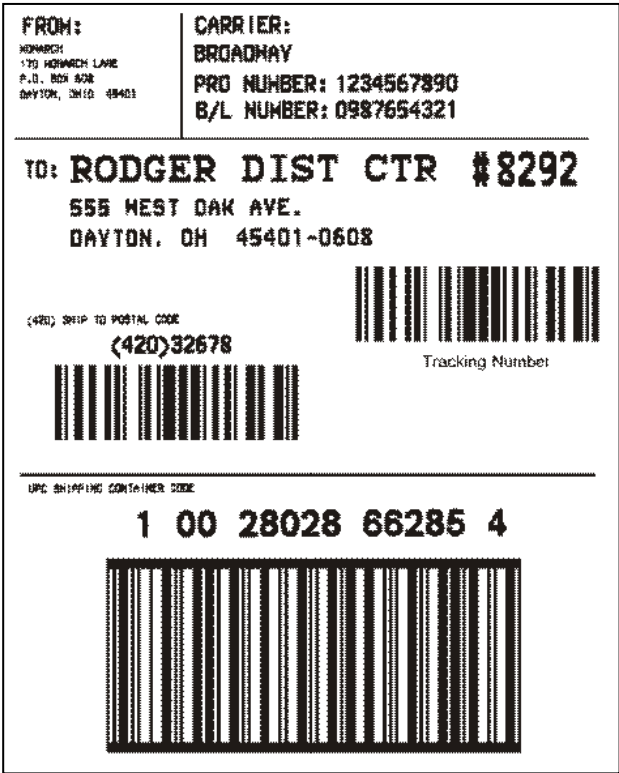
Option 62 can be applied to any bar code on this format.
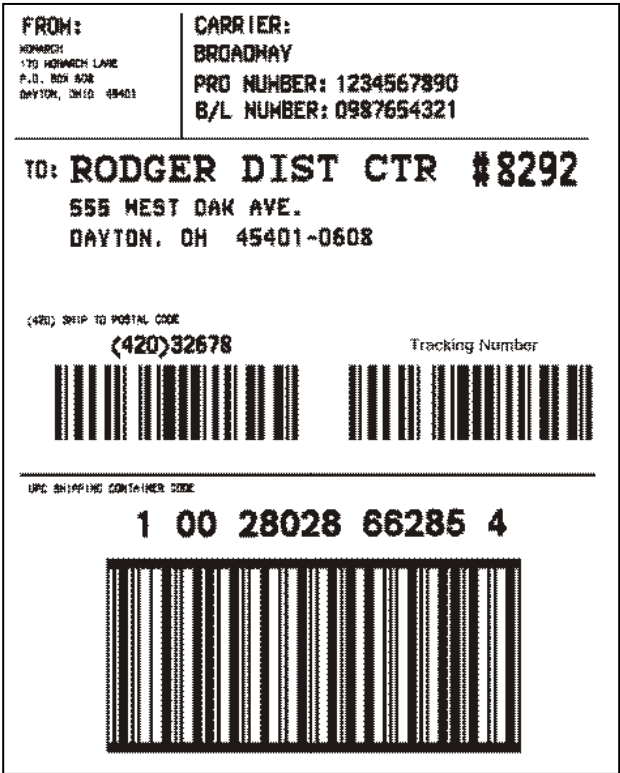
Option 62 must be applied to BOTH adjacent bar codes.

Example   **B,1,12,F,110,115,1,2,120,5,L,0 |**

        R,62 |

The bar code verifier does not scan or verify this UPCA bar code on the format.

Bar Code Verifier Information

♦ There is a 0.50-inch no scan zone on the trailing edge of each label.

♦ You cannot verify adjacent bar codes that do not start and end on the same print row. See the following graphic.

B,3,13,V,310,28,8,4,50,8,L,0| **Postal Code Bar code**
B,4,13,V,355,200,8,4,50,8,L,0 | **Tracking Number Bar code**
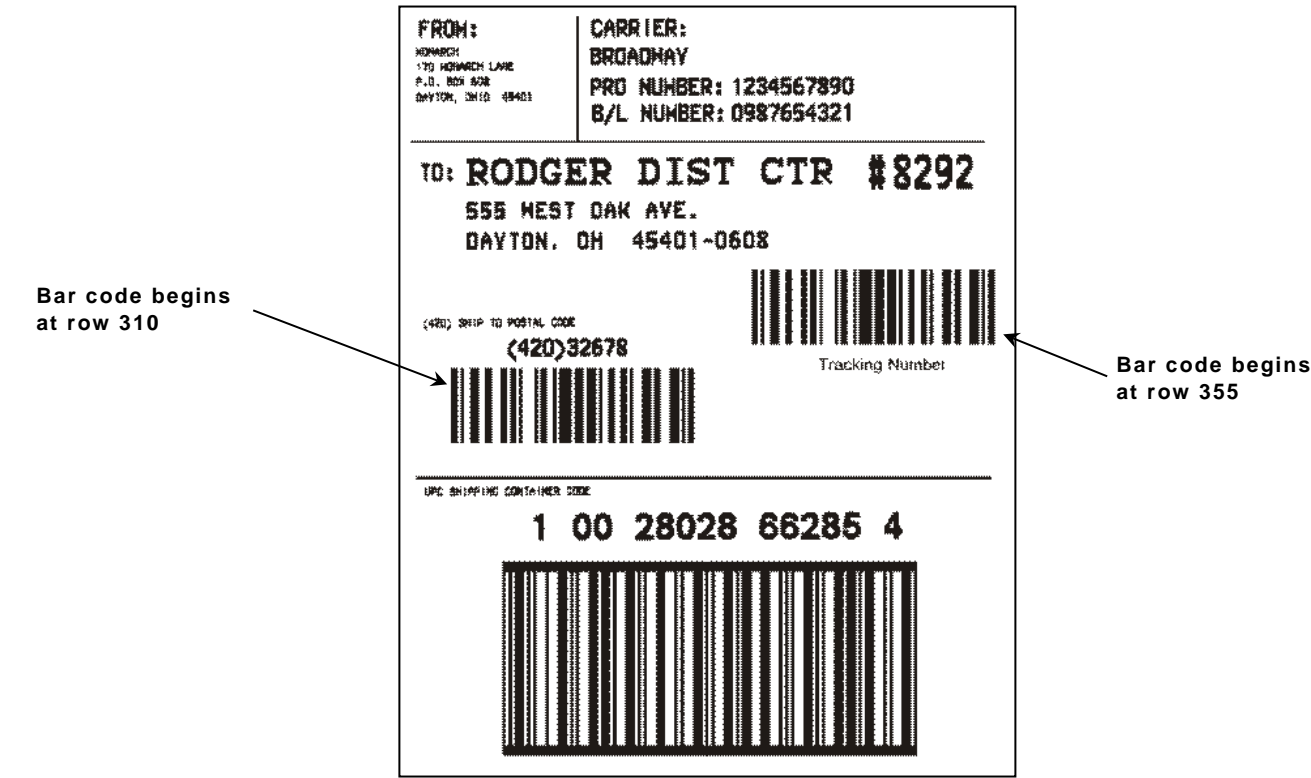
The bar code verifier cannot verify the "postal code" and "tracking number" bar codes, because they do not start and end on the same row.

Using Option 65 (Serialized EPC Data)

Using Version 1.7 or greater firmware, the printer supports serialized EPC data, which combines EPC data with a unique serial number. The unique serial number is always 38 bits of data. The EPC data can contain a Global Trade Item Number (GTIN), but it is not limited only to GTIN data.

The printer calculates the serial number using Multi-Vendor Chip-Based Serialization (MCS). MCS is an industry standard scheme that ensures the uniqueness (no duplication) of serial numbers.

The printer supports serialized EPC data using the following microchips:

- Impinj Monza 4
- Impinj Monza 5
- Impinj Monza 6
- NXP G2iL
- NXP G2iL+
- NXP G2XM
- NXP G2XL
- NXP UCODE7

Note: Refer to the chip's manufacturer for additional details and specifications about serialized EPC data.

When serializing EPC data, the RFID data entered in the batch data is modified before encoding. The printer calculates the serial number and inserts the serial number into the lower 38 bits before encoding. The original data in those lower 38 bits (if there was any) is replaced. We recommend padding the RFID data with zeros to use as placeholders for those 38 bits.

The minimum EPC length is 48 bits (10 bits of EPC data, plus the 38-bit serial number); the maximum EPC length is 496 bits (458 bits of EPC data, plus the 38-bit serial number).

Note: Set the RFID printer for stop-to-encode mode for serialized EPC data. Non-stop encode mode is not supported.

Syntax   **R,65,serial |**

R1. R	Option Header.
R2. 65	Option **65**.
R3. serial	Enter **1** to create serialized EPC data from the RFID data field.

Example   **R,65,1 |**

Enables serialized EPC data for the RFID data field.

Example   **X,5,24 |**
            R,65,1 |

Enables serialized EPC data for the RFID data field #5 that contains 24 ASCII Hex characters, which is a 96-bit RFID inlay (58 bits of EPC data with the last 38 bits as the unique serial number).

RFID Data Field with Serialized EPC Data Option

```
{F,2,A,R,E,400,400,"SerEPC" |
T,1,50,V,10,10,0,1,1,1,B,L,0,0 |
X,2,24,0 |
  R,65,1 | }
{B,2,N,1 |
1,"31323334353637383930" |
2,"31323334353637000000000" |}
```

The RFID data field includes zeroes so the field's maximum number of characters is 96 bits of data (24 ASCII Hex characters). The encoded RFID data contains the first 58 bits of provided EPC data and 38 bits of the calculated serial number.

Using Check Digits

Check digits are typically used to ensure that a text or bar code field scans correctly. The printer calculates a check digit when Option 31 is applied. A check digit scheme determines how the printer calculates a check digit. When defining a check digit scheme, assign a number to identify it. This number is later entered in **R4** (**check digit** #) when applying Option 31 to a field. Use check digits with text or bar code fields. Check digit calculations are performed on numeric data only.

Do not use check digits with price fields. Do not define a check digit scheme for UPC, EAN, Code 39 (with the MOD43 check digit), and Code 93 bar codes, because they have predefined check digits.

Syntax   {A,selector,action,device,modulus,fld_length, D/P,
           "weights" | }

A1. A	Check Digit Header.
A2. selector	Assign a number from **1** to **10** to this check digit formula. **1** is the default.
A3. action	The action to perform. Enter **A** to add the check digit scheme.
A4. device	Format storage device. **R** is the default . Options:
	F Flash (must be formatted first. See "Formatting Flash" and "Flash Memory Guidelines" in Chapter 2 for more information.)
	R Volatile RAM
Note:	Check digits stored in flash are saved when the printer is turned off.
A5. modulus	Number from **2** to **11**. The modulus is used to divide the sum of products or the sum of digits.
A6. fld_length	The maximum number of characters the field will contain. Range: **0** to **2710**. **2710** is the default.
A7. D/P	Algorithm. The algorithm determines how the check digit is calculated. Options:
	D sum of digits
	P sum of products
A8. "weights"	String of digits used for calculation. A weight string is a group of two or more numbers that is applied to a field. The number of digits in this string should equal the number in fld_length. Enclose in quotation marks. Range: **0** to **2710**. **""** is the default.

Example   {A,1,A,R,10,5,P,"65432" | }

Adds check digit scheme number 1 to the printer's memory. The modulus is 10, the maximum number of characters in the field is 5. The check digit is calculated by using the Sum of Products and the string of digits used in the calculation is "65432."

Sum of Products Calculation

This is an example of how the printer uses Sum of Products to calculate a check digit for this data:

5 2 3 2 4 5 2 1 9

1. Weights are applied to each digit, starting with the last digit in the weight string. They are applied right to left, beginning at the right-most position of the field. Remember, a weight string must contain at least two different numbers. This example has a weight string of **1,2,3,4**:

    field: **5 2 3 2 4 5 2 1 9**
    weight string: **4 1 2 3 4 1 2 3 4**

2. Each digit in the field is multiplied by the weight assigned to it:

    field: **5 2 3 2 4 5 2 1 9**
    weight string: **4 1 2 3 4 1 2 3 4**
    products: **20 2 6 6 16 5 4 3 36**

3. Next, the product of each digit is added together. This is the sum of the products.

    20 + 2 + 6 + 6 + 16 + 5 + 4 + 3 + 36 = 98

4. Divide the sum of the products by the modulus (10 in this case), only to the whole number. The balance is called the remainder.

    ```
        9
    10 )98
       90
        8
    ```

5. Subtract the remainder from the modulus.
    The result becomes the check digit. In this case, the check digit is 2.

    10 - 8 = 2

Sum of Digits Calculation

This is an example of how the printer uses Sum of Digits to calculate a check digit for this data:

5 2 3 2 4 5 2 1 9

1. Weights are applied to each digit, starting with the last digit in the weight string. They are applied right to left, beginning at the right-most position of the field. Remember, a weight string must contain at least two different numbers. This example has a weight string of **1,2,3,4**:

field:	5 2 3 2 4 5 2 1 9
weight string:	4 1 2 3 4 1 2 3 4

2. Each digit in the field is multiplied by the weight assigned to it:

field:	5 2 3 2 4 5 2 1 9
weight string:	4 1 2 3 4 1 2 3 4
products:	20 2 6 6 16 5 4 3 36

3. Next, the digits of the products are added together. Two-digit products are treated as two separate digits. This is the sum of the digits.

   2 + 0 + 2 + 6 + 6 + 1 + 6 + 5 + 4 + 3 + 3 + 6 = 44

4. Divide the sum of the digits by the modulus (10 in this case), only to the whole number. The balance is called the remainder.

   $$\begin{array}{r} 4 \\ 10\overline{)44} \\ \underline{40} \\ 4 \end{array}$$

5. Subtract the remainder from the modulus. The result becomes the check digit. In this case, the check digit is 6.

   10 - 4 = 6

CREATING GRAPHICS

This chapter provides information on how to

♦ map out the graphic image using the hexadecimal (hex) or run length method.

♦ create a graphic packet using a graphic header, bitmap, duplicate, next-bitmap, text, constant text, line, and box fields.

♦ place a graphic image into a format.

♦ use the compressed graphics packet.

Use graphic packets to create bitmapped images. To include a graphic packet within a format, the format **must** contain a graphic field. See "Placing the Graphic in a Format" for more information.

Overview of Bitmapped Images

A printed image is formed through a series of dots. Each square on the grid below represents a dot on the printhead. The graphic image is created by blackening dots in a specific pattern. Print shades of gray according to the concentration of dots on the image. When the dots are printed together, the end result is a graphic image.
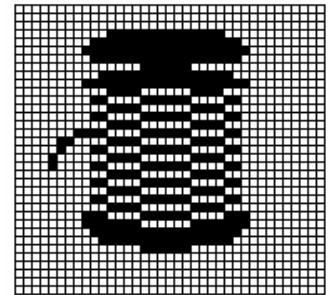
Determining a Method

Use one of two methods to map out the graphic image:

Hex Method
The dot sequences are segmented into binary numbers and then converted to hex numbers.
A graphic using gray-scaling, several slanted lines, or several vertical lines typically translates more efficiently with hex representation.

Run Length Encoding Method
The dot sequences are segmented into black and white strings within a row. The total count of each white string is converted to a lower-case letter, corresponding to numeric value. The total count of each black string is converted to an uppercase letter, corresponding to numeric value. This method can reduce imaging time for graphics that contain repetitive rows of dots. A graphic with horizontal lines or very few white-to-black borders typically translates more efficiently with run length encoding.

The most efficient encoding method depends on how complicated the graphic image is and whether or not imaging time is a concern. Experiment with both encoding methods to get optimal performance.

Designing Bitmapped Images

After determining the encoding method to use, map out the graphic image.

Note: The mapped image must be an upside down mirror image of the final result.
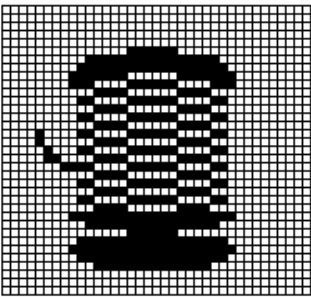
Special Considerations

Solid black print cannot exceed 30% of any given square inch of the supply. If the black print exceeds this limit, you may lose data or damage the printhead.

In the first label, the large "M" logo and thick black line exceed the allowed black to white print ratio. In the second label, the large "M" logo does not exceed the black to white print ratio.

2 inches

Exceeds limit →

Exceeds limit →

MERRY MART
$189.95/ea
LG WINTER COAT

37000 62872

2 inches

Does not exceed limit →

Does not exceed limit →

0.5"

MERRY MART
$189.95/ea
LG WINTER COAT

37000 62872

Using the Hex Method
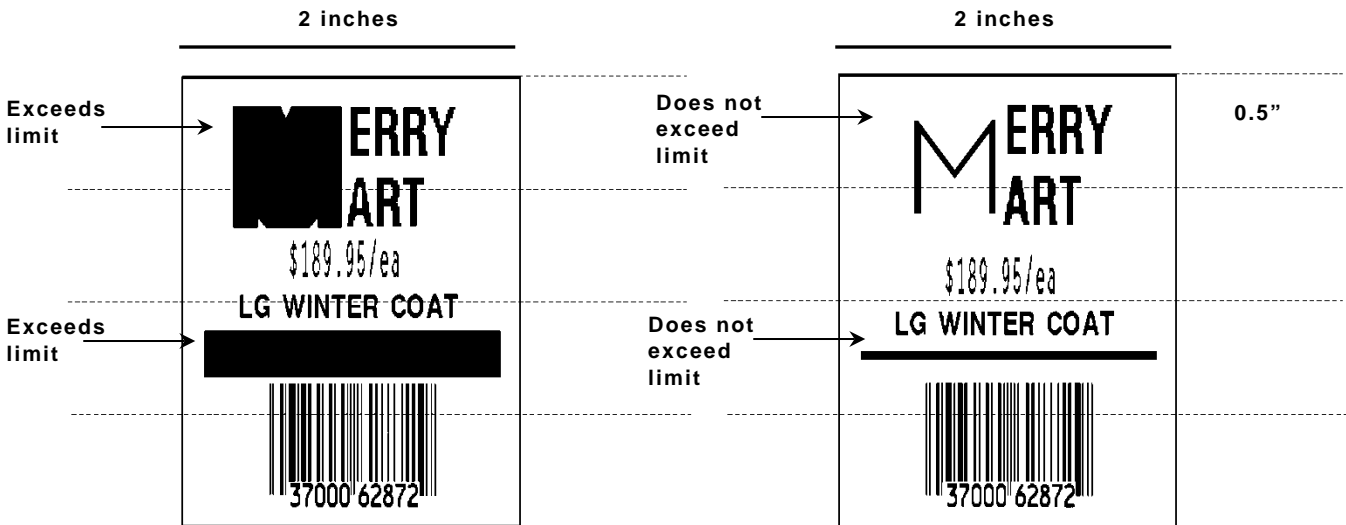
The following steps explain how to derive a hex character string from a bitmapped graphic. Each square on the grid represents a dot. A black square indicates the dot is ON, and a white square indicates the dot is OFF. A sequence of binary numbers, called a bit pattern or bitmap, determines what dots are on and off. The numbers "0" and "1" are used for this purpose. The number "1" turns a dot on and "0" turns a dot off.

All hex numbers must be two digits. For example, write hex 0 as 00, or hex E as 0E.

1. Assign 1 to every black square and 0 to every white square.

```
0000000000000000000000000000000000000000000000011111111111111111111111110000000000000000000000
0000000000000000000000000000000000000001111111111111111111111111111111111111110000000000000000
0000000000000000000000000000000000011111100000000000000000000000000000000000001111110000000000
0000000000000000000000000000000111110000000111111111111111111111111111111110000000111100000000
0000000000000000000000000011111000011111111111111111111111111111111111111111111000001111100000
0000000000000000000000001100001111111111111111111111111111111111111111111111111111000001100000
0000000000000000000001100011111111111111111111111111111111111111111111111111111111111111111110
0000000000000000000111111111111111111111110000000000000000001111111111111111111111111111111100
0000000000000000011111111111111111111000000000000000000000000001111111111111111111111111111000
0000000000000000111110000011111111111111111111111111111111111100000011000000000000
0000000000000011110000000111111111111111111111111111111111111110000000111000000000
0000000000000011000000111111111111111111111111111111111111111110000001111100000000
0000000000000011111111111111111111111111111111111111111111111111000001111100000000
0000000000000011100000111111111111111111111111111111111111111110000001111000000000
0000000000000000011111111111111111111111111111111111111111111000000000011111000000
0000000000000000111110000011111111111111111111111111111111111110000001111100000000
0000000000000011110000000111111111111111111111111111111111111110000001111100000000
0000000000000011000000111111111111111111111111111111111111111110000001111000000000
0000000000000011111111111111111111111111111111111111111111111111000001111100000000
0000000000000011100000111111111111111111111111111111111111111110000001111000000000
0000000000000000011111111111111111111111111111111111111111111000000000011111000000
0000000000000000111110000011111111111111111111111111111111111110000001111100000000
0000000000000011110000000111111111111111111111111111111111111110000001111100000000
0000000010000011000000111111111111111111111111111111111111111110000001111000000000
0000000111000011111111111111111111111111111111111111111111111111000001111100000000
0000001111110000000000111111111111111111111111111111111111111110000001111000000000
0000000111110000000000011111111111111111111111111111111111111111000001111000000000
```

2. Section off the grid in columns of eight. If any rows are not divisible by 8, add enough 0's to complete a column.

```
00000000|00000000|00000000|00000000|00000001|11111111|11000000|00000000|00000000|00001111|11111000|00000000|00000000
00000000|00000000|00000000|00000000|00000000|00111110|00000000|00111111|11111111|11100000|00000111|11100000|00000000
00000000|00000000|00000000|00000011|11000000|00000000|00111111|11111111|11111111|11100000|00000000|00011110|00000000
00000000|00000000|00000001|01111100|00111111|11111111|11111111|11111111|11111111|11111111|11111111|11100001|11110000
00000000|00000000|00000001|10000011|11111111|11111111|11111111|11111111|11111111|11111111|11111111|11111110|00001100
00000000|00000000|00000001|10001111|11111111|11111111|11111111|11111111|11111111|11111111|11111111|11111111|11111110
00000000|00000000|00000001|11111111|11111111|11111111|11111000|00000000|00000000|11111111|11111111|11111111|11111110
00000000|00000000|00000001|11111111|11111111|11110000|00000000|00000111|11111111|11111111|01111111|11111111|11111100
00000000|00000000|00000000|01111111|11000000|00000000|00000111|11111111|11111111|00000000|00000000|00111111|11110000
00000000|00000000|00000000|00011111|11000000|00000000|00000111|11111111|11111111|00000000|00000000|00111111|11000000
00000000|00000000|00000000|00011100|00000011|11111111|11111111|11111111|11111111|11111110|00000001|11000000
00000000|00000000|00000011|11111111|11100000|00000000|00000000|00000000|01111111|11111110|00000000
00000000|00000000|00000011|00011111|11000000|00000011|11111111|11111111|11111111|11111110|00011111|11000000
00000000|00000000|00000000|00011100|00000011|11111111|11111111|11111111|11111111|11111110|11000001|11000000
00000000|00000000|00000000|00000000|00111111|11111111|11111111|11111111|11111111|11100001|11000000
00000000|00000000|00000011|11111111|11100000|00000000|00000000|00000000|01111111|11111110|00000000
00000000|00000000|00000000|00011111|11000000|00000011|11111111|11111111|11111111|11111110|00011111|11000000
00000000|00000000|00000000|00011100|00000011|11111111|11111111|11111111|11111111|11111110|11000001|11000000
00000000|00000000|00000000|00000000|00111111|11111111|11111111|11111111|11111111|11100001|11000000
00000000|00000000|00000011|11111111|11100000|00000000|00000000|00000000|01111111|11111110|00000000
00000000|00000000|00000000|00011111|11000000|00000011|11111111|11111111|11111111|11111110|00011111|11000000
00000000|00000000|00000000|00011100|00000011|11111111|11111111|11111111|11111111|11111110|11000001|11000000
00000000|00000000|00000000|00000000|00111111|11111111|11111111|11111111|11111111|11100001|11000000
00000000|00000000|00000011|11111111|11100000|00000000|00000000|00000000|01111111|11111110|00000000
00000011|00000000|00000011|11000000|00011100|00000011|11111111|11111111|11111111|11111110|00011111|11000000
00000001|11100000|00000011|11000000|00000011|11111111|11111111|11111111|11111111|11111110|11000001|11000000
00000000|11111100|00000000|00011100|00000011|11111111|11111111|11111111|11111111|11111111|11100011|11000000
00000000|11111110|00000000|00000000|00111111|11111111|11111111|11111111|11111111|11100011|11000000
```

3. One row at a time, convert each group of eight binary digits to hex.

starting at position 49 ...
00111111 = 3F
11111111 = FF
11111111 = FF
11110000 = F0

4. Write the hex values for each row as a continuous string.

row 1, position 49 = 03FFFFFF00000

5. Repeat steps **3** through **4** for each row on the grid.

6. Insert the hex values in syntax format.

Using the Run Length Encoding Method

The following steps explain how to derive a run length character string from a bitmapped graphic.

Each square on the grid represents a dot. A black square indicates the dot is ON, and a white square indicates the dot is OFF.

Note: The following example shows "1" to indicate when a square is ON, and "0" to indicate when a square is OFF. Dots do not have to be converted when using the run length method.

```
00000000 00000000 00000000 00000000 00000000 00000000 00111111 11111111 11111111 11110000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000001 11111111 11000000 00000000 00000000 00001111 11111000 00000000
00000000 00000000 00000000 00000000 00111110 00000000 00111111 00000000 00000000 00000000 00001111 11000000 00000000
00000000 00000000 00000000 00000011 11000000 00111111 11111111 11111111 11111111 11111111 00000000 00011110 00000000
00000000 00000000 00000000 01111100 00111111 11111111 11111111 11111111 11111111 11111111 11111111 11100001 11110000
00000000 00000000 00000001 10000011 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111110 00001100
00000000 00000000 00000001 10001111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111110
00000000 00000000 00000001 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111110
00000000 00000000 00000001 11111111 11111111 11110000 00000000 00000000 11111111 11111111 11111111 11111111 11111110
00000000 00000000 00000001 11111111 11111111 11110000 00000000 00000111 11111111 11111111 00000000 11111111 11111100
00000000 00000000 00000000 01111111 10000000 00000000 00000111 11111111 11111111 00000000 00000000 00111111 11110000
00000000 00000000 00000000 00011111 11000000 00000011 11111111 11111111 11111111 11111111 00000000 00011111 11000000
00000000 00000000 00000000 00011100 00000011 11111111 11111111 11111111 11111111 11111110 00000001 11000000
00000000 00000000 00000000 00000000 00111111 11111111 11111111 11111111 11111111 11111111 11111110 00000000
00000000 00000000 00000000 00000011 11111111 11100000 00000000 00000000 00000000 00000111 11111111 11111110 00000000
00000000 00000000 00000000 00011111 11000000 00000000 00000111 11111111 11111111 00000000 00000000 00011111 11000000
00000000 00000000 00000000 00011100 00000011 11111111 11111111 11111111 11111111 11111111 11111110 00000001 11000000
00000000 00000000 00000000 00000011 11111111 11111111 11111111 11111111 11111111 11111111 11100001 11111110
00000000 00000000 00000000 00000001 11111111 11110000 00000000 00000111 11111111 11111111 00000000 00000000 00011111 11000000
00000000 00000000 00000000 00011111 11000000 00000000 00000111 11111111 11111111 00000000 00000000 00011111 11000000
00000000 00000000 00000000 00011100 00000011 11111111 11111111 11111111 11111111 11111111 11111110 00000001 11000000
00000000 00000000 00000000 00000011 11111111 11100000 00000000 00000000 00000000 00000111 11111111 11111110 00000000
00000000 00000000 00000000 00011111 11000000 00000000 00000111 11111111 11111111 00000000 00000000 00011111 11000000
00000000 00000000 00000000 00011100 00000011 11111111 11111111 11111111 11111111 11111111 11111110 00000001 11000000
00000000 00000000 00000000 00000011 11111111 11111111 11111111 11111111 11111111 11111111 11100001 11111110
00000000 00000000 00000000 00000001 11111111 11110000 00000000 00000111 11111111 11111111 00000000 00000000 00011111 11000000
00000000 00000000 00000000 00011111 11000000 00000000 00000111 11111111 11111111 00000000 00000000 00011111 11000000
00000000 00000000 00000000 00011100 00000011 11111111 11111111 11111111 11111111 11111111 11111110 00000001 11000000
00000010 00000000 00000000 00011111 11000000 00000000 00000111 11111111 11111111 00000000 00000000 00011111 11000000
00000011 00000000 00000000 00011100 00000011 11111111 11111111 11111111 11111111 11111111 11111110 00000001 11000000
00000001 11100000 00000000 00000000 00111111 11111111 11111111 11111111 11111111 11111111 11100001 11000000
00000000 11100000 00000000 00000011 11111111 11100000 00000000 00000000 00000000 00000111 11111110 00000000
00000000 11111100 00000000 00011100 00000011 11111111 11111111 11111111 11111111 11111110 00000011 11000000
00000000 11111110 00000000 00000000 00111111 11111111 11111111 11111111 11111111 11111111 11100011 11000000
```

1. Count the number of consecutive OFF or ON dots in a row. Write the number of consecutive dots in sequence for the first row on the grid. Write "ON" or "OFF" after each number to indicate ON or OFF dots.

    (row 1, position 50) 26 on
    (row 2, position 39) 11 on, 26 off, 9 on
    (row 3, position 34) 5 on, 45 off, 6 on

2. Replace each number with its corresponding code from the Dot to Run Length Encoding Chart provided at the end of this chapter. Use CAPITAL letters for black dots and lower-case letters for white dots.

    26 on (Z)11 on (K), 26 off (z), 9 on (I)..

    If the number is greater than 26, write z, followed by the letter corresponding to the amount over 26. For example, to represent 45 off dots, write zs.

    5 on (E), 45 off (zs), 6 on (F)..

3. Write the letter codes in sequence, uninterrupted, for each row.

    (row 1,position 50) Z
    (row 2,position 39) KzI
    (row 3,position 34) EzsF
    (row 4,position 30) DpZoD..

Note: If the end of the line specifies OFF dots (lower-case letters), the ending lower-case letters can be omitted. For example, uZFu can be written as uZF.

4. Repeat steps **1** through **5** for each row on the grid.

5. Insert the code values in syntax format.

Determining How to Store the Image

Once the image is mapped out, determine how to store it:

♦ Flash

♦ Volatile RAM

♦ Temporary Storage

Using Flash

Use flash memory when the graphic image is used by several formats, because the graphic only has to be sent once. This eliminates the need to send the graphic image repeatedly. See "Placing the Graphic in a Format," for more information about using the graphic packet in a format.

Flash memory must be formatted before use. From the Main Menu, select Setup, Flash Memory, then, Format Flash. The process takes a few minutes. **Graphics stored in flash memory are saved when the printer is turned off**. To clear flash memory, refer to the *System Administrator's Guide*.

Using Volatile RAM

Use RAM when the graphic image is used by several formats, because the graphic is sent once. This eliminates the need to send the graphic image repeatedly. See "Placing the Graphic in a Format," for more information about using the graphic packet in a format. Store graphics smaller than approximately 1/2 inch by 1/2 inch in printer RAM and reference by the graphic ID number.

Note: Graphics are stored in the format buffer and remain there until another graphic packet is sent or the printer is turned off.

Using Temporary Storage

Use temporary storage when the graphic image is used only in one format or the graphic image is very large. Graphic data in temporary storage is held in the image buffer until the graphic is printed and then it is cleared from memory. Temporary graphics are also cleared from memory when a new batch is sent or updated. Use the same graphic image multiple times on a format. Send the graphic image to the printer after the format to which it applies.

If a graphic is stored in temporary storage, do not place a graphic field in the format. This will cause an error. Instead, position the graphic image by using the row and column locations in the graphic packet header. Image memory (temporary storage) will accept a graphic packet 1218 rows long with 811 dots per row.

Creating a Graphic Packet

A graphic packet can contain

♦ bitmapped fields (for bitmapped images)

♦ constant text fields

♦ lines

♦ boxes

Images using hex representation or run length encoding are bitmapped images. See "Designing Bitmapped Images" to design a bitmapped image.

Once the image is designed, define a graphic packet. This packet generates the graphic image to use in a format.

Positioning the Graphic Image

This section explains how to position the graphic image within a graphic packet header, a field of a graphic packet, or within a format.

Within the Graphic Packet Header

When using RAM, the row and column parameters in the graphic header are usually **0,0**, because placement is controlled by the graphic field in the format. This is especially true when designing a compliance label overlay.

When using temporary storage, these parameters control the placement of the graphic image on the supply.

The area enclosed within the dotted lines represents the graphic image starting at **0,0** (as defined in the graphic header).
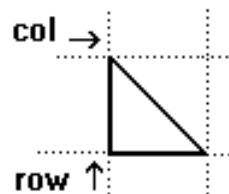
To create a fixed amount of white space around the graphic image, use something other than 0 for row and/or column. The area enclosed within the dotted lines represents the graphic image starting at **0,0** with a fixed amount of white space (**10,10**) around the graphic image.
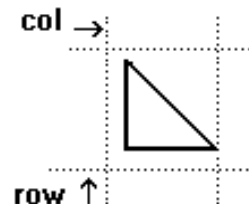
Within the Field

In a bitmap, constant text, line, or box field, the row and column parameters control where an individual field or bitmapped row begins in relation to the coordinates defined in the graphic header.

The bottom of the triangle in this example represents the first field of the graphic packet starting at **10,0**.
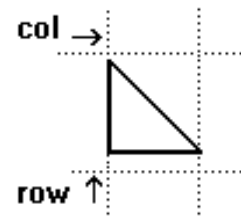
Within a Format

When defining the graphic field within the format, the *row* and *column* parameters represent where on the **format** to place the graphic image.

To place a graphic (a logo, for example) within a certain area on the supply, enter the starting position (bottom left corner) of the graphic image.

This label shows the triangle "logo" beginning (the bottom left corner) at **400, 75** as defined in the graphic field.

Defining the Graphic Header

Every graphic packet must contain a graphic header. It identifies and provides important measurement and formatting information for the graphic. Bitmap, duplicate, next-bitmap, constant text, line, and box fields follow the graphic header, if they are used.

Syntax   {G,graphID,action,device,units,row,col, mode,"name" |

G1. G	Graphic Header.	
G2. graphID	Unique number from **1** to **999** to identify the graphic image. **1** is the default.	
G3. action	Use **A** to add the graphic to the printer.	
G4. device	Graphic storage device. **R** is the default.	
	F	Flash (saved when the printer is turned off)
	R	Volatile RAM
	T	Temporary storage
G5. units	Unit of measure. **G** (dots) is the default.	
	E	(English - in 1/100 inches)
	M	(Metric -in 1/10 mm)
	G	(Graphic - in dots)

G6. row   Distance between the *bottom* of the graphic image area and the first bitmap line. This is usually **0**, unless you want a fixed amount of white space around the graphic image. **10** is the default. See "Positioning the Graphic Image," for more information.

English	**0 – 1599**
Metric	**0 – 4063**
203 Dots	**0 – 3247**
300 Dots	**0 – 3599**

G7. col   Distance between the *left edge* of the graphic image area and the left edge of first bitmap line. This is usually **0**, unless you want a fixed amount of white space around the graphic image. **10** is the default. See "Positioning the Graphic Image," for more information.

English	**0 – 409**
Metric	**0 – 1039**
203 Dots	**0 – 831**
300 Dots	**0 – 1247**

G8. *mode*	Imaging mode. Enter **0**.	
G9. *"name"*	Graphic name (optional), **0** to **8** characters, enclose within quotation marks. "" is the default.	

Example   {G,99,A,R,G,0,0,0,"99Wire" |

Adds a graphic image identified by number 99 to volatile RAM. The graphic uses dot measurement. The image will be placed according to the row and column parameters in the graphic field. The imaging mode is 0 and the image is called 99Wire.

Creating Bitmap Fields

This defines one row of dots, starting at a specific row and column within the graphic image. Each unique row of dots requires a bitmap field. Repeat bitmap fields with a duplicate field.

Syntax   B,row,col,algorithm,"data" |

B1. *B*	Bitmap Field.	
B2. *row*	Distance (in dots) from the graphic image's bottom margin to the bitmap line. **10** is the default.	
	English	**0 – 1599**
	Metric	**0 – 4063**
	203 Dots	**0 – 3247**
	300 Dots	**0 – 3599**
B3. *col*	Distance (in dots) from the graphic image's left edge to the bitmap line. **10** is the default.	
	English	**0 – 409**
	Metric	**0 – 1039**
	203 Dots	**0 – 831**
	300 Dots	**0 – 1247**
B4. *algorithm*	Coding method for bitmap data:	
	H Hex Representation	
	R Run Length Encoding	
B5. *"data"*	Character string made up of hex or run length encoding. Do not put spaces or any other character between the numbers. Range: **0** to **2710**. **2710** is the default.	

Example   B,39,56,H,"3FFFFFF0" |

Defines a bitmapped graphic field. The image begins 39 dots from the bottom and 56 dots from the left edge of the graphic area. Hex representation is used.

Creating Next-Bitmap Fields

This field uses the previous field's row and column locations. Use the bitmap or duplicate field data without having to recalculate row and column locations. This field represents one row of dots on the image.

Syntax   **N,adjdir,adjamt,algorithm,"data" |**

N1. N	Next-Bitmap Field.	
N2. adjdir	Increments or decrements the row count. Inserts the duplicate line after or before the current row. **0** is the default.	

                                               0   Increments (inserts duplicate line after)
                                               1   Decrements (inserts duplicate line before) For example:
                                             B,50,35,R,"GsSsG" |
                                           N,0,1,R,"DpZoD" |
                                           prints a next-bitmap field on row 51 at column 35.

N3. adjamt	Row adjustment in dot rows. Using **0** overwrites the same line. Range: **0** to **999**.	
N4. algorithm	Coding method for bitmap data:	

                                               H   Hex Representation
                                             R   Run Length Encoding

N5. "data"	Character string made up of hex or run length encoding. Do not put spaces or any other character between the hex numbers or run length code letters. Range: **0** to **2710**. **2710** is the default.	

Example   **B,39,56,H,"3FFFFFF0" |**
                   N,0,1,H,"000000E00000" |

Defines a next-bitmap graphic field beginning on row 40. The row count increments by 1. Hex representation is used.

Creating Duplicate Fields

If a line of data is identical to a previous bitmap or next-bitmap field, the duplicate field allows repeating the dot sequence without retyping the data. A duplicate field represents one row of dots on the image.

Note: Duplicate fields are useful when a graphic has a lot of repetition.

Syntax   **D,adjdir,adjamt,count |**

D1. D	Duplicate Field.	
D2. adjdir	Increments or decrements the row count. Inserts the duplicate line after or before the current row.	

                                               0   Increments (inserts after)
                                             1   Decrements (inserts before)
                                           For example:
                                           B,50,35,R,"GsSsG" |
                                           D,0,20,2 |
                                           inserts row 50 again at row 70 and row 90. Rows 70 and 90 do not have to be defined later.

D3. adjamt	Amount of row adjustment in dot rows. Range: **0** to **999**. **1** is the default. The above example adjusts the duplicate field to image on row 70 and 90 (adding 20 to the current row count).	
D4. count	Number of times to duplicate the line. Range: **0** to **999**.	

Example **B,117,24,H,"03FFFFFFFFFFFFFFFFFFFC" |**
 D,0,1,2 |

Defines a duplicate field that is imaged after the bitmap line. This field duplicates the preceding bitmap line twice (at row 118 and 119).

Use constant text, line, or box fields in a graphic packet to create a compliance label overlay. See Chapter 3, "Defining Fields," for more information about these fields.

Sample Hex Graphic Packet

{G,99,A,R,G,0,0,0,"99WIRE" |
B,39,48,H,"3FFFFFF0" |
B,40,32,H,"01FFC000000FF8" |
B,41,32,H,"3E00000000000FC0" |
B,42,24,H,"03C0003FFFFFF0000F" |
B,43,24,H,"7C3FFFFFFFFFFFFFFE1F0 |
B,44,16,H,"0183FFFFFFFFFFFFFFFF06" |
B,45,16,H,"018FFFFFFFFFFFFFFFFFFE" |
B,46,16,H,"01FFFFFFFFFFFFFFFFFFFE" |
B,47,16,H,"01FFFFFF80001FFFFFFFFE" |
B,48,16,H,"01FFFF0000000007FFFFC" |
B,49,24,H,"7F800007FFFF00003FF0" |
B,50,24,H,"1FC00007FFFF00001FC0" |
D,0,4,4 |
B,51,24,H,"1C03FFFFFFFFFFFE01C0" |
D,0,4,4 |
B,52,32,H,"3FFFFFFFFFFFFFE1C0" |
D,0,4,4 |
B,53,24,H,"03FFF0000000007FFE" |
D,0,4,4 |
B,70,0,H,"0400001FC00007FFFF00001FC0" |
B,71,0,H,"0600001C03FFFFFFFFFFFE01C0" |
B,72,0,H,"030000003FFFFFFFFFFFFFE1C0" |
B,73,0,H,"01000003FFF0000000007FFE" |
B,74,8,H,"FC001C03FFFFFFFFFFFE00C0" |
B,75,8,H,"FE00003FFFFFFFFFFFFFE0C0" |
B,76,8,H,"1FF803FFF0000000007FFE" |
B,77,8,H,"0FFFCFFC0000000000001C0" |
B,78,16,H,"FFDF000FFFFFFFFF8003C0" |
B,79,16,H,"7FFFC00007FFFF00001FC0" |
B,80,24,H,"1C03FFFFFFFFFFFE01C0" |
D,0,4,4 |
B,81,32,H,"3FFFFFFFFFFFFFE1C0" |
D,0,4,4 |

B,82,24,H,"03FFF0000000007FFE" |
D,0,4,3 |
B,83,24,H,"1FC00007FFFF00001FC0" |
D,0,4,3 |
B,98,24,H,"03FFFFFFFFFFFFFFF0" |
B,99,24,H,"07FFFFFFFFFFFFFFFC" |
B,100,24,H,"1FF9FFFFFFFFFFFFFF" |
B,101,24,H,"3FFE0007FFFF8000FF80" |
B,102,24,H,"391E0027FFFF803FFC0" |
B,103,24,H,"1C7FFFFFFFFFFFFFFFFC0" |
B,104,24,H,"1FC1FFFFFFFFFFFF1FC0" |
B,105,24,H,"0FFDFFFFFFFFFFE0FF" |
B,106,24,H,"FFFFFFFFFFFFFF8" |
B,107,32,H,"3FFFFFFFFFFFFFE0" |
B,108,32,H,"03FFFFFFFFFFFF" |
B,109,48,H,"07FFFF80" |
D,0,1,2 |
B,111,48,H,"FFFFFFFF" |
B,112,32,H,"FFFF00000000FFE0" |
B,113,24,H,"078000FFFFFFFF001F" |
B,114,24,H,"78FFFFFFFFFFFFFE060" |
B,115,16,H,"0187FFFFFFFFFFFFFFFC18" |
B,116,16,H,"027FFFFFFFFFFFFFFFFFF2" |
B,117,16,H,"03FFFFFFFFFFFFFFFFFFFC" |
D,0,1,2 |
B,120,16,H,"01FFFFFFFFFFFFFFFFFFF8" |
B,121,24,H,"FEFFFFFFFFFFFFFFFE0" |
B,122,24,H,"07FFFFFFFFFFFFFFFC" |
B,123,32,H,"FFFFFFFFFFFFFC0" |

Sample Run Length Graphic Packet

```
{G,99,A,R,G,0,0,0,"99WIRE" |           B,81,34,R,"ZZEdC" |
B,39,50,R,"Z" |                        D,0,4,4 |
B,40,39,R,"KzI" |                      B,82,30,R,"NzkN" |
B,41,34,R,"EzsF" |                     D,0,4,4 |
B,42,30,R,"DpZoD" |                    B,83,27,R,"GsSsG" |
B,43,25,R,"EdZZEdE" |                  D,0,4,4 |
B,44,23,R,"BeZZMeB" |                  B,98,30,R,"ZZJ" |
B,45,23,R,"BcZZW" |                    B,99,29,R,"ZZM" |
B,46,23,R,"ZZZA" |                     B,100,27,R,"JbZZE" |
B,47,23,R,"ZDsZE" |                    B,101,26,R,"MnToI" |
B,48,24,R,"TzkU" |                     B,102,26,R,"CbHnTiP" |
B,49,25,R,"HtRqJ" |                    B,103,27,R,"CcZZC" |
B,50,27,R,"GsSsG" |                    B,104,27,R,"GeZWcG" |
D,0,4,4 |                              B,105,28,R,"JaZReH" |
B,51,27,R,"ChZWgC" |                   B,106,32,R,"ZZI" |
D,0,4,4 |                              B,107,34,R,"ZZE" |
B,52,34,R,"ZZEdC" |                    B,108,38,R,"ZQ" |
D,0,4,4 |                              B,109,53,R,"T" |
B,53,30,R,"NzkN" |                     D,0,1,2 |
D,0,4,4 |                              B,111,48,R,"ZF" |
B,70,5,R,"AuGsSsG" |                   B,112,33,R,"PzfK" |
B,71,5,R,"BtChZWgC" |                  B,113,29,R,"CpZBoE" |
B,72,6,R,"DxZZEdC" |                   B,114,25,R,"DcZZGfB" |
B,73,7,R,"CtNzkN" |                    B,115,23,R,"BdZZMeB" |
B,74,8,R,"FmChZWhC" |                  B,116,22,R,"AbZZVbA" |
B,75,8,R,"GsZZEdC" |                   B,117,22,R,"ZZZB" |
B,76,11,R,"JiNzkN" |                   D,0,1,2 |
B,77,12,R,"NbJzzeC" |                  B,120,23,R,"ZZZ" |
B,78,16,R,"JaElZKmD" |                 B,121,25,R,"ZZV" |
B,79,17,R,"QsSsG" |                    B,122,29,R,"ZZM" |
B,80,27,R,"ChZWgC" |                   B,123,32,R,"ZZF" |
D,0,4,4 |                              B,124,39,R,"ZT" | }
```


Placing the Graphic in a Format

To include a graphic within a format:

1. Design the graphic image as shown in "Designing Bitmapped Images."

2. If using RAM, place a graphic field in the format file to reference the graphic. See the following section, "Defining the Graphic Field," for more information.

Note: If using temporary storage, there is no need to add a graphic field to reference the graphic image.

3. Download all the necessary packets (check digit, format, etc.).

4. Send the graphic file to the printer. See "Creating a Graphic Packet" for more information.

Defining the Graphic Field

The graphic field in a format references the graphic image by the graphID in the graphic header. This field is required only if the graphic will be stored in RAM.

Syntax   **G,graphID,row,col,mode,rotation |**

G1. G	Graphic Field.
G2. graphID	Unique number from **1** to **999** to identify the graphic image. **1** is the default.
G3. row	Distance between the *bottom* of the print area on the supply to the bottom of the graphic image. **10** is the default. Measured in selected units.

English	**0 – 1599**
Metric	**0 – 4063**
203 Dots	**0 – 3247**
300 Dots	**0 – 3599**

The row specified in the constant text, bitmap, line, or box field is added to the *row* value above to determine the actual position in the format.

G4. column	Distance between the *left edge* of the print area on the supply and the left edge of the graphic. **10** is the default. Measured in selected units. The column specified in the constant text, bitmap, line, or box field is added to the *col* value above to determine the actual position in the format.

English	**0 – 409**
Metric	**0 – 1039**
203 Dots	**0 – 831**
300 Dots	**0 – 1247**

G5. mode	Imaging mode. Enter **0**.
G6. rotation	The orientation of the graphic on the supply. Enter **0**.

Example   **G,57,0,0,0,0 |**

Defines a graphic field that is identified by the number 57. The image begins at 0,0. The imaging mode is 0 and there is no rotation.

Sample Bitmap Graphic Image

The following format shows the wire hex graphic packet included.

{F,2,A,R,E,400,400,"FMT2" |
G,99,227,35,0,0 | ⟵——————— **Reference to the graphic packet**
Q,240,15,300,125,10," " |
T,1,5,V,285,137,0,10,2,2,B,L,0,0,0 |
T,2,5,V,255,137,0,10,2,2,B,L,0,0,0 |
T,3,15,V,180,25,0,10,1,2,B,L,0,0,0 |
T,4,15,V,121,35,0,1,3,1,B,L,0,0,0 |
L,S,94,15,94,235,10,"" |
B,5,12,F,50,65,1,2,40,1,L,0 | }

Sample Batch Packet

{B,2,N,1 |
1,"Pat's" |
2,"Parts" | **4,"3.55/8 Pack" |**
3,"3/8 inch Wire" | **5,"345911871209" | }**
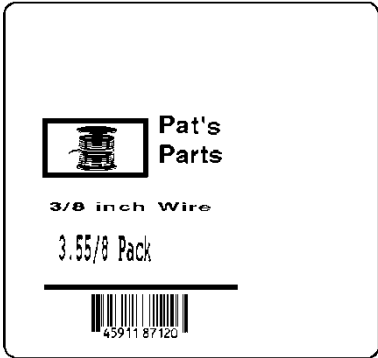
About the Compressed Graphics Packet

The compressed graphics packet is created by label design software when a format is saved in LZ77 compressed format with a MIME wrapper. Some label design software sends data to the printer as a graphic, so using the compressed graphics packet provides faster imaging and printing.

Note: This packet is included for reference only since it is created by label design software.

Syntax **{E,graph_id,action,device,units,row,col,#row,#col, size, *"name"* |**
D,*"LZ77 data or MIME data"* | }

E1. E	Enhanced (Compressed) Graphics Packet	
E2. graph_id	Unique number from **1** to **999** to identify the graphic image. **1** is the default.	
E3. action	Use **A** to add the graphic to the printer.	
E4. device	Use **T** to add the graphic to temporary storage.	
E5. units	Unit of measure. **G** (dots) is the default.	

     E   (English - in 1/100 inches)
     M   (Metric -in 1/10 mm)
     G   (Graphic - in dots)

E6. row   Distance between the *bottom* of the print area on the supply to the bottom of the graphic image. **10** is the default. Measured in selected units.

English	**0 – 1599**
Metric	**0 – 4063**
203 Dots	**0 – 3247**
300 Dots	**0 – 3599**

E7. col   Distance between the *left edge* of the print area on the supply and the left edge of the graphic. **10** is the default. Measured in selected units. The column specified in the constant text, bitmap, line, or box field is added to the *col* value above to determine the actual position in the format.

English	**0 – 409**
Metric	**0 – 1039**
203 Dots	**0 – 831**
300 Dots	**0 – 1247**

E8. #row	Number of rows in the data. Range: **1** to **32000**. **1** is the default.
E9. #col	Number of columns in the data. Range: **1** to **32000**. **1** is the default.
E10. size	Size of the data in bytes. Range: **1** to **32000**. **1** is the default.
E11. "name"	Graphic name (optional), **0** to **8** characters, enclose within quotation marks. **""** is the default.

Example **{E,99,A,T,E,10,10,25,50,256, *"LOGO"* |**
D,"datadatadata" | }

Adds a graphic image, identified by number 99 to temporary storage. The graphic uses English measurements. The image is placed at row 10 and column 10. The image contains 25 rows and 50 columns of data. The file size is 256 bytes and the image is called LOGO.

Dot to Run Length Encoding Chart

ON (Black) Dots

# of Dots	Code	# of Dots	Code
1	A	14	N
2	B	15	O
3	C	16	P
4	D	17	Q
5	E	18	R
6	F	19	S
7	G	20	T
8	H	21	U
9	I	22	V
10	J	23	W
11	K	24	X
12	L	25	Y
13	M	26	Z

Off (White Dots)

# of Dots	Code	# of Dots	Code
1	a	14	n
2	b	15	o
3	c	16	p
4	d	17	q
5	e	18	r
6	f	19	s
7	g	20	t
8	h	21	u
9	i	22	v
10	j	23	w
11	k	24	x
12	l	25	y
13	m	26	z

PRINTING

This chapter describes how to

♦ download files to the printer

♦ define the batch header, batch control, and batch data files

♦ create batch files. The printer accepts files in MPCLII, MLI, or XML format.

Turn on the printer and make sure it is ready to receive data before downloading files. See "Downloading Methods" for information on ways to download.

When downloading, send packets in this order:

1. Memory configuration packet (M)

2. Configuration packets (A-X)

3. Any of the following:

♦ Check digit packets (see Chapter 4)

♦ Format packets (see Chapter 3)

♦ Graphic packets (see Chapter 5)

4. Batch data (see "Defining the Batch Header")

Downloading Files

There are several ways to download files to the printer.

Make sure the communication settings at the host match those at the printer.

Port	Connection
♦ Serial port	Connect the printer to the PC with a serial cable. Send the communication settings packet to select the printer's communication settings. See "Defining the Communication Settings Packet" in Chapter 2 for more information.
	Use the Command prompt, terminal emulator (communications) software, such as Tera Term or our MPCL Toolbox software to download files.
	Command Prompt Example
	COPY LABEL1.FMT COM1 **Transmits a file called "LABEL1.FMT" to COM1.**
	COPY LABEL1.BCH COM1 **Transmits a batch called "LABEL1.BCH" to COM1.**
	If using the COPY command to download formats, set flow control to DTR (not XON/XOFF).
♦ USB port	Connect the printer to the PC with a USB cable. Use the Command prompt or terminal emulator (communications) software, such as Tera Term to download files.
♦ Ethernet	Connect the printer with an Ethernet cable and use MonarchNet2 or other software to download files.
♦ 802.11 a/b/g/n	Connect to the printer using MonarchNet2™ or other software to download files. Refer to the *MonarchNet2 Operating Instructions* for more information

About Batch Packets (Print Jobs)

A batch packet containing batch data is required for all print jobs.

Batch data is the actual information printed on the supply. Batch data fills in the format's text, bar code, and non-printable text fields.

A batch packet consists of the following:

batch header	identifies the format and how many labels to print.	For example: **{B,1,N,1|**
batch control	defines the print job. For example:	For example: **E,0,1,1,1,3,0,0,0,1|**
batch data (optional)	defines the information printed on the label.	For example: **1, #,"Size 5"	}**

Label Design software may create this packet.

Defining the Batch Header

The batch header specifies which format the batch uses and how many labels to print.

Syntax	**{B,format#,N/U,quantity |**	
	B1. B	Batch Header.
	B2. format#	Format number (**1** to **999**) to use. **1** is the default.
	B3. N/U	Controls how image is generated. **N** is the default.
		N New. Erase image and re-image all fields using online data. Any missing fields will be blank.
		U Update last image with one or more fields. All other fields remain the same as the last queued batch.
	B4. quantity	Quantity to print (0 to 32000). **1** is the default. Using **0** pre-images the field to reduce the imaging time for labels. See "Batch Quantity Zero Method" for more information.

Example **{B,1,N,5 |**

Defines a batch header that uses format #1 and reimages all fields using the online data. Five labels are printed in this batch.

Defining the Batch Control Field

The batch header must precede this field. The batch control field defines the print job and applies only to the batch that immediately follows.

The batch control field overrides the settings made at the printer. For example, the printer's cut mode is set to disabled; however, the batch control field specifies to cut each tag, so each tag in the batch is cut.

The batch control field also overrides settings made in the online configuration packets. For example, the Supply Setup Packet is set for liner take-up mode; however, the batch control field specifies continuous mode, so continuous mode is used for the batch.

Syntax   **E,feed_mode,batch_sep,print_mult,multi_part,cut_mode,cut_mult, ver_mode,cable_det,rotate_img |**

E1. E		Batch Control Field.
E2. feed_mode		Feed Mode. **0** is the default. Options:

0   Continuous Feed
1   On-Demand
2   Liner take-up (optional)

Note: For peel mode, the minimum supply length is 1.50 inches.

E3. batch_sep   Batch Separator (striped label in between batches). **0** is the default. Options:

0   Does not print a separator
1   Prints a separator
2   Double-length separator – prints 2 tags

Note: Do not use batch separators with continuous (non-indexed) supply. If using a stacker, the batch separator is always 3.66 inches long.

E4. print_mult   Number of tags (**1** to **999**) with the same image. **1** is the default.

E5. multi_part   Number of identical parts on one tag (**1** to **5**). **1** is the default.

E6. cut_mode   Enables or disables the optional cutter. Since the cutter is 3.4 inches away from the printhead, printed tags may be left between the printhead and cutter. **0** (does not cut) is the default.

Printed tags left between the printhead and cutter - Options:

1   Cuts each tag.
2   Cuts in strips, not each tag.
4   Does not cut before first tag, cuts each tag.

None *or* ***one*** *printed tag left between the printhead and cutter - Options:*

3   Cuts each tag.
5   Cuts in strips, not each tag.

Description of cut mode operation:

Cut Mode	Cut Before first tag in Batch?	Cut Between Tags?	Cut After Batch?	Printed tags left between printhead and cutter?
0	No	No	No	No
1	Yes	Yes	Yes*	Yes*
2	Yes	No (Feeds strips)	Yes*	Yes
3	Yes	Yes	Yes**	No*
4	No	Yes	Yes*	Yes
5	Yes	No (Feeds strips)	Yes**	No*

No* Cut Modes 3 and 5 minimize the number of tags left between the printhead and the cutter. Depending on the length of supply being cut, there may be one tag left between the printhead and the cutter.

Yes* The last tag in the batch is queued to be cut once it reaches the cutter; however, there may be printed tags left between the printhead and the cutter. The user must send another batch to feed the last tag out far enough to be cut.

Note: Using "overstrike/continue" as the Error Action minimizes the number of RFID tags left between the printhead and cutter. To use the least amount of RFID tags, set overstrike to five.

Recommended Cutter Operation:

Printhead Density	With RFID Supply	With Stacker Attached	Recommended Cut Modes
203	No	No	1, 2, 4 – **Note**: Cut Modes 3 and 5 may cause ribbon tearing
203	No	Yes	1, 2, 4 – **Note:** Cut Modes 3 and 5 may cause ribbon tearing
203	RFID is not supported with 203 dpi		
203	RFID is not supported with 203 dpi		
300*	No	No	1, 2, 3, 4, 5
300*	No	Yes	1, 2, 3, 4, 5
300*	Yes	No	1, 2, 4
300*	Yes	Yes	3, 5 – **Note:** Some RFID supplies may curl and cause a jam (with cut mode 3). If this happens, use cut mode 1.

* Contact Technical Support to order replacement 300 dpi printheads (KST) for use with the cutter and RFID supplies.

> *E7. cut_mult*   Number of tags to print before cutting. A cut multiple of one cuts after each tag. The range is 0 to 32,000. **0** is the default. The cut multiple is a multiple of the print quantity. If the cut multiple is three and the print quantity is 16, then five sets of three tags and one set of one tag is produced.

> **Note**:   The last tag in the batch is always cut, regardless of the multiple.

E8. ver_mode	Bar code verifier mode. Enables or disables the bar code verifier. For more information about enabling the bar code verifier, refer to the bar code Verifier's *Operating Instructions*. **0** is the default. Options:	
	0	Disable bar code verifier - not using a bar code verifier
	1	Enable bar code verifier
E9. cable_det	Bar code verifier cable detect. The printer can be set to detect when a bar code verifier's cable is disconnected. For more information about connecting the bar code verifier's cables, refer to the bar code Verifier's *Operating Instructions*. **0** is the default. Options:	
	0	The printer does not detect when a cable is disconnected
	1	I/O cable is disconnected
	2	I/O and/or data cables are disconnected
E10.rotate_img	Image rotation. Rotates the printed image 180°. Design formats as needed, so image rotation is not required. **0** is the default. Options:	
	0	No rotation
	1	180° rotation
Note:	The format *length* and *width* must exactly match the loaded supply or the rotated image does not print correctly.	

Example **E,0,1,4,2,3,0,0,0,0** |

Defines a batch control field. Continuous feed mode is used and a separator prints between batches. Four tags have the same image and there are two identical parts on one tag. The cutter cuts after every tag. The image is not rotated.

Defining Batch Data Fields

Batch data fields should be sent in field number order. Use continuation fields for large amounts of data. If using **N** (New) in the batch header, list all fields with the data in sequence. If using **U**, list only those fields and data that changes from the last printed batch.

Syntax **field#,"data string"** |
 C,"continuation" |

field#	Identifies the text, bar code, or non-printable text field in which to insert the following data. Range: **1** to **999**.
"*data string*"	Provides the actual information to appear in fields. Enclose in quotation marks. Length: **0** to **2710** characters. "" is the default.
Note:	When using Option 65 (Serialized EPC Data), the amount of data entered here matches the number of bits for the final output (including the 38-bit serial number). Pad data with zeros to ensure the correct amount of bits.
C	Identifies information to be appended to the data string. This parameter is optional.
"*continuation*"	Provides the actual information to be added to the batch packet. Enclose in quotation marks. Use this option to break up longer fields. Length: **0** to **2710** characters. "" is the default. This parameter is optional.

Example   1,"Size 12" |
              2,"" |
              3,"Blue" |
              C,"and this would be appended." |

Defines a batch data field. "Size 12" prints in field #1, a blank line appears in field #2, "Blue and this would be appended" prints in field #3.

Merged or Sub-Fields

If a field is completely filled by data copied from other fields, use quotation marks without spaces between them for the "*data string*" parameter.

Incrementing Fields

In incrementing fields, the first number in the sequence must contain the same number of digits as the highest number to be counted. For example, to increment the numbers in a field from **1** to **999**, enter the starting number in the batch as **001**.

Creating Gen2 RFID Data

Printing over the RFID inlay may cause printing irregularity.

Do not use skip index with RFID supplies. Unexpected results may occur.

The RFID printer supports EPC Gen2 data, which contains these fields:

- ♦ EPC Data   The printer supports EPC Class-1 Generation-2 UHF (C1Gen2) protocol encoding following the *EPC™ Radio-Frequency Identity Protocols Class 1 Generation 2 UHF RFID Protocol for Communications at 860-960MHz Standards* (GS1 EPCglobal™ Specification for RFID Air Interface).

- ♦ User   Many different RFID supplies (tags) are available and the amount of
  Memory   programmable user memory varies with the chip embedded in the tag. Depending on the tag type, all memory fields may not be available. Refer to the *Monarch® RFID Printer Setup Utility* (http://printers.averydennison.com/en/home/resources/service-and-support/rfid-printer-setup-utility.html) for details about the available user memory for each chip.

- ♦ TID Field   A portion of the TID field contains a unique number that identifies the
  Tag Identifier   manufacturer of the tag. This field is readable and writable.

- ♦ Access   Controls when new data can be written to a field.
  Password

- ♦ Kill   Sets a tag up to be inoperable.
  Password

- ♦ Lock Code   Contains the locking method for each field. There are four locking methods

Lock Code Methods:

Value	EPC Lock	Description
0	No lock (**un**secure)	The selected memory fields (EPC, user memory, access password and kill password) are readable and writable. The tag can be programmed multiple times.
1	Permalock (permanently **un**secure)	Permanently locked in a writable state. The tag can be programmed multiple times. User memory and kill password are also readable.
2	Password lock (secure)	Requires the access password to rewrite the selected memory fields. The tag can be programmed multiple times *with the password*. EPC and user memory are writable with password and readable. Access password is never readable. Kill password is only writable with password.
3	Permalock and Password lock (permanently locked)	Never rewritable, but always readable. The tag can never be rewritten, once locked. EPC and user memory are never rewritable, but are readable. Access password and kill password are never readable or rewritable.

Note: The EPC field is always readable, no matter what locking method is assigned.

The RFID settings in the Easy Setup Menu and the RFID Printer Setup Utility are based on encoding 96-bits of EPC data with a lock code. If your RFID data is more complex than 96-bits of EPC data with a lock code, you may need to decrease the printer's speed in non-stop encode mode. See "Defining the Print Control Packet" to decrease the print speed. The following examples may require a decreased print speed:

♦ encoding 96-bits of EPC data and programming user memory

♦ encoding more than 96-bits of EPC data with a lock code

♦ encoding more than 96-bits of EPC data and programming user memory

To use Gen2 data, modify the RFID Field's batch data.

Syntax   **field#,"EPC_data~028" |**
        C,"User_Mem~028" |
        C,"TID_Mem~028" |
        C,"Acs_Pwd~028" |
        C,"Kill_Pwd~028" |
        C,"Lock_Code" |

field#	Identifies the RFID Data Field number for the following data. Range: **1** to **999**. **1** is the default.
"EPC_data~028"	EPC data. Enclose in quotation marks. To create EPC data, follow the guidelines in the *EPC Global Generation 1 Tag Data Standards Specification*. The data **must** be in ASCII Hex format specified in the RFID Data Field (*data_type*). The data must end with the ASCII field separator (decimal 028). The printer can program up to 496 data bits (124 ASCII Hex characters) and increment EPC data.
Note:	The amount of programmable EPC data, user memory, access password, and lock code data varies by *print speed* and RFID chip. If the printer cannot program all the RFID data within the allotted time, an error occurs.

C,"User_Mem~028"   User Memory. Enclose in quotation marks. The amount of programmable user memory varies with the tag types.   Refer to the Monarch® RFID Printer Setup Utility for details about the available user memory for each chip.

The printer programs up to 512-bits of data (128 ASCII Hex characters). The data must be ASCII Hex characters and end with the ASCII field separator (decimal 028).

According to the *Tag Data Standards Specification* requirements, an even number of characters (bytes) must be sent for user memory. For more information, refer to the *Tag Data Standards Specification* on the GS1 website: https://www.gs1.org/epc-rfid

Using version 2.1 or greater firmware, if an odd number of characters are sent, the last position is **automatically** encoded with a Hex 0x00 or a Null character (this character is padded on the right to make an even number of characters).

Prior to version 2.1 firmware, if an odd number of characters (bytes) are sent, the printer *errors*, because it did not receive the expected even number of characters. Prior to version 2.1 firmware, the printer does **not** automatically pad data.

Note: The amount of programmable EPC data, user memory, access password, and lock code data varies by *print speed* and RFID chip. If the printer cannot program all the RFID data within the allotted time, an error occurs.

C,"TID_Mem~028"   Identifies information to be appended. Reserved for future use. Only include the field separator (decimal 028) in this field.

C,"Acs_Pwd~028"   Access Password. This must be 8 ASCII Hex characters. No password is assigned if this field is left blank. The data must end with the ASCII field separator (decimal 028).

C,"Kill_Pwd~028"   Kill Password. This must be 8 ASCII Hex characters. No password is assigned if this field is left blank. The data must end with the ASCII field separator (decimal 028).

C,"Lock_Code"   Five-digit locking method for each field in this order:
EPC Data, User Memory, TID, Access password, and Kill password. Locking options:
0   No Lock (unsecure)
1   Permalock (permanently unsecure)
2   Password lock (secure)
3   Permalock & password lock (permanently locked)

Note: Use only one locking method per field. The printer is not capable of unlocking a field.

Do not use the ASCII file separator character (1C in ASCII Hex) in RFID batch data!

Example   **F,2,A,R,E,400,400,"PERMLOCK" |**
 X,1,100,0 |
 B,2,N,1 |
 1,"31323334353637383 1323334~028" |
 C,"ABCDEF~028" |
 C,"~028" |
 C,"73737373~028" |
 C,"CAD01234~028" |
 C,"11101" | }

The EPC data is 31323334353637383 1323334, the user memory data is ABCDEF, the access password is 73737373, and the kill password is CAD01234. Selects permalock (1) as the locking method for the EPC, user memory, TID, and kill field. Selects no lock (0) for the access password field. The EPC, user memory, TID, and kill fields are **permanently** readable/writable.

Entering Batch Data for QR Code

QR Code requires certain parameters at the beginning of all batch data.

Syntax   **"error_cor mask# data_input,char"**

	error_cor	Level of error correction. Some damaged or dirty bar codes may still be scannable if the error correction is high enough. Options:

    H   Ultra high reliability level
    Q   High reliability level
    M   Standard level
    L   High density level (more data in the bar code)

    Note: When increasing the error correction level, the maximum number of characters (in the field) decreases.

mask#   Mask number. Undefined. Use **0**.

data_input   Type of data input. Options:

    A   Automatic
    M   Manual

    Note: When using **A**utomatic, do not end with a comma or specify the next parameter for *char* (type of characters).

char   Type of characters. This parameter is only required when *data_input* is **M**anual. Options:

    A   Alphanumeric
    B   Binary
    K   Kanji
    N   Numeric

    Note: In binary mode, the number of characters must be represented by the 4-digit number in decimal.

Example

{F,1,A,R,E,200,200,"QRURL" |
B,2,200,V,75,50,36,0,100,2,B,0 | }
{B,1,N,1 |
2,"LA testdatainAutomode0987654321" | }

Sets the QR Code's error correction level to L, which provides high density (more data in bar code); leaves the mask number blank; sets the data input mode to Automatic and the data is testdatainAutomode0987654321.

Example   1,"HM,N0123456789012345" |

Sets the QR Code's error correction level to H, which provides very high reliability; leaves the mask number blank; sets the data input mode to Manual; the type of characters are Numeric and the data is 0123456789012345.

Structured Append Mode

QR Code offers a mode called structured append (or concatenated) that collects data from multiple QR Code symbols and uses that data elsewhere. For example, the components of a sub-assembly can have individual QR Codes and the QR Code for the entire assembly contains all the data from the individual codes. This mode also requires certain parameters at the beginning of all batch data.

Syntax   **"mode_id code# #of_div parity, error_cor mask# data_input char"**

mode_id	Mode identifier. Use **D** to indicate the structured append (or concatenated) mode.
code#	Code number of the individual symbol in the concatenated set. Use a two-digit number in decimal.
#of_div	Total number of symbols in this concatenated set. Use a two-digit number in decimal.
parity	Parity byte. Use a two-digit number in hexadecimal. There is no standard parity byte.
error_cor	Level of error correction. Some damaged bar codes may still be scannable if the error correction is high enough. Options:

    H   Ultra high reliability level
    Q   High reliability level
    M   Standard level
    L   High density level

Note: When increasing the error correction level, the maximum number of characters (in the field) decreases.

mask#	Mask number. Undefined. Leave blank or use **0**.
data_input	Type of data input. When using **A**utomatic, do not specify the next parameter for *char* (type of characters). Options:

    A   Automatic
    M   Manual

char	Type of characters. This parameter is only required when *data_input* is **M**anual. Options:

    A   Alphanumeric
    B   Binary
    K   Kanji
    N   Numeric

Note: In binary mode, the number of characters must be represented by the 4-digit number in decimal.

Example   1,"D0205E9,Q0A," |
           C,"B006qrcode," | }

Defines the structured append mode (D) for QR Code. This is symbol (02) of a concatenated set containing (05) symbols. The parity byte is E9. The error correction level is Q, which provides a high reliability. Use 0 for the mask number. The data input mode is Automatic. The type of characters are binary (B) and there will be six (06) data characters (qrcode).

Structured Append QR Code Packet

{F,2,A,R,E,200,200,"QRCODE2" |
B,1,200,V,50,50,36,0,100,2,B,0 | }
{B,2,N,1 | 1,"D0202E9,Q0A" |
C,"0123456789ABCD+__âôû
~129~064~159~252~224~064" | }

Using Special Characters in Batch Data

There are two ways to specify special characters in batch data:

♦ Place a tilde (~) before each character

♦ Use a tilde with the decimal ASCII equivalent

For example, use "" or ~034 to print the " character in the batch data; otherwise, the tilde characters are ignored. You can also use ~XXX where XXX is the decimal equivalent of an unprintable character.

Sample Batch Data with Special Characters

{B,1,N,1		Decimal Character	What Prints
1,"123~034456789"		~034 is "	123"456789
2,"~094983~'126LG4451"		~094 is ^~126 is ~	^983~'LG4451

Downloading Methods

Download the format and batch data using one of three methods: sequential, batch, and batch quantity zero.

Sequential Method

Using the sequential method, send all format and batch data at one time. Use this method when the application does not require operator intervention to input data. All data is sent down at one time, and the printer then images each field. As soon as the last field is imaged, labels begin to print.

Example   {Format}
            {Batch Packet}

Batch Method

This is similar to the sequential method, but it is used to send multiple batches. All data for the first batch is sent at one time, and the printer then images each field. As soon as the last field for the first batch is imaged, labels begin to print. This process is repeated for each subsequent batch.

Example   {Format}
            {Batch Packet}
            {Batch Packet}

Batch Quantity Zero Method

Use the batch quantity zero method when the application requires operator intervention to enter data. While the operator is entering data, the previous field is sent with a batch quantity of zero. The printer images the field, but does not print it. After the operator enters the data for the last field, the batch quantity can be specified. The last remaining field is imaged, and the label prints almost immediately.

To use the batch quantity zero method:

1. Send the format and a batch header in one file. The first time the batch header is sent, use the parameter **N** (new batch), and the parameter **0** for (zero quantity). This ensures the label is properly positioned.

    The printer images constant text, line, and box fields, but does not print them.

2. Input data for each field, and send it with a batch header using the parameter **U** (batch update) and a quantity of zero. When the printer receives the data, it immediately images the field, but does not print it.

    At this time, the printer is imaging all associated fields, including fields that copy from other fields.

3. Repeat step 2 for each field except the last one.

4. For the last field, input data and send a batch header with the quantity of labels to print. When the printer receives input for the last field, it immediately prints the labels. See "Reducing Imaging Time" in Chapter 9 for an example using the batch quantity zero method.

Modifying Formats

The optional entry method is a quick way to modify format fields, check digit fields and configuration packets.

Optional Entry Method

This method resets only the parameters that change. Commas act as placeholders for unchanged parameters. The optional entry method reduces file size and increases the speed at which files are sent to the printer.

Creating Batch Files for Downloading

If downloading from a Command prompt, create batch files to set communication values and download formats. It is a good idea to create a subdirectory to hold the format files.

Here is a batch file that sets a serial port, changes to a subdirectory, and downloads a check digit file, format file, and batch data file.

```
MODE COM1: 9600,N,8,1,|
CD\MONARCH
COPY LABEL1.CDS COM1
COPY LABEL1.FMT COM1
COPY LABEL1.BCH COM1
```

STATUS POLLING

This chapter explains how to use status polling.

There are two types of Status Polling:

♦ Inquiry Request—information about the readiness of the printer.

♦ Job Request—information about the current (or last received) job downloaded to the printer.

Inquiry Request (ENQ)

An ENQ character acts as a request for printer status information. Send an ENQ in front of, in the middle of, or immediately following any packet downloaded to the printer. An ENQ is a command that can be executed as part of a packet or sent on its own (using a communications program). An ENQ is processed immediately. The ENQ character is user defined.

The ENQ character does not appear as a visible character; however, this manual represents the ENQ character as ▓.

Inquiry Response

Printer status is returned to the host in a 3-byte (3-character) sequence. The first byte is the non-printable user-defined ENQ character, which is not visible on the response. The second and third bytes are printer status codes. See the ENQ Reference Tables for the meaning of bytes 2 and 3.

Example: ▓AB

The status codes (A and B in this case) are ASCII equivalents to the hexadecimal bits that represent the various types of status responses. This response indicates that the printer is online (Character A) and that there is a stock fault (Character B).

Example: ▓??

Indicates that this is the first ENQ response since the printer was turned on. Send another ENQ immediately to receive the printer's status.

Example: ▓@@

Indicates the printer is offline.

The following graphics can be used as a quick reference for the Status of Byte #2 and Byte #3. Byte #1 is the non-printable user-defined ENQ character.

Status Byte #2

bit 7 6 5 4 3 2 1 0

| 0 | 1 | × | × | × | × | × | × |

→ online
→ active
→ busy
→ online data error
→ operator correctable error
→ hardware failure
→ constant ON
→ constant OFF

Status Byte #3

bit 7 6 5 4 3 2 1 0

| 0 | 1 | × | × | × | × | × | × |

→ online error
→ stock error
→ ribbon error
→ waiting to dispense label
→ format error
→ low battery
→ constant ON
→ constant OFF

Char	Const. OFF	Const. ON	Comp. Failure	Corr. Error	Online Data Error	Busy	Active	Online
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
@	0	1	0	0	0	0	0	0
A	0	1	0	0	0	0	0	1
B	0	1	0	0	0	0	1	0
C	0	1	0	0	0	0	1	1
D	0	1	0	0	0	1	0	0
E	0	1	0	0	0	1	0	1
F	0	1	0	0	0	1	1	0
G	0	1	0	0	0	1	1	1
H	0	1	0	0	1	0	0	0
I	0	1	0	0	1	0	0	1
J	0	1	0	0	1	0	1	0
K	0	1	0	0	1	0	1	1
L	0	1	0	0	1	1	0	0
M	0	1	0	0	1	1	0	1
N	0	1	0	0	1	1	1	0
O	0	1	0	0	1	1	1	1
P	0	1	0	1	0	0	0	0
Q	0	1	0	1	0	0	0	1
R	0	1	0	1	0	0	1	0
S	0	1	0	1	0	0	1	1
T	0	1	0	1	0	1	0	0
U	0	1	0	1	0	1	0	1
V	0	1	0	1	0	1	1	0
W	0	1	0	1	0	1	1	1
X	0	1	0	1	1	0	0	0
Y	0	1	0	1	1	0	0	1
Z	0	1	0	1	1	0	1	0
[	0	1	0	1	1	0	1	1
\	0	1	0	1	1	1	0	0
]	0	1	0	1	1	1	0	1
^	0	1	0	1	1	1	1	0
_	0	1	0	1	1	1	1	1
`	0	1	1	0	0	0	0	0

Note: A "1" indicates the bit is turned on. A "0" indicates the bit is off.

Char	Const. OFF	Const. ON	Comp. Failure	Corr. Error	Online Data Error	Busy	Active	Online
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
a	0	1	1	0	0	0	0	1
b	0	1	1	0	0	0	1	0
c	0	1	1	0	0	0	1	1
d	0	1	1	0	0	1	0	0
e	0	1	1	0	0	1	0	1
f	0	1	1	0	0	1	1	0
g	0	1	1	0	0	1	1	1
h	0	1	1	0	1	0	0	0
i	0	1	1	0	1	0	0	1
j	0	1	1	0	1	0	1	0
k	0	1	1	0	1	0	1	1
l	0	1	1	0	1	1	0	0
m	0	1	1	0	1	1	0	1
n	0	1	1	0	1	1	1	0
o	0	1	1	0	1	1	1	1
p	0	1	1	1	0	0	0	0
q	0	1	1	1	0	0	0	1
r	0	1	1	1	0	0	1	0
s	0	1	1	1	0	0	1	1
t	0	1	1	1	0	1	0	0
u	0	1	1	1	0	1	0	1
v	0	1	1	1	0	1	1	0
w	0	1	1	1	0	1	1	1
x	0	1	1	1	1	0	0	0
y	0	1	1	1	1	0	0	1
z	0	1	1	1	1	0	1	0
{	0	1	1	1	1	0	1	1
|	0	1	1	1	1	1	0	0
}	0	1	1	1	1	1	0	1
~	0	1	1	1	1	1	1	0
Dec 127	0	1	1	1	1	1	1	1

Note: A "1" indicates the bit is turned on. A "0" indicates the bit is off.

Char	Const. OFF	Const. ON	Low Battery	Format Error	Waiting to Dispense Label	Ribbon Fault	Stock Fault	Online Error
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
@	0	1	0	0	0	0	0	0
A	0	1	0	0	0	0	0	1
B	0	1	0	0	0	0	1	0
C	0	1	0	0	0	0	1	1
D	0	1	0	0	0	1	0	0
E	0	1	0	0	0	1	0	1
F	0	1	0	0	0	1	1	0
G	0	1	0	0	0	1	1	1
H	0	1	0	0	1	0	0	0
I	0	1	0	0	1	0	0	1
J	0	1	0	0	1	0	1	0
K	0	1	0	0	1	0	1	1
L	0	1	0	0	1	1	0	0
M	0	1	0	1	1	0	0	1
N	0	1	0	0	1	1	1	0
O	0	1	0	0	1	1	1	1
P	0	1	0	1	0	0	0	0
Q	0	1	0	1	0	0	0	1
R	0	1	0	1	0	0	1	0
S	0	1	0	1	0	0	1	1
T	0	1	0	1	0	1	0	0
U	0	1	0	1	0	1	0	1
V	0	1	0	1	0	1	1	0
W	0	1	0	1	0	1	1	1
X	0	1	0	1	1	0	0	0
Y	0	1	0	1	1	0	0	1
Z	0	1	0	1	1	0	1	0
[	0	1	0	1	1	0	1	1
\	0	1	0	1	1	1	0	0
]	0	1	0	1	1	1	0	1
^	0	1	0	1	1	1	1	0
_	0	1	0	1	1	1	1	1
`	0	1	1	0	0	0	0	0

Note: A "1" indicates the bit is turned on. A "0" indicates the bit is off.

Char	Const. OFF	Const. ON	Low Battery	Format Error	Waiting to Dispense Label	Ribbon Fault	Stock Fault	Online Error
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
a	0	1	1	0	0	0	0	1
b	0	1	1	0	0	0	1	0
c	0	1	1	0	0	0	1	1
d	0	1	1	0	0	1	0	0
e	0	1	1	0	0	1	0	1
f	0	1	1	0	0	1	1	0
g	0	1	1	0	0	1	1	1
h	0	1	1	0	1	0	0	0
i	0	1	1	0	1	0	0	1
j	0	1	1	0	1	0	1	0
k	0	1	1	0	1	0	1	1
l	0	1	1	0	1	1	0	0
m	0	1	1	0	1	1	0	1
n	0	1	1	0	1	1	1	0
o	0	1	1	0	1	1	1	1
p	0	1	1	1	0	0	0	0
q	0	1	1	1	0	0	0	1
r	0	1	1	1	0	0	1	0
s	0	1	1	1	0	0	1	1
t	0	1	1	1	0	1	0	0
u	0	1	1	1	0	1	0	1
v	0	1	1	1	0	1	1	0
w	0	1	1	1	0	1	1	1
x	0	1	1	1	1	0	0	0
y	0	1	1	1	1	0	0	1
z	0	1	1	1	1	0	1	0
{	0	1	1	1	1	0	1	1
|	0	1	1	1	1	1	0	0
}	0	1	1	1	1	1	0	1
~	0	1	1	1	1	1	1	0
Dec 127	0	1	1	1	1	1	1	1

Note: A "1" indicates the bit is turned on. A "0" indicates the bit is off.

A Job Request returns status information about the most recently processed print job. Send a job request after an ENQ or batch. There are two levels of Job Requests:

♦ Numeric Error Codes Only (0, 1, or 2)

♦ Verbose (3 or 4)

Syntax   **{J,#}**

J1. J		Job Request Identifier.
J2. #		Request number. Options:
	0	Returns ASCII coded strings or numeric error codes
	1	Returns ASCII coded strings or numeric error codes
	2	Returns ASCII coded strings or numeric error codes
	3	Returns error number
	4	Returns number of labels printed in batch

Example **{J,3}**

The job response may not be immediate. If the printer has an error (out of supplies, ribbon problem, etc.), is in pause mode, or has insufficient memory, correct the problem and then resend the job request. If the problem is not corrected, no response is returned. If a formatting error has occurred, the job request returns the status. The printer interprets the format and batch data before returning the response.

Note: Press **FEED** (if necessary) to return the job response.

To clear an error, press **ENTER**. An ENQ can also clear errors numbered less than 500. Once the error is corrected, a job request can be sent. The printer cannot accept another job request until the error is resolved.

Job Response

The Job Response varies, depending on the type of request sent to the printer. The following syntax is the response for a Job 0, 1,or 2 request.

Syntax   **{J,Status1,Status2,"FMT-1","BCH-2"}**

J1. J	Job Response Identifier.
J2. Status1	These errors stop the print job. Examples include out of stock, supply faults, or data formatting errors. See "Job Status Responses."
J3. Status2	These are errors in the syntax of the MPCL data stream. Printing does not stop, but the information may not print properly. See "Job Status Responses."
J4. "FMT-1"	Returns the format number.
J5. "BCH-2"	Returns the batch number.

Here is an example of a response returned to a J 0, 1, 2 request:

   {J,8,0,"FMT-1","BCH-2"}

Indicates that a portion of the format extends off the tag in format 1, batch 2. Refer to the "Job Status 0, 1, 2 Response Table" later in this chapter for brief explanations for J, 0, 1, 2 requests. In the above example, refer to error 8 for an explanation.

The following syntax is the response for a Job 3 request. Press **FEED** (if necessary) to return the job response.

Syntax   {J,"Status1 A,B","Status2
                A,B,C,D,E","FMT-1","BCH-2"}

   J1. J   Job Response Identifier.

   J2. "Status1 A,B"   Status 1A contains the field number, in the format or batch, where an error was found. If the error is not in the format or batch, a "0" is returned.

                                                       Status1 B contains the error number. See Chapter 8, "Diagnostics and Errors."

   J3. "Status2 A,B,C,D,E"   Status2 A contains the packet type, field type, field number, parameter, and error number:

                                            A   The MPCLII packet that has an error - Format (**F**), Batch (**B**), Check Digit (**A**), Graphic (**G**), or Font (**W**).

                                            B   The MPCLII field that has an error. If the packet has no fields, Status2 A is replicated. If the error occurs before the field is identified a question mark is sent. Since the batch data is variable, a D is sent to indicate data.

                                            C   The field number within each packet. The packet header is the first field and each subsequent field is indicated by the field separator.

                                            D   The parameter within the field that has an error. The numbering begins after the field identifier.

                                            E   The error number. See Chapter 8, "Diagnostics and Errors."

   J4. "FMT-1"   Returns the format number.

   J5. "BCH-2"   Returns the batch number.

Example   {J,"2,612",}

2 is the field number where an error was found. 612 is the error number, indicating that data is missing or does not match the format definition for that field.

Note: If more than one error occurred, only the most serious error is acknowledged.

Example   {J,"","F,B,4,6,33","FMT-1","BCH-2"}

Indicates that an error occurred on a bar code (**B**) field within a format (**F**) packet. The bar code field is the fourth (**4**) field in the packet. The error occurred in the sixth (**6**) parameter of the field. Error number **33** means the bar code density is invalid.

To clear an error, press **ENTER**. An ENQ can also clear errors numbered less than 500. Once the error is corrected, a job request can be sent. The printer cannot accept another job request until the error is resolved.

The following syntax is the response for a Job 4 request.

Syntax   {J,printed,total,"FMT-1","BCH-2"}

       Printed   the number of tags or labels *remaining* in the batch to print.

       Total   the total number of tags or labels to be printed in the current batch.

       "FMT-1/BCH-2"   The format or batch number is returned.

Example   {J,8,25,"FMT-3","Bch-2"}

8 out of **25** tags or labels are left to print from format number 3.

Use a Job Request 4 when printing in the on-demand mode with a large number of tags or labels from a single batch. A Job Request 4 may not be accurate if tags or labels are printed in continuous mode, because of the response time involved. A Job Request 4 is not useful in single ticket batches (printing 1 of 1) or multiple single ticket batches.

Note: A batch has to be printing when sending the job request. Do not use this job request on batches with incrementing fields.

Job Status Responses

Status 1 Codes		Status 2 Codes	
No.	**Description**	**No.**	**Description**
0	No error	51	Invalid command
1	Stacker Fault	52	Invalid separator value
2	Supply problem	53	Graphic not found
4	Hot printhead	54	Format for batch not found
5	Printhead open	55	Quantity/Multiples out-of-range
6	Insufficient memory	56	Name descriptor too long
7	Ribbon problem	57	Invalid cut value
8	Field number extends off tag	58	Invalid number-of-parts
9	Field number has a bad font/bar code	59	Invalid orientation value
10	Field number contains invalid data	60	Invalid thickness value
11	Field number has a graphic missing	61	Invalid text field
12	Invalid communication channel	62	Invalid bar code file
13	Invalid file type	63	Data string too long
14	All communication channels are busy	64	Invalid data field
15	Receive overrun error	65	Row greater than stock length
16	Receive parity error	66	Row greater than format length
17	Receive framing error	67	Column greater than printhead width
18	Receive buffer full	68	Column greater than format width
19	Label waiting	69	Invalid label length
21	Bad dots (bar code verifier failure)	70	Invalid label width
23	Low battery	71	Invalid increment/decrement value
24	Memory configuration packet error	72	Identifier out-of-range
Numbers 25 through 50 are not currently in use.		73	No field to create format
		74	Stop location of a line is out of range
		75	Syntax error
		80	Cannot use offline format for online batch
		81	Cannot queue graphic batch
		82	Cannot store online format
		83	Cannot queue online batch
		84	Cannot queue online clear command

Status Polling Considerations for Script Mode

The following table shows the effects of the **I** Packet and the printer's front panel on status polling and immediate commands.

	I Packet Printer Configuration Packet (inside the script or sent online to the printer)	**While Running in Script Mode ONLY (set through printer's front panel)**	
		Disabled	Enabled
Status Polling	**OFF** (status polling not enabled by **I** packet)	**OFF** (status polling not enabled through printer's front panel)	**OFF** (status polling enabled through printer's front panel)
	ON (status polling enabled by **I** packet)	**OFF** (status polling not enabled through printer's front panel)	**ON** (status polling enabled through printer's front panel)
Immediate Commands	**OFF** (immediate commands not enabled by **I** packet)	**OFF** (immediate commands not enabled through printer's front panel)	**OFF** (immediate commands enabled through printer's front panel)
	ON (immediate commands enabled by **I** packet)	**OFF** (immediate commands not enabled through printer's front panel)	**ON** (immediate commands enabled through printer's front panel)

If status polling and immediate commands are NOT enabled through the **I** Packet, enabling status polling and immediate commands at the printer's front panel has NO effect.

DIAGNOSTICS & ERRORS

This chapter explains how to

♦ print diagnostics labels

♦ reset the printer

♦ call Technical Support.

Before calling Service, print a test label. The label contains information to help diagnose mechanical and setup problems.

This chapter also provides explanations of the printer's errors. The errors are classified by type and are listed in order. To load supplies or perform maintenance, refer to the *Operator's Handbook*. See Chapter 7, "Status Polling," for information on requesting printer and job status.

Some errors numbered 400-438 and 500-574 are internal software errors. Errors numbered 703-758 are supply errors. Follow the directions provided with the error description to correct the problem. Errors numbered 900-999 are Hard Printer Failures. If an error cannot be cleared, turn off the printer, wait several seconds and then turn on the printer. Call Technical Support when receiving any error message not listed in this chapter.

To clear an error, correct the problem (load supply, etc.) and press **ENTER** to reprint/continue. If a formatting error occurs, the label prints; but data may be missing. Correct the format or batch and resend them to the printer.

Printing a Test Label

From the Main Menu select **TLABEL**. Select from the test label options:

Printer Information	**MPCL Label**
Contains generic information, including inch counts and printhead resolution.	Contains the printer's MPCL packet configuration.
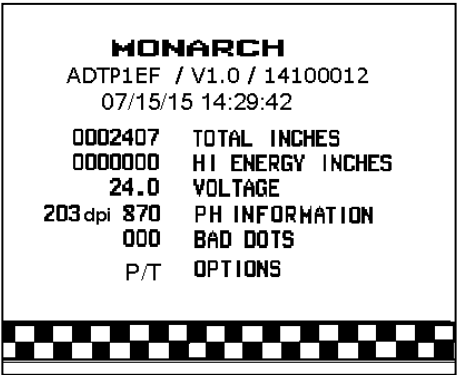

```
MONARCH
ADTP1EF / V1.0 / 14100012
    07/15/15 14:29:42

0002407    TOTAL INCHES
0000000    HI ENERGY INCHES
   24.0    VOLTAGE
203dpi 870 PH INFORMATION
    000    BAD DOTS
    P/T    OPTIONS
```

```
MONARCH
ADTP1EF / V1.0 / 14100012
A,0,0,0,0,0¦
B,1,1,0,0,0,0¦
C,0,0,0,0,0,0¦
D,1,0,2¦
E,{,,,",¦,},,--,--,0d/0a¦
F,3,1,0,0,1¦
G,0,65,65¦
Y,0,0,-15,-15,0,0,100,200,0,0,100,1,-15¦
M,D,R,640¦ M,F,R,1500¦
M,I,R,3300¦
```
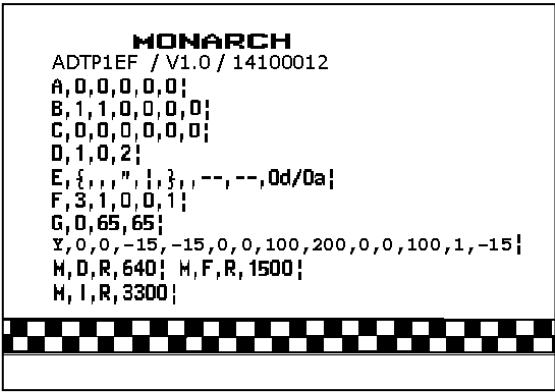
The Printer Info label shows the model number, software version, total number of inches printed, number of inches printed by current printhead, voltage, print contrast, printhead resistance, number of bad dots, and installed options (see the following table). The test label shows an inch count for the high energy ribbon. The one-dot rule line at the bottom of the test label indicates the vertical 0,0 point.

The MPCL label shows the printer's configuration by packet (**A-Y**). See Chapter 2, "Configuring the Printer" for more information. The lines beginning with **M** show the printers memory allocation, which can be changed. The units for supply position, etc. are displayed in dots, even if entered in English or Metric units.

Installed Options	Description
B	Battery
C	Cutter
P	Peel mode
S	Stacker
T	Liner Take-up
V	Bar Code Verifier

Additional labels show wireless network settings and RFID settings if the printer has those capabilities.

Clearing an Error

To clear an error:

♦ Correct the problem (load supply, etc.), then press **ENTER** to reprint/continue.

♦ Press **ESCAPE**. Use the printer's menu to correct the problem (when possible). For example, change the printer's supply type, etc., then press **SELECT** to reprint/continue.

When receiving a message that is not described in this manual or the recommended action does not solve the problem, call Technical Support. Some errors are the result of communication problems. In this case, verify network settings or reset the printer and reboot the computer. If any online configuration packets changed, resend the format packet to the printer, so the configuration changes take effect.

If the PC and Printer Are Not Communicating

If the PC is not communicating with the printer, follow these steps:

♦ Check any messages that occur at the printer and at the computer. See the following error message listing in this chapter for more information.

♦ Use the correct printer cable.

♦ Make sure the cable is plugged into the correct port on the computer.

♦ Compare the printer's communications settings (especially flow control) with the settings on the PC. They must match. Print a test label to identify the printer's communication settings.

♦ Make sure the printer is online (ready to receive data).

If all of the above are correct, reset the printer. Try the function again. Call Technical Support if the printer does not establish communications.

Resetting the Printer

Sometimes, the printer receives mixed signals and loses its ability to communicate. If this happens, reset the printer and attempt communication again. To reset the printer, turn off the printer, wait 15 seconds, and turn it back on.

Whenever the printer is turned off, all the information set through the online configuration packets (A-Y) is saved. See the sections in Chapter 2, "Configuring the Printer," for more information about each packet.

Calling Technical Support

Technical support representatives are available Monday through Friday during regular business hours. Follow these steps before calling:

1. Make sure the PC and printer are properly connected.

2. Record any error messages that occurred.

3. Recreate the problem, if possible.

4. Check the communication port settings and change if necessary.

5. List any recent changes to the system. Record what you were doing when the problem occurred.

6. Reset the printer, see "Resetting the Printer."

7. Reboot the computer. Refer to the computer documentation for specific instructions.

8. Print a test label, see "Printing a Test Label" for more information.

Have the following information ready before calling: computer brand name and model, printer model, other peripheral devices on your system, support agreement, contract number, or invoice information, customer number, and printer serial number.

Data Errors

Call Technical Support if any error message is received that is not listed below.

Errors 001 to 499 are data errors. These errors are returned in response to a J,3 request. See Chapter 7, "Status Polling," for more information. A data error indicates that incorrect data was received from the host, causing the printer to ignore the entire print job. After checking the packet and correcting the problem, transmit the print job again.

The following is a list of data errors. These errors occur because data in the format, batch, check digit, font, or graphic packet is invalid.

Format Errors

Error Code	Description
001	Packet ID number must be 1 to 999.
002	Name must be 1 to 8 characters (excluding non-printable control characters) inside quotes. The range is 1 to 256 characters for the XML-enabled printer.
003	Action must be A (add) or C (clear).
004	Print length is invalid. See "Defining the Format Header" in Chapter 3 for valid lengths.
005	Print width is invalid. See "Defining the Format Header" in Chapter 3 for valid widths.
006	Storage device must be F (Flash), R (volatile RAM), or T (temporary for graphics).
007	Unit of measure must be E (English), M (Metric), or G (Dots). See "Defining the Format Header" in Chapter 3 for more information.
010	Field ID number is outside the range 0 to 999.
011	Field length exceeds 2710.
012	Row field position is greater than the maximum stock dimension. See "Defining Text Fields" in Chapter 3 for valid row lengths.

Error Code	Description
013	Column field position is greater than the maximum stock dimension. See "Defining Text Fields" in Chapter 3 for valid column widths.
014	Font style must be 1, 2, 3, 4, 5, 6, 10, 11, 15, 16, 17, 18, 50, 56; or a valid downloaded font selector. See "Defining Text Fields" in Chapter 3 or Appendix B, "Fonts," for more information.
015	Character rotation must be 0 (0 degree), 1 (90 degree), 2 (180 degree), or 3 (270 degree). See "Defining Text Fields" in Chapter 3 for more information.
016	Field rotation must be 0 (0 degree), 1 (90 degree), 2 (180 degree), or 3 (270 degree). See "Defining Text Fields" in Chapter 3 for more information.
017	Field restriction must be V (variable) or F (fixed).
018	Code page/symbol set selection defined in the field must be 0 (Internal), 1 (ANSI), 100 (Macintosh), 101 (Wingdings), 102 (Unicode), 103 (BIG5 for Unicode), 104 (GB2312 for Unicode), 105 (SJIS for Unicode), 106 (GB2312), 107 (BIG5), 108 (KSC5601), 110 (Unicode UTF-8), 437 (DOS Page 437), 850 (DOS Page 850), 852 (Latin 2), 855 (Russian), 857 (Turkish), 860 (Portuguese), 1250 (Latin 2), 1251 (Cyrillic), 1252 (Latin 1), 1253 (Greek), 1254 (Turkish), 1255 (Hebrew), 1256 (Arabic), 1257 (Baltic), or 1258 (Vietnamese). See "Defining Text Fields" in Chapter 3 for more information.
020	Vertical magnification must be 1 to 7 or 4 to 255 (point size) for the scalable or downloaded TrueType fonts.
021	Horizontal magnification must be 1 to 7 or 4 to 255 (point size) for the scalable or downloaded TrueType fonts.
022	Color must be A, B, D, E, F, N, O, R, S, T, or W. See "Defining Text Fields" in Chapter 3 for more information.
023	Intercharacter gap must be 0 to 99 dots.
024	Field justification must be B (balanced), C (centered), E (end), L (left), or R (right). See "Defining Text Fields" in Chapter 3 for more information.
025	String length is outside the range 0 to 2710.
030	Bar code height must be at least 19 (English), 48 (Metric), 38 (203 dpi Dots), 57 (300 dpi Dots), or is not within the supply dimensions.
031	Human readable option must be 0 default 1 no CD or NS 5 NS at bottom, no CD 6 CD at bottom, no NS 7 CD and NS at bottom 8 no text
032	Bar code type is invalid. See "Defining Bar Code Fields" in Chapter 3 for valid options.
033	Bar code density is invalid. See "Defining Bar Code Fields" in Chapter 3 for the bar code density chart.
034	GS1 DataBar type is invalid. See "Defining Bar Code Fields" in Chapter 3 for more information.
035	GS1 DataBar separator height is invalid. See "Defining Bar Code Fields" in Chapter 3 for more information.
036	GS1 DataBar segment width is invalid. See "Defining Bar Code Fields" in Chapter 3 for more information.

Error Code	Description
040	Line thickness must be 0 to 99 dots.
041	Line direction must be 0, 90, 180, or 270.
042	End row is invalid. Line segment or box end row is defined outside of printable area.
043	End column is invalid. Line segment or box end column is defined outside of printable area.
044	Dot pattern for line or box must be "".
045	Line length is defined beyond the maximum length. See "Defining Line Fields" in Chapter 3 for valid lengths.
046	Line type must be S (segment) or V (vector).
051	Imaging mode in the graphic header must be 0.
052	Data type in the RFID Data Field must be 0.

Batch Errors

Error Code	Description
101	The format referenced by batch is not in memory.
102	Print quantity is outside the range 0 to 32000.
104	Batch mode must be N (new) or U (update).
105	Batch separator in a batch control field must be 0 (Off), 1 (On), or 2 (Double-length).
106	Print multiple is outside the range 1 to 999.
107	Cut multiple is outside the range 0 to 999. Only valid for printers with a cutter.
108	Multiple part supply is outside the range 1 to 5.
109	Cut type is outside the range 0 to 5.
110	Image rotation must be 0 (no rotation) or 1 (180° rotation) in the batch control field.

Option Errors

Error Code	Description
200	Option number must be 1 - 7, 20, 21, 30, 31, 42, 50, 51, 52, 53, 60, 61 or 62.
201	Copy length is outside the range 0 to 2710.
202	Copy start position must be 1 to 2710.
203	Destination start position must be 1 to 2710.
204	Source field must be 0 to 999.
205	Copy type must be 1 (copy after rules) or 2 (copy before rules).
206	Increment/Decrement selection must be I (increment) or D (decrement).
207	Incrementing start position must be 0 to 2710.
208	Incrementing end position must be 0 to 2710.
209	The incrementing amount must be 0 to 999.
210	Security value for a PDF417 bar code must be 0 to 8. Correct the value and resend the format to the printer.

Error Code	Description
211	Narrow element value is less than 1 or greater than 99. Correct the value and resend the format to the printer.
212	Wide element value is less than 1 or greater than 99. Correct the value and resend the format to the printer.
213	Dimension must be 1 to 30 for a column or 3 to 90 for a row on a PDF417 bar code.
214	Truncation code must be S (standard) or T (truncated bar code).
215	Aspect code must be C (columns) or R (rows).
216	Option definition must be S (set) or T (template).
217	Input device must be D (Default), H (Host), K (Keyboard), N (None), or S (Scanner).
218	Pad direction must be L (from left) or R (from right).
219	Pad character is outside the range 0 to 255.
220	Check digit selection must be G to generate check digit.
221	Primary or secondary price format is outside the range 1 to 16.
222	Data type restriction is outside the range of 1 to 6.
223	Option is not valid for the field. An example is applying Option 6 (**R,6,H,T**|) to a non-RFID field.
224	Bar code inter character gap must be 0 to 99 dots. (This is also known as the additional character gap when using Option 50, Defining Bar Code Densities.)
226	Rule (option) Record Line xx. The upload device must be H (host) for Option 6.
231	Error control level must be 0, 1-99, 101-104, 201-232, or 300 for Option 53, Optional Settings for Aztec bar code.
232	Invalid ECI data flag for Option 53, Optional Settings for Aztec bar code.
233	Invalid menu symbol flag for Option 53, Optional Settings for Aztec bar code.
234	Invalid structured append information for Option 53, Optional Settings for Aztec bar code.
235	Invalid string to append for Option 53, Optional Settings for Aztec bar code.

Online Configuration Errors

Error Code	Description
251	Power up mode must be 0 (online) or 1 (offline).
252	Language selection must be 0 (English), 1 (Français), 2 (Deutsch), 3 (Español-ES), 4 (日本語), 5 (Português), 6 (Italiano), 7 (Svenska), 8 (Español-MX), 9 (Dansk), 10 (Nederlands), 13 (Polski), 14 (Türkçe), 15 (简体中文), or 16 (Français-CA).
253	Batch separator code in a system setup packet must be 0 (off), 1 (on) or 2 (double).
254	Slash zero selection must be 0 (standard zero) or 1 (slashed zero).
255	Supply type must be 0 (black mark), 1 (die cut), 2 (continuous), or 3 (aperture)
256	Ribbon selection must be 0 (direct), 1 (transfer), or 2 (high energy).
257	Feed mode must be 0 (continuous) or 1 (on-demand).
258	Supply position must be -300 to 300 dots.
259	Contrast adjustment must be -699 to 699 dots.

Error Code	Description
260	Print adjustment (position) must be -450 to 450 dots.
261	Margin adjustment (position) must be -99 to 99 dots.
262	Speed adjustment must be 0 (default), 25 (2.5 ips), 40 (4.0 ips), 60 (6.0 ips), 80 (8.0 ips), 100 (10.0 ips), or 120 (12.0 ips).
263	Primary monetary symbol must be 0 (None), 1 (Dollar), 2 (Pound), 3 (Yen), 4 (Deutsche Mark), 5 (Franc), 6 (Peseta), 7 (Lira), 8 (Krona), 9 (Markka), 10 (Schilling), 11 (Rupee), 12 (Ruble), 13 (Won), 14 (Baht), 15 (Yuan), or 16 (Euro).
264	Secondary symbol selection must be 0 (none) or 1 (print secondary sign).
265	Monetary decimal places must be 0 to 3.
266	Character string length in the control characters packet must be 5 (MPCL control characters) or 7 (ENQ/IMD command character).
267	Baud rate selection must be 0 (1200), 1 (2400), 2 (4800), 3 (9600), 4 (19200), 5 (38400), 6 (57600), or 7 (115200). Resend the communication settings packet.
268	Word length selection must be 0 (7 bits), or 1 (8 bits). Resend the communication settings packet.
269	Stop bits selection must be 0 (1 bit), or 1 (2 bits). Resend the communication settings packet.
270	Parity selection must be 0 (none), 1 (odd), or 2 (even). Resend the communication settings packet.
271	Flow control selection must be 0 (none), 1 (DTR), 2 (CTS/RTS), or 3 (XON/XOFF). Resend the communication settings packet.
272	Symbol set (code page) in the System Setup packet must be 0 (Internal), 1 (ANSI), 2 (DOS 437), 3 (DOS 850), 4 (1250- Latin 2), 5 (1251- Cyrillic), 6 (1252- Latin 1), 7 (1253- Greek), 8 (1254- Turkish), 9 (1255- Hebrew), 10 (1256- Arabic), 11 (1257- Baltic), 12 (1258- Vietnamese), 13 (852- Latin 2), 14 (855- Russian), 15 (857- IBM Turkish), 16 (860- DOS Portuguese), 17 (Wingdings), 18 (Macintosh), 19 (Unicode), 20 (BIG5), 21 (GB), 22 (SJIS to SJIS), 23 (GB to GB), 24 (BIG5 to BIG5), 25 (KSC to KSC) or 26 (UTF-8).
273	Cut adjustment must be -300 to 300 dots.
282	RS232 Trailer string is too long. Use a maximum of 3 characters.
283	ENQ Trailer string is too long. Use a maximum of 3 characters.
284	The buffer type must be D (Downloadable Fonts), F (Format), or I (Image).
285	The storage device type in the memory configuration packet must be F (flash) or R (volatile RAM).
286	The buffer size is invalid.
287	The printhead width must be 0.
288	The battery voltage must be 0 (15-volt battery) or 1 (12-volt battery).
289	The printer address specified in the communications settings packet must use exactly six characters.
290	Action must be 0 (disable), 1 (enable), or 2 (extended) for the backfeed control packet or the printer is active.
291	Dispense position must be 50 to 200 dots or the printer is active.
292	Backfeed distance must be 10 to 200 dots or the printer is active. The backfeed distance can not be greater than the dispense position.

Error Code	Description
293	The skip index selection defined in the field must be 0 (disabled) or 1 (enabled). Correct the packet and resend to the printer. **Do not use skip index with RFID supplies**. Unexpected results may occur.
294	The bar code verifier mode must be 0 (disable) or 1 (enable).
295	The bar code verifier's cable detect mode must be 0 (no detect), 1 (I/O cable detect), or 2 (I/O and data cable detect).
298	Read power must be -15 to 23. See "Defining the RFID Setup Packet" for more information.
299	Write power must be -15 to 23. See "Defining the RFID Setup Packet" for more information.
301	Image rotation must be 0 (no rotation) or 1 (180° rotation).
302	The hours in the time/date packet must be 0 to 23.
303	The minutes in the time/date packet must be 0 to 59.
304	The day in the time/date packet must be 1 to 31.
305	The month in the time/date packet must be 1 to 12.
306	The year in the time/date packet must be 0 to 99.

Check Digit Errors

Error Code	Description
310	Check digit scheme number must be 1 to 10.
311	Modulus must be 2 to11.
312	Multi Tag Check must be 0 (Check the first tag) or 1 (Check each tag). See "Defining the RFID Setup Packet" for more information.
313	Inlay Position is invalid. Use **0**.
314	Check digit algorithm must be D (sum of digits) or P (sum of products).

Graphic Errors

Error Code	Description
325	Duplicating direction must be 0 (insert after) or 1 (insert before) in duplicate fields for graphics.
327	Amount of row adjustment must be 0 to 999 dots in duplicate fields for graphics.
328	Duplicate count must be 0 to 999.
340	Bitmap line encoding must be H (hex) or R (run length).
350	Font selector must be 1 to 9999.
351	Font data length must be 68 to 16384.
352	Insufficient font memory is available for the downloaded font.
360	Prompt name is invalid.
362	Prompt number is invalid.
363	Prompt string is invalid.

Error Code	Description
370	RFID encode position must be between 0 and 2000. See "Defining the RFID Setup Packet" for more information.
371	RFID stop to encode mode must be 0 (disabled) or 1 (enabled). See "Defining the RFID Setup Packet" for more information.
372	RFID singulate mode must be 0 (disabled) or 1 (enabled). See "Defining the RFID Setup Packet" for more information.
373	RFID start encode position must be between 10 and 2000. See "Defining the RFID Setup Packet" for more information.
374	RFID encode zone must be between 50 and 2000. See "Defining the RFID Setup Packet" for more information.
375	RFID tag saver mode must be 0 (disabled) or 1 (enabled). See "Defining the RFID Setup Packet" for more information.
376	RFID tag verify must be 0 (none), 1 (internal), 2 (external), or 3 (both). See "Defining the RFID Setup Packet" for more information.
377	RFID verify position be between 10 and 2000. See "Defining the RFID Setup Packet" for more information.
378	RFID TID position must be between 0 to 2000. See "Defining the RFID Setup Packet" for more information.
379	Verify power must be -15 to 23. See "Defining the RFID Setup Packet" for more information.
380	Job request is outside the range 0 to 4.
400	The character immediately following { is invalid.
401	Internal software failure. Call Technical Support.
402	Field separator is not in the expected location.
403	Field separator was not found.
404	The number (greater than five digits/732,000), string (greater than 2710), or tilde sequence (greater than 255) currently being processed is out of range.
405	Too many fields exist in the format. 1000 fields is the maximum amount per format. Lines, boxes, and constant text fields count as fields.
406	Parser error. Format or graphic is already in use. Make sure the packet is not missing **|** or **{}** characters.
407	Parser timed out. Turn off the printer and resend the packets. If that does not solve the problem, call Technical Support.
408	No Data. Resend packet to printer.

Communication Errors

Error Code	Description
409	The printer memory is full. Delete unnecessary formats or graphics from memory. Use run length encoding to map large graphics. To use flash memory, format it first. See "Formatting Flash Memory" in Chapter 2 for more information.
410	Parity on the printer does not match the parity on the host. Check the parity setting under SETUP options.
411	Framing error. The printer cannot communicate with the host. Make sure the host is turned on, communication cables are connected correctly, port settings are correct, and communications are active. Check the baud rate, word length, and stop bits to make sure they match those at the host. Re-transmit the data.
412	There is a problem with flow control between the printer and the host. Make sure the printer and the host flow control settings match (both are DTR or both are XON/XOFF). If the error persists, call Technical Support.
413	Online receive buffer is full. Check the printer's XON/XOFF or DTR SETUP values to be sure there is not a flow control problem.
414	The internal keyboard buffer is full or replace the keypad. Call Technical Support.
415	The defined buffer size exceeds the total available memory.
416	Flash/PC board error. Call Technical Support.
417	Flash memory is full. Clear flash. Resend the format, graphic, and check digit packets. If the error reappears, call Technical Support.
420	Internal software error relating to uploading.
421	Internal software error relating to location name.
422	Internal software error relating to duplicate name.
423	Internal software error.
424	Item in use.
425	Item already checked out.
426	Item not checked out.
427	Format name is invalid. Valid name is 1 to 8 characters inside quotes or "" for a printer-assigned name. Press **ESCAPE** and try to continue. If the error reappears, call Technical Support.
428	Batch name is invalid or graphic not found. Press **ESCAPE** and try to continue. If the error reappears, call Technical Support.
429	A field number appears more than once in a format.
430	The format uses a graphic packet that cannot be found.
431	Internal software failure. Format name in task does not match the task name added.
432	Internal software failure. Graphic name in task does not match the task name added.
433	The batch references a field number that does not exist in the format.
434	Internal software error caused by a downloaded task that is not executable. Turn off the printer, wait ten seconds and turn it back on.
435	Internal software error caused by a downloaded task that already exists. Turn off the printer, wait ten seconds and turn it back on.
497	An error occurred during the loop back test on the parallel port. Call Service.
499	An error occurred during the loop back test on the serial port. Call Service.

Data Formatting Failures

Formatting errors indicate that a field prints incorrectly. After checking the data stream and correcting the data, retransmit the format and batch.

Note: For errors 571-622, the batch still prints, but the field, font, bar code, or density may be incomplete, missing or contain incorrect data.

Error Code **Description**

571   UPC or EAN bar code data length is invalid. The bar code data length in the batch does not fit the format.

572   A copy field, padded field, or incrementing field length is invalid. The field length in the batch does not fit the format, the field contains blanks, the destination field does not have enough characters to hold the copied data, the copy count exceeds the maximum listed value, or the fixed length field does not contain the specified number of characters.

573   Price field length is invalid. The price field length in the batch does not fit the format or the field contains blanks.

574   No CD scheme or room for CD. The CD scheme in the batch does not fit the format or the field contains blanks.

575   The graphic included in the format could not be found. Resend the graphic packet. If the error persists, call Technical Support.

600   The printer could not image the batch, because the batch was refused. Call Technical Support.

601   An error occurred while the batch was imaging. Resend the format, batch, font, and/or any graphic packets. If the error persists, call Technical Support.

603   The batch was not found while imaging. Resend the batch packet. If the error persists, call Technical Support.

611   Font, bar code or density is invalid. The font, bar code or density in the batch does not fit the format.

612   The data in this line of the batch is either missing or does not match the format. This error also occurs if the EPC Gen2 RFID data length is invalid. See "Using Gen2 RFID Data" for more information.

613   Reference point off tag.

614   Portion of field off tag. Position the entire field within the printable area. There may be an invalid character in the packet. Check for O instead of a 0 (zero). Resend the format.

615   Bar code width is greater than 16 inches or the number of keywords for the PDF417 bar code exceeds 928. Decrease the density or shorten the amount of data to print the bar code.

616   A bad dot falls on a bar code and the dot cannot be shifted. A gray box is printed instead of the bar code. Check or replace the printhead.

618   Magnification must be 1 to 7 or 4 to 255 (point size) for the scalable font.

619   The JIS (16-bit) code for Kanji is invalid.

620   Font and printhead dot density mismatch. A 300 dpi font is specified in the format, but a 203 dpi printhead is used in the printer or vice-versa. Check the font or change the printhead to correct the error.

621   An error occurred opening the TrueType font file. Select a different font to download. If the error message persists, call Technical Support.

Error Code	Description
622	Not enough memory to create the downloaded TrueType characters in the scalable fonts buffer. Save the font into the printer's flash memory instead. If the error message persists, call Technical Support.
623	There are more than 10 bar codes on one label for the bar code verifier to scan. Redesign the format or disable the bar code verifier.
700	Error pending and the printer cannot continue with batch.
701	Printer received a command that it cannot execute while it is running.
702	SETUP setting is invalid.

Machine Faults

These errors occur when there is a problem with the printer.

Error Code	Description
703	The printer sensed a calibration of different-sized black marks. Make sure the correct supply is loaded.
704	Printer has not sensed a supply mark within the specified number of inches or out of supplies. Check the ♦ supply tracking ♦ supply marks ♦ black mark sensor position ♦ supply roll for binding. Press **ENTER**. If the error continues to appear, change the supply and/or clean the supply sensor.
706	The motor could not get up to speed within the allotted time.
707	The motor reached desired speed but could not maintain it.
708	The motor has stalled or is running too slow to measure the speed.
710	Band out of data.
711	UART receiving communication error (framing, parity, etc.).
712	Receiving failure or command overrun error.
713	Receiving failure or illegal command.
714	Receiving failure or undefined command.
730	A memory allocation error occurred during initialization.

RFID Errors

These errors occur when there is an RFID problem. If using Easy Setup, contact Technical Support for assistance with these errors.

Note: To correct an RFID error, turn off the printer to clear the error. Turn on the printer, then adjust the RFID settings.

Error Code	Description
715	Invalid data length/data mismatch. The data in the RFID Data Field has an incorrect data length or there is a data type mismatch between the selected data type and the actual data entered. See "Defining the RFID Data Field" in Chapter 3 for more information about the data length. This error also occurs if any of the Gen2 fields is invalid.
727	Too many inlays within the programmable range. Make sure the read power is correct for the selected inlay. If the read power is correct and this error occurs, lower the read power by one and try again.
729	Invalid inlay. The chip (inlay) embedded in the RFID supply does not support serialized TID. Make sure the RFID supply supports serialized TID. This may also be a bad RFID inlay.
732	No RFID optional hardware installed. A non-RFID printer received a format containing an RFID Data Field. Send the format to an RFID printer with RFID supplies loaded.
735	Error in the RFID module – hardware, software, or firmware. The RFID module may need to be replaced or has become disconnected. If this error continues, call Technical Support. Message will say "reboot", press button; printer reboots.
736	RFID inlay encoding failed. The RFID inlay was found in the RF Field, but could not be read or programmed. The RFID module has an error. This may also be a bad RFID inlay. If this error continues, call Technical Support.

737 RFID Unsupported. The printer detected an unsupported or illegal RFID setting:

♦ Make sure the start encode position is before or "leading" the first TID position.

♦ Reduce the tag verify position so the end of the label being verified is not within 0.2" from the outside edge of the external RFID verifier.

♦ When using RFID verifier overstrike, the printer detected more than one active label after the errored label.

♦ Singulate mode (defined in the RFID Setup Packet) must be enabled to upload TID data using Option 6.

♦ Option 65 (Serialized EPC Data) is only supported in stop-to-encode mode.
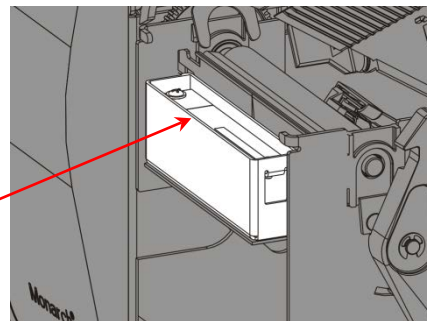
If this error continues, call Technical Support.

738 Inlay communication failure. This error can occur when

♦ The printer cannot read the TID field.

♦ The inlay cannot be found within the encode zone.

♦ Invalid read power.

Call technical support. The start encode position may need to be decreased or increased. The read power level may also need to be increased by 1.

Error Code	Description
740	RFID Tag Failure. The RFID module detected a problem during encoding. This error also occurs when TID singulation fails. Call Technical Support.
741	RFID inlay missing. Inlay not found in RF Field (area inside the printer where RFID inlay is programmed.) This may be a bad RFID inlay. ♦ For continuous 741 messages: ♦ The inlay may be moving too fast to encode. Decrease the printer's speed and try again. ♦ Make sure the correct values are set for the selected inlay and try again. When using non-stop encode, adjust the start encode position by 10 and try again. If using stop-to-encode, adjust the encode position by 10 and try again. ♦ Call Technical Support.
744	Inlay locked fail. The RFID inlay is unable to be programmed, because it is already locked. This is considered a bad RFID inlay.
748	Verify mismatch. The external RFID verifier read the EPC data and it does not match the programmed data so it failed RFID verification. If this message appears frequently, adjust the tag verify position and try again. Make sure the printer has the external verify option installed. This is a bad RFID inlay. The operator must not use this inlay. If tag saver is set to yes, this inlay may have an overstrike pattern. If tag saver is set to no, this inlay will not have an overstrike pattern
749	RFID verify fail. The RFID verifier could not read the inlay. Adjust the tag verify position and try again. Make sure verification is enabled and the printer has the external verify option installed. This may also be a bad RFID inlay.

Machine Faults (Continued)

Error Code	Description
750	Printhead is overheated. Turn off the printer to let the printhead cool. If the error persists, call Technical Support.
751	Printer did not sense a black mark when expected. The supply may be jammed. Check the supply tracking, supply marks, black mark sensor position, and supply roll for binding. Press **ENTER** to continue printing. Clean the sensor. If the error continues to appear, change the supply.
752	Printer sensed a mark in the wrong place. Check the supply tracking, supply marks, black mark sensor position, and supply roll for binding. Press **ENTER** to continue printing. Clean the sensor. If the error continues to appear, change the supply.
753	Printer sensed a mark that is too long. Check the supply tracking, supply marks, black mark sensor position, and supply roll for binding. Press **ENTER** to continue printing. Clean the sensor. If the error continues to appear, change the supply.
754	Printer has a ribbon problem. Check the ribbon for a jam or remove any slack by turning the take-up reel clockwise. Load a new ribbon. Press **ENTER** to continue printing.
755	Printhead is open. Close the printhead before continuing. If the error persists, call Technical Support.
756	The printer is out of supplies. Load supplies and/or clean the supply sensor. Press **ENTER** to continue printing.

Error Code	Description
758	Check supply. Either the supply is not seen, or the on-demand sensor is broken (purchase optional). Check for a label jam. Clear the supply path or reload supplies. This error may occur when removing a label too quickly in on-demand mode. The printer does not recalibrate after this error.
759	Cutter is not moving. Close the cutter's door.
760	Check cutter. Close the cutter's door. Call Technical Support.
761	The stacker is full or jammed. Empty the stacker or clear the jam before continuing. The printer does not recalibrate after this error.
762	Low Voltage. For DC-powered printers, recharge the external battery. For AC-powered printers, check the power inlet where the printer is connected. The acceptable voltage range is between 90-132 V_{RMS} and 180-264 V_{RMS}. If more or less power is received, unpredictable results may occur.
763	Waiting to dispense label. Press **FEED**.
764	Bar code verifier scan error (bad scan) on label when using normal (no overstrike) mode. Press **ENTER** to clear the error and continue printing.
765	The printhead has less than 8 bad dots. The printer can shift some bar code fields to avoid bad dots on the printhead. See "Defining Bar Code Fields" for more information. Press **ENTER** to continue printing. Print a test label to confirm the number of bad dots.
766	Backfeed/overfeed error. There is a problem with the backfeed or overfeed distance.
767	Incorrect AD value. Call Technical Support.
768	Printhead has more than 8 bad dots within the format area or is not connected. Make sure the printhead is connected. Replace the printhead.
769	The printhead is not receiving enough volts. Call Technical Support.
770	The print motor is not ready. Call Technical Support.
771	The format specified by the application was not found. Reload the application and format, and try again. If the problem continues, call Technical Support.
772	Bar code verifier scan error (bad scan) on label or multiple labels in a row when using overstrike and continue mode. Press **ENTER** to clear the error and continue printing.
773	Bar code verifier Failure – the bar code verifier is enabled, but not connected. The printer can detect if the I/O and/or serial cables are connected to the bar code verifier. Refer to the bar code verifier's *Operating Instructions* for more information.
781	The voltage is too low. Call Technical Support.
782	The voltage is too high. Call Technical Support.
783	The printer is printing too fast for the bar code verifier to accurately scan the bar codes. Increase the bar code height in the format or slow the print speed.
784	No power to stacker. Check connections between the stacker and the printer. Make sure the printer is connected to an AC outlet.
785	The internal liner take-up reel is full. Remove the liner take-up reel and dispose of the liner. Press **ENTER** to continue printing.
786	Internal liner take-up error. The liner has separated from the core or is not properly winding on the core. Remove the liner from the take-up core and reload the liner into the take-up core.
790	Wait until the printer is idle (no batch waiting to print or not receiving data) before sending any packets. This error may occur when printing a test label if the printer is busy.

Error Code	Description
791	The printer has an error pending. Turn off the printer. Wait 15 seconds and turn it back on. Resend the packets. If the problem continues, call Technical Support.
792	The printer is not initialized. Call Technical Support.
793	The printer job queue is full. Turn off the printer. Wait 15 seconds and turn it back on. Resend the packets. If the problem continues, call Technical Support.
794	The backfeed mode is not set.
795	Print engine lock-up. Call Technical Support.
796	No job quantity for the job setup. If this error message appears, call Technical Support.
797	No sub-job quantity for the job setup. If this error message appears, call Technical Support.
799	Job quantity mismatch. If this error message appears, call Technical Support.

Flash Memory Errors

Error Code	Description
800	A directory in flash memory is full.
801	Flash memory is full.
802	A directory in flash memory can't be found.
803	There is no directory in flash memory.
810	A file in flash memory is not open.
811	A file in flash memory is already open.
812	The file in flash memory is full.
813	Flash memory is not accessible.
820	An error occurred in the flash program.
821	An error occurred while erasing flash memory.
822	Flash ID error.

Script Errors

These errors occur when there is a script (application) problem.

Error Code	Description
850	Invalid script packet or not enough flash memory for the script. The script download failed. Format flash memory and download the script using the Command prompt.
851	Invalid lookup table or not enough flash memory for the script. The lookup table download failed. Check the data in the lookup table. Format flash memory and download the script using the Command prompt.
852	Invalid buffer definition packet or not enough flash memory for the script. Invalid input buffer, temporary buffer, lookup buffer, printer, or array buffer. Check the buffer definitions. Format flash memory and download the script using the Command prompt.
853	Invalid version string or not enough flash memory for the script. The version information in the APPVERSION command is missing or invalid. Make sure the information is enclosed in double quotation marks. Format flash memory and download the script using the Command prompt.

Error Code	Description
854	Invalid table (string and numeric) data or not enough flash memory for the script. Check the data. Format flash memory and download the script using the Command prompt.
855	Invalid start/clear command. The script is missing the start or clear command {S space}.
856	Invalid quit command. The script is missing the quit command {Q space}.
857	Illegal command in the script. The script may include commands that have been removed from the ADK2 scripting language.
858	Check sum error. Each packet has a four-digit checksum at the beginning of the packet. Recompile the script.
859	Script download buffer overflow. The image buffer is used as the SCRATCH buffer. Increase the size of the image buffer.
860	Error opening a font file on the print engine memory expansion option.
861	Error writing to a file on the print engine memory expansion option.
862	Invalid condition for the print engine memory expansion option.

Hard Printer Failure Errors

Errors numbered 930-940 are errors that may occur during the loading of flash memory. Errors numbered 950-999 may indicate a circuit board failure. These errors are hard printer failures. Call Technical Support if errors numbered 900- 938 are received.

Error Code	Description
900	RAM test failure.
901	ROM/EPROM checksum failure.
902	Software timer failure.
903	Software interrupt failure.
904	No memory for native layer.
905	Illegal interrupt.
906	Non-maskable interrupt. Check format packet for font clear or format clear packets.
907	Low RAM error.
908	Non Volatile RAM checksum failure.
909	NVRAM reset.
910	Warm restart.
911	Version string mismatch.
930	Flash erase error.
931	Flash write error.
932	RAM write error.
933	Parity error.
934	Framing error.
935	Buffer overrun.

Error Code	Description
936	Invalid record type.
937	Invalid hex data.
938	Invalid checksum.
939	Invalid record count.
940	Invalid flash address.

"Waiting for Host" Message

While the printer logs data, if five consecutive messages have been kept in RAM without receiving an ACK response, the printer stops with the error message "**Waiting for Host**". The operator cannot clear this error and no printer buttons are active.

Depending on the host's condition, the host must send either an ACK or a NAK (hex 15) to clear the printer's "**Waiting for Host**" message and restart the ACK/NAK process. Once the printer receives an ACK or a NAK, it automatically processes any pending print jobs. If no unacknowledged log messages are saved, the printer responds with a NAK to any host ACK or NAK message.

PRINTER OPTIMIZATION

This chapter provides information on how to improve the printer's performance by

♦ adjusting the print quality

♦ reducing the imaging time for printing

♦ providing general tips and hints for designing formats.

This printer uses "smart imaging" to image and print fields on supplies. Smart imaging remembers the exact boundaries and locations of each field and places a boundary box (white space) around each field. When a field changes that particular boundary box is cleared and the new field data is imaged. However, the new field data may require a larger boundary box than the previous field did. In some cases, neighboring fields that do not change may be covered with white space from the changing field's boundary box. To prevent existing fields from being covered by a changing field, see "Using Option 61 (Reimage Field)" in Chapter 4.

Note: Smart imaging is automatically disabled on formats with a Data Matrix bar code.

Adjusting the Print Quality

Many factors affect print quality: type of supplies, print speed, print contrast, and the type of printer's application. This printer supports both thermal transfer and thermal direct supplies. The type of supply should match the printer's application.

♦ To print at high speeds, use premium supplies. Using premium supplies reduces smudged images, hard to read labels, and faded print. Supply type, print speed, and print contrast work together to improve the print quality of labels. Contact your Sales Representative for more information.

♦ Select the print speed based on desired throughput and print quality. If print quality is more important, reduce the print speed, because a lower print speed increases the print quality of labels. If throughput is more important, increase the print speed. See "Increasing Throughput" for more information.

♦ If the print quality is too light or too dark, adjust the print contrast. The correct contrast setting is important because it effects how well the bar codes scan and how long the printhead lasts. Solid black print cannot exceed 30% of any given square inch of the supply.

Check the print quality of bar codes with a bar code verifier or scanner. If you do not have a bar code verifier or scanner, check the bar code visually. A bar code that is **IN SPEC** will have complete bars and clear spaces. Small alphanumeric characters will look complete. A bar code that is **IN SPEC** may not look as good as one that is too dark, but it will have the highest scan rate.

Dark	IN SPEC	Light

Note: For highest scan rates, make sure there is adequate white space before and after the bar code. Also, a darker bar code does not mean it will scan better.

The following non-print zones are recommended:

Left and Right   0.025 inches (0.6mm or 5 dots) on either edge of the label.

Top and Bottom   0.025 inches (0.6mm or 5 dots) on the leading and trailing edge of the label.

The leading edge is the edge of the label that exits the printer first; the trailing edge is the edge of the label that exits the printer last; regardless of how the format is designed on the label. If using the optional bar code verifier, allow a minimum no-scan zone on the trailing edge of the label of 0.5 inch (13 mm).

Reducing Imaging Time

Imaging time is the time it takes the printer to image the data for the first label after the printer receives the format and batch packet. There are several ways to reduce the imaging time: send formats and configurations once, use a batch quantity of zero, or update batch fields.

♦ If the formats use the same check digit scheme, only send the check digit scheme once.

♦ Send formats once and use the batch update field to change information on the label. Using a batch update field reduces the imaging time, because only the fields that change are imaged. All other fields remain the same as the last queued batch.

♦ Use the batch quantity zero method when the application requires operator intervention to enter data. While the operator is entering data, the previous field is sent with a batch quantity of zero. The printer images the field, but does not print it. After the operator enters the data for the last field, the batch quantity can be specified. The last remaining field is imaged, and the label prints almost immediately.

To pre-image a label:

1. Send the format and a batch header in one file. The first time the batch header is sent, use the parameter N (new batch), and the parameter 0 for (zero quantity).

Example   **{B,1,N,0 | }**

The printer images constant text, line, box, and graphic fields, but does not print them.

2. Input data for each field, and send it with a batch header using the parameter U (update) and a quantity of zero. When the printer receives the data, it images the field, but does not print it.

Example   **{B,1,U,0 |**
1,"RODGER DIST CTR" | }
{B,1,U,0 |
2,"8292" | }

At this time, the printer is imaging all associated fields, including fields that copy from other fields.

3. Repeat step 2 for each field except the last one.

{B,1,U,0 |
3,"BROADWAY" | }
{B,1,U,0 |
4,"555 WEST OAK AVE." | }

4. For the last field, input data and send it with the quantity of labels to print. When the printer receives input for the last field, it immediately prints the labels.

Example   **{B,1,U,10 |**
5,"DAYTON, OHIO" | }

Increasing Throughput

Reducing the imaging time increases throughput. Increase the baud rate to increase the transmission time and increase throughput. Make sure the communication settings at the printer match those at the host. Using a baud rate of 19200 is almost twice as fast as 9600 baud. Using a baud rate of 38400 is almost twice as fast as 19200 baud.

General Format Tips and Hints

The following tips and hints are helpful to keep in mind when designing MPCLII formats.

With Packets

♦ Leave parameters blank that do not change when sending online configuration packets. For example, {**I**,A,,,,1 | } prints a slashed zero and uses the last sent online system setup parameters.

Group fields with similar parameters. For example

> **T,1,10,V,250,50,1,1,1,1,B,C,0,0 |**
> **T,2,15,,,75 |**
> **T,3,,,,100 |**

The first text field sets all the parameters for that field. The second text field's number of characters and column location changes from what was defined in the first field. In the third text field, only the column location is changed. This method can be used on bar code and constant text fields as well.

Note: Understand the basics of each field before using this method.

After modifying any fields or parameters with the optional entry method, resend the format, batch, or configuration packet to the printer.

With Bar Codes

♦ Be careful when rotating or placing a UPC/EAN bar code with human readable characters, because the bottom reference point is at the bottom of the bars, not at the bottom of the human readable characters.

With Fields

♦ Data that remains the same for each label should be in a constant text field. Data that varies for each label should be in a text field.

♦ Check for trailing spaces in text or constant text fields if a "field off tag" error appears. An easy way to see trailing spaces is to print the field in the reverse font.

When fields are magnified, they may go off the label or cover another field. Magnifying a field increases the distance between the printed character and the edge of the cell.

SAMPLES

This appendix contains sample formats. Customize any of these formats.

The field separator is the split vertical bar (|). The decimal value is 124. To enter this character, use the Shift key plus the Split Vertical Bar key on the computer's keyboard. Depending on the text editor, it may appear as a solid vertical bar or as a split vertical bar.

Sample Aztec Packet

Aztec is a two-dimensional bar code consisting of square modules arranged around a square bulls eye pattern. For more information about the Aztec bar code, refer to the *ANSI/AIM BC13 ISS - Aztec Code Specification* from AIM Global.

```
{F,1,A,R,E,200,200,"TEST" |
B,1,25,V,20,20,37,7,0,0,L,0 |
R,53,0,0,0,1,"" | }
{B,1,N,1 |
1,"Monarch Brand Printers" | }
```

Sample Codabar Packet

Codabar is a linear one-dimensional bar code.

```
{F,101,A,F,E,600,400,"Codabar"|
B,3,12,V,55,50,5,8,85,8,L,0|
  R,1,"1234567890" | }
{B,101,N,1|}
```

Sample Code 16K Packet

Code 16K is a multi-row bar code.

```
{F,101,A,F,E,600,400,"Code16K"|
B,2,100,V,250,50,31,4,0,8,L,0|
  R,1,"1234567890ABCDEF" | }
{B,101,N,1 | }
```

Sample Code 39 Packet

Code 39 is a linear one-dimensional bar code.

```
{F,5,A,R,G,576,768,"1HDREC1" |
L,S,19,39,499,39,38,"" |
L,S,19,93,499,93,15,"" |
L,S,19,124,499,124,15,"" |
L,S,19,155,499,155,38,"" |
C,461,232,0,1,1,1,B,L,0,0,"PO NUMBER ",1 |
C,426,232,0,1,1,1,B,L,0,0,"STORE " |
C,391,232,0,1,1,1,B,L,0,0,"CTNS SHOPPED" |
C,357,232,0,1,1,1,B,L,0,0,"CTNS RECVD " |
C,320,232,0,1,1,1,B,L,0,0,"CARRIER " |
C,259,232,0,1,1,1,B,L,0,0,"FRT BILL # " |
C,196,232,0,1,1,1,B,L,0,0,"FRT TERMS " |
C,159,232,0,1,1,1,B,L,0,0,"KEYREC# " |
C,125,232,0,1,1,1,B,L,0,0,"FRT CLAIM # " |
C,090,232,0,1,1,1,B,L,0,0,"RECVD BY " |
```

```
C,056,232,0,1,1,1,B,L,0,0,"DATE RECVD " |
C,461,445,0,1,1,1,B,L,0,0,":" |
C,426,445,0,1,1,1,B,L,0,0,":" |
C,391,445,0,1,1,1,B,L,0,0,":" |
C,357,445,0,1,1,1,B,L,0,0,":" |
C,320,445,0,1,1,1,B,L,0,0,":" |
C,259,445,0,1,1,1,B,L,0,0,":" |
C,196,445,0,1,1,1,B,L,0,0,":" |
C,159,445,0,1,1,1,B,L,0,0,":" |
C,125,445,0,1,1,1,B,L,0,0,":" |
C,090,445,0,1,1,1,B,L,0,0,":" |
C,056,445,0,1,1,1,B,L,0,0,":" |
T,01,08,V,461,458,0,1,1,1,B,L,0,0 |
T,02,08,V,426,458,0,1,1,1,B,L,0,0 |
T,03,08,V,391,458,0,1,1,1,B,L,0,0 |
T,04,08,V,357,458,0,1,1,1,B,L,0,0 |
T,05,18,V,290,289,0,1,1,1,B,R,0,0 |
T,06,12,V,230,395,0,1,1,1,B,R,0,0 |
T,07,08,V,196,458,0,1,1,1,B,L,0,0 |
T,08,08,V,159,458,0,1,1,1,B,L,0,0 |
T,09,08,V,125,458,0,1,1,1,B,L,0,0 |
T,10,08,V,090,458,0,1,1,1,B,L,0,0 |
T,11,08,V,056,458,0,1,1,1,B,L,0,0 |
T,12,11,V,259,762,0,1,2,2,B,B,0,1 |
  R,1,"*_____*" |
B,13,09,V,259,720,4,4,145,8,B,1 |
  R,4,12,2,9,1,1 |
  R,50,3,8 | }
```

Sample Zero Batch Packet

```
{B,5,N,0 | }
{B,5,U,1 |
1,"6005710" |
2,"106" |
3,"3" |
4,"3" |
5,"ALLIED FREIGHT1234" |
6,"123456789012" |
7,"P" |                    10,"VIC" |
8,"10650337" |             11,"1/6/94" |
9,"0" |                    12,"106503378" | }
```


Sample Code 93 Packet

Code 93 is a linear one-dimensional bar code that provides higher density than Code 39.

```
{F,101,A,F,E,600,400,"EAN"|
B,2,12,V,225,50,23,7,85,8,L,0|
  R,1,"1234567890" | }
{B,101,N,1|}
```
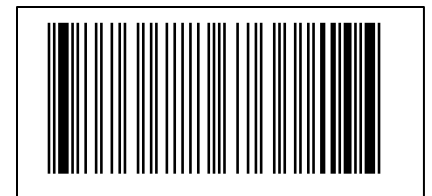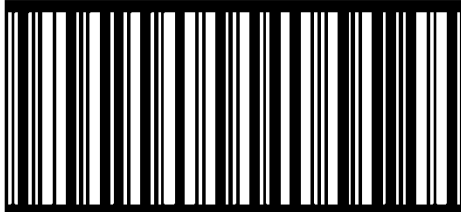
Code 128 is a high density linear bar code for alphanumeric or numeric data.
Codabar is a linear one-dimensional bar code.

```
{F,1,A,R,E,600,400,"RDCI" |
L,V,500,115,90,85,3 |
L,V,298,245,90,102,3 |
L,V,500,2,0,390,3 |
L,V,400,2,0,390,3 |
L,V,298,2,0,390,3 |
L,V,200,2,0,390,5 |
C,568,8,0,2,2,2,B,L,0,0,"FROM:",0 |
C,568,125,0,2,2,2,B,L,0,0,"CARRIER:",0 |
C,529,124,0,2,2,2,B,L,0,0,"PRO NUMBER:",0 |
C,511,125,0,2,2,2,B,L,0,0,"B/L NUMBER:",0 |
C,472,8,0,2,2,2,B,L,0,0,"TO:",0 |
C,387,8,0,2,1,1,B,L,0,0,"(420) SHIP TO POSTAL CODE",0 |
C,391,250,0,2,1,1,B,L,0,0,"APPOINTMENT NUMBER:",0 |
C,358,250,0,2,1,1,B,L,0,0,"ORDER TYPE:",0 |
C,327,250,0,2,1,1,B,L,0,0,"ITEM:",0 |
C,190,8,0,2,1,1,B,L,0,0,"UPC SHIPPING CONTAINER CODE",0 |
C,557,6,0,2,1,1,B,L,0,0,"MONARCH",0 |
C,547,6,0,2,1,1,B,L,0,0,"170 MONARCH LANE",0 |
C,537,6,0,2,1,1,B,L,0,0,"P.O. BOX 608",0 |
C,527,6,0,2,1,1,B,L,0,0,"DAYTON, OHIO 45401",0 |
C,462,313,0,2,4,3,B,L,0,0,"#",0 |
T,1,15,V,529,220,0,2,2,2,B,L,0,0,0 |
T,2,15,V,511,220,0,2,2,2,B,L,0,0,0 |
B,3,13,V,311,28,8,4,50,8,L,0 |
B,4,14,V,17,60,50,5,130,8,L,0 |
T,5,30,V,161,080,0,3,1,1,B,L,0,0,0 |
T,6,15,V,467,40,4,1,2,2,B,L,0,0,0 |
T,7,10,V,462,330,6,2,4,3,B,L,0,0,0 |
T,8,20,V,549,124,0,2,2,2,B,L,0,0,0 |
T,9,30,V,446,40,4,2,2,2,B,L,0,0,0 |
T,10,30,V,426,40,4,2,2,2,B,L,0,0,0 |
T,11,30,V,406,40,4,2,2,2,B,L,0,0,0 |
T,12,20,V,368,255,0,2,3,2,B,L,0,0,0 |
T,13,5,V,335,270,0,2,3,2,B,L,0,0,0 |
T,14,15,V,304,270,0,2,3,2,B,L,0,0,0 |
T,15,15,V,366,65,0,2,2,2,B,L,0,0,0 |
 R,1,"(420)ÿ" | T,16,27,V,270,10,0,3,1,1,B,C,0,0,0 |
T,17,27,V,240,10,0,3,1,1,B,C,0,0,0 |
 R,1,"WELCOMES GUEST # " |
T,18,27,V,210,10,0,3,1,1,B,C,0,0,0 |}
```

FROM:
MONARCH
170 MONARCH LANE
P.O. BOX 608
DAYTON, OHIO 45401

CARRIER:
BROADWAY
PRO NUMBER: 1234567890
B/L NUMBER: 0987654321

TO: RODGER DIST CTR #8292
555 WEST OAK AVE.
DAYTON, OH 45401-0608

(420) SHIP TO POSTAL CODE
(420)32678

APPOINTMENT NUMBER:
08292 123456-123
ORDER TYPE:
AR
ITEM:
999-999999-99

WYSIWYG
WELCOMES GUEST #99999
TO A PRODUCT DEMO

UPC SHIPPING CONTAINER CODE
1 00 28028 66285 4

Sample Batch Packet

{B,1,N,1 |
1,"1234567890" |
2,"0987654321" |
3,"~20142032678" |
4,"10028028662854" |
5,"1 00 28028 66285 4" |
6,"RODGER DIST CTR" |
7,"8292" | 8,"BROADWAY" |
9,"555 WEST OAK AVE." |
10,"DAYTON, OH 45401-0608" |
11,"" |
12,"08292 123456-123" |

13,"AR" |
14,"999-999999-99" |
15,"32678" |
16,"WYSIWYG" |
17,"99999" |
18,"TO A PRODUCT DEMO" | }

Sample Data Matrix Packets

Data Matrix (ECC-200) is a two-dimensional bar code which is made up of square modules arranged within a perimeter finder pattern. There are 24 square symbol sizes available ranging from 10 rows by 10 columns to 144 rows by 144 columns. There are six rectangular symbol sizes available ranging from 8 rows by 8 columns to 16 rows by 48 columns. The symbol size is data dependent. Data Matrix automatically pads data.

Do not overlay other fields when designing the Data Matrix symbol. Smart imaging is automatically disabled on formats with a Data Matrix bar code. Allow a three or four-dot "quiet zone" (blank space around the bar code's perimeter) for scanning. See "Defining a Bar Code Field" for more information.

Sample Batch Data with Special Characters

To use this character in the bar code	enter these characters in the batch data
null character	~~@
~ (tilde)	~126~126
FNC1	~~1

Square Data Matrix Packet

{F,36,A,R,E,400,400,"DTMTRX1" |
B,1,50,V,50,100,35,0,100,8,L,0 | }
{B,36,N,1 |
1,"1234567890ABCDEFGHIJKLMNOPQRST" | }

This example prints a one-inch wide by one-inch tall (100) square Data Matrix symbol using the default density (0) without any field rotation (0).

Rectangular Data Matrix Packet

{F,36,A,R,E,400,400,"DTMTRX2" |
B,1,400,V,100,200,35,29,50,8,L,1 | }
{B,36,N,1 |
1,"1234567890ABCDEFGHIJKLMNOPQRST" | }

This example prints a one-inch by a half-inch tall (50) rectangular 16 rows by 36 columns (density 29) Data Matrix symbol rotated 90 (1).

Sample Data Matrix with Function 1

{F,36,A,R,E,400,400,"DTMTRX1" |
B,1,50,V,10,50,35,0,50,8,L,0 | }
{B,36,N,1 |
1,"~~110012345678902" | }

This example prints a 0.50-inch wide by 0.50-inch tall (50) square Data Matrix symbol using the default density (0) without any field rotation (0). FNC1 appears in the batch data as ~~1.

Sample EAN Packet

European Article Number is a linear bar code for numeric data.

{F,101,A,F,E,600,400,"EAN"|
B,1,10,F,395,50,14,2,85,7,L,0|
** R,1,"1234567890" | }**
{B,101,N,1|}

Sample GS1 DataBar Packets

GS1 DataBar encodes brand identification and dynamic data. This family of bar codes can contain more information than current EAN/UPC bar codes. Seven symbologies encode a Global Trade Item Number (GTIN). GS1 DataBar codes can be stacked, truncated, or omni directional.

GS1 DataBar with Function 1

{F,1,A,R,E,400,400,"UCCAB" |
B,1,30,V,5,65,38,4,35,0,L,0,11,2,22 | }
{B,1,N,1 |
1,"#10ABC|#Monarch Brand Printers" | }

GS1 DataBar EAN13 with Composite

{F,1,A,R,E,600,400,"GS1EAN13" |
B,0,39,V,285,100,38,3,60,8,L,0,9,1,22 |
R,1,"123456789012|#910123456#011234567890123" | }
{B,1,N,1 | }

GS1 DataBar 14 Stacked Omni Directional

{F,17,A,R,E,400,400,"STACKOM" |
B,1,120,V,50,100,38,3,100,0,L,0,4,2 | }
{B,17,N,1 |
1,"1234567890123" | }

GS1 DataBar Expanded with CC

{F,1,A,R,E,400,400,""|
B,1,75,V,15,25, 38,2,0,0,L,0,6,1,4|}
{B,1,N,1|
1,"0193712345678904 3103001234|911A2B3C4D5E"|}

GS1 DataBar Expanded

{F,27,A,R,E,400,400,"EXPANDED" |
B,1,120,V,50,100,38,3,100,0,L,0,6,,6 | }
{B,27,N,1 |
1,"1234567890123456789012345678901234567890123456789 0
12345678901234567890" | }
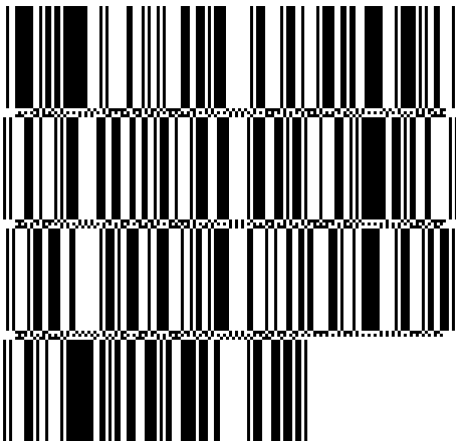
GS1 DataBar Expanded (no composite)

{F,1,A,R,E,600,400,"GS1EXPD" |
B,0,16,V,425,25,38,3,50,8,L,0,6,1,22 |
R,1,"10123456#21ABCD" | }
{B,1,N,1 | }

Sample Interleaved Two of Five (I 2of5) Packet

Interleaved Two of Five is a linear bar code for numeric data.

{F,11,A,R,E,300,400,"IOF5" |
B,4,12,V,125,40,3,12,80,8,L,0 | }
{B,11,N,1|
4,"028028123456"| }
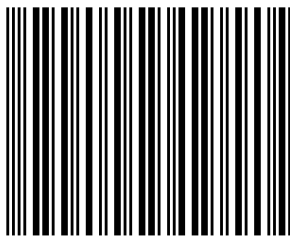
Sample Intelligent Mail® Packet

For the Intelligent Mail® bar code, the maximum number of characters is 31, including commas between the data. For more information, refer to the *USPS® Intelligent Mail® Specification*.

{F,1,A,R,G,1218,812,"intmail"|
B,1,35,V,100,100,24,0,0,8,L,0|}
{B,1,N,1|
1, "11,222,333333,444444444,55555555555"| }

In the above sample, the batch data indicates the following:

Batch Data	Description
11	**Bar code ID:** First digit = 0 to 9; Second digit = 0 to 4
222	**Service Type ID:** Range: 000 to 999
333333	**Mailer ID:** Range: 000000 to 999999
444444	**Serial Number 6 or 9 digits:** Range: 000000000 to 999999999
555555555	**Delivery Point ZIP code:** 0, 5, 9, or 11-digit ZIP code

Sample MaxiCode Packets

MaxiCode is a two-dimensional bar code developed by UPS (United Parcel Service, Inc.). Data must be defined in a specific way for UPS. Refer to the *Guide to Bar Coding with UPS* or the *AIM MaxiCode Specification* for more details about data requirements.

The printer supports modes 0, 1, 2, and 3. Contact Avery Dennison for information about additional MaxiCode modes.

Mode	Description
0	Obsolete
1	Obsolete
2	Structured Message
3	Structured Message

Select which mode to use in the bar code field or allow the printer to auto-select the mode (0, 2, or 3) based on the data. See "Defining a Bar Code Field" for more information. MaxiCode automatically pads data with the "!" character.

Note: MaxiCode does not support the NULL character.

Modes 2 and 3 are defined by the way the postal code, class of service, and country code fields are arranged. (The postal code, class of service, and country code are required fields.) Begin with the message header, then the primary data (15 characters), followed by the secondary message (up to 78 characters). Or, begin with the primary data, then the message header, followed by the secondary data. If the postal code data characters are all numeric then the MaxiCode symbol is set to Mode 2. If the characters are alphanumeric, or only contain ASCII characters 65 to 90, then the MaxiCode symbol is set to Mode 3.

If error 612 appears, check the MaxiCode data. It may not be correctly structured or missing one of the three required fields (postal code, class of service, and country code) or the "~029" character.

Mode 0 (Obsolete) Sample

{F,1,A,R,E,0200,0200,"MAXICODE" |	
**B,1,93,V,020,20,33,7,0,8,L,0	}**	MaxiCode bar code (33)
{B,1,N,1 |	Batch header
1,"450660000" |	Postal code- zip code
	(This field determines Mode)
C,"001" |	Country code
C,"840" |	Class of service
C,"[)~030" |	Message header
C,"01~02996" |	Transportation header
C,"1Z12345678~029" |	Tracking number
C,"UPSN~029" |	Origin carrier SCAC
C,"12345A~029" |	UPS shipper number
C,"070~029" |	Julian day of pickup
C,"~029" |	Shipment ID (empty)
C,"1/1~029" |	Package count
C,"15~029" |	Weight (lb.)
C,"Y~029" |	Address validation
C,"60 SADDLEBROOK CT.~029" |	Street address (empty)
C,"DAYTON~029" |	City (empty)
C,"OH~030" |	State
**C,"~004"	}**	EOT
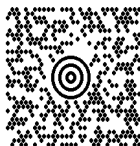
Mode 2 Sample

{F,1,A,R,E,200,200,"MAXI_M2" |
B,1,93,V,020,020,33,7,0,8,L,0 | }   MaxiCode bar code (33)
{B,1,N,1 |
1,"[)>~030" | Message header
C,"01~02996" | Transportation header
C,"068100000~029" | Postal Code
          (This field determines Mode)

C,"840~029" | Country code
C,"001~029" | Class of service
C,"1Z12345675~029" | Tracking number
C,"UPSN~029" | Origin carrier SCAC
C,"12345E~029" | UPS shipper number
C,"089~029" | Julian day of pickup
C,"~029" | Shipment ID (empty)
C,"1/1~029" | Package count
C,"10~029" | Weight (lb.)
C,"Y~029" | Address validation
C,"~029" | Street address (empty)
C,"~029" | City (empty)
C,"CT~030" | State
C,"~004" | }   EOT
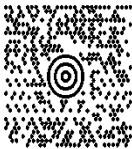
Mode 3 Sample

```
{F,1,A,R,E,200,200,"MAXI_M3" |
B,1,93,V,020,020,33,7,0,8,L,0 | }          MaxiCode bar code (33)
{B,1,N,1 |
1,"[)>~030" |                              Message header
C,"01~02996" |                             Transportation header
C,"M5E1G45~029" |                          Postal Code
                                           (This field determines Mode)

C,"124~029" |                              Country code
C,"066~029" |                              Class of service
C,"1Z12345679~029" |                       Tracking number
C,"UPSN~029" |                             Origin carrier SCAC
C,"12345E~029" |                           UPS shipper number
C,"089~029" |                              Julian day of pickup
C,"~029" |                                 Shipment ID (empty)
C,"1/1~029" |                              Package count
C,"10~029" |                               Weight (lb.)
C,"Y~029" |                                Address validation
C,"~029" |                                 Street address (empty)
C,"TORONTO~029" |                          City (empty)
C,"ON~030" |                               State
C,"~004" | }                               EOT
```
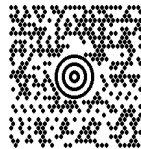
MaxiCode Compression Sample

Once the data is properly encoded per the compressed format, the UPS scanners and software do the translating of the compressed data when reading the symbol. Our printers take the input data as-is and encode it as a standard MaxiCode symbol. Standard scanners interpret the encoded data exactly as it appears in the symbology without decompressing its contents.

Note: Compressed data can not contain carriage return or line feed (CRLF) characters.

```
{F,1,A,R,E,600,400,"Compress"|
B,01,120,V,272,15,33,7,0,8,L,0|}          appearance (8) for compressed data
{B,1,N,1|
1,"[)>~030"|
C,"01~029"|
C,"96453420000~029"|
C,"840~029"|
C,"013~029"|
C,"1Z12121212~029"|
C,"UPSN~029"|
C,"564W65~030"|                          start of compressed data
C,"07'4N%KKL8X5~029"|
C,"9T#VO:Z+Z 0~029"|
C,"SFBH3R#&ZAT/~029"|
C,"87EPJ)~029"|                           end of compressed data
C,"S~013"|
C,"~030"|
C,"~004"|}
```
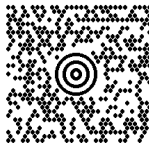
Batch data can also be included as one long string without carriage return or line feed (CRLF) characters.

```
{B,1,N,1|
1,"[)>~03001~02996142010000~029840~029003~0291Z00000014~029UPSN~029111111~03007#P36
(AWO'$6,X3&W6HMJAL-7WK0 8YU,)92+'#I%~029#S~013~030~004"|}
```

Sample MSI Packet

MSI is a linear bar code for numeric data.

{F,101,A,F,E,600,400,"MSI"|
B,3,12,V,55,50,9,7,85,8,L,0|
R,1,"1234567890" | }
{B,101,N,1 | }
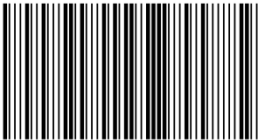
Sample PDF417 Packet

PDF417 is two-dimensional barcode that contains alphanumeric or numeric data.

{F,101,A,F,E,600,400,"POSTNET"|
B,2,100,V,225,50,32,7,0,8,L,0|
R,1,"1234567890ThisisasamplepdF417barcodeAveryDennison2015" | }
{B,101,N,1 | }

Sample POSTNET Packet

POSTNET (**Pos**tal **N**umeric **E**ncoding **T**echnique) is a barcode that contains numeric data.

{F,101,A,F,E,600,400,"POSTNET"|
B,1,9,F,395,50,22,0,0,8,L,0|
R,1,"123456789" | }
{B,101,N,1 | }

Sample Quick Response Packets

Quick Response (QR Code) is a two-dimensional bar code, which is made up of square modules arranged in an overall square pattern. A unique finder pattern is located at three corners of the symbol. Four levels of error correction are available, along with a wide range of symbol sizes.

♦ Model 1 is the original specification.

♦ Model 2 is an enhanced form that includes additional features.

The maximum number of characters depends on the type of characters entered for the batch data and differs for the two models of QR Code.

Data Type	Model1	Model2
Numeric Data	1167	2710
Alphanumeric Data	707	2710
8-Bit data	486	2710
Kanji data	299	1817

QR Code can accommodate Japanese *Kana* and *Kanji* characters and has a variety of applications, including marking spark plugs, radiators, printed circuit boards, and test tubes. Refer to the *AIM International Symbology Specification* for more details about data requirements.

Sample QR Code Packet

{F,1,A,R,E,200,200,"QRCODE" |
B,1,200,V,75,50,36,0,100,2,B,0 | }
{B,1,N,1 |
1,"HM,N0123456789012345" |}

Sample QR Code with URL Packet

{F,1,A,R,E,200,200,"QRURL" |
B,2,200,V,75,50,36,0,100,2,B,0 | }
{B,1,N,1 |
2,"MA,http://www.monarch.averydennison.com" | }

Structured Append QR Code Packet

{F,2,A,R,E,200,200,"QRCODE2" |
B,1,200,V,50,50,36,0,100,2,B,0 | }
{B,2,N,1 | 1,"D0202E9,Q0A" |
C,"0123456789ABCD+__âôû
~129~064~159~252~224~064" | }

Sample UPCA Format Packet

{F,25,A,R,E,200,200,"Fmt 25" |
C,140,40,0,1,2,1,W,C,0,0,"SAMPLE FORMAT" |
B,1,12,F,85,40,1,2,40,5,L,0 |
T,2,18,V,50,50,1,3,1,1,B,L,0,0 | }

Sample Batch Packet

{B,25,N,1 |
1,"02802811111" |
2,"TEXT FIELD" | }
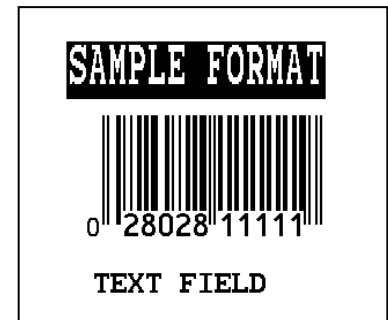
Sample Data Entry Format Packet (UPCA Bar Code)

This packet uses Option 5 (Data Entry Sources) and Option 20 (Data Entry Prompts). After the printer receives the format packet, the operator can go into Batch Entry Mode at the printer and enter the batch data.

{F,1,A,R,E,300,200,"TEXTILES" |
T,1,10,V,250,50,0,1,1,1,B,C,0,0,0 |
 R,20,"PART NUMBER:" |
 R,5,K |
B,2,12,V,150,40,1,2,80,7,L,0 |
 R,20,"UPC NUMBER:" |
 R,5,K |
D,3,20 |
 R,20,"FIBER#1:" |
 R,5,K |
D,4,3 |
 R,20,"PERCENTAGE:" |
 R,5,K |
T,5,30,V,80,10,0,1,1,1,B,L,0,0,0 |
 R,1," % " |
 R,4,3,1,20,5,1 |
 R,4,4,1,3,1,1 |
 R,5,N | D,6,20 |
 R,20,"FIBER #2:" |
 R,5,K |
D,7,3 |
 R,20,"PERCENTAGE:" |
 R,5,K |
T,8,30,V,65,10,0,1,1,1,B,L,0,0,0 |
 R,1," % " |
 R,4,6,1,20,5,1 |
 R,4,7,1,3,1,1 |
 R,5,N |
C,30,10,0,1,1,1,B,L,0,0,"MADE IN USA",0 |
L,S,110,30,110,150,10," " |
Q,240,30,270,150,3," " | }

Depending on the operator-entered batch data, a label similar to this one prints.
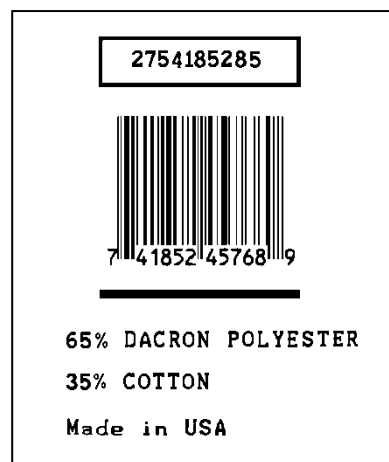
{F,1,A,R,E,200,200,"EPCFONT" |
T,40,1,F,10,10,0,56,40,40,B,L,0,0|}
{B,1,N,1 |
40,"A"|}

RFID Data Field with a Copy Option

F,1,A,R,E,600,400,"RDCI" |
L,V,500,115,90,85,3 |
L,V,298,245,90,102,3 |
C,568,8,0,2,2,2,B,L,0,0,"FROM:",0 |
C,568,125,0,2,2,2,B,L,0,0,"CARRIER:",0 |
C,387,8,0,2,1,1,B,L,0,0,"(420) SHIP TO POSTAL CODE",0 |
C,391,250,0,2,1,1,B,L,0,0,"APPOINTMENT NUMBER:",0 | C,327,250,0,2,1,1,B,L,0,0,"ITEM:",0 |
C,190,8,0,2,1,1,B,L,0,0,"UPC SHIPPING CONTAINER CODE",0 | T,1,15,V,529,220,0,2,2,2,B,L,0,0,0
|
B,3,13,V,311,28,8,4,50,8,L,0 |
B,4,14,V,17,60,50,5,130,8,L,0 |
X,5,24,0 |
T,6,20,V,415,270,0,50,15,15,B,L,0,2 |
R,4,5,1,16,1,0 |

Sample Batch Packet

B,1,N,1 |
1,"RFID TEST" |
3,"1005678" |
4,"67-90-32" |
5,"3123456789ABCDEF12345678" |
6,"ABCDEFG" |

ASCII HEX

{F,2,A,R,E,400,400,"ASCIIHEX" |
T,1,50,V,10,10,0,1,1,1,B,L,0,0 |
X,2,24,0 |}
{B,2,N,1 |
1,"31323334353637383930 3132" |
2,"31323334353637383930 3132" |}

ASCII HEX Incrementing Field

{F,2,A,R,E,400,400,"ASCIIHEX" |
T,1,50,V,10,10,0,1,1,1,B,L,0,0 |
X,2,24,0 |
** R,60,I,1 |}**
{B,2,N,5 |
1,"31323334353637383930 3132" |
2,"31323334353637383930 3132" | }

SSCC96

```
{F,45,A,R,E,600,400,"SSCC96" |
C,45,220,0,50,10,10,B,L,0,2,"®" |
C,75,385,0,50,40,30,B,L,0,2,"Monarch RFID" |
C,110,385,0,50,6,6,B,L,0,2,"SHIP TO RFID USER" |
C,110,150,0,50,6,6,B,L,0,2,"CARRIER" |
C,150,150,0,50,9,9,B,L,0,2,"PRO:" |
C,165,150,0,50,9,9,B,L,0,2,"B/L:" |
C,200,380,0,50,18,15,B,L,0,2,"AVERY DENNISON" |
C,235,380,0,50,14,12,B,L,0,2,"EMAIL: RFID@MONARCH.COM" |
C,255,380,0,50,14,12,B,L,0,2,"PHONE: 1 800 543-6650" |
C,275,345,0,50,6,6,B,L,0,2,"VENDOR STK NO:" |
C,275,110,0,50,6,6,B,L,0,2,"PACK/UNITS:" |
C,360,330,0,50,10,10,B,L,0,2,"EPC Pure Identity:" |
C,275,190,0,50,6,6,B,L,0,2,"COLOR:" |
C,315,110,0,50,6,6,B,L,0,2,"SIZE/STYLE:" |
C,415,390,0,50,12,12,B,L,0,2,"EPC#:" |
L,S,098,005,098,395,6,"" |
L,S,170,005,170,395,6,"" |
L,S,098,155,170,155,6,"" |
L,S,260,005,260,395,6,"" |
L,S,390,005,390,395,6,"" |
T,1,15,V,130,335,0,50,13,12,B,L,0,2 |
B,2,15,V,165,365,8,4,25,0,L,2 |
T,3,20,V,130,150,0,50,13,12,B,L,0,2 |
T,4,20,V,150,110,0,50,10,12,B,L,0,2 |
T,5,20,V,165,110,0,50,10,12,B,L,0,2 |
T,12,10,V,295,370,0,50,15,18,B,L,0,2 |
T,13,10,V,295,110,0,50,15,18,B,L,0,2 |
T,14,50,V,380,385,0,50,12,11,B,L,0,2 |
T,15,10,V,295,200,0,50,15,18,B,L,0,2 |
T,16,10,V,335,155,0,50,15,18,B,L,0,2 |
T,18,30,V,550,330,0,50,15,18,B,L,0,2 |
B,17,20,V,530,340,50,6,110,0,L,2 |
X,19,24 |
T,20,40,V,415,330,0,50,12,13,B,L,0,2 |
R,4,19,1,40,1,1 |}

{B,45,N,1 |
1,"VENDOR USA" |
2,"42060512" |
3,"AVERY DENNISON" |
4,"0987764356" |
5,"0020545640" |
12,"0075687332" |
13,"3600" |
14,"urn:epc:tag:sscc-96:1.0028028.0000001235" |
15,"RED" |
16,"48~"/TOUGH" |
17,"009280287586887" |
18,"0  09 28028  75688  7" |
19,"313401B5F0000004D3000000" |
20,"0" |}
```

SGTIN 96

```
{F,46,A,R,E,600,400,"SGTIN96" |
C,45,220,0,50,10,10,B,L,0,2,"®" |
C,75,385,0,50,40,30,B,L,0,2,"Monarch RFID" |
C,110,385,0,50,6,6,B,L,0,2,"SHIP TO RFID USER" |
C,110,150,0,50,6,6,B,L,0,2,"CARRIER" |
C,150,150,0,50,9,9,B,L,0,2,"PRO:" |
C,165,150,0,50,9,9,B,L,0,2,"B/L:" |
C,200,380,0,50,18,15,B,L,0,2,"AVERY DENNISON" |
C,235,380,0,50,14,12,B,L,0,2,"EMAIL: RFID@MONARCH.COM" |
C,255,380,0,50,14,12,B,L,0,2,"PHONE: 1 800 543-6650" |
C,275,345,0,50,6,6,B,L,0,2,"VENDOR STK NO:" |
C,275,110,0,50,6,6,B,L,0,2,"PACK/UNITS:" |
C,360,330,0,50,10,10,B,L,0,2,"EPC Pure Identity:" |
C,275,190,0,50,6,6,B,L,0,2,"COLOR:" |
C,315,110,0,50,6,6,B,L,0,2,"SIZE/STYLE:" |
C,415,390,0,50,12,12,B,L,0,2,"EPC#:" |
L,S,098,005,098,395,6,"" |
L,S,170,005,170,395,6,"" |
L,S,098,155,170,155,6,"" |
L,S,260,005,260,395,6,"" |
L,S,390,005,390,395,6,"" |
T,1,15,V,130,335,0,50,13,12,B,L,0,2 |
B,2,15,V,165,365,8,4,25,0,L,2 |
T,3,20,V,130,150,0,50,13,12,B,L,0,2 |
T,4,20,V,150,110,0,50,10,12,B,L,0,2 |
T,5,20,V,165,110,0,50,10,12,B,L,0,2 |
T,12,10,V,295,370,0,50,15,18,B,L,0,2 |
T,13,10,V,295,110,0,50,15,18,B,L,0,2 |
T,14,50,V,380,385,0,50,12,11,B,L,0,2 |
T,15,10,V,295,200,0,50,15,18,B,L,0,2 |
T,16,10,V,335,155,0,50,15,18,B,L,0,2 |
T,18,30,V,550,330,0,50,15,18,B,L,0,2 |
B,17,20,V,530,340,50,6,110,0,L,2 |
X,19,24 |
T,20,40,V,415,330,0,50,12,13,B,L,0,2 |
R,4,19,1,40,1,1 |}

{B,46,N,1 |
1,"VENDOR USA" |
2,"42060512" |
3,"AVERY DENNISON" |
4,"0987764356" |
5,"0020545640" |
12,"0075687332" |
13,"3600" |
14,"urn:epc:tag:sgtin-96:1.0028028.001234.2" |
15,"RED" |
16,"48~"/TOUGH" |
17,"009280287586887" |
18,"0  09 28028  75688  7" |
19,"303401B5F001348000000002" |
20,"0" |}
```

Password Lock

{F,1,A,R,E,400,400,"PWDLOCK" |
X,1,100,0 |}
{B,1,N,1 |
1,"31323334353637383132334~028" |
C,"ABCDEFGH01234567UserMemorySample~028" |
C,"~028" |
C,"73737373~028" |
C,"CAD01234~028" |
C,"22022" |}

Permalock and Password Lock

{F,1,A,R,E,400,400,"BOTHLOCK" |
X,1,100,0 |}
{B,1,N,1 |
1,"31323334353637383132334~028" |
C,"ABCDEF ~028" |
C,"~028" |
C,"73737373~028" |
C,"CAD01234~028" |
C,"33033" |}

RFID Data Field with Serialized EPC Data Option

{F,2,A,R,E,400,400,"SerEPC" |
T,1,50,V,10,10,0,1,1,1,B,L,0,0 |
X,2,24,0 |
 R,65,1 | }
{B,2,N,1 |
1,"313233343536373839303132" |
2,"313233343536370000000000" |}

The RFID data field includes zeroes so the field's maximum number of characters is 96 bits of data (24 ASCII Hex characters). The encoded RFID data contains the first 58 bits of provided EPC data and 38 bits of the calculated serial number.

FONTS

Our printers support two types of fonts: Bitmapped (traditional printer fonts such as Standard and Reduced) and Scalable/TrueType® (Font 50). This appendix gives a brief overview of each type of font and how the printer interprets fonts. It also shows examples of the printer's installed fonts.

Number	Font Size and Appearance	Type of Spacing	# of Dots Between Characters
1	Standard	Monospaced	3 (203 dpi) 5 (300 dpi)
2	Reduced	Monospaced	1 (203 dpi) 2 (300 dpi)
3	Bold	Monospaced	3 (203 dpi) 5 (300 dpi)
4	OCRA-like	Monospaced	3 (203 dpi) 5 (300 dpi)
5	HR1 – only for numeric data	Monospaced	2 (203 dpi) 3 (300 dpi)
6	HR2 – only for numeric data	Monospaced	1 (203 dpi) 2 (300 dpi)
10*	CG Triumvirate™ Typeface Bold (9pt. at 203 dpi or 8 pt. at 300 dpi)	Proportional	Varies with each letter
11*	6 pt. CG Triumvirate™ Typeface	Proportional	Varies with each letter
15*	7 pt. CG Triumvirate™ Typeface	Proportional	Varies with each letter
16*	9 pt. CG Triumvirate™ Typeface	Proportional	Varies with each letter
17*	11 pt. CG Triumvirate™ Typeface	Proportional	Varies with each letter
18*	15 pt. CG Triumvirate™ Typeface	Proportional	Varies with each letter
50	EFF Swiss Bold	Scalable	Varies with each letter
56	PaxarSymbols	Proportional	Varies with each letter
70	15 pt. Paxar	Proportional	Varies with each letter
71	18 pt. Paxar	Proportional	Varies with each letter
72	15 pt. NAFTA	Proportional	Varies with each letter
73	18 pt. NAFTA	Proportional	Varies with each letter

* The CG Triumvirate™ typefaces support only ANSI and DOS Code Page 437 and 850 Symbol Sets. The scalable font does not support Code Page 1256 (Arabic). The Euro symbol(€) at position ~192 is only available in the Standard, Reduced, and Bold fonts.

These samples were printed using the Internal Symbol set.

Standard Font

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;<
=>?@!"#$%&'()
*+,-./[\]^_`{
¦}~ÇüéÉæÆáíóú
ñÑªº¿⌐¬½¼¡«»
$£¥₧FPL.K̃MϼℛℰW
℟¥€_Øøαß™

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;<
=>?@!"#$%&'()
*+,-./[\]^_`{
¦}~ÇüéÉæÆáíóú
ñÑªº¿⌐¬½¼¡«»
$£¥₧FPL.K̃MϼℛℰW
℟¥€_Øøαß™

Reduced Font

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;<
=>?@!"#$%&'()
*+,-./[\]^_`{
¦}~ÇüéÉæÆáíóú
ñÑªº¿⌐¬½¼¡«»
$£¥₧FPL.K̃MϼℛℰW
℟¥€_Øøαß™

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;<
=>?@!"#$%&'()
*+,-./[\]^_`{
¦}~ÇüéÉæÆáíóú
ñÑªº¿⌐¬½¼¡«»
$£¥₧FPL.K̃MϼℛℰW
℟¥€_Øøαß™

Bold Font

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
0123456789:;<
=>?@!"#$%&'()
*+,-./[\]^
$£¥₧FPL.K̃MϼℛℰW
℟¥€_

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
0123456789:;<
=>?@!"#$%&'()
*+,-./[\]^
$£¥₧FPL.K̃MϼℛℰW
℟¥€_

OCRA-like Font

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
0123456789<>"
$⁒+¬-.⁄$£¥₧FP
L.K̃MϼℛℰWℬ¥

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
0123456789<>"
$⁒+¬-.⁄$£¥₧FP
L.K̃MϼℛℰWℬ¥

EFF Swiss Bold Font*

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;<=>?
@!"#$%&'()*+,-./
[\]^_`{|}~€ ‚
ƒ„…†‡ˆ‰Š‹Œ Ž
''""•––—˜™š›œ
 žŸ ¡¢£¤¥¦§¨©
ª«¬-®¯°±²³´µ¶
·¸¹º»¼½¾¿ÀÁÂÃ
ÄÅÆÇÈÉÊËÌÍÎÏÐ
ÑÒÓÔÕÖ×ØÙÚÛÜÝ
Þßàáâãäåæçèéê
ëìíîïðñòóôõö÷
øùúûüýþ

* Printed with ANSI Symbol Set

These samples were printed using Code Page 437.

CG Triumvirate™ Typeface Bold

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;< = >?
@!"#$%&'()* + ,-./
[\]^_`{¦}˜Çüé
âäàåçêëèïîìÄÅ
ÉæÆôöòûùÿÖÜ¢£
¥PtƒáíóúñÑªº¿⌐

¬ ½ ¼ ¡«»░▒▓│┤╡
╢╖╕╣║╗╝╜╛┐└┴┬├─
┼╞╟╚╔╩╦╠═╬╧╨╤╥
╙╘╒╓╫╪┘┌█▄▌▐▀

αßΓπΣσµτΦΘΩ
δ∞φε∩ ≡ ± ≥ ≤ ⌠⌡ ÷ ≈
° · ·√ⁿ²■

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;< = >?
@!"#$%&'()* + ,-./
[\]^_`{¦}˜Çüé
âäàåçêëèïîìÄÅ
ÉæÆôöòûùÿÖÜ¢£
¥PtƒáíóúñÑªº¿⌐

¬ ½ ¼ ¡«»░▒▓│┤╡
╢╖╕╣║╗╝╜╛┐└┴┬├─
┼╞╟╚╔╩╦╠═╬╧╨╤╥
╙╘╒╓╫╪┘┌█▄▌▐▀

αßΓπΣσµτΦΘΩ
δ∞φε∩ ≡ ± ≥ ≤ ⌠⌡ ÷ ≈
° · ·√ⁿ²■

CG Triumvirate™ Typeface

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;< = >?
@!"#$%&'()* + ,-./
[\]^_`{¦}˜Çüé
âäàåçêëèïîìÄÅ
ÉæÆôöòûùÿÖÜ¢£
¥PtƒáíóúñÑªº¿⌐

¬ ½ ¼ ¡«»░▒▓│┤╡
╢╖╕╣║╗╝╜╛┐└┴┬├─
┼╞╟╚╔╩╦╠═╬╧╨╤╥
╙╘╒╓╫╪┘┌█▄▌▐▀

αßΓπΣσµτΦΘΩ
δ∞φε∩ ≡ ± ≥ ≤ ⌠⌡ ÷ ≈
° · ·√ⁿ²■

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;< = >?
@!"#$%&'()* + ,-./
[\]^_`{¦}˜Çüé
âäàåçêëèïîìÄÅ
ÉæÆôöòûùÿÖÜ¢£
¥PtƒáíóúñÑªº¿⌐

¬ ½ ¼ ¡«»░▒▓│┤╡
╢╖╕╣║╗╝╜╛┐└┴┬├─
┼╞╟╚╔╩╦╠═╬╧╨╤╥
╙╘╒╓╫╪┘┌█▄▌▐▀

αßΓπΣσµτΦΘΩ
δ∞φε∩ ≡ ± ≥ ≤ ⌠⌡ ÷ ≈
° · ·√ⁿ²■

PaxarSymbols Font 56 Characters

A (decimal 65) =

{F,1,A,R,E,200,200,"epc" |
T,1,10,V,30,35,0,56,20,20,B,L,0,0|}
{B,1,N,1|
1,"~065   A"|}
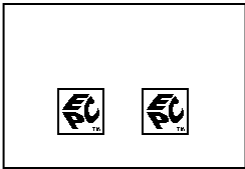
Paxar Font 70 and Font 71 Characters

32	33	34	35	36	37	38	39
	○	Ⓐ	Ⓕ	Ⓟ professional dry clean	⊗	Ⓕ	Ⓟ
40	**41**	**42**	**43**	**44**	**45**	**46**	**47**
Cl	Cl	No bleach	Tumble dry	No tumble dry	Tumble dry low heat	Tumble dry high heat	
48	**49**	**50**	**51**	**52**	**53**	**54**	**55**
		Cool iron	Warm iron	Hot iron	No iron	Hand Wash	Do Not Wash
56	**57**	**58**	**59**	**60**	**61**	**62**	**63**
30 Cotton Wash	30 Synthetics Wash	40 Cotton Wash	40 Synthetics Wash	50 Cotton Wash	50 Synthetics Wash	60 Cotton Wash	60 Synthetics Wash
64	**65**	**66**	**67**	**68**	**69**	**70**	**71**
70 Cotton Wash	70 Synthetics Wash	95 Cotton Wash	95 Synthetics Wash	1 95	1 100	2 60	3 60
72	**73**	**74**	**75**	**76**	**77**	**78**	**79**
4 50	5 40	6 40	7 40	8 30	9 95	Ⓢ	Ⓜ
80	**81**	**82**	**83**	**84**	**85**	**86**	**87**
Ⓛ	ⓍⓁ	110	150	200	40 Wool Wash	85	30

Use the tilde sequences shown in this table to produce the desired character on tags/labels.

{F,1,A,R,E,200,200,"Font70" |
T,1,10,V,25,25,0,70,2,2,B,L,0,0|
T,2,10,V,65,65,0,70,1,1,B,L,0,0|}
{B,1,N,1|
1,"~042 ~064"|
2,"~052   ~054"|}

B-4 Packet Reference Manual
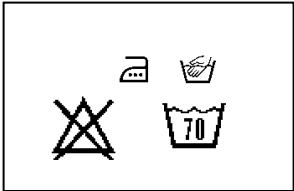
NAFTA Font 72 and Font 73 Characters

32	33	34	35	36	37	38	39
	Machine Wash	Machine Wash	Machine Wash	Machine Wash	Machine Wash	Machine Wash	Machine Wash
40	**41**	**42**	**43**	**44**	**45**	**46**	**47**
Hand Wash	Hand Wash	Hand Wash	Hand Wash	Hand Wash	Hand Wash	Machine Wash, Perm. Press	Machine Wash, Perm. Press
48	**49**	**50**	**51**	**52**	**53**	**54**	**55**
Machine Wash, Perm. Press	Machine Wash, Perm. Press	Machine Wash, Perm. Press	Machine Wash, Perm. Press	Machine Wash, Gentle	Machine Wash, Gentle	Machine Wash, Gentle	Machine Wash, Gentle
56	**57**	**58**	**59**	**60**	**61**	**62**	**63**
Machine Wash, Gentle	Machine Wash, Gentle	Do Not Wash	Do Not Bleach	Do Not Dry	No Iron	Bleach When Needed	Non-Chlorine Bleach When Needed
64	**65**	**66**	**67**	**68**	**69**	**70**	**71**
	Tumble Dry	Tumble Dry, High Heat	Tumble Dry, Medium Heat	Tumble Dry, Low Heat	Tumble Dry, No Heat	Tumble Dry, Perm. Press	Tumble Dry, Perm. Press, High Heat
72	**73**	**74**	**75**	**76**	**77**	**78**	**79**
Tumble Dry, Perm. Press, Medium Heat	Tumble Dry, Perm. Press, Low Heat	Tumble Dry, Perm. Press, No Heat	Tumble Dry, Gentle	Tumble Dry, Gentle, High Heat	Tumble Dry, Gentle, Medium Heat	Tumble Dry, Gentle, Low Heat	Tumble Dry, Gentle, No Heat
80	**81**	**82**	**83**	**84**	**85**	**86**	**87**
Line Dry	Drip Dry	Dry Flat	Do Not Wring	Do Not Tumble Dry	Dry In Shade	Iron	Iron, High Heat
88	**89**	**90**	**91**	**92**	**93**	**94**	**95**
Iron, Medium Heat	Iron, Low Heat	Iron, High Heat, No Steam	Iron, Medium Heat, No Steam	Iron, Low Heat, No Steam	Professional dry clean	Do Not Dry clean	Dry clean, Any Solvent

96	97	98	99	100	101	102	103
(P)	(F)	(A)	(P)	(F)	(A)	(P)	(F)
Dry clean, No Trichloro-ethylene	Dry clean, Petroleum Solvent Only	Dry clean, Any Solvent, Short Cycle	Dry clean, No Trichloro-ethylene, Short Cycle	Dry clean, Petroleum Solvent Only, Short Cycle	Dry clean, Any Solvent, Reduced Moisture	Dry clean, No Trichloro-ethylene, Reduced Moisture	Dry clean, Petroleum Solvent Only, Reduced Moisture

104	105	106	107	108	109		
(A)	(P)	(F)	(A)	(P)	(F)		
Dry clean, Any Solvent, Low Heat	Dry clean, No Trichloro ethylene, Low Heat	Dry clean, Petroleum Solvent Only, Low Heat	Dry clean, Any Solvent, No Steam	Dry clean, No Trichloro ethylene, No Steam	Dry clean, Petroleum Solvent Only, No Steam		

Use the tilde sequences shown in this table to produce the desired character on tags/labels.

{F,1,A,R,E,200,200,"Font72" |
T,1,10,V,25,25,0,72,2,2,B,L,0,0|
T,2,10,V,65,65,0,72,1,1,B,L,0,0|}
{B,1,N,1|
1,"~042 ~067"|
2,"~080 ~087"|}

Bitmap Font Information

Our bitmap fonts are either monospaced (each character occupies the same amount of space) or proportional (each character is a different width). Use monospaced fonts for price fields and data to list in a column. With proportionally spaced fonts, more characters may be placed per line. Experiment with these fonts and adjust field measurements as necessary. The bitmapped fonts (either monospaced or proportional) appear jagged when magnified. The magnification range is 1 to 7.

Use the MONARCH® MPCL™ Toolbox (Font Utility), available on our Web site, to convert a bitmap font to Hex or Run-Length encoding for the printer. Select the point size and characters to print. Bitmap fonts may image faster than a TrueType font, but they are limited to the downloaded point size and characters.

Monospaced Font Magnification

Monospaced characters occupy the same amount of space within a magnification. Use monospaced fonts for price fields and data to list in a column. Decide how wide and tall the characters should appear on the labels.

Only the 1x width can be scanned with the OCRA-like font. Using a printhead with 203 dpi, the character widths are as follows: 7.9 (English), 20.1 (Metric), and 16 (Dots).

To calculate other font widths, multiply the font dots (14 dots for Standard, 7 dots for Reduced, 24 dots for Bold) by the magnification and add the default spacing (3 dots for Standard, 1 dot for Reduced, 3 dots for Bold) between characters.

Example

14 (Standard font dots) x 5 (magnification) = 70 + 3 (default spacing between characters). There are 73 dots in the Standard font at 5x.

Proportional Font Magnification

Each character in a proportionally spaced font is a different height and width. More characters may be placed per line. Experiment with these fonts and adjust field measurements as necessary. The bitmapped fonts (either monospaced or proportional) appear jagged when magnified. The magnification range is 1 to 7.

To calculate other font widths, multiply the font dots (3 dots for Minimum, 13 dots for Average, 22 dots for Maximum) by the magnification.

Example

13 (Average font dots) x 5 (magnification) = 65 dots in an average letter of the CG Triumvirate™ Typeface Bold at 5x.

Scalable Font Information

The scalable font is smooth at any point size. There are no jagged edges because the font is created from an equation every time it is used. The field width varies with each letter.

When defining formats using scalable fonts, set the character rotation to 0 (it is not supported). However, field rotation is supported for text or constant text fields using the scalable font. **The scalable font does not print a slashed zero**.

Scalable fonts perform better in constant text fields, because those fields are imaged only once per batch, not once per label as in text fields. The transparent overlay allows closer field placement when using scalable fonts.

The height and width magnification are defined in point size. 72 points = one inch.
One inch = cell size. The cell size is the built-in space around the individual characters of the scalable font. The point size range is **4** to **255**. If the height and width are not set to the same point size, the printed characters look tall and thin or short and thick, which allows for greater flexibility in the appearance of the font.

The 72 point EFF Swiss Bold sample shows the one inch cell size.

6pt   Sample

10pt   Sample

24pt   Sample

48pt Sample

72pt Sample
1"

TrueType Font Information

TrueType fonts follow the TrueType outline font standard. These fonts are smooth at any point size. There are no jagged edges, because the font is created from an equation every time it is used. The height and width magnification are defined in point size.
72 points = one inch. One inch = cell size. The cell size is the built-in space around the individual characters of the scalable font. The point size range is 4 to 255. The field width varies with each letter. The printer accepts downloaded TrueType fonts.

Downloading TrueType Fonts

The MONARCH® MPCL™ Toolbox (Font Utility) is available on our Web site and converts TrueType fonts to Hex or Run-Length encoding for the printer.

When downloading a TrueType font, download the entire font, not particular characters or one point size. A variety of symbol sets can be printed with International (Turkish, Latin, Spanish, etc.) characters. TrueType fonts are designed to be regionally specific; therefore, all symbol sets may not be supported in a given font.

Save the fonts to flash memory for optimal performance with downloaded fonts. No memory configuration packets are necessary since flash memory cannot be reallocated.

Using International Fonts

International fonts are available as bitmap or TrueType fonts. See "Bitmap Font Information" or "TrueType Font Information" for more details. Save the font into the printer's flash memory. Use the symbol set (110 - Unicode UTF-8) with a downloaded International TrueType font.

To use International fonts, consider the following information:

♦ All fonts contain an internal character mapping. The mapping is organized by one or more standards, such as BIG5. These mapping standards can provide over 65,000 characters, which are not represented in this manual. The printer supports several mapping standards: Unicode (UCS-2), BIG5, KSC, GB2312, and SJIS.

♦ Specify a symbol set based on the characters to print and one that is compatible with the font's character mapping. For example, to print Japanese characters, select symbol set 932 (Japanese Shift JIS) and a font compatible with that symbol set.

Enter batch data specified by the font's character mapping and compatible symbol set.

Selecting a Symbol Set

Specify a symbol set based on the characters to print and one that is compatible with the font's character mapping. The symbol set parameter identifies the character mapping used in the text field or constant text field, for example, Unicode, BIG5, etc. If no symbol set is selected, the default symbol set (Internal Symbol Set) is used.

The printer automatically translates some character mappings to others. For example, if you need a BIG5 font, it is possible to use Unicode text data. Use Unicode in the symbol set parameter to indicate the text mapping and select the BIG5 font needed in the font parameter (T8 or C5). The printer automatically translates the Unicode character values into BIG5 values before printing the character.

The following table lists the compatible mappings and symbol sets.

Font Character Mapping		Symbol Set Parameter (T15, C13, or A6)
Batch Data*	True Type Font Character Mapping	Use in text or constant text fields
Unicode	BIG5	102 – Unicode Use this symbol set, because the printer automatically translates the character mappings.
Unicode	SJIS	
Unicode	KSC5601	
Unicode	GB2312	
Unicode	Unicode	
BIG5	Unicode	103 – BIG5
GB2312	Unicode	104 – GB2312
SJIS	SJIS	105 – SJIS (Code Page 932 – Japanese Shift JIS)
GB2312	GB2312	106 – GB2312 (Code Page 936 – Simplified Chinese)
BIG5	BIG5	107 – BIG5 (Code Page 950 – Traditional Chinese)
KSC	KSC	108 – KSC (Code Page 5601 – Korean)
UTF-8	Unicode BIG5 SJIS KSC5601 GB2312	110 – Unicode UTF-8 The printer automatically translates the character mappings. **Note:** UTF-8 is very common. Use it for compatibility with different host applications and data streams.

* Characters in batch data must be entered based on their mapping (Unicode, BIG5, etc.).

Note: Symbol sets 102-110 require a downloaded International TrueType font.

International Font Sample

{F,3,A,R,E,150,200,"SIMPLE" |
T,1,5,V,10,10,0,100,30,30,B,L,0,0,102 | }
{B,3,U,1 |
1,"~125~000~125~002~125~004~125~005" | }

This example prints these four characters with Unicode batch data of ~125~000, ~125~002, ~125~004 and ~125~005.

Symbol Set Parameter

Font Number

紀紵約紅

Arabic Font Sample

{F,3,A,R,E,220,400,"SIMPLE" |
T,1,250,V,40,380,0,1000,24,24,B,E,0,0,110 |
T,2,250,V,90,380,0,1000,24,24,B,E,0,0,110 | }
{B,3,N,1 |
**1,"%5 بوليستر 95% قطب | **
2,"5% polyester 95% cotton" | }

This example prints on the printer with a downloaded Arabic font and the printer's Numeric Format set as Default.

Refer to the Internet for a listing of the characters in each code page. Search on a particular code page, such as "codepage 936" to view the characters in that code page.

5% polyester 95% cotton

قطب 95% بوليستر 5%

Licensing Fonts

Avery Dennison provides tools to create and download TrueType fonts. However, it is your responsibility to purchase and license any fonts you download to the printer. Contact a font supplier for licensing information. Additional fonts that are compatible with the printer can be purchased from:

The Electronic Font Foundry   thefonts.com

DynaComware (Korean, Chinese, and Japanese fonts)   www.dynalab.com

Locating the Font Number in a Font Packet

The font number is the second parameter in the packet. Software is available to create the font data and packet. Call Technical Support for more information.

Font Number

Example   **{W,200,A,M,68 |**
           font data |
           font data | }

Use this number in T8 (**font**) or in C5 (**font**). See "Defining Text Fields" or "Defining Constant Text Fields" in Chapter 2 for more information.

Font Number

Example   **T,1,10,V,30,10,0,200,1,1,B,L,0,0,0 |**
           C,50,30,0,200,1,1,B,L,0,0,"MONARCH",0 |

Font Number

Defines a text and constant text field using the downloaded (#200) font.

SYMBOL SETS/CODE PAGES

This appendix contains a listing of the symbol sets, code pages, and extended character sets the printer supports.

This appendix also includes an ASCII to Hexadecimal Conversion Chart.

Supported Symbol Sets and Code Pages

The printers support these symbol sets and code pages: Internal, ANSI, Bold, OCRA Character Set, DOS Code Page 437 and 850. Additional Code Pages are supported with downloaded TrueType or Unicode (International) fonts.

The printer defaults to the internal symbol set. See "Defining the System Setup Packet" in Chapter 2 to change the symbol set.

Selecting a Symbol Set or Code Page

The CG Triumvirate™ typefaces support only the ANSI and DOS Code Page 437 and 850 Symbol Sets. These fonts print a slashed zero when using the ANSI symbol set.

Internal	Use this symbol set to print international monetary symbols, the trademark (™) symbol, and for formats that may be used on other MPCLII printers.
ANSI	Use this symbol set with proportionally spaced fonts.
DOS CP 437/850	Use this symbol set for extended and international characters with proportionally spaced fonts.

Using Code 128 Function Codes

This table lists the characters for Bar Code 128 function codes. These functions are used with scanners.

Code	Function Code
~201	F1
~202	F2
~203	F3
~204	F4

Entering Extended Characters

When using extended characters in batch data file, type a tilde in front of the three-digit code. For example, to include the character Ä in a text field using the Internal Symbol Set, type:

   1,"~142" |

Using International Character Sets/Code Pages

Symbol sets 100, 101, 852-860, and 1250-1258 may only be used with the scalable font (font#50) or downloaded TrueType fonts. TrueType fonts are designed to be regionally specific; therefore, all symbol sets may not be supported in a given font. For example, to print Hebrew characters, find a font (such as Arial) that supports Hebrew characters; convert, and then download the font to the printer. Make sure the correct symbol set for Hebrew characters is selected.

Font 50 does not print the slashed zero or support Code Pages 101 (Wingdings) and 1256 (Arabic). The Euro symbol at position **~192** is only available in the Standard, Reduced, and Bold fonts.

Code pages 102-107, 932, 936, and 950 contain thousands of characters, which are not represented in this manual. These code pages require the memory expansion option and a downloaded International TrueType font.

The Code Pages (100 and greater) on the following pages were printed using Arial or a similar downloaded TrueType font.

Internal Symbol Set

Note: To determine the character code, add the column number and row number for the character. For example, to produce the ™ character with the Internal character set, press Alt 255 (column 15 + row 240) or use ~255 in the data stream.

ANSI Symbol Set

Bold Symbol Set

OCRA Symbol Set

Code Page 100 (Macintosh)

Code Page 101 (Wingdings)

Code Page 437 (Latin U.S.)

Code Page 850 (Latin 1)

Code Page 852 (Latin 2)

Code Page 855 (Russian)

Code Page 857 (Turkish)

Note: To determine the character code, add the column number and row number for the character.
For example, to produce the ÿ character with the ANSI character set, press Alt 255 (column 15 + row 240)
or use ~255 in the data stream.

Code Page 860 (Portuguese)

Code Page 1250 (Latin 2)

Code Page 1251 (Cyrillic)

Code Page 1252 (Latin 1)

Code Page 1253 (Greek)

Code Page 1254 (Turkish)

Code Page 1255 (Hebrew)

Code Page 1256 (Arabic)

Code Page 1257 (Baltic)

Code Page 1258 (Vietnamese)

Note: To determine the character code, add the column number and row number for the character.
For example, to produce the ü character with the Code Page 1250 character set, press Alt 252 (column 12 + row 240)
or use ~252 in the data stream. To determine the character code, add the column number and row number for the character.

ASCII to Hexadecimal Conversion Chart

Use the chart below to translate the characters printed on the test label. The chart lists ASCII characters and their hexadecimal and decimal equivalents.

Char.	Hex	Decimal	Char.	Hex	Decimal
NUL	00	0	DC2	12	18
SOH	01	1	DC3	13	19
STX	02	2	DC4	14	20
ETX	03	3	NAK	15	21
EOT	04	4	SYN	16	22
ENG	05	5	ETB	17	23
ACK	06	6	CAN	18	24
BEL	07	7	EM	19	25
Backspace	08	8	SUB	1A	26
Tab	09	9	Escape	1B	27
Linefeed	0A	10	File separator	1C	28
Vertical tab	0B	11	Group separator	1D	29
Form feed	0C	12	Record separator	1E	30
Carriage return	0D	13	Unit separator	1F	31
SO	0E	14	Space	20	32
SI	0F	15	!	21	33
DLE	10	16	„	22	34
DC1	11	17	#	23	35

Char.	Hex	Decimal	Char.	Hex	Decimal
$	24	36	;	3B	59
%	25	37	<	3C	60
&	26	38	=	3D	61
,	27	39	>	3E	62
(	28	40	?	3F	63
)	29	41	@	40	64
*	2A	42	A	41	65
+	2B	43	B	42	66
,	2C	44	C	43	67
-	2D	45	D	44	68
.	2E	46	E	45	69
/	2F	47	F	46	70
0	30	48	G	47	71
1	31	49	H	48	72
2	32	50	I	49	73
3	33	51	J	4A	74
4	34	52	K	4B	75
5	35	53	L	4C	76
6	36	54	M	4D	77
7	37	55	N	4E	78
8	38	56	O	4F	79
9	39	57	P	50	80
:	3A	58	Q	51	81

Char.	Hex	Decimal	Char.	Hex	Decimal
R	52	82	i	69	105
S	53	83	j	6A	106
T	54	84	k	6B	107
U	55	85	l	6C	108
V	56	86	m	6D	109
W	57	87	n	6E	110
X	58	88	o	6F	111
Y	59	89	p	70	112
Z	5A	90	q	71	113
[	5B	91	r	72	114
\	5C	92	s	73	115
]	5D	93	t	74	116
^	5E	94	u	75	117
_	5F	95	v	76	118
`	60	96	w	77	119
a	61	97	x	78	120
b	62	98	y	79	121
c	63	99	z	7A	122
d	64	100	{	7B	123
e	65	101	|	7C	124
f	66	102	}	7D	125
g	67	103	~	7E	126
h	68	104	Delete	7F	127

FORMAT DESIGN TOOLS

Use copies of these worksheets and grids to create online configuration packets and design formats:

♦ Online Configuration Worksheet

♦ Supply Layout Grids (English, Metric, Dots)

ONLINE HEADER
- (I) HEADER

SYSTEM SETUP
- (A) HEADER
- POWERUP MODE
- LANGUAGE
- SEPARATOR
- SLASH ZERO
- SYMBOL SET

SUPPLY SETUP
- (B) HEADER
- SUPPLY TYPE
- RIBBON
- FEED MODE
- SUPPLY POSN
- CUT POSN
- SKIP INDEX
- CUT TYPE

PRINT CONTROL
- (C) HEADER
- CONTRAST
- PRINT ADJUST
- MARGIN ADJUST
- SPEED ADJUST
- (O) PRINTHD WIDTH
- (O) BATTERY VOLT
- ROTATE IMAGE

MONETARY FORMATTING
- (D) HEADER
- CURRENCY SYMBOL
- SECONDARY
- DECIMALS

CONTROL CHARACTERS
- (E) HEADER
- START OF HEADER
- PARAMETER SEPARATOR
- CHAR. STRING
- FIELD SEPARATOR
- END OF TRANSMISSION
- DATA ESCAPE
- IMMED. COMMAND
- STATUS REQUEST
- JOB REQUEST

COMMUNICATION SETUP
- (F) HEADER
- BAUD
- WORD LENGTH
- STOP BITS
- PARITY
- FLOW CONTROL

BACKFEED CONTROL
- (G) HEADER
- ACTION
- DISPENSE POSITION
- BACKFEED DISTANCE

BUFFER ALLOCATION
- (M) HEADER
- BUFFER
- DEVICE
- BUFFER SIZE

NETWORK CONSOLE
- (N) HEADER
- NUMBER
- ACTION
- DEVICE
- "NAME"
- (C) COMMAND FIELD
- CON COMDS

TIME/DATE CONFIGURATION
- (T) HEADER
- HOUR
- MINUTES
- DAY
- MONTH
- YEAR

VERIFIER CONFIGURATION
- (V) HEADER
- FORMAT #
- ACTION
- DEVICE
- "NAME"
- (D) DATA FIELD
- VFR COMDS

FONT PACKET
- (W) HEADER
- FONT NUMBER
- ACTION
- DEVICE
- DATA LENGTH
- DATA RECORD

RFID SETUP
- (Y) HEADER
- ENCODE MODE
- SINGULATE
- WRITE POWER
- READ POWER
- TAG SAVER
- FIRST TID POSN
- START ENC POSN
- ENCODE ZONE
- ENCODE POSN
- TAG VERIFY
- TAG VERIFY POSN
- RETRY ENCODE

SUPPLY LAYOUT (Inches)

FEED

Indicates No-Print Area   0.025 inches on each side and top and bottom

SUPPLY LAYOUT (Metric)

Indicates
No-Print   0.6 mm on each side and top and bottom
Area

FEED

203   **300**

SUPPLY LAYOUT (Dots)

203	300
1218	1800
1117	1650
1015	1500
914	1350
812	1200
711	1050
609	900
508	750
406	600
305	450
203	300
102	150
0	0

FEED

203	0	102	203	305	406	508	609	711	830	863
300	0	150	300	450	600	750	900	1050	1227	1275

Indicates No-Print Area

5 dots (for 203) and 7.5 dots (for 300) on each side and top and bottom

GLOSSARY

Batch Data
2,"Monarch" |

Defines the actual information (as fields within **{ }**) printed on the label.

Batch Control
E,0,0,4,2,1,0 |

Defines the print job (as a field).

Batch Header
{B,1,N,1 | }

First line of a batch, immediately following (**{**). Identifies the format and batch quantity.

Batch Packet
{B,1,N,1 |
2,"Monarch" | }

Contains a batch header and the batch data. Enclose within **{ }**.

Bitmapped Fonts

Reside in the printer's memory. If the point size is changed, so has the font. Magnifying these fonts causes jaggedness to occur.

Buffer

Storage area in the printer's memory that holds specific data (images, formats, etc).

EPC

The Electronic Product Code, which is a numbering standard for items, similar to the UPC code for bar coding. The EPC is divided into several sections: Header, Manager Number, Object Class, and Serial Number. One of the memory fields reserved for EPC programming. This memory is separate from the user memory and the amount of EPC memory varies with the tag types.

Field

Can be text, bar codes, lines, boxes, constant, or non-printable text. It is the result of a field definition.

Downloaded Fonts

Reside in the printers RAM and deleted when the printer is turned off.

Field Definition

Any string of parameters that pertain to one field. A field definition begins with a field identifier (such as T, B, D, C, etc.).

    T,1,10,V,250,50,0,1,1,1,B,C,0 |

Field Parameters

Parameters that apply to a field and are separated by commas. (In the above example, **B** is a field element for black print on a white background.)

Flash Memory

Contains information that is SAVED when the printer is turned off. Flash memory needs to be formatted before it can be used.

Format

Layout or design for a printed label.

Format Header

First line of a format, immediately following the start of packet (**{**). A format header must begin with **F**, followed by various header elements.

    F,1,A,R,E,600,400,"Fmt-1" |

Inlay

A type of media that contains a transponder and is converted for use in Monarch® RFID supplies (tags). Inlays can be made with different types of transponders.

Interrogator

The electronics module that programs the RFID inlays through the antenna.

Monospaced Fonts

All characters have the same width and are easy to center justify. (Standard, bold, and reduced are monospaced.)

Non-volatile RAM

Contains information that is SAVED when the printer is turned off.

Option **R,4,6,1,3,1 |**	Any line within a format that applies special formatting to a field. This line begins with **R** and must immediately follow the field it applies to.
Packet **B,1,N,1 |** **2,"Monarch" |**	Any string of characters within (**{ }**).
Pre-image	A way to optimize the printer, because it images the fields while data is collected. After the last field is imaged, the label prints almost immediately.
Proportionally Spaced Fonts	All characters have different widths and are difficult to center justify (CG Triumvirate™ Typefaces).
Read Power	Increase or decreases the strength of the RF field emitted by the printer's antenna to read the RFID inlays.
RFID Data Field	The Monarch® Printer Control Language II (MPCLII) data field containing the information to program into an RFID inlay.
RF Field	Area inside the printer where the RFID inlay is programmed. The RF field area is controlled by the RFID power level and the antenna. **Note:** The printer's antenna is located between the platen roller and supply guide inside a bracket.
RFID Reader	An *optional* device that reads the RFID inlays after they are programmed.
RFID Inlays	Supplies that contain an embedded programmable chip and antenna.
Scalable Fonts	All characters are scalable and smooth at any point size. There are no jagged edges at any point size because the font is created from an equation every time it is used.
TID	The Transponder Identification Number, which contains the chip type, features, and available custom commands supported for tag authentication.
Transponder	The combination of the embedded programmable chip with an antenna on some type of media (film, paper, etc.).
TrueType Fonts	All characters follow the TrueType outline font standard. All characters are scalable and smooth at any point size.
User Memory	One of the memory fields reserved for user programming. This memory is separate from the EPC memory and the amount of programmable user memory varies with the tag types.
Volatile RAM	Contains information that is LOST when the printer is turned off.
Write Power	Increase or decreases the strength of the RF field emitted by the printer's antenna to program the RFID inlays.

AVERY DENNISON | Printer Solutions