Packet Reference Manual

Monarch® Printers
• 9416® XL®

AVERY DENNISON

TC941XPM Rev. AH 3/18

Each product and program carries a respective written warranty, the only warranty on which the customer can rely. Avery Dennison Corp. reserves the right to make changes in the product, the programs, and their availability at any time and without notice. Although Avery Dennison Corp. has made every effort to provide complete and accurate information in this manual, Avery Dennison Corp. shall not be liable for any omissions or inaccuracies. Any update will be incorporated in a later edition of this manual.

Trademarks

Avery Dennison
170 Monarch Lane
Miamisburg, OH 45342

TABLE OF CONTENTS

GETTING STARTED

1

This manual provides the necessary information to design, write and print a Monarch® Printer Control Language II (MPCLII) format on the Monarch® 9416 ®XL ®printer. The printer supports both thermal direct and thermal transfer printing. Review the printer information in the *Quick Reference* or *Equipment Manual*.

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.
- use the Supply Layout Grid and Format Worksheet.
- categorize data into field types and select fonts to use in a format.

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.

About the Printers

The following features are standard on the printer.

Feature	**9416XL**
Scalable Font	Version 2.0 or greater
International Fonts	Version 2.0 or greater
Prints 2D bar codes	PDF417, MaxiCode, Quick Response, and Data Matrix using Version 2.1 or greater
Memory Card	Yes
Flash Memory	2 MB

Before You Begin

1. Connect the printer to the host. Refer to the *Quick Reference* for more information.
2. Load supplies in the printer. Refer to the *Quick Reference* 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.

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:

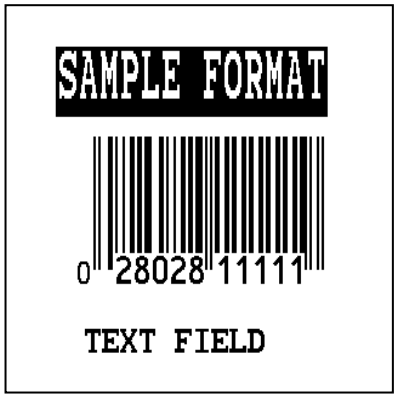
   2,"TEXT FIELD" | }

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

Determine the supply size, fonts, bar codes, and graphics being used. 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 printer. See Appendix B, "Fonts," for samples of each font.
4. Fill out the Format Worksheet. See "Using the Format Worksheet" for more information.

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

5. Create a format packet, based on the completed format worksheet. See Chapter 3, "Defining Fields," for more information.
6. 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.

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 varies, depending on the size of your supply. When designing formats, the following non-print zone is recommended: 0.04 inches at the top and bottom of the label.

Unit of Measure	Minimum Supply (Wid x Len)	Maximum Supply (Wid x Len)	Maximum Print Area (Wid x Len)
English (1/100 inch)	25 x 75	425 x 1000	400 x 1000
Metric (1/10 mm)	63 x 191	1080 x 2540	1016 x 2540
Dots (203)	51 x 152	864 x 2030	812 x 2030
Dots (300)	75 x 225	1275 x 2700	1200 x 2700

The peel mode minimum feed length is 0.75 inches (19 mm).

For exact print area measurements of your supply, see the supply layout grids in Appendix D, "Format Design Tools."

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, which help accurately position information on the label. Decide whether to design formats using English, Metric, or Dot measurements.

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

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.

Using the Format Worksheet

The Format Worksheet is divided into sections that list the field types. Each section has boxes to fill in with parameters that define a format. A format worksheet is included in Appendix D, "Format Design Tools."

Filling in the Format Worksheet

Decide what type of field to use on the label.

1. Make a copy of the Format Worksheet.
2. Define the Format Header. See "Defining the Format Header" in Chapter 3 for more information.
3. Define all non-printable text fields before defining printable ones. See "Defining Non-Printable Text Fields" in Chapter 3 for more information.
4. Define options (copy data, pad data, etc.) as needed. See Chapter 4, "Defining Field Options" for more information.

CONFIGURING THE PRINTER

2

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

Use the following information if you are connecting to the printer's 9-pin serial port. 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`**

Sets the host's communication values to: 9600 baud, no parity, an 8 bit word length, 1 stop bit.

Using Parallel Communications

If your printer supports parallel communications, the parallel port is Centronics® mode. The communication settings are automatically configured for you. There are no operaor settings required.

We recommend waiting at least two seconds (or longer) when switching between the serial and parallel ports to send data, because data may be lost. Be careful when using print spoolers, because data transmission occurs in the background of the operating system. This makes data transmission completion difficult to determine when switching between ports.

Using MPCLII Conventions

Follow these guidelines with MPCLII.

MPCLII Punctuation

Use the following *default* symbols when creating MPCLII packets:

Default 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.

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. Packets A-M 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 - 9 optional records
A, parameter 1...parameter 5 |          System Setup
B, parameter 1...parameter 5 |          Supply Setup
C, parameter 1...parameter 5 |          Print Control
D, parameter 1...parameter 3 |          Monetary Formatting
E, parameter 1...parameter 9 |          Control Characters
F, parameter 1...parameter 5 |          Communication Settings
}                                       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).

Example **{I,0,A,R |**
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,0,0,0,0,1 |
B,2,0,0,0,0 |
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 |
```

The parameters for each packet (**A-F**) are displayed. See each packet's description later in this chapter for more information.

Configuration Syntax Guidelines

When creating a printer configuration packet:

- Follow the Standard Syntax Guidelines listed at the beginning of this chapter.
- 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. Enter **0**:

A3. language — Display Language. Enter **0.**

A4. sep_on — Batch Separators. Enter **0**. The printer does not print batch separators.

A5. slash_zero — Slash Zero. Enter **0**. The printer does not print a zero with a slash through it.

A6. symbol_set — Symbol Set. **0** is the default. Options:

0	Internal	**9**	Code Page 1255 (Hebrew)
1	ANSI	**10**	Code Page 1256 (Arabic)
2	Code Page 437 (Latin U.S.)	**11**	Code Page 1257 (Baltic)
3	Code Page 850 (Latin 1)	**12**	Code Page 1258 (Vietnamese)
4	Code Page 1250 (Latin 2)	**13**	DOS Code Page 852 (Latin 2)
5	Code Page 1251 (Cyrillic)	**14**	DOS Code Page 855 (Russian)
6	Code Page 1252 (Latin 1)	**15**	DOS Code Page 857 (IBM Turkish)
7	Code Page 1253 (Greek)	**16**	DOS Code Page 860 (Portuguese)
8	Code Page 1254 (Turkish)	**19**	Unicode

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 page 19 requires a downloaded International TrueType font (stored on a memory card). 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,0,0,0,0,0 | }`**

Powers up the printer in the online mode, displays prompts in English, does not print a separator after each batch, prints standard zeros (without a slash), 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 | }**

B1. B Supply Setup Packet

B2. supply_type Supply Type. **1** is the default. Options:

0 Black mark
1 Gap/Die Cut
2 Continuous (non-indexed)

Use continuous supply in continuous mode.
Print contrast adjustments may be necessary depending on the supply.
See "Defining the Print Control Packet" for more information.

B3. ribbon Ribbon. The printer automatically senses if a ribbon is installed and switches to thermal transfer mode. **0** is the default. Options:

0 Ribbon not installed (thermal direct)
1 Ribbon installed (thermal transfer)

Note: If "ribbon installed" is sent to the printer, but no ribbon is installed, an error occurs. If "ribbon not installed" is sent to the printer, but a ribbon is installed, no error occurs.

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

0 Continuous operation
1 On-demand mode

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 or 1/300 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.

You cannot change the supply position while the printer is active. Changing the supply position affects the print position. Once the supply position is set, use the print control packet to adjust the printer 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 black marks on the supply. Your supply may need adjustments. Increase to move the cut up, decrease to move the cut down.

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

Indicates black mark and thermal direct supply is 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), and feeds the supply .25 inches (50/203 inches) before cutting.

Defining the Print Control Packet

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

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

C1. C Print Control Packet

C2. contrast Print Contrast. Range: **-390** to **156**. **0** is the default. You may need to adjust this value depending on the type of supplies you are using. To make the print darker, use increments of 13 (for example, 0, 13, 26, 39, 52, etc.). To make the print lighter, use increments of -129 (for example, -129, -258, or -387). You need to use these incremental values to see a difference in the print contrast. For example, values 1 to 13 produce the same result. This is true for values -1 to -130. Solid black print cannot exceed 25% 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: **-99** to **99** 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.

C5. speed_adj Print Speed in inches per second (ips). The only valid settings for 300 dpi are 20 and 30. Options:

0 This is the default – the printer prints at 3.0 ips
20 uses a print speed of 2.0 ips
30 uses a print speed of 3.0 ips
40 uses a print speed of 4.0 ips
50 uses a print speed of 5.0 ips

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

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

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), the printer prints at the default speed (3.0 ips), and uses the default printhead width.

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.

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 when you turn 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.
^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. 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. 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.
^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.
^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. M**41** = 9416
^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)
^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. **Note:** Printer ignores this command if printing.
^RS	Resynchronizes supply when supply roll is changed. **Note:** Printer ignores this command if printing.
^TP	Prints a test label. **Note:** Printer ignores this command if printing.

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. To set parallel communications, see “Using Parallel Communications.”

Changing the communication settings takes approximately two seconds. Communications sent during this interval will be lost. Make sure the host communication values match the values on the printer and the host is 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. **3** is the default. Options:

0	1200	**3**	9600
1	2400	**4**	19200
2	4800	**5**	38400

F3. word_length — Word Length. **1** is the default. Options:

0 7-bit word length
1 8-bit word length

F4. stop_bits — Stop Bits. **0** is the default. Options:

0 1-stop bit
1 2-stop bits

F5. parity — Parity. **0** is the default. Options:

0 None
1 ODD parity
2 EVEN parity

F6. flow_control — Flow Control. **1** is the default. Options:

0	None	**2**	(CTS)
1	DTR	**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.

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. Use R (Volatile RAM).

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

Clears Format #1 from volatile RAM.

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.
W4. device	Device. Options: **F** Flash **M** Memory card (optional) **R** Volatile RAM. **Z** all devices (use for upload)
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|
0,1,437,"Standard",0,0,0,21,33,21,33,5,1|
0,2,0,"Reduced",0,0,0,7,14,7,14,1|
0,2,437,"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|
0,6,0,"HR2",0,0,0,10,16,10,16,1|
0,10,0,"CGTriBd9",1,0,7,25,31,10,15,0|
0,10,1,"CGTriBd9",1,0,7,25,31,10,15,0|
0,10,437,"CGTriBd9",1,0,7,25,31,10,15,0|
0,10,850,"CGTriBd9",1,0,7,25,31,10,15,0|
0,11,0,"CGTriumv6",1,0,5,17,21,5,10,0|
0,11,1,"CGTriumv6",1,0,5,17,21,5,10,0|
0,11,437,"CGTriumv6",1,0,5,17,21,5,10,0|
0,11,850,"CGTriumv6",1,0,5,17,21,5,10,0|
0,15,0,"CGTriumv7",1,0,7,21,28,9,14,0|
0,15,1,"CGTriumv7",1,0,7,21,28,9,14,0|
0,15,437,"CGTriumv7",1,0,7,21,28,9,14,0|
0,15,850,"CGTriumv7",1,0,7,21,28,9,14,0|
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|
0,17,0,"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|
0,18,0,"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|
0,50,0,"EffSwissBold",1,1,92248 | }
```

Font Style/Number
Symbol Set
Font Name
Spacing
Type
Cell Width
Nominal Width
Cell Height
Nominal Height
Inter-Character Gap
Baseline

Note: The CG Triumvirate™ typefaces are trademarks of Monotype Imaging, Inc.

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: **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 | }
```

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 FIELDS

3

This chapter provides a reference for defining

- the format header
- bar code fields
- line and box fields
- text and constant text fields
- non-printable text 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.)
M Memory card (optional)
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. Measure supply from the leading edge of one label to the leading edge of the next label.

English	25 – 1000
Metric	63 – 2540
203 Dots	51 – 2030
300 Dots	75 – 3000

In peel mode, the minimum label length is 0.75 inches (19 mm). For 300 dpi, the maximum label length is 9.0 inches (229 mm).

Make sure the format length matches the actual label size exactly for correct printer performance. This is especially true for shorter feed length supply and formats. If an error occurs, recalibrate the supplies in the printer. Refer to your *Equipment Manual* for more information.

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

English	75 – 400
Metric	191 – 1016
203 Dots	152 – 812
300 Dots	225 – 1200

F8. "name" Format name (optional), **0** to **8** characters, enclose within quotation marks. **""** is the default.

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.

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.

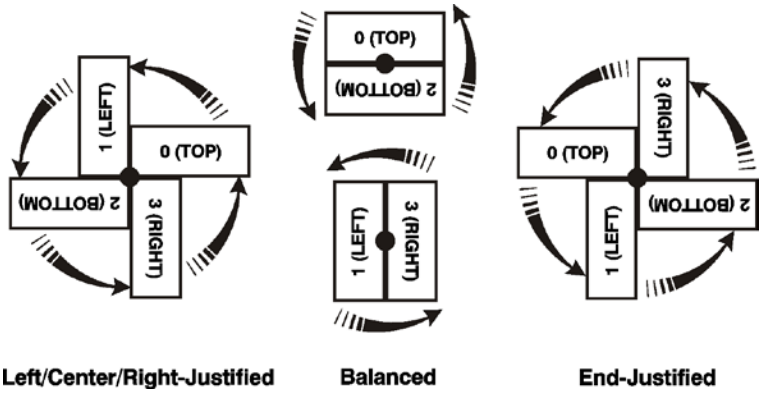
T2. field# — Unique number from **1** to **999** to identify this field. **1** is the default.

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:

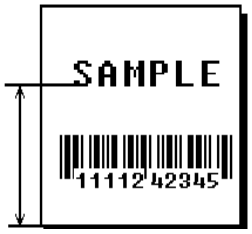
F Fixed length
V Variable length

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.



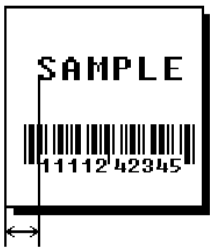
For proportionally spaced fonts, distance from bottom of print area to baseline of characters in field.

English	**0 – 999**
Metric	**0 – 2539**
203 Dots	**0 – 2029**
300 Dots	**0 – 2999**



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 – 399**
Metric	**0 – 1015**
203 Dots	**0 – 811**
300 Dots	**0 – 1199**



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 (scalable)

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."

Note: To use larger point sizes (greater than 60 point), you must reconfigure memory and increase the size of the scalable (vector) fonts buffer.

T11. color

Field color overlay attributes. 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.

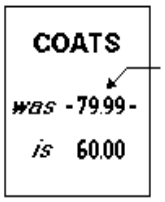
Options for standard printer fonts:

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

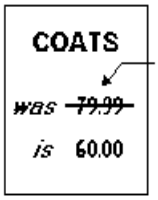
Options for scalable fonts:

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

Note: Solid black print should not exceed 25% on a given square inch of the label, or the printhead life may be decreased.



Line field blocked out by opaque field using attribute B



Line field not blocked out by transparent field using attribute O

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.

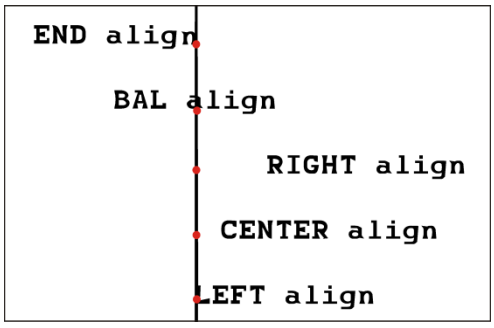
T12. alignment

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.

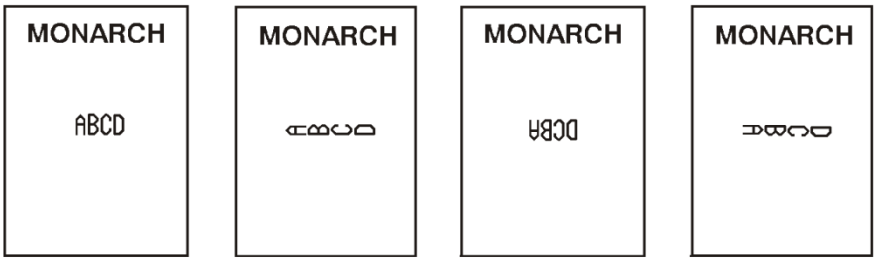
The red dot indicates the field origin and the line indicates the column position for each field in the following graphic.



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

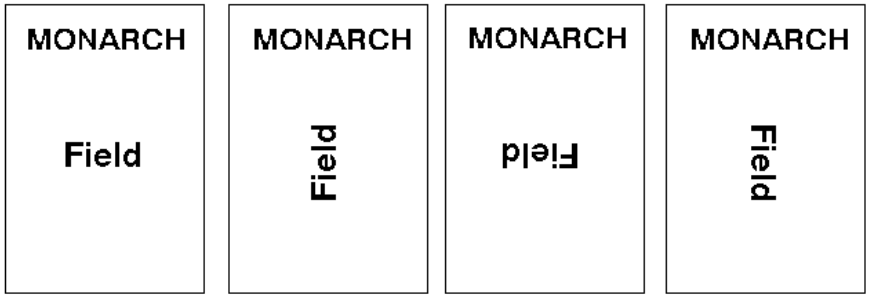


Note: Font 50 and downloaded TrueType fonts do not support character rotation.

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



T15. sym set

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

1 ANSI Symbol Set
102 Unicode (user input) for particular mapping
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. 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 **`T,2,10,V,50,80,0,1,1,1,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 50, 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.

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

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
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
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

Bar Code	Number of Characters	
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
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

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

B5. row

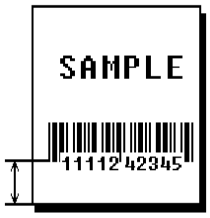
Row location -distance from bottom of the print area to the pivot point of the field. The pivot point varies, depending on how the field is justified. **10** is the default. Remember to include text or numbers that may appear with the bar code for the row measurement. Pivot points

Left/Center/Right-Justified Fields

Balanced Fields

End-Justified Fields

English	**0 – 999**
Metric	**0 – 2539**
203 Dots	**0 – 2029**
300 Dots	**0 – 2999**



B6. column

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

English	**0 – 399**
Metric	**0 – 1015**
203 Dots	**0 – 811**
300 Dots	**0 – 1199**

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

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
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 to FFH

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.

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
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

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

English	**20**
Metric	**51**
203 Dots	**40**
300 Dots	**60**

Height	Bar Code Type	Description
0	Code 16K 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.
20	Code 93	**20** is the default for Code 93.
40	UPCA/UPCE/EAN 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:

Appear.	Bar Code Type	Description
0	MaxiCode QR Code	MaxiCode Mode 0 (obsolete) QR Code Model 2
1	UPC/EAN QR Code	No check digit or number system QR Code Model 1
2	MaxiCode QR Code	MaxiCode Mode 2 (Numeric Postal Code) QR Code Model 2
3	MaxiCode	MaxiCode Mode 3 (Alphanumeric Postal Code)
5	UPC/EAN	Number system at bottom, no check digit
6	UPC/EAN	Check digit at bottom, no number system
7	UPC/EAN	Check digit and number system at bottom (default)
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 Data Matrix, 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.

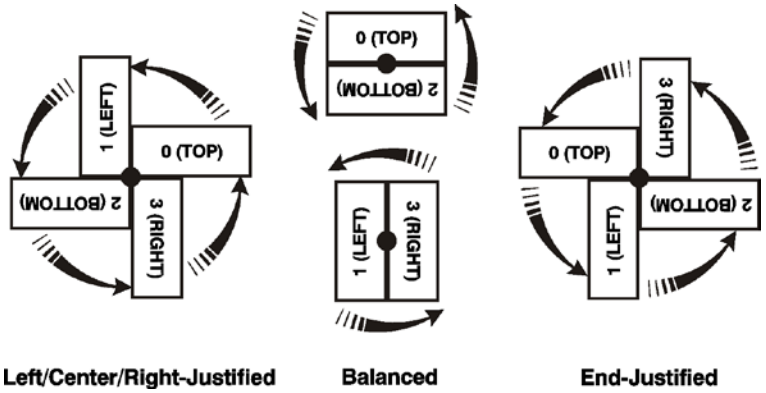
The red dot indicates the field origin and the line indicates the column position for each field in the following graphic.

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 printed at speeds greater than 3.0 IPS may not scan properly.



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 50, 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.

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 |`**

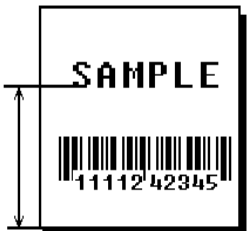
Defines a non-printable text field (field #4) with a maximum of 20 characters.

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 – 999**
Metric	**0 – 2539**
203 Dots	**0 – 2029**
300 Dots	**0 – 2999**

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 – 399**
Metric	**0 – 1015**
203 Dots	**0 – 811**
300 Dots	**0 – 1199**

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 (scalable)

Or a valid downloaded font selector number.
Fonts 5 and 6 are for numeric data only.
Fonts 15 – 18 are only for 300 dpi and do not support Code page 0, 1, 437, and 850. 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.

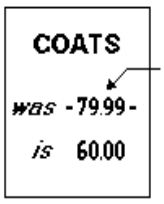
Options for standard printer fonts:

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

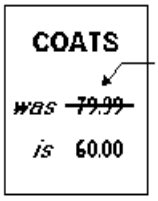
Options for scalable fonts:

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

Note: Solid black print should not exceed 25% on a given square inch of the label, or the printhead life may be decreased.



Line field blocked out by opaque field using attribute B



Line field not blocked out by transparent field using attribute O

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.

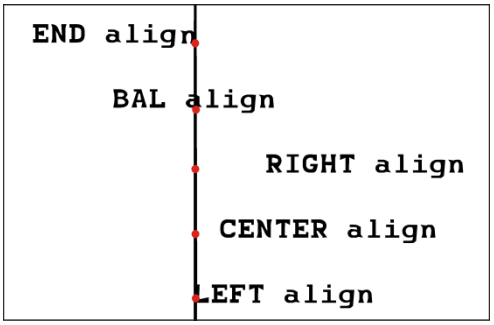
C9. alignment

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.

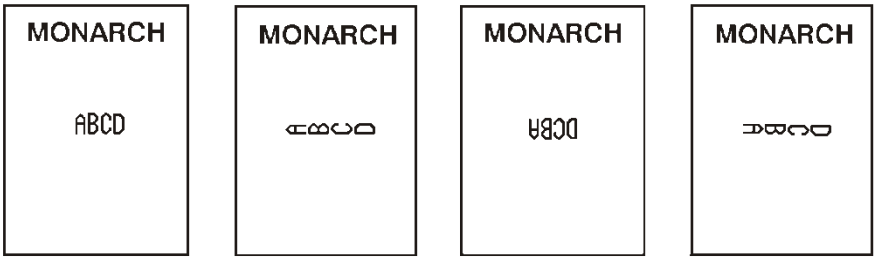
The red dot indicates the field origin and the line indicates the column position for each field in the following graphic.



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

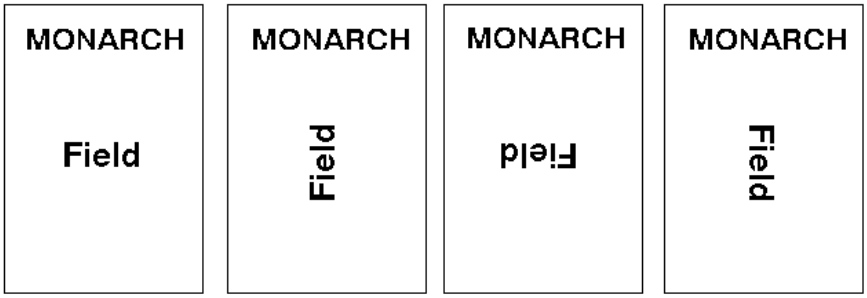


Note: Font 50 and downloaded TrueType fonts do not support character rotation.

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



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
102 Unicode (user input) for particular mapping
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. Symbol Set 102 (Unicode) requires a downloable International TrueType font (stored on a memory card). 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 **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 25 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" |**

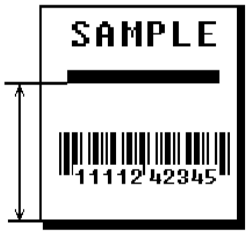
L1. L Line Field.

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.

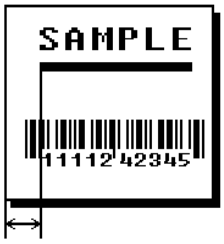
L3. row Row location - distance from bottom of print area to the starting point. **10** is the default.

English	**0 – 999**
Metric	**0 – 2539**
203 Dots	**0 – 2029**
300 Dots	**0 – 2999**



L4. column Column location - distance from left edge of the print area to line origin. **10** is the default.

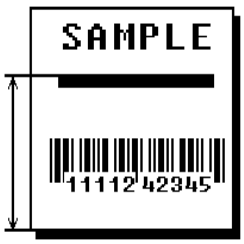
English	**0 – 399**
Metric	**0 – 1015**
203 Dots	**0 – 811**
300 Dots	**0 – 1199**



L5. angle/end row If Using *Segments*:

Row location of ending point. Measure from bottom of print area. The ranges are the same as *row* above. On horizontal lines, this value must match item **L3**. **100** is the default.

If Using *Vectors*:
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. The ranges are the same as *column* above. On vertical lines, this value must match parameter **L4**. **100** is the default.

If Using *Vectors*:
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

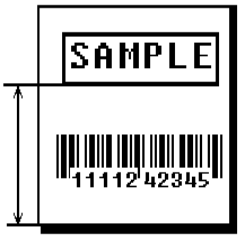
Use boxes to form borders or highlight items of interest. Define each box field separately. This field is not assigned a field number, but is counted as a field (keep this in mind, as the printer allows a maximum of **1000** fields per format). Define any line length and a thickness up to 99 dots, as long as the solid black print does not exceed 25 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.

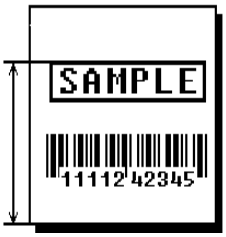
English	**0 – 999**
Metric	**0 – 2539**
203 Dots	**0 – 2029**
300 Dots	**0 – 2999**



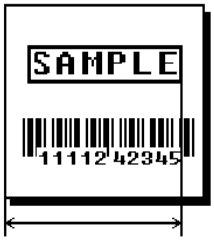
Q3. column Column location - distance from left edge of print area to lower left corner of box. **10** is the default.

English	**0 – 399**
Metric	**0 – 1015**
203 Dots	**0 – 811**
300 Dots	**0 – 1199**

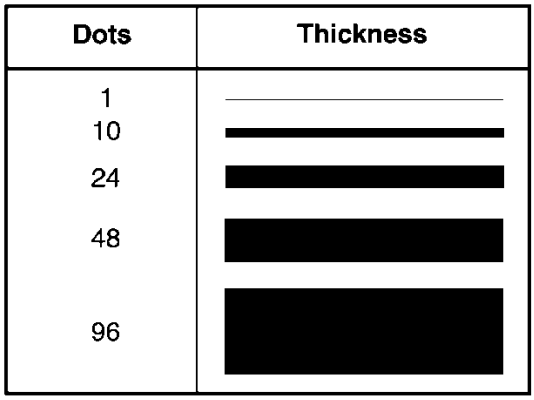
Q4. end row Distance from bottom of print area to upper right corner of box. **100** is the default. The ranges are the same as *row* above.



Q5. end col Distance from left edge of print area to upper right corner of box. **100** is the default. The ranges are the same as *column* above.



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.



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

Example **`Q,40,30,70,150,3,"" |`**

Defines a box field starting at row 40, column 30. It ends at row 70, column 150. It has a thickness of 3 dots.

DEFINING FIELD OPTIONS

4

This chapter provides a reference for defining

- field options in formats
- check digit packets.

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. 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
4	Copy data from previous field
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
60	Define incrementing or decrementing field
61	Reimage fields

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 "Defining Batch Data Fields" in Chapter 6. As an alternative, apply Option 4 to copy data into the non-fixed character positions.

Using Option 4 (Copy Data)

Use option 4 to create a field that uses data from another field. This is useful for creating 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 number of characters that will be placed in the field, including any price, check digit, or fixed characters inserted by the printer. The maximum number of characters in the field into which data is copied cannot exceed **2710** or the maximum number of characters permitted by the **bar code**.

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.
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. **1** Copy field as is (including price symbols, pad characters, check digits, etc.). **2** Copy unformatted data (without price characters, pad characters, etc.).

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.

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.

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 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.

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.

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 “Using Check Digits.” 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 "Defining the Monetary Formatting Packet" for more information. See Appendix C, "Symbol Sets/Code Pages," 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
- Codabar
- I 2of5
- 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 |`**

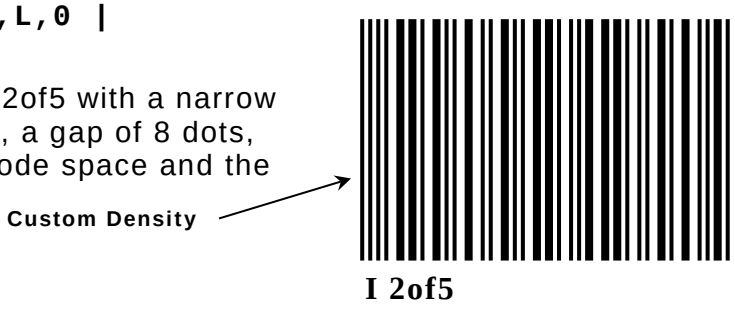
R1. R	Field Option Header.
R2. 50	Option **50**.
R3. narrow	Dot width of the narrow element. Range: **1** to **99**. **1** is the default.
R4. wide	Dot width of the wide element. Range: **1** to **99**. **1** is the default.
R5. gap	Additional dot space between characters. Enter a value of **1** to **99**. (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.

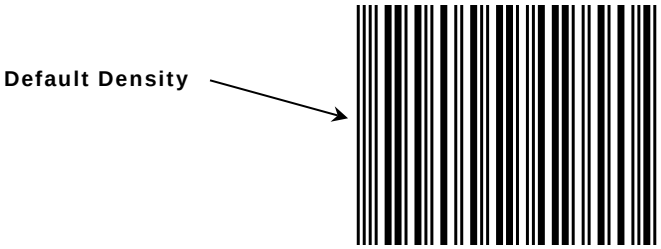
R7. wide_space Additional dot width of the wide bar code space/height of each individual stacked bar code. (Code 39, Codabar, I 2of5, PDF417, MSI only). Range: **1** to **99**. **1** is the default.

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 |**



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 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).

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
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.

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.

The printer does 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 |**

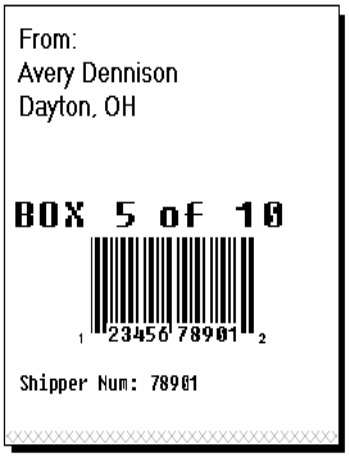
R1. R Option Header.

R2. 61 Option **61**.

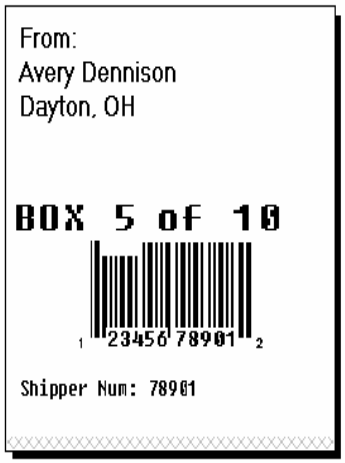
Example **R,61 |**

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 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** (Volatile RAM).
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.

   ```
      4
   10 | 44
        40
        4
   ```

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

5

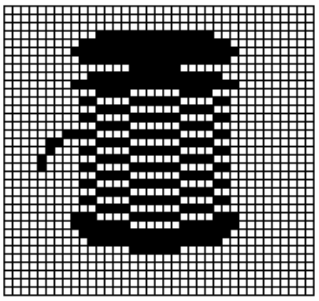
This chapter provides information on how to

- 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

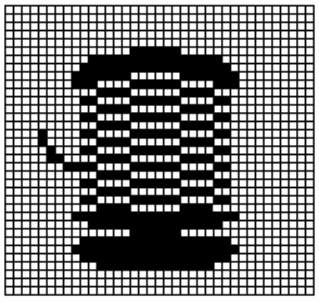
The dot sequences are segmented into black and white strings within a row. The total count of each white string is converted to a lower-case letter, corresponding to numeric value. The total count of each black string is converted to an uppercase letter, corresponding to numeric value. This method can 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.

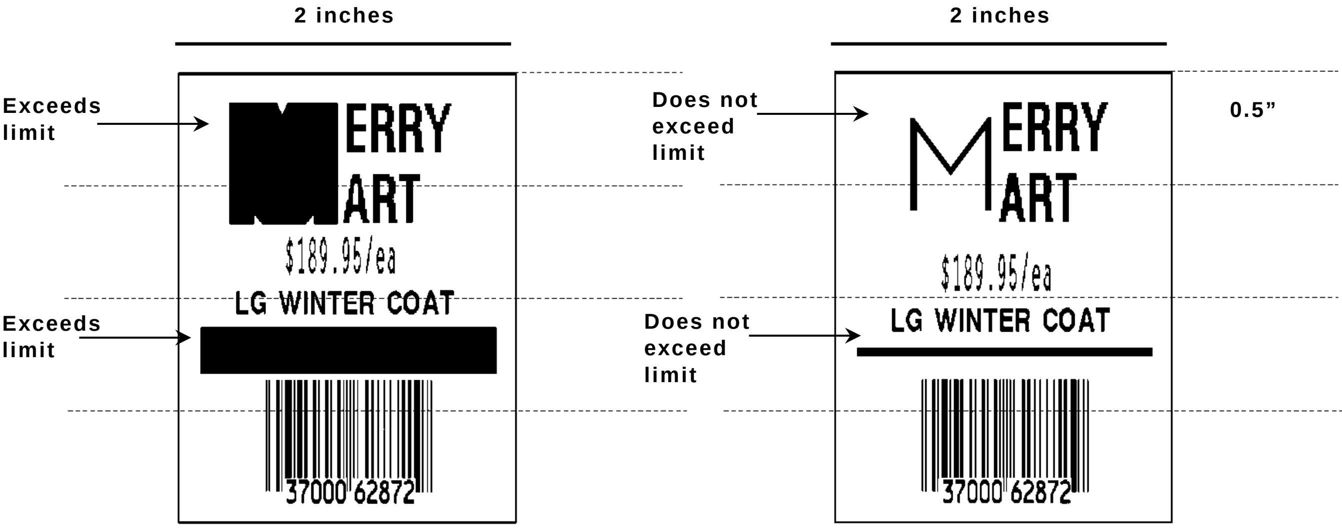
Note: The mapped image must be an upside down mirror image of the final result.



Special Considerations

Solid black print cannot exceed 25% 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.



Using the Hex Method

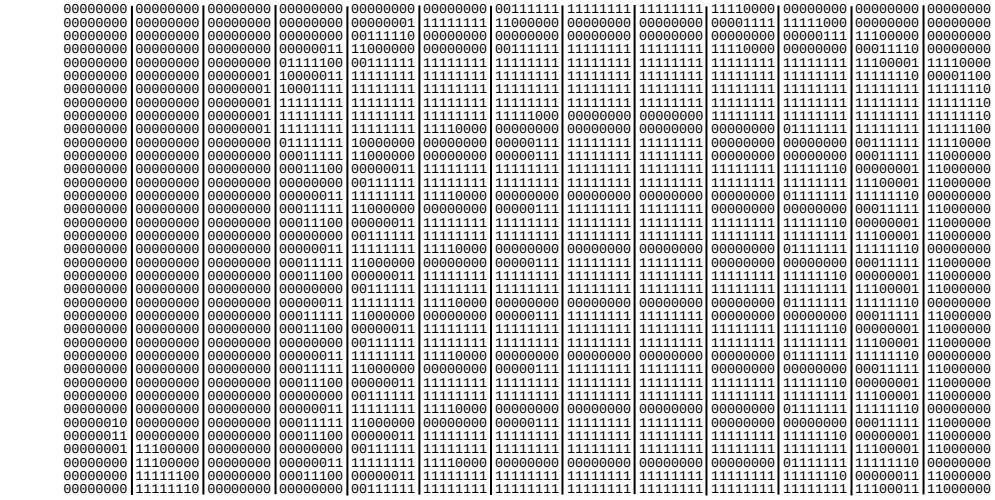
The following steps explain how to derive a hex character string from a bitmapped graphic. Each square on the grid represents a dot. A black square indicates the dot is ON, and a white square indicates the dot is OFF. A sequence of binary numbers, called a bit pattern or bitmap, determines what dots 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.

```
00000000000000000000000000000000000000000000000000111111111111111111111111100000000000000000000000000000
00000000000000000000000000000000000000011111111111000000000000000000000000001111111100000000000000000000
00000000000000000000000000000000001111100000000000000000000000000000000000000000000001111100000000000000
00000000000000000000000000000001110000000000000000111111111111111111111111100000000000000001110000000000
00000000000000000000000001111100001111111111111111111111111111111111111111111111111111111000011111000000
00000000000000000000000110000011111111111111111111111111111111111111111111111111111111111111100000110000
00000000000000000000000110001111111111111111111111111111111111111111111111111111111111111111111111111110
00000000000000000000000111111111111111111111111111111111111111111111111111111111111111111111111111111110
00000000000000000000000111111111111111111111111111111000000000000000000011111111111111111111111111111110
00000000000000000000000111111111111111111111000000000000000000000000000000000000011111111111111111111100
00000000000000000000000001111111100000000000000000000111111111111111111100000000000000000011111111110000
00000000000000000000000000011111100000000000000000000111111111111111111100000000000000000000111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111100000000111000000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111000011100000000
00000000000000000000000000000011111111111110000000000000000000000000000000000000011111111111110000000000
00000000000000000000000000011111100000000000000000000111111111111111111100000000000000000000111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111100000000111000000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111000011100000000
00000000000000000000000000000011111111111110000000000000000000000000000000000000011111111111110000000000
00000000000000000000000000011111100000000000000000000111111111111111111100000000000000000000111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111100000000111000000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111000011100000000
00000000000000000000000000000011111111111110000000000000000000000000000000000000011111111111110000000000
00000000000000000000000000011111100000000000000000000111111111111111111100000000000000000000111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111100000000111000000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111000011100000000
00000000000000000000000000000011111111111110000000000000000000000000000000000000011111111111110000000000
00000000000000000000000000011111100000000000000000000111111111111111111100000000000000000000111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111100000000111000000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111000011100000000
00000000000000000000000000000011111111111110000000000000000000000000000000000000011111111111110000000000
00000010000000000000000000011111100000000000000000000111111111111111111100000000000000000000111111000000
00000011000000000000000000011100000000111111111111111111111111111111111111111111111111100000000111000000
00000000111000000000000000000000001111111111111111111111111111111111111111111111111111111000011100000000
00000000111000000000000000000011111111111110000000000000000000000000000000000000011111111111110000000000
00000000111111000000000000011100000000111111111111111111111111111111111111111111111111000000011110000000
00000000111111000000000000000000001111111111111111111111111111111111111111111111111111110001111000000000
```

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.



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	00000001	11111111	11000000	00000000	00000000	00001111	11111000	00000000	00000000
00000000	00000000	00000000	00000000	00111110	00000000	00000000	00000000	00000000	00000000	00001111	11000000	00000000
00000000	00000000	00000000	00000011	11000000	00000000	00111111	11111111	11111111	11110000	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	11111111	11111000	00000000	00000000	11111111	11111111	11111111	11111110
00000000	00000000	00000001	11111111	11111111	11110000	00000000	00000000	00000000	00000000	01111111	11111111	11111100
00000000	00000000	00000000	01111111	10000000	00000000	00000111	11111111	11111111	00000000	00000000	00111111	11110000
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	00000000	00111111	11111111	11111111	11111111	11111111	11111111	11111111	11100001	11000000
00000000	00000000	00000000	00000011	11111111	11110000	00000000	00000000	00000000	00000000	01111111	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	00000000	00111111	11111111	11111111	11111111	11111111	11111111	11111111	11100001	11000000
00000000	00000000	00000000	00000011	11111111	11110000	00000000	00000000	00000000	00000000	01111111	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	00000000	00111111	11111111	11111111	11111111	11111111	11111111	11111111	11100001	11000000
00000000	00000000	00000000	00000011	11111111	11110000	00000000	00000000	00000000	00000000	01111111	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	00000000	00111111	11111111	11111111	11111111	11111111	11111111	11111111	11100001	11000000
00000000	00000000	00000000	00000011	11111111	11110000	00000000	00000000	00000000	00000000	01111111	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	00000000	00111111	11111111	11111111	11111111	11111111	11111111	11111111	11100001	11000000
00000000	00000000	00000000	00000011	11111111	11110000	00000000	00000000	00000000	00000000	01111111	11111110	00000000
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	11111111	11100001	11000000
00000000	11100000	00000000	00000011	11111111	11110000	00000000	00000000	00000000	00000000	01111111	11111110	00000000
00000000	11111100	00000000	00011100	00000011	11111111	11111111	11111111	11111111	11111111	11111110	00000011	11000000
00000000	11111110	00000000	00000000	00111111	11111111	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 in Appendix C, “Symbol Sets/Code Pages.” Be sure to 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:

- Volatile RAM
- Memory card (optional)
- Temporary Storage
- Flash Memory

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.

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.

Using a Memory Card

The 9416 XL printer has an optional memory card that allows storage of formats, graphics, and fonts. Graphics stored on the memory card are saved when the printer is turned off. Use **M** in the Graphic header. See "Defining the Graphic Header" for more information.

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. **Graphics stored in flash memory are saved when the printer is turned off**.

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

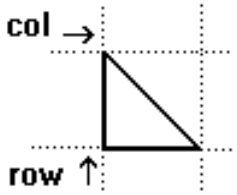
This section explains how to position the graphic image within a graphic packet header, a field of a graphic packet, or within a format.

Within the Graphic Packet Header

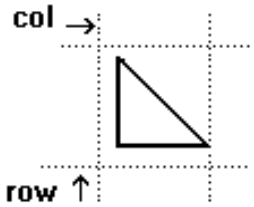
When using RAM, the row and column parameters in the graphic header are usually **0,0**, because placement is controlled by the graphic field in the format. This is especially true when designing a compliance label overlay.

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).



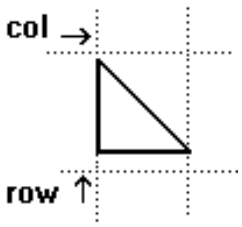
To create a fixed amount of white space around the graphic image, use something other than 0 for row and/or column. The area enclosed within the dotted lines represents the graphic image starting at **0,0** with a fixed amount of white space (**10,10**) around the graphic image.



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 **155, 33** 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)
- **M** Memory card (optional)
- **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 – 999**
Metric	**0 – 2539**
203 Dots	**0 – 2029**
300 Dots	**0 – 2999**

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 – 399**
Metric	**0 – 1015**
203 Dots	**0 – 811**
300 Dots	**0 – 1199**

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 – 999**
Metric	**0 – 2539**
203 Dots	**0 – 2029**
300 Dots	**0 – 2999**

B3. col — Distance (in dots) from the graphic image's left edge to the bitmap line. **10** is the default.

English	**0 – 399**
Metric	**0 – 1015**
203 Dots	**0 – 811**
300 Dots	**0 – 1199**

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. Use duplicate fields 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,"03FFFFFFFFFFFFFFFFFC" |**
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,"3FFFFF0" |
B,40,32,H,"01FFC000000FF8" |
B,41,32,H,"3E00000000000FC0" |
B,42,24,H,"03C0003FFFFF0000F" |
B,43,24,H,"7C3FFFFFFFFFFFE1F0 |
B,44,16,H,"0183FFFFFFFFFFFFFF06" |
B,45,16,H,"018FFFFFFFFFFFFFFFFE" |
B,46,16,H,"01FFFFFFFFFFFFFFFFFE" |
B,47,16,H,"01FFFFFF80001FFFFFFFE" |
B,48,16,H,"01FFFFF0000000007FFFC" |
B,49,24,H,"7F800007FFFF00003FF0" |
B,50,24,H,"1FC00007FFFF00001FC0" |
D,0,4,4 |
B,51,24,H,"1C03FFFFFFFFFFE01C0" |
D,0,4,4 |
B,52,32,H,"3FFFFFFFFFFFFE1C0" |
D,0,4,4 |
B,53,24,H,"03FFF0000000007FFE" |
D,0,4,4 |
B,70,0,H,"0400001FC00007FFFF00001FC0" |
B,71,0,H,"0600001C03FFFFFFFFFFE01C0" |
B,72,0,H,"030000003FFFFFFFFFFFFE1C0" |
B,73,0,H,"01000003FFF0000000007FFE" |
B,74,8,H,"FC001C03FFFFFFFFFFE00C0" |
B,75,8,H,"FE00003FFFFFFFFFFFFE0C0" |
B,76,8,H,"1FF803FFF0000000007FFE" |
B,77,8,H,"0FFFCFFC00000000000001C0" |
B,78,16,H,"FFDF000FFFFFFFF8003C0" |
B,79,16,H,"7FFFC00007FFFF00001FC0" |
B,80,24,H,"1C03FFFFFFFFFFE01C0" |
D,0,4,4 |
B,81,32,H,"3FFFFFFFFFFFFE1C0" |
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,"03FFFFFFFFFFFFF0" |
B,99,24,H,"07FFFFFFFFFFFFFC" |
B,100,24,H,"1FF9FFFFFFFFFFFF" |
B,101,24,H,"3FFE0007FFFF8000FF80" |
B,102,24,H,"391E0027FFFF803FFFC0" |
B,103,24,H,"1C7FFFFFFFFFFFFFC0" |
B,104,24,H,"1FC1FFFFFFFFFF1FC0" |
B,105,24,H,"0FFDFFFFFFFFFE0FF" |
B,106,24,H,"FFFFFFFFFFFFFF8" |
B,107,32,H,"3FFFFFFFFFFFFE0" |
B,108,32,H,"03FFFFFFFFFFF" |
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,"0187FFFFFFFFFFFFFC18" |
B,116,16,H,"027FFFFFFFFFFFFFFFF2" |
B,117,16,H,"03FFFFFFFFFFFFFFFFFC" |
D,0,1,2 |
B,120,16,H,"01FFFFFFFFFFFFFFFFF8" |
B,121,24,H,"FEFFFFFFFFFFFFFFFE0" |
B,122,24,H,"07FFFFFFFFFFFFFFC" |
B,123,32,H,"FFFFFFFFFFFFFC0" |
```

Sample Run Length Graphic Packet

```
{G,99,A,R,G,0,0,0,"99WIRE" |
B,39,50,R,"Z" |
B,40,39,R,"KzI" |
B,41,34,R,"EzsF" |
B,42,30,R,"DpZoD" |
B,43,25,R,"EdZZEdE" |
B,44,23,R,"BeZZMeB" |
B,45,23,R,"BcZZW" |
B,46,23,R,"ZZZA" |
B,47,23,R,"ZDsZE" |
B,48,24,R,"TzkU" |
B,49,25,R,"HtRqJ" |
B,50,27,R,"GsSsG" |
D,0,4,4 |
B,51,27,R,"ChZWgC" |
D,0,4,4 |
B,52,34,R,"ZZEdC" |
D,0,4,4 |
B,53,30,R,"NzkN" |
D,0,4,4 |
B,70,5,R,"AuGsSsG" |
B,71,5,R,"BtChZWgC" |
B,72,6,R,"DxZZEdC" |
B,73,7,R,"CtNzkN" |
B,74,8,R,"FmChZWhC" |
B,75,8,R,"GsZZEdC" |
B,76,11,R,"JiNzkN" |
B,77,12,R,"NbJzzeC" |
B,78,16,R,"JaElZKmD" |
B,79,17,R,"QsSsG" |
B,80,27,R,"ChZWgC" |
D,0,4,4 |
B,81,34,R,"ZZEdC" |
D,0,4,4 |
B,82,30,R,"NzkN" |
D,0,4,4 |
B,83,27,R,"GsSsG" |
D,0,4,4 |
B,98,30,R,"ZZJ" |
B,99,29,R,"ZZM" |
B,100,27,R,"JbZZE" |
B,101,26,R,"MnToI" |
B,102,26,R,"CbHnTiP" |
B,103,27,R,"CcZZC" |
B,104,27,R,"GeZWcG" |
B,105,28,R,"JaZReH" |
B,106,32,R,"ZZI" |
B,107,34,R,"ZZE" |
B,108,38,R,"ZQ" |
B,109,53,R,"T" |
D,0,1,2 |
B,111,48,R,"ZF" |
B,112,33,R,"PzfK" |
B,113,29,R,"CpZBoE" |
B,114,25,R,"DcZZGfB" |
B,115,23,R,"BdZZMeB" |
B,116,22,R,"AbZZVbA" |
B,117,22,R,"ZZZB" |
D,0,1,2 |
B,120,23,R,"ZZZ" |
B,121,25,R,"ZZV" |
B,122,29,R,"ZZM" |
B,123,32,R,"ZZF" |
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.

   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 – 999**
Metric	**0 – 2539**
203 Dots	**0 – 2029**
300 Dots	**0 – 2999**

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 – 399**
Metric	**0 – 1015**
203 Dots	**0 – 811**
300 Dots	**0 – 1199**

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 |
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 | }
```

Reference to the graphic packet

Sample Batch Packet

```
{B,2,N,1 |
1,"Pat's" |
2,"Parts" |                4,"3.55/8 Pack" |
3,"3/8 inch Wire" |        5,"345911871209" | }
```

PRINTING

6

This chapter describes how to

- download files to the printer
- define the batch header, batch control, and batch data files
- create batch files.

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. Configuration packets (A-F)
2. Any of the following:
   - Check digit packets (see Chapter 4)
   - Format packets (see Chapter 3)
   - Graphic packets (see Chapter 5)
3. Batch data (see "Defining the Batch Header")

Downloading Files

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, or other communications 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).

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.

To record batch data, make a copy of the worksheet in Appendix D, "Format Design Tools."

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 **999**). **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.

Syntax `E,feed_mode,batch_sep,print_mult,multi_part,cut_type,cut_mult |`

E1. E — Batch Control Field.

E2. feed_mode — Feed Mode. **0** is the default. Options:

0 Continuous Feed
1 On-Demand

E3. batch_sep — Batch Separator (striped label in between batches). Use **0** (no separator).

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_type — Enables or disables the optional cutter. **0** (does not cut) is the default. Options:

0 Does not cut.
1 Stops to cut each tag.

E7. cut_mult — Number of tags to print before cutting. A cut multiple of one cuts after each tag. The range is 0 to 999. **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.

Example **`E,0,0,0,0,1,0 |`**

Defines a batch control field. Continuous feed mode is used and no separator prints between batches. The print multiple, multi-part supply, and cut multiple are set to 0 (not used). The cutter cuts after every tag. The printer stops while the cutter is active.

Example **`{B,1,N,3 |`**

Defines a batch header that prints three tags. The batch control field uses continuous feed mode and no separator prints between batches. The print multiple and multi-part supply are not used. The cutter cuts after the strip of three tags, producing one set of three tags (print quantity = 3; cut multiple = 3).

Example
```
{B,1,N,3 |
E,0,0,4,2,1,0 |
```

Defines a batch header that prints 12 tags. The batch control field uses continuous feed mode and no separator prints between batches. The print multiple is four and batch quantity is three, so 12 tags are printed. There are two identical parts on each tag. The cutter cuts after each tag.

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.
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.

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

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**.

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) 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 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 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 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 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" | }
```

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
```

DIAGNOSTICS

7

This chapter explains how to

- print diagnostics labels
- reset the printer
- call Technical Support.

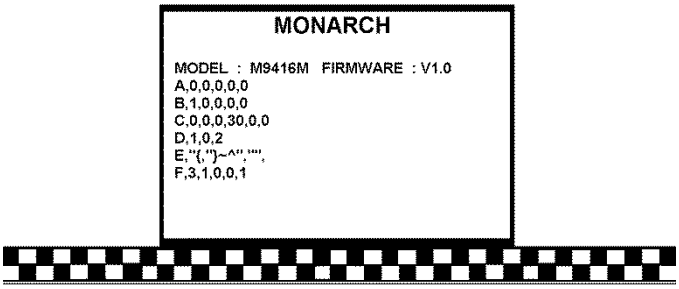
Before calling Service, print a test label. The label contains information to help diagnose mechanical and setup problems.

Printing a Test Label

1. Turn off the printer.
2. Press and hold the **FEED** button while turning on the printer.
3. Release the **FEED** button when the LED is flashing orange and the printer starts advancing lines.

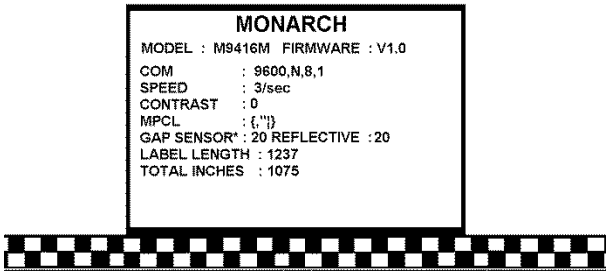
MPCL Label

Contains the printer's MPCL packet configuration. See Chapter 2, "Configuring the Printer" for more information.



Printer Information

Contains generic information, including speed, contrast, and inch counts.



Using Dump Mode

After printing test labels, the printer is in “data dum” mode. Dump mode requires 4.0 inch (101.6mm) wide supply.

To use data dump mode:

1. Turn off the printer.
2. Press and hold the **FEED** button while turning on the printer.
3. Release the **FEED** button when the printer starts advancing lines.

   The test labels print.

   Your printer is now in data dump mode.
4. Download the data stream you want to ‘dump’ (print out on a label).
5. The data stream prints automatically when there is enough data to fill a label. Press the **FEED** button to print any remaining data on another label.

```
,"06/11/94" |}
0 10 D AD 1111" |
o2,"G111111" p 03,"Smith" | , “12345678" p 05
3,"CHECKER ID: " | }0{B,1,N,1 |01,”G111111
D D 200,100,0,1,2,1,B,L,0,3 | OC,330,20,0,1,1,1,B,L,0,D
0,100,0,1,1,1,B,L,0,3,"DATE RECEIVED:" | OT,5,8,V,D
“VIN #:” | OT,4,17,V,260,130,0,1,2,B,L,0,3 | OC,33D
D,160,0,1,2,1,B,L,0,3 | OC,330,130,0,1,1,1,B,L,0,3,D
C,330,160,0,1,1,1,B,L,0,3,"OWNER:" | OT,3,15,F,260
D,3,3,2,B,L,0,3 | 08,2,9,V,330,220,4,11,080,8,L,3 |
0D D {F,1,A,R,E,400,400,"WINDSHIE" | OT,1,9,V,310,340,0D
```

6. Download another data stream to dump or press the **FEED** button to exit data dump mode.

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-F) 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.

PRINTER OPTIMIZATION 8

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.

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 25% 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.

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.

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

A

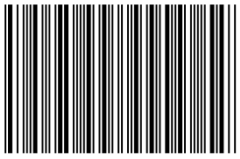
This appendix contains sample formats. Customize any of these formats.

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 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

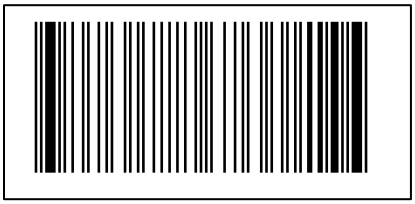
Code 39 is a linear one-dimensional bar code.

```
{F,1,A,R,E,300,150,"1LAB1530" |
C,100,90,0,50,10,10,A,L,0,1,"BATTERY PACK",1 |
C,20,130,0,50,10,10,A,L,0,1,"1452-99311",1 |
C,230,128,0,50,12,10,A,L,0,1,"$5.99",1 |
B,1,9,F,75,55,4,7,40,8,L,1 |
R,5,N |
R,1,"031535512" |
T,2,9,V,125,67,0,50,8,8,A,L,0,1,1 |
R,4,1,1,9,1,1 | }
{B,1,N,1|}
```

Sample Code 93 Packet

Code 93 is a linear one-dimensional bar code that provides higher density than Code 39.

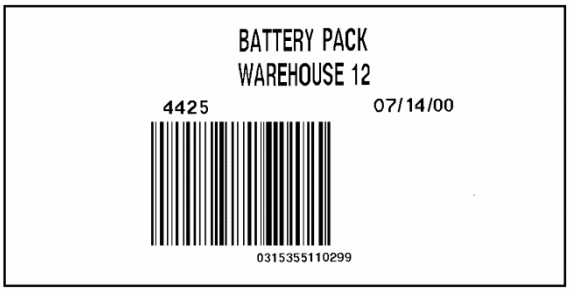
```
{F,101,A,F,E,600,400,"EAN"|
B,2,12,V,225,50,23,7,85,8,L,0|
  R,1,"1234567890" | }
{B,101,N,1|}
```



Sample Code 128 Packet

Code 128 is a high density linear bar code for alphanumeric or numeric data.

```
{F,10,A,R,E,400,200,"1LAB2040" |
C,150,21,0,50,14,12,A,L,0,1,"BATTERY PACK",1 |
C,150,46,0,50,14,12,A,L,0,1,"WAREHOUSE 12",1 |
C,285,70,0,50,10,10,A,L,0,1,"07/14/00",1 |
C,110,70,0,50,10,10,A,L,0,1,"4425",1 |
B,1,13,F,95,165,8,6,90,8,L,1 |
R,5,N |
R,1,"0315355110299" |
T,2,13,V,214,176,0,50,7,9,A,L,0,1,1 |
R,4,1,1,13,1,1 |}
{B,10,N,3|}
```



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,"~~1100123456789O2" | }
```

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 I 2 of 5 with Barrier Bar Packet

Interleaved Two of Five is a linear bar code for numeric data.


```
{F,45,A,R,E,200,400,"SHIPPER" |
C,190,8,0,2,1,1,B,L,0,0,"SHIPPING CONTAINER CODE",0 |
B,1,14,V,17,60,50,5,130,8,L,0 |
T,2,30,V,161,080,0,3,1,1,B,L,0,0,0 |}
{B,45,N,1|
1,"10028028662854" |
2,"1 00 28028 66285 4" |}
```

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.

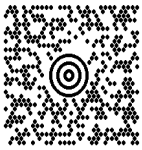
Note: MaxiCode does not support the NULL character.

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.

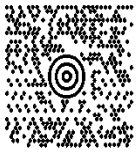
Mode 0 (Obsolete) Sample

{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



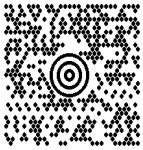
Mode 2 Sample

{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

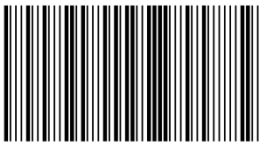
{F,1,A,R,E,200,200,"MAXI_M3" |	
B,1,93,V,020,020,33,7,0,8,L,0 | **}**	MaxiCode bar code (33)
{B,1,N,1 |	
1,"[)~030" |	Message header
C,"01~02996" |	Transportation header
C,"M5E1G45~029" |	Postal Code (This field determines Mode)
C,"124~029" |	Country code
C,"066~029" |	Class of service
C,"1Z12345679~029" |	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



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

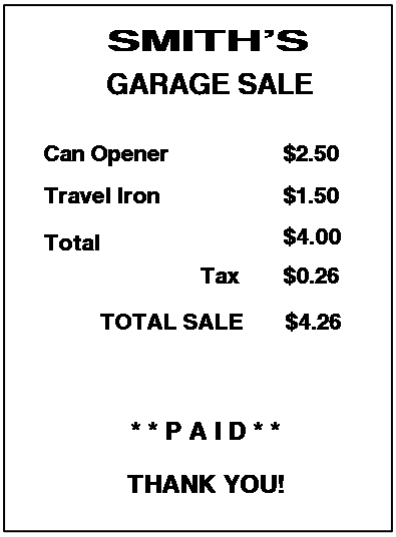
```
{F,1,A,R,E,200,150,"1LAB1520" |
T,1,20,V,44,40,0,50,9,9,A,L,0,0,1 |
B,2,12,F,125,25,1,2,50,7,L,0 |
R,1,"028400067362" |
T,3,20,V,20,34,0,50,8,8,A,L,0,0,1 |
C,84,45,0,50,14,14,A,L,0,0,"$1.19",1 | }
```

Sample Batch Packet

```
{B,1,N,1 |
1,"PEANUTS"|
3,"*SALT FREE*"|}
```

Sample Receipt Format

```
{F,1,A,R,E,300,175,"1Garage" |
C,277,35,0,50,10,18,A,L,0,0,"SMITH'S",1 |
C,223,4,0,50,8,8,A,L,0,0,"Can Opener",1 |
C,202,4,0,50,8,8,A,L,0,0,"Travel Iron",1 |
C,179,4,0,50,8,8,A,L,0,0,"Total",1 |
C,163,81,0,50,8,8,A,L,0,0,"Tax",1 |
C,140,32,0,50,8,8,A,L,0,0,"TOTAL SALE",1 |
C,86,47,0,50,9,9,A,L,0,0,"* * P A I D * *",1 |
C,60,45,0,50,9,9,A,L,0,0,"THANK YOU!",1 |
C,256,35,0,50,10,10,A,L,0,0,"GARAGE SALE",1 |
C,223,122,0,50,8,8,A,L,0,0,"$2.50",1 |
C,202,122,0,50,8,8,A,L,0,0,"$1.50",1 |
C,182,122,0,50,8,8,A,L,0,0,"$4.00",1 |
C,163,122,0,50,8,8,A,L,0,0,"$0.26",1 |
C,140,123,0,50,8,8,A,L,0,0,"$4.26",1 |}
{B,1,N,1 | }
```



Sample Label

```
{F,1,A,R,E,110,200,"1LAB2011" |
C,92,70,0,50,7,7,A,L,0,0,"PRETZELS",1 |
B,1,12,F,45,50,1,2,40,7,L,0 |
R,1,"028400067362" |
C,18,105,0,50,10,10,A,L,0,0,"$.79",1 |}
{B,1,N,1 |}
```

FONTS

B

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

***** 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. Fonts **15** through **18** are only for 300 dpi and do not support Code pages 0, 1, 437, and 850.

These samples were printed using the Internal Symbol set.

Standard Font

Reduced Font

Bold Font

OCRA-like Font

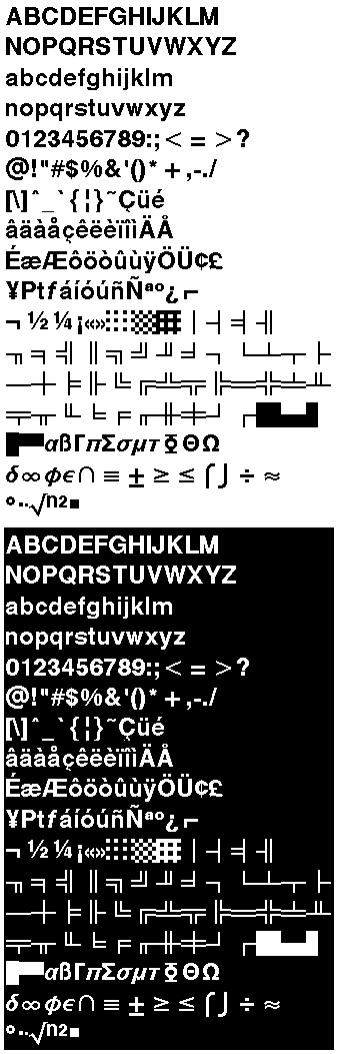
EFF Swiss Bold Font*

ABCDEFGHIJKLM
NOPQRSTUVWXYZ
abcdefghijklm
nopqrstuvwxyz
0123456789:;<=>?
@!"#$%&'()*+,-./
[\]^_`{|}~€ ,
ƒ„…†‡ˆ‰Š‹Œ Ž
‘’“”•—˜™š›œ
žŸ ¡¢£¤¥¦§¨©
ª«¬-®¯°±²³´µ¶
·¸¹º»¼½¾¿ÀÁÂÃ
ÄÅÆÇÈÉÊËÌÍÎÐ
ÑÒÓÔÕÖ×ØÙÚÛÜÝ
Þßàáâãäåæçèéê
ëìíîðñòóôõö÷
øùúûüýþÿ

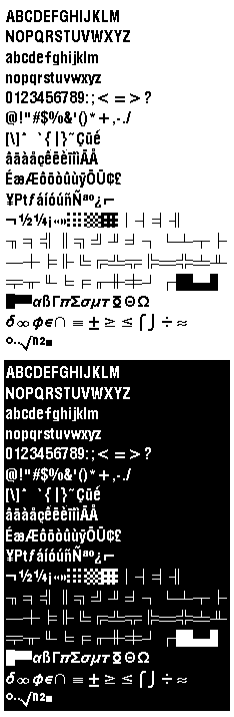
* Printed with ANSI Symbol Set

These samples were printed using Code Page 437.

CG Triumvirate™ Typeface Bold



CG Triumvirate™ Typeface



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

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.

International fonts are available as bitmap or TrueType fonts. See "Bitmap Font Information" or "TrueType Font Information" for more details. Purchase the optional memory card and download the font to the memory card.

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	

***** Characters in batch data must be entered based on their mapping (Unicode, BIG5, etc.).

Note: Symbol set 102 requires 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" | }
```

Symbol Set Parameter

Font Number

This example prints these four characters with Unicode batch data of ~125~000, ~125~002, ~125~004 and ~125~005.

紀紂約紅

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.

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

C

This appendix contains a listing of the symbol sets, code pages, and extended character sets the printer supports.

Use the charts in this appendix to convert dot sequences from the image dot pattern to codes usable in the fields. Use the Binary to Hex Conversion Chart to convert Binary dot sequences to Hexadecimal numbers for bitmap files. Use the Dot to Run Length Encoding Chart to convert dot sequences to alphabetic characters for bitmap files.

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. Fonts 15 through 18 are only for 300 dpi and do not support Code pages 0, 1, 437, and 850.

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

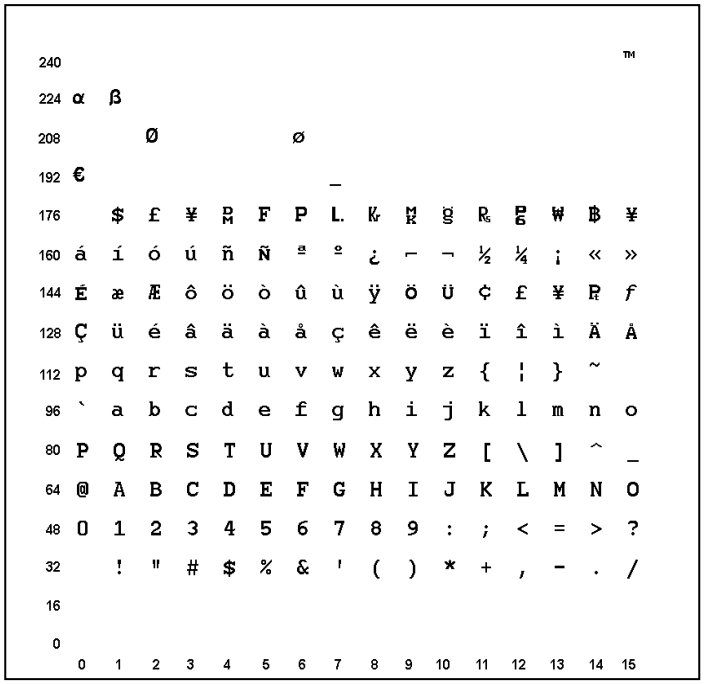
Symbol sets 852-860, and 1250-1258 may only be used with the scalable font (font#50) or downloaded TrueType fonts. TrueType fonts are designed to be regionally specific; therefore, all symbol sets may not be supported in a given font. For example, to print Hebrew characters, find a font (such as Arial) 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 Page 1256 (Arabic). The Euro symbol at position **~192** is only available in the Standard, Reduced, and Bold fonts.

Code pages 102 contains thousands of characters, which are not represented in this manual. These code pages require the memory expansion option and a downloaded International TrueType font.

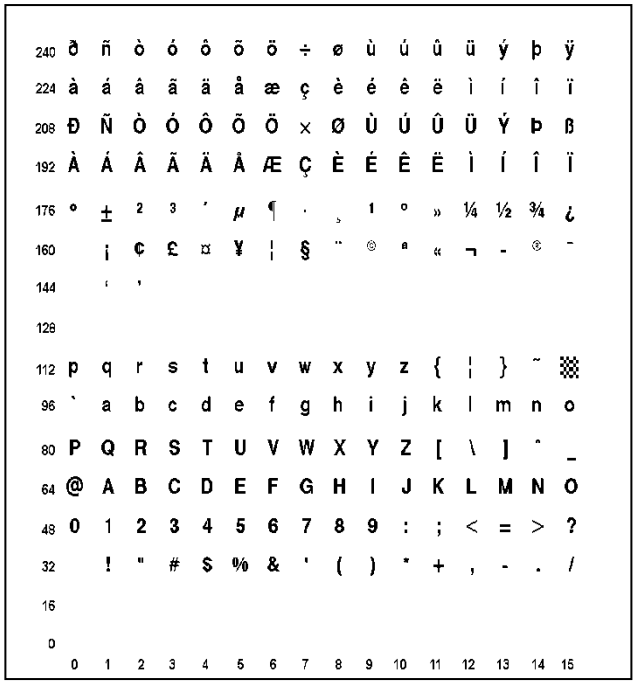
The Code Pages (437 and greater) on the following pages were printed using Arial or a similar downloaded TrueType font.

Internal Symbol Set



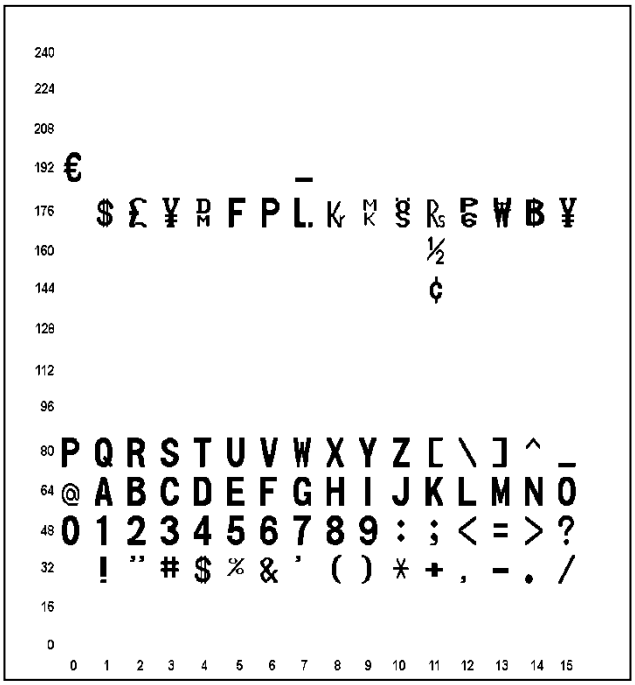
Note: To determine the character code, add the column number and row number for the character. For example, to produce the ™ character with the Internal character set, press Alt 255 (column 15 + row 240) or use ~255 in the data stream.

ANSI Symbol Set

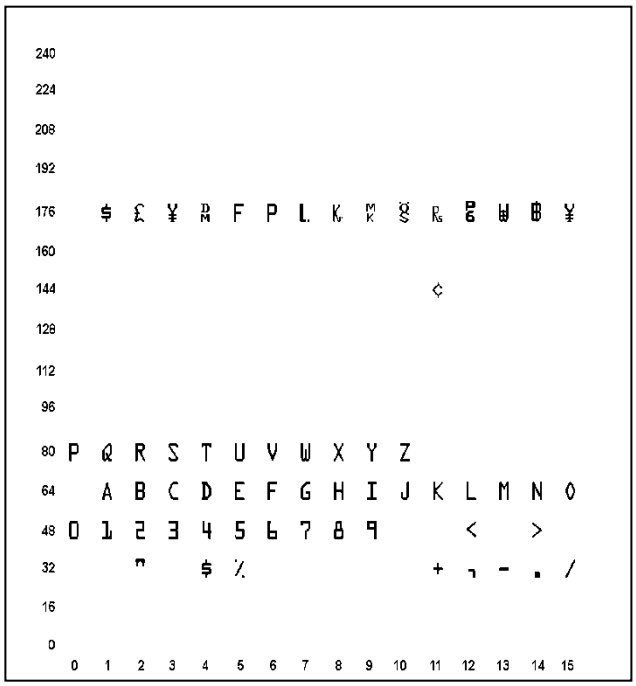


Note: To determine the character code, add the column number and row number for the character. For example, to produce the ÿ character with the ANSI character set, press Alt 255 (column 15 + row 240) or use ~255 in the data stream.

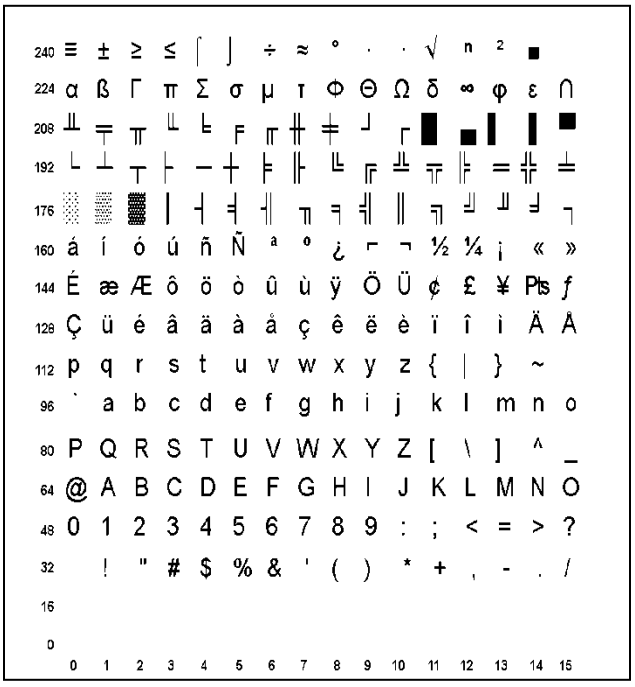
Bold Character Set



OCRA Character Set

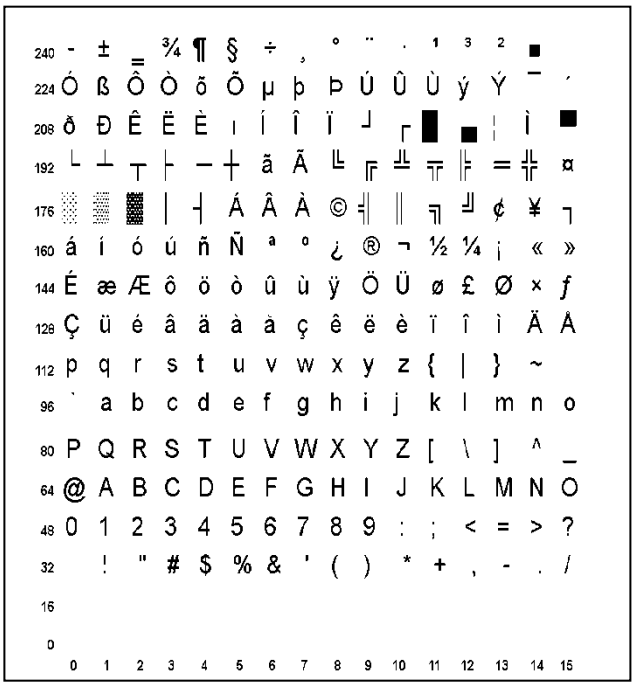


Code Page 437 (Latin U.S.)

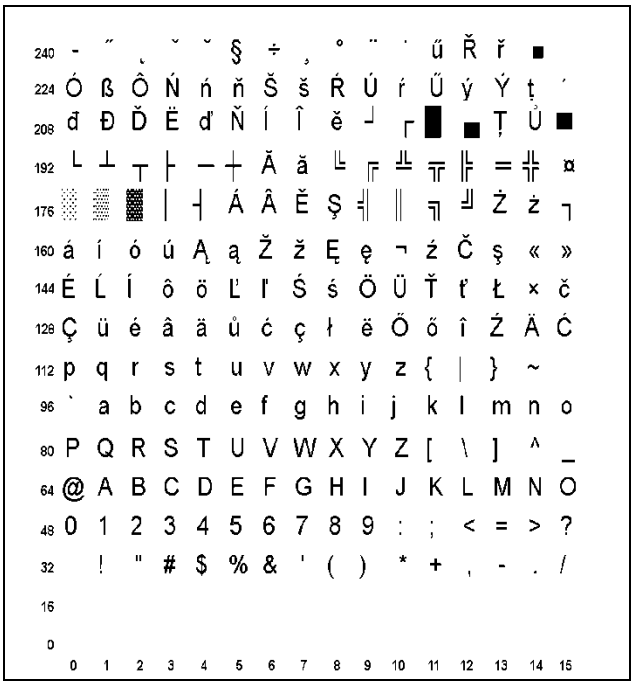


Note: To determine the character code, add the column number and row number for the character. For example, to produce the û character with the Code Page 437 character set, press Alt 150 (column 6 + row 144) or use ~150 in the data stream.

Code Page 850 (Latin 1)

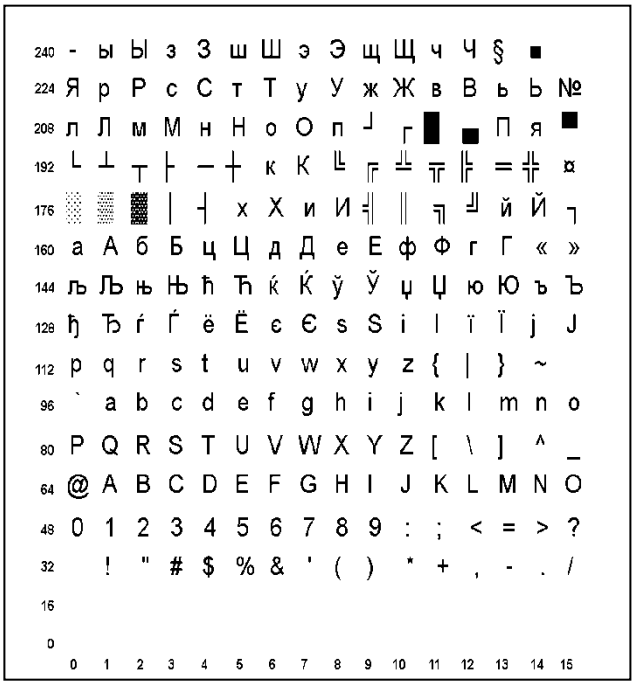


Code Page 852 (Latin 2)

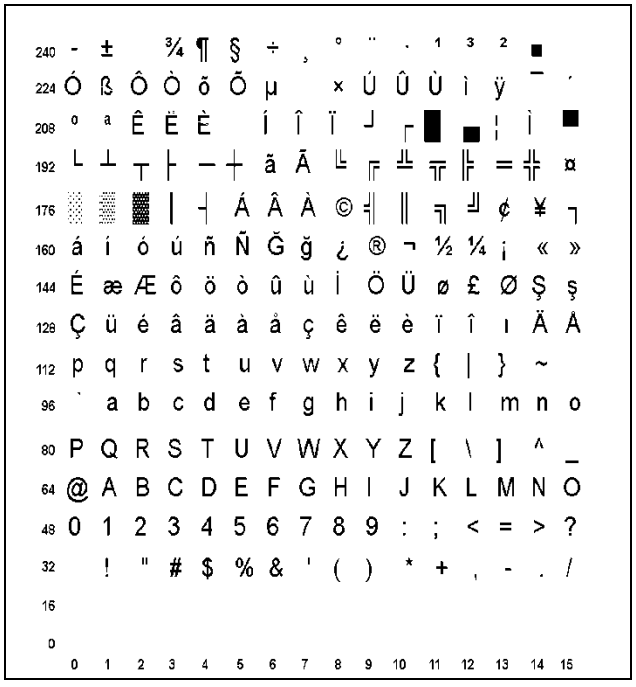


Note: To determine the character code, add the column number and row number for the character. For example, to produce the ľ character with the Code Page 852 character set, press Alt 150 (column 6 + row 144) or use ~150 in the data stream.

Code Page 855 (Russian)

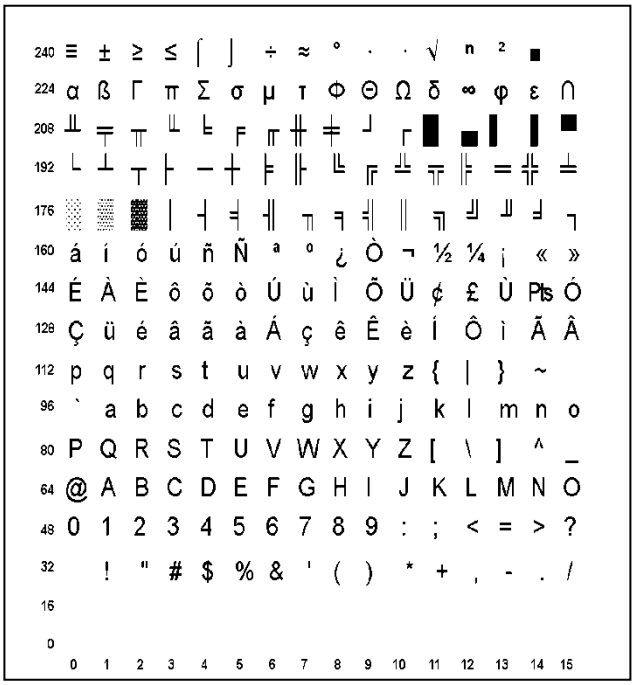


Code Page 857 (IBM Turkish)

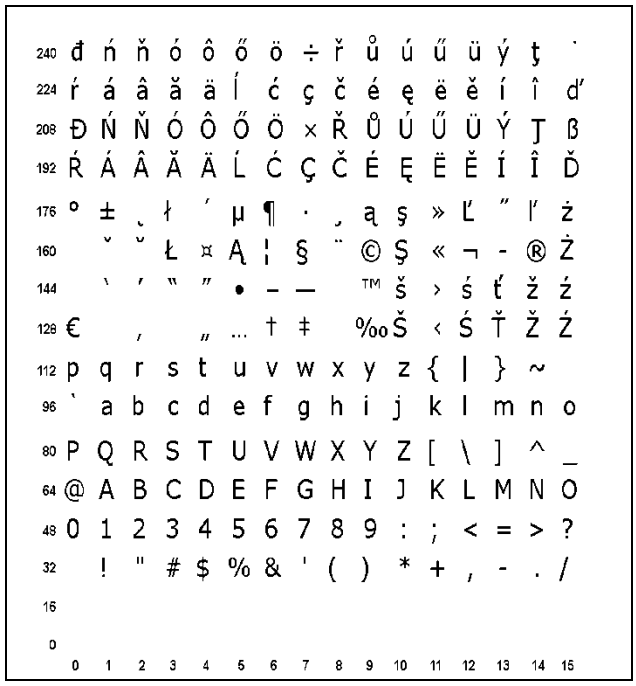


Note: To determine the character code, add the column number and row number for the character. For example, to produce the û character with the Code Page 857 character set, press Alt 150 (column 6 + row 144) or use ~150 in the data stream.

Code Page 860 (MS-DOS Portuguese)

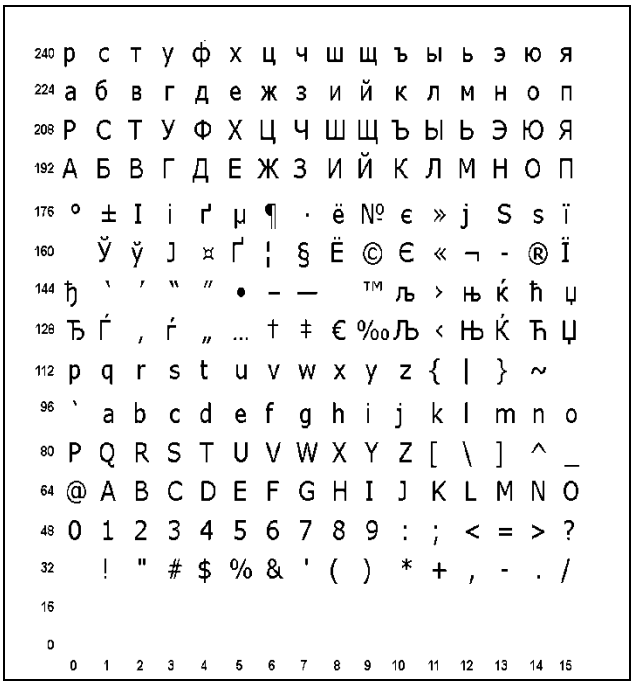


Code Page 1250 (Latin 2)

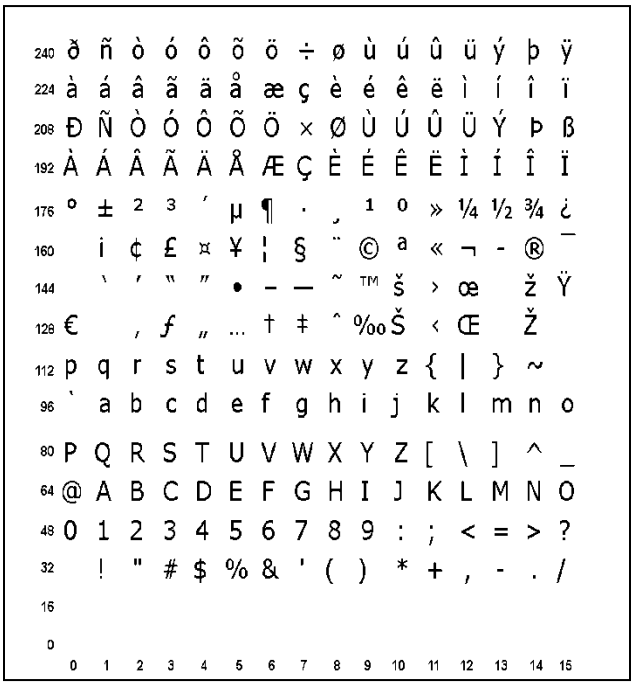


Note: To determine the character code, add the column number and row number for the character. For example, to produce the ü character with the Code Page 1250 character set, press Alt 252 (column 12 + row 240) or use ~252 in the data stream.

Code Page 1251 (Cyrillic)

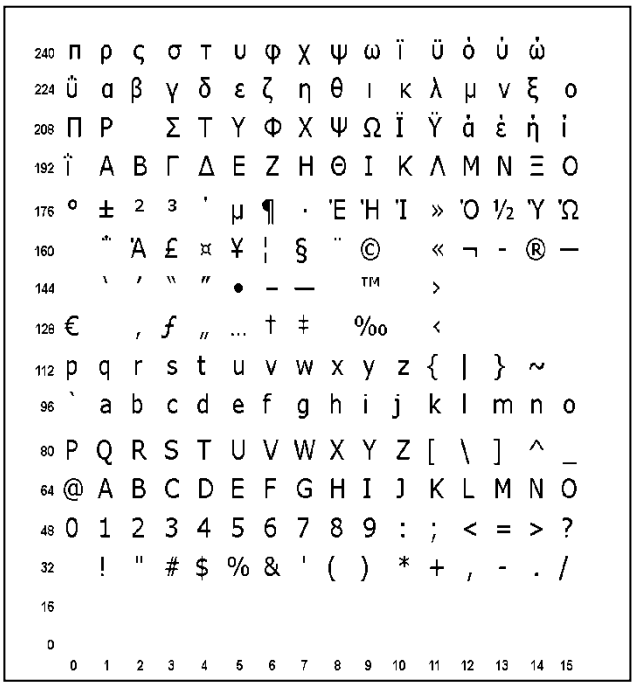


Code Page 1252 (Latin 1)

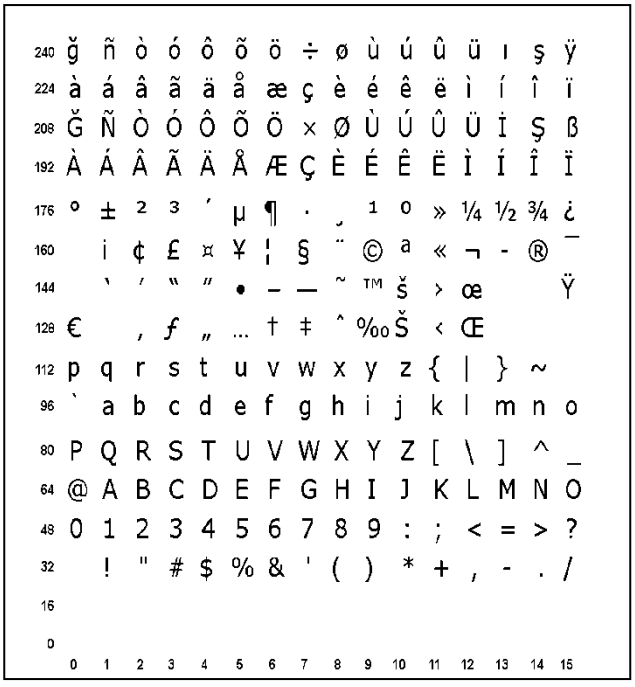


Note: To determine the character code, add the column number and row number for the character. For example, to produce the ü character with the Code Page 1252 character set, press Alt 252 (column 12 + row 240) or use ~252 in the data stream.

Code Page 1253 (Greek)

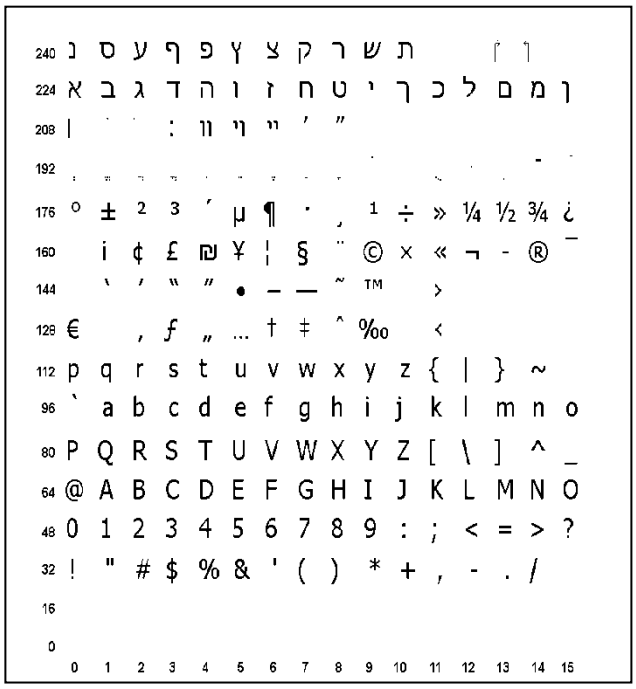


Code Page 1254 (Turkish)

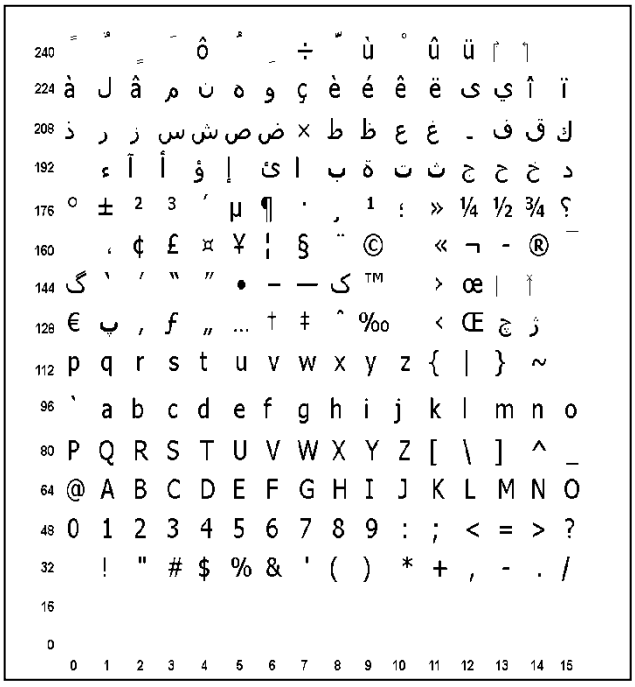


Note: To determine the character code, add the column number and row number for the character. For example, to produce the ü character with the Code Page 1254 character set, would press Alt 252 (column 12 + row 240) or use ~252 in the data stream.

Code Page 1255 (Hebrew)

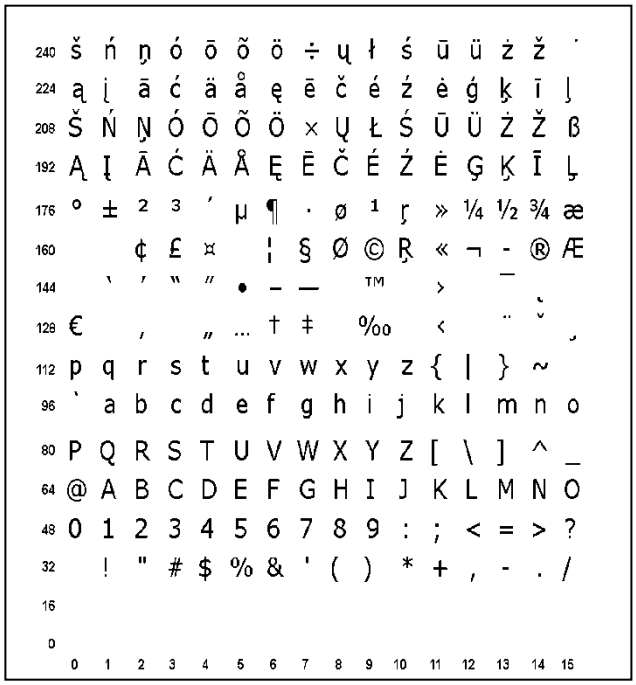


Code Page 1256 (Arabic)

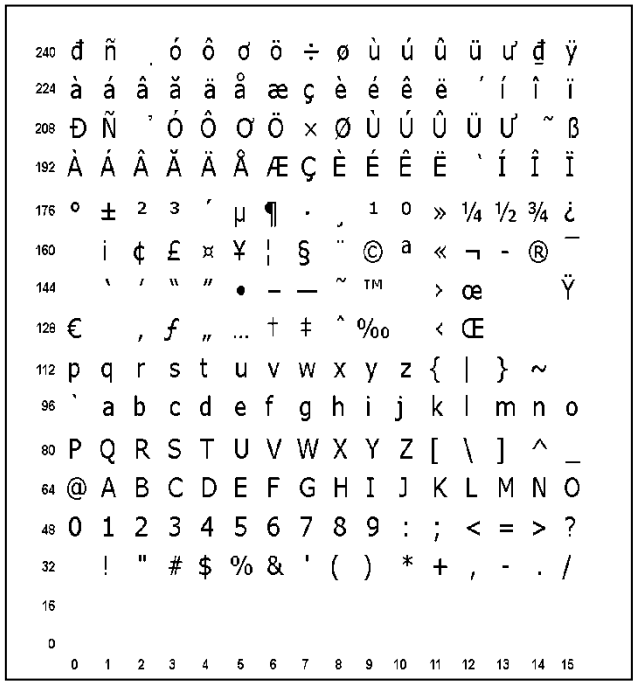


Note: To determine the character code, add the column number and row number for the character. For example, to produce the ü character with the Code Page 1256 character set, press Alt 252 (column 12 + row 240) or use ~252 in the data stream.

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 1258 character set, press Alt 252 (column 12 + row 240) or use ~252 in the data stream.

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

Binary to Hex Conversion Chart

Binary	Hex
00000000	0
00000001	1
00000010	2
00000011	3
00000100	4
00000101	5
00000110	6
00000111	7
00001000	8
00001001	9
00001010	a
00001011	b
00001100	c
00001101	d
00001110	e
00001111	f
00010000	10
00010001	11
00010010	12
00010011	13
00010100	14
00010101	15
00010110	16
00010111	17
00011000	18
00011001	19
00011010	1a
00011011	1b
00011100	1c
00011101	1d
00011110	1e
00011111	1f
00100000	20
00100001	21
00100010	22
00100011	23
00100100	24
00100101	25
00100110	26

Binary	Hex
01000000	40
01000001	41
01000010	42
01000011	43
01000100	44
01000101	45
01000110	46
01000111	47
01001000	48
01001001	49
01001010	4a
01001011	4b
01001100	4c
01001101	4d
01001110	4e
01001111	4f
01010000	50
01010001	51
01010010	52
01010011	53
01010100	54
01010101	55
01010110	56
01010111	57
01011000	58
01011001	59
01011010	5a
01011011	5b
01011100	5c
01011101	5d
01011110	5e
01011111	5f
01100000	60
01100001	61
01100010	62
01100011	63
01100100	64
01100101	65
01100110	66

Binary	Hex
10000000	80
10000001	81
10000010	82
10000011	83
10000100	84
10000101	85
10000110	86
10000111	87
10001000	88
10001001	89
10001010	8a
10001011	8b
10001100	8c
10001101	8d
10001110	8e
10001111	8f
10010000	90
10010001	91
10010010	92
10010011	93
10010100	94
10010101	95
10010110	96
10010111	97
10011000	98
10011001	99
10011010	9a
10011011	9b
10011100	9c
10011101	9d
10011110	9e
10011111	9f
10100000	a0
10100001	a1
10100010	a2
10100011	a3
10100100	a4
10100101	a5
10100110	a6

Binary	Hex
11000000	c0
11000001	c1
11000010	c2
11000011	c3
11000100	c4
11000101	c5
11000110	c6
11000111	c7
11001000	c8
11001001	c9
11001010	ca
11001011	cb
11001100	cc
11001101	cd
11001110	ce
11001111	cf
11010000	d0
11010001	d1
11010010	d2
11010011	d3
11010100	d4
11010101	d5
11010110	d6
11010111	d7
11011000	d8
11011001	d9
11011010	da
11011011	db
11011100	dc
11011101	dd
11011110	de
11011111	df
11100000	e0
11100001	e1
11100010	e2
11100011	e3
11100100	e4
11100101	e5
11100110	e6

Binary to Hex Conversion Chart (continued)

Binary	Hex
00100111	27
00101000	28
00101001	29
00101010	2a
00101011	2b
00101100	2c
00101101	2d
00101110	2e
00101111	2f
00110000	30
00110001	31
00110010	32
00110011	33
00110100	34
00110101	35
00110110	36
00110111	37
00111000	38
00111001	39
00111010	3a
00111011	3b
00111100	3c
00111101	3d
00111110	3e
00111111	3f

Binary	Hex
01100111	67
01101000	68
01101001	69
01101010	6a
01101011	6b
01101100	6c
01101101	6d
01101110	6e
01101111	6f
01110000	70
01110001	71
01110010	72
01110011	73
01110100	74
01110101	75
01110110	76
01110111	77
01111000	78
01111001	79
01111010	7a
01111011	7b
01111100	7c
01111101	7d
01111110	7e
01111111	7f

Binary	Hex
10100111	a7
10101000	a8
10101001	a9
10101010	aa
10101011	ab
10101100	ac
10101101	ad
10101110	ae
10101111	af
10110000	b0
10110001	b1
10110010	b2
10110011	b3
10110100	b4
10110101	b5
10110110	b6
10110111	b7
10111000	b8
10111001	b9
10111010	ba
10111011	bb
10111100	bc
10111101	bd
10111110	be
10111111	bf

Binary	Hex
11100111	e7
11101000	e8
11101001	e9
11101010	ea
11101011	eb
11101100	ec
11101101	ed
11101110	ee
11101111	ef
11110000	f0
11110001	f1
11110010	f2
11110011	f3
11110100	f4
11110101	f5
11110110	f6
11110111	f7
11111000	f8
11111001	f9
11111010	fa
11111011	fb
11111100	fc
11111101	fd
11111110	fe
11111111	ff

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

FORMAT DESIGN TOOLS

D

Use copies of these worksheets and grids to create formats, batch data, and check digit schemes. Keep copies of the completed forms:

- Online Configuration Worksheet
- Batch Worksheet
- Check Digit Worksheet
- Supply Layout Grids (English, Metric, Dots)
- Format Worksheet
- Sample Format Worksheet

ONLINE HEADER

I	HEADER

SYSTEM SETUP

A	HEADER
	POWERUP MODE
	LANGUAGE
O	SEPARATOR
O	SLASH ZERO
	SYMBOL SET

SUPPLY SETUP

B	HEADER
	SUPPLY TYPE
	RIBBON
	FEED MODE
	SUPPLY POSN
	CUT POSN

PRINT CONTROL

C	HEADER
	CONTRAST
	PRINT ADJUST
	MARGIN ADJUST
	SPEED ADJUST
O	PRINTHD WIDTH

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

T	HEADER
	BAUD
	WORD LENGTH
	STOP BITS
	PARITY
	FLOW CONTROL

BATCH HEADER

HEADER	FORMAT #	NEW / UPDATE	QUANTITY
B			

BATCH CONTROL

HEADER	FEED MODE	BATCH SEP	PRINT MULT	MULTI PART	CUT TYPE	CUT MULT
m		0				

BATCH DATA

FIELD #	DATA
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	

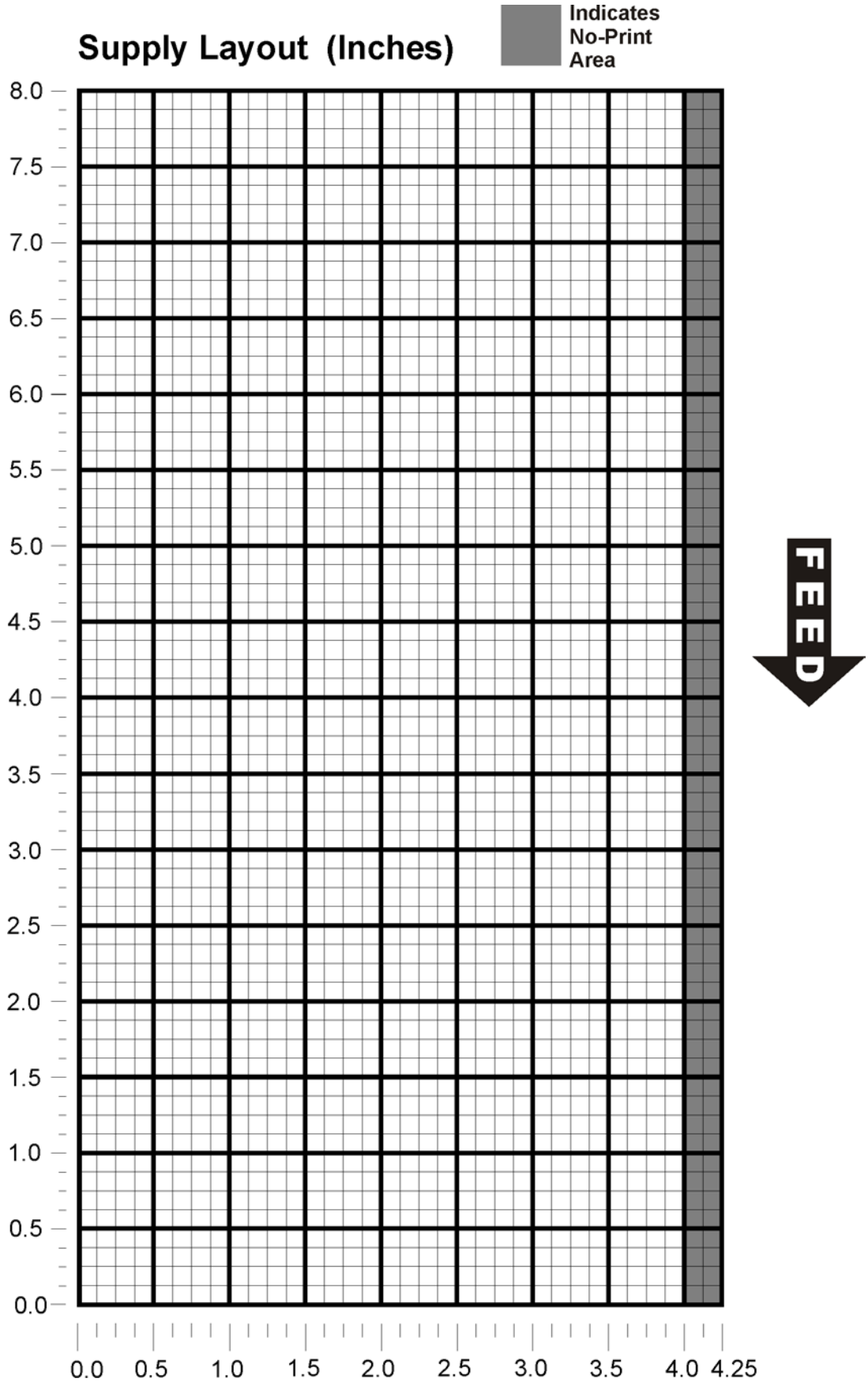
BATCH CONTINUATION

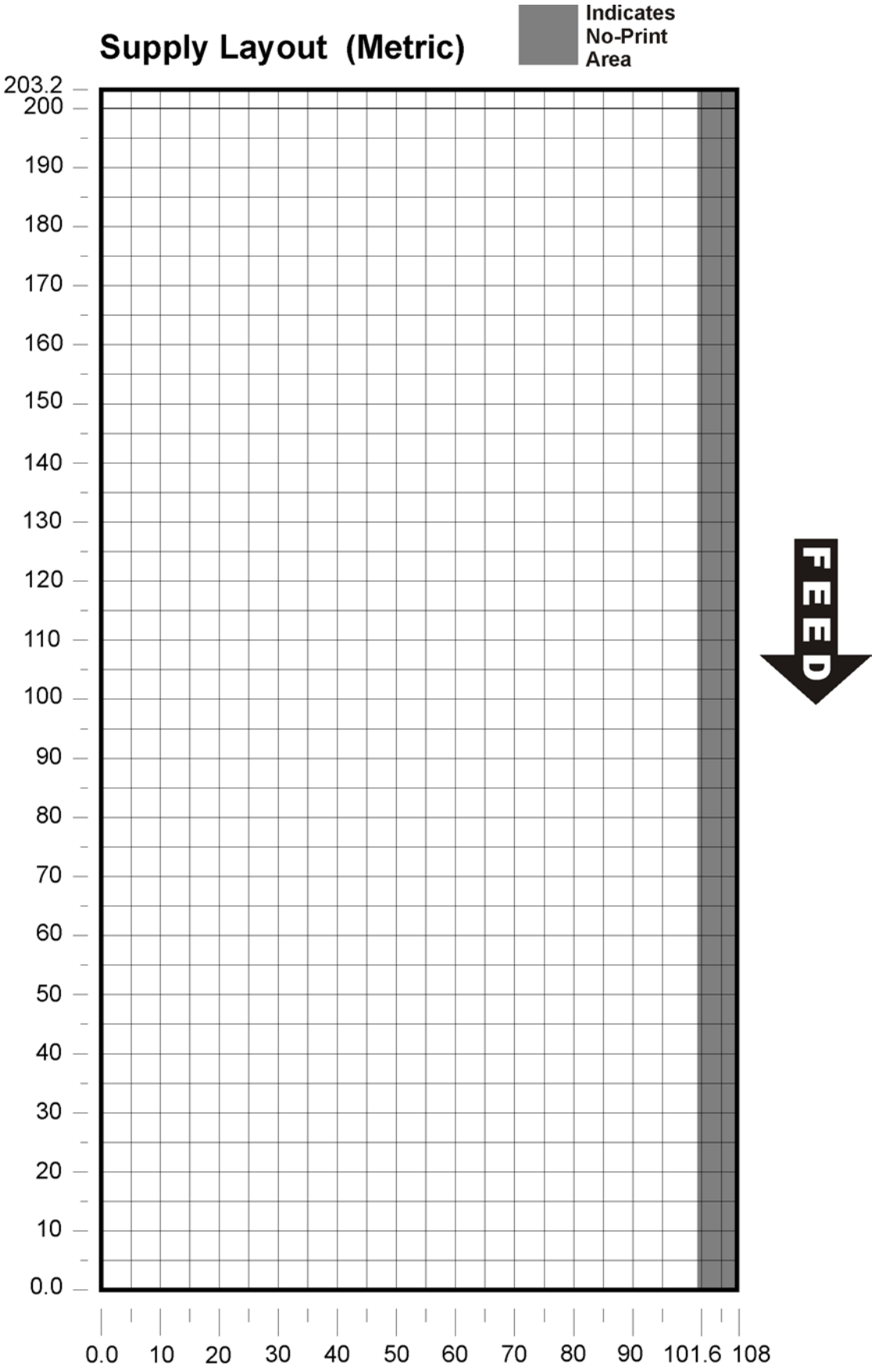
C	DATA
C	
C	
C	
C	
C	
C	
C	
C	
C	
C	
C	
C	
C	
C	

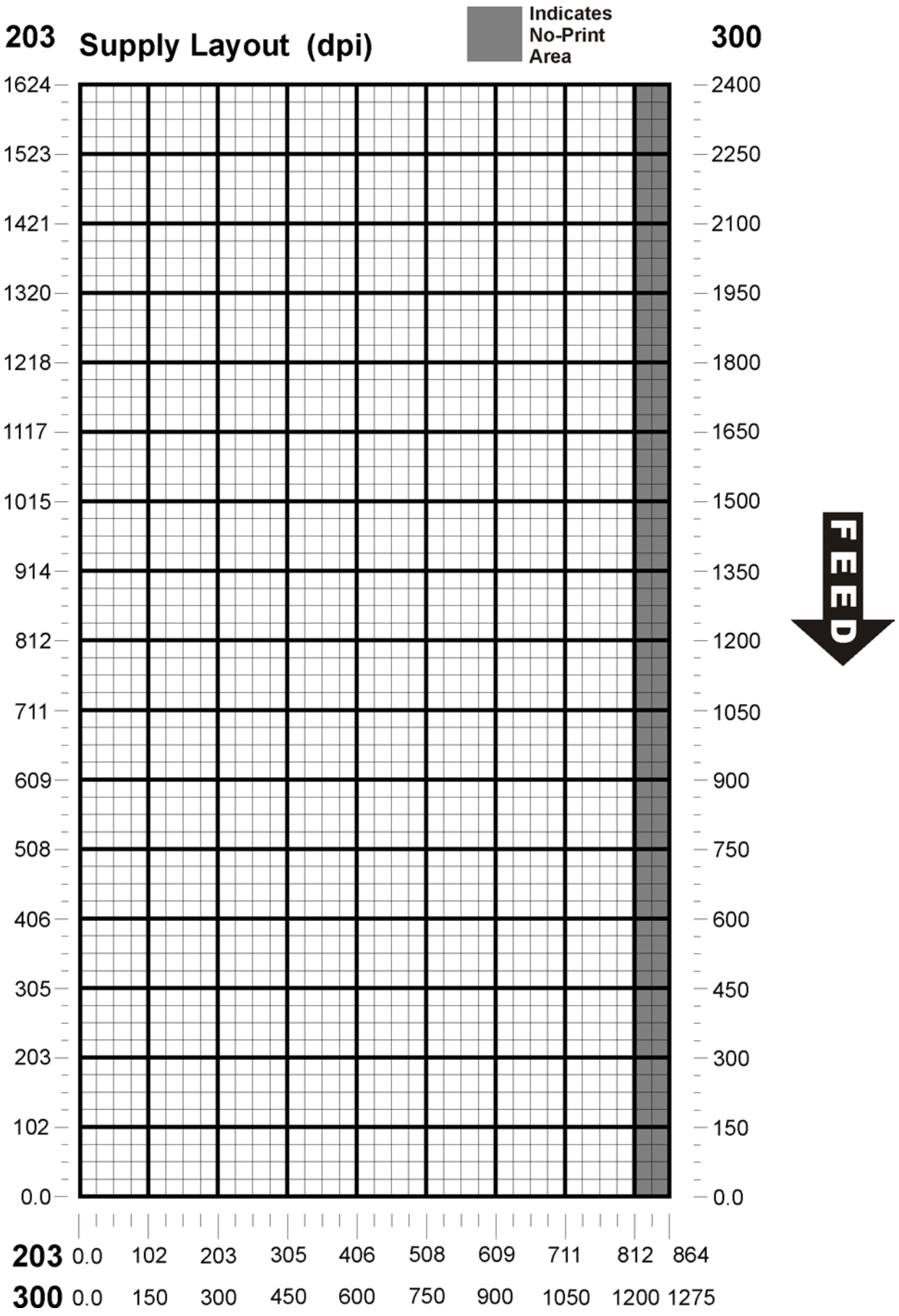
A1 HEADER	A2 SELECTOR #	A3 ACTION	A4 DEVICE	A5 MODULUS	A6 LENGTH	A7 ALGORITHM	WEIGHTS A8
A			R				

A1 HEADER	A2 SELECTOR #	A3 ACTION	A4 DEVICE	A5 MODULUS	A6 LENGTH	A7 ALGORITHM	WEIGHTS A8
A			R				

A1 HEADER	A2 SELECTOR #	A3 ACTION	A4 DEVICE	A5 MODULUS	A6 LENGTH	A7 ALGORITHM	WEIGHTS A8
A			R				


Supply Layout (Inches)
Indicates No-Print Area
8.0
7.5
7.0
6.5
6.0
5.5
5.0
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0.0 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.25
FEED


Supply Layout (Metric)
Indicates
No-Print
Area
203.2
200
190
180
170
160
150
140
130
120
110
100
90
80
70
60
50
40
30
20
10
0.0
0.0
10
20
30
40
50
60
70
80
90
101.6
108
FEED


203
Supply Layout (dpi)
Indicates No-Print Area
300
1624
1523
1421
1320
1218
1117
1015
914
812
711
609
508
406
305
203
102
0.0
2400
2250
2100
1950
1800
1650
1500
1350
1200
1050
900
750
600
450
300
150
0.0
FEED
203 0.0 102 203 305 406 508 609 711 812 864
300 0.0 150 300 450 600 750 900 1050 1200 1275

FORMAT WORKSHEET

FORMAT HEADER

F1 HEADER	F2 FORMAT #	F3 ACTION	F4 DEVICE	F5 MEASURE	F6 LENGTH	F7 WIDTH	R8 NAME. ("IN QUOTES")
F			R				

TEXT FIELDS

															OPTION #1 Fixed Characters			OPTION #4 Copy Data from Previous Field							OPTION #30 Pad Data				OPTION #42 Price Field			OPTION #61 Reimage Field		
T1 HEADER	T2 FIELD #	T3 # OF CHAR.	T4 FIX/VAR	T5 ROW	T6 COLUMN	T7 GAP	T8 FONT	T9 HGT. MAG.	T10 WID. MAG.	T11 COLOR	T12 ALIGNMENT	T13 CHAR. ROT.	T14 FIELD ROT.	T15 SYM. SET	R1 HEADER	R2 CODE #	R3 FIXED CHAR. ("IN QUOTES")	R1 HEADER	R2 CODE #	R3 SRC FIELD	R4 SRC START	R5 # TO COPY	R6 DEST. START	R7 COPY CODE	R1 HEADER	R2 CODE #	R3 L / R	R4 CHARACTER	R1 HEADER	R2 CODE #	R3 APP CODE	R1 HEADER	R2 CODE #	R3 INPUT (√)
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	

BAR CODE FIELDS

												OPTION #1 Fixed Characters			OPTION #4 Copy Data from Previous Field							OPTION #30 Pad Data				OPTION #31 Define Check Digit				OPTION #50 Bar Code Densities							OPTION #51 PDF417 Security / Truncation				OPTION #52 PDF417 Aspect Ratio				OPTION #60 Incrementing Data						OPTION #61 Reimage Field		
B1 HEADER	B2 FIELD #	B3 # OF CHAR.	B4 FIX/VAR	B5 ROW	B6 COLUMN	B7 FONT	B8 DENSITY	B9 HEIGHT	B10 TEXT	B11 ALIGNMENT	B12 FIELD ROT.	R1 HEADER	R2 CODE #	R3 FIXED CHAR. ("IN QUOTES")	R1 HEADER	R2 CODE #	R3 SRC FIELD	R4 SRC START	R5 # TO COPY	R6 DEST. START	R7 COPY CODE	R1 HEADER	R2 CODE #	R3 L / R	R4 CHARACTER	R1 HEADER	R2 CODE #	R3 GEN/VER	R4 CHECK DIGIT #	R1 HEADER	R2 CODE #	R3 DOT WIDTH NAR. ELEMENT	R4 DOT WIDTH WIDE ELEMENT	R5 ADDITIONAL CHAR. GAP	R6 ADDITIONAL NAR. SPACE	R7 ADDITIONAL WIDE SPACE	R1 HEADER	R2 CODE #	R3 SECURITY LEVEL	R4 STANDARD /DEFAULT	R1 HEADER	R2 CODE #	R3 ROW /COLUMN	R4 DIMENSION NUMBER	R1 HEADER	R2 CODE #	R3 I/D	R4 AMOUNT	R5 L POS	R6 R POS	R1 HEADER	R2 CODE #	R3 INPUT (√)
B												R	1		R	4						R	30			R	31			R	50						R	51			R	52			R	60					R	61	
B												R	1		R	4						R	30			R	31			R	50						R	51			R	52			R	60					R	61	
B												R	1		R	4						R	30			R	31			R	50						R	51			R	52			R	60					R	61	
B												R	1		R	4						R	30			R	31			R	50						R	51			R	52			R	60					R	61	

CONSTANT TEXT FIELDS

C1 HEADER	C2 ROW	C3 COLUMN	C4 GAP	C5 FONT	C6 HGT. MAG.	C7 WID. MAG.	C8 COLOR	C9 ALIGNMENT	C10 CHAR. ROT.	C11 FIELD ROT.	C12 FIXED CHAR. ("IN QUOTES")	C13 SYM. SET
C												
C												
C												
C												
C												
C												
C												

NON - PRINTABLE TEXT FIELDS

			OPTION #1 Fixed Characters			OPTION #4 Copy Data from Previous Field						
D1 HEADER	D2 FIELD #	D3 # OF CHAR.	R1 HEADER	R2 CODE #	R3 FIXED CHAR. ("IN QUOTES")	R1 HEADER	R2 CODE #	R3 SRC FIELD	R4 SRC START	R5 # TO COPY	R6 DEST. START	R7 COPY CODE
D			R	1		R	4					
D			R	1		R	4					
D			R	1		R	4					
D			R	1		R	4					

LINES

L1 HEADER	L2 TYPE	L3 ROW	L4 COLUMN	L5 ANGLE/ END ROW	L6 LENGTH/ END COL.	L7 THICKNESS	L8 PATTERN
L							
L							
L							
L							

BOXES

Q1 HEADER	Q2 ROW	Q3 COLUMN	Q4 END ROW	Q5 END COLUMN	Q6 THICKNESS	Q7 PATTERN
Q						
Q						
Q						
Q						

GRAPHICS FIELDS

G1 HEADER	G2 GRAPH ID	G3 ROW	G4 COLUMN	G5 MODE	G6 ROTATION
G					
G					
G					
G					

Format Name ______________________

Format # ______________________

Date ______________________

Supply Size ______________________

Supply Type ______________________

Customer Name ______________________

Software Version ______________________

FORMAT HEADER

F1 HEADER	F2 FORMAT #	F3 ACTION	F4 DEVICE	F5 MEASURE	F6 LENGTH	F7 WIDTH	F8 NAME ("IN QUOTES")
F	1	A	R	E	300	400	"TEXTILES"

TEXT FIELDS

T1 HEADER	T2 FIELD #	T3 # OF CHAR.	T4 FIX/VAR	T5 ROW	T6 COLUMN	T7 GAP	T8 FONT	T9 HGT. MAG.	T10 WID. MAG.	T11 COLOR	T12 ALIGNMENT	T13 CHAR. ROT.	T14 FIELD ROT.	T15 SYM. SET	OPTION #1 Fixed Characters			OPTION #4 Copy Data from Previous Field							OPTION #30 Pad Data				OPTION #42 Price Field			OPTION #61 Reimage Field		
															R1 HEADER	R2 CODE #	R3 FIXED CHAR. ("IN QUOTES")	R1 HEADER	R2 CODE #	R3 SRC FIELD	R4 SRC START	R5 # TO COPY	R6 DEST. START	R7 COPY CODE	R1 HEADER	R2 CODE #	R3 L / R	R4 CHARACTER	R1 HEADER	R2 CODE #	R3 APP CODE	R1 HEADER	R2 CODE #	R3 INPUT (√)
T	2	10	V	250	050	0	1	1	1	B	C	0	0	0	R	1		R	4	3	1	3	1	1	R	30			R	42		R	61	
T	5	25	V	080	010	0	1	1	1	B	L	0	0	0	R	1	"___%________________"	R	4	4	1	20	6	1	R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	
T															R	1		R	4						R	30			R	42		R	61	

BAR CODE FIELDS

B1 HEADER	B2 FIELD #	B3 # OF CHAR.	B4 FIX/VAR	B5 ROW	B6 COLUMN	B7 FONT	B8 DENSITY	B9 HEIGHT	B10 TEXT	B11 ALIGNMENT	B12 FIELD ROT.	OPTION #1 Fixed Characters			OPTION #4 Copy Data from Previous Field							OPTION #30 Pad Data				OPTION #31 Define Check Digit			
												R1 HEADER	R2 CODE #	R3 FIXED CHAR. ("IN QUOTES")	R1 HEADER	R2 CODE #	R3 SRC FIELD	R4 SRC START	R5 # TO COPY	R6 DEST. START	R7 COPY CODE	R1 HEADER	R2 CODE #	R3 L / R	R4 CHARACTER	R1 HEADER	R2 CODE #	R3 GEN/VER	R4 CHECK DIGIT #
B	3	12	V	150	040	1	2	80	7	L	0	R	1		R	4						R	30			R	31		
B												R	1		R	4						R	30			R	31		
B												R	1		R	4						R	30			R	31		
B												R	1		R	4						R	30			R	31		

OPTION #50 Bar Code Densities							OPTION #51 PDF417 Security / Truncation				OPTION #52 PDF417 Aspect Ratio				OPTION #60 Incrementing Data						OPTION #61 Reimage Field		
R1 HEADER	R2 CODE #	R3 DOT WIDTH NAR. ELEMENT	R4 DOT WIDTH WIDE ELEMENT	R5 ADDITIONAL CHAR. GAP	R6 ADDITIONAL NAR. SPACE	R7 ADDITIONAL WIDE SPACE	R1 HEADER	R2 CODE #	R3 SECURITY LEVEL	R4 STANDARD /DEFAULT	R1 HEADER	R2 CODE #	R3 ROW /COLUMN	R4 DIMENSION NUMBER	R1 HEADER	R2 CODE #	R3 I/D	R4 AMOUNT	R5 L POS	R6 R POS	R1 HEADER	R2 CODE #	R3 INPUT (√)
R	50						R	51			R	52			R	60					R	61	
R	50						R	51			R	52			R	60					R	61	
R	50						R	51			R	52			R	60					R	61	
R	50						R	51			R	52			R	60					R	61	

CONSTANT TEXT FIELDS

C1 HEADER	C2 ROW	C3 COLUMN	C4 GAP	C5 FONT	C6 HGT. MAG.	C7 WID. MAG.	C8 COLOR	C9 ALIGNMENT	C10 CHAR. ROT.	C11 FIELD ROT.	C12 FIXED CHAR. ("IN QUOTES")	C13 SYM. SET
C	030	010	0	1	1	1	B	L	0	0	"MADE IN USA"	0
C												
C												
C												
C												
C												
C												

NON-PRINTABLE TEXT FIELDS

D1 HEADER	D2 FIELD #	D3 # OF CHAR.	OPTION #1 Fixed Characters			OPTION #4 Copy Data from Previous Field						
			R1 HEADER	R2 CODE #	R3 FIXED CHAR. ("IN QUOTES")	R1 HEADER	R2 CODE #	R3 SRC FIELD	R4 SRC START	R5 # TO COPY	R6 DEST. START	R7 COPY CODE
D	3	3	R	1		R	4					
D	4	20	R	1		R	4					
D			R	1		R	4					
D			R	1		R	4					

(SAMPLE) FORMAT WORKSHEET

Format Name TEXTILES

Format # 1

Date 01/08/08

Supply Size 4" x 3"

Supply Type THERMAL DIRECT

Customer Name ACE INDUSTRIES

Software Version V2.0

LINES

L1 HEADER	L2 TYPE	L3 ROW	L4 COLUMN	L5 ANGLE/ END ROW	L6 LENGTH/ END COL.	L7 THICKNESS	L8 PATTERN
L	S	110	030	110	150	10	" "
L							
L							
L							

BOXES

Q1 HEADER	Q2 ROW	Q3 COLUMN	Q4 END ROW	Q5 END COLUMN	Q6 THICKNESS	Q7 PATTERN
Q	240	030	270	150	3	" "
Q						
Q						
Q						

GRAPHICS FIELDS

G1 HEADER	G2 GRAPH ID	G3 ROW	G4 COLUMN	G5 MODE	G6 ROTATION
G	S	010	200	0	0
G					
G					
G					

GLOSSARY

G

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).

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.

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" |
```

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).

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.

TrueType Fonts
All characters follow the TrueType outline font standard. All characters are scalable and smooth at any point size.

Volatile RAM
Contains information that is LOST when the printer is turned off.

Avery Dennison Printer Systems Division
170 Monarch Lane
Miamisburg, OH 45342

1-800-543-6650 (In the U.S.A.)
1-800-387-4740 (In Canada)
printers.averydennison.com

AVERY DENNISON