Packet Reference Manual

Monarch® Pathfinder® Ultra® *Silver* and *Gold* Printers

AVERY DENNISON

TC6037PR Rev. AF 1/08

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

WARNING

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

CANADIAN D.O.C. WARNING

This digital apparatus does not exceed the Class A limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.

Le présent appareil numérique n'émet pas de bruits radioélectriques dépassant les limites applicables aux appareils numériques de la classe A prescrites dans le Réglement sur le brouillage radioélectrique édicte par le ministère des Communications du Canada.

Trademarks
Monarch®, Pathfinder®, Ultra®, 6032, and 6037 are trademarks of Paxar Americas, Inc.
Paxar® is a trademark of Paxar Corporation.
Avery Dennison® is a trademark of Avery Dennison Corporation.
Microsoft, Windows, and NT are trademarks of Microsoft Corporation.
Hewlett-Packard is a trademark of Hewlett-Packard Company.
Adobe and Acrobat are trademarks of Adobe Systems Incorporated.
UFST, Monotype, the Monotype logo, and CG Triumvirate are trademarks of Monotype Imaging, Inc.

LITHIUM - ION RECHARGEABLE BATTERY.
FOR PROPER RECYCLING OR DISPOSAL, CALL YOUR LOCAL SERVICE OFFICE.

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

Monotype Imaging

Li-Ion

TABLE OF CONTENTS

GETTING STARTED

1

Before you read this manual, review the printer information in the *Equipment Manual*.

About This Manual

This manual is for the developer who is creating and designing custom formats for the Monarch® Pathfinder® Ultra® Silver 6032™ and Monarch® Pathfinder® Ultra® Gold 6037™ printers.

Note: You must develop an application that uses the packets you create. Refer to the Programmer's Manual for information about developing an application.

This chapter creates a sample Monarch® Printer Control Language II (MPCLII) packet.

Standard Features

The following features are standard on the printer:

Feature	6032	6037
DPI	203	203
Max. Print Image	1.89" x 3.84"	1.89" x 3.84"
Print Speed	Up to 2" per second	Up to 4" per second
LCD	2-line graphical with backlight	4-line or 8-line graphical with backlight
Number of Keys	24	38
Memory	4 MB RAM 4 MB Flash	1 MB RAM 2 MB Flash
Prints 2D Bar Codes	No	Yes
On-Demand Sensor	Not available	Yes
Supports International fonts	Yes	No

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. This section describes how to create a sample MPCLII format packet.

Make sure supplies are loaded, you have a fully charged battery, the printer is connected to a host and ready to receive data, and you have software in the printer. Refer to your *Equipment Manual* for more information.

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",1  ¦
```

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:

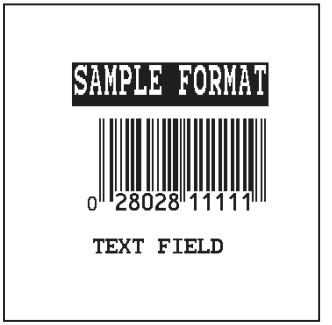
```
T,2,18,V,50,50,1,1,1,1,B,L,0,0,1  ¦}
```

For detailed information about the format header, text, constant text, and bar code fields, see Chapter 2, “Defining Fields.” For information about batch packets, see Chapter 5, “Printing.”

You have created a format packet for your MPCLII printer. Now, a batch packet must be created before you can print 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 your file as SAMPLE.FMT.

9. For your application to use the format, use the PCL library functions in the *Programmer Manual*.

Using MPCLII Conventions

Here are some guidelines to follow when using MPCLII.

MPCLII Punctuation

Use the following symbols when creating MPCLII packets:

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

Note: These MPCL characters are the default.

* The field separator is the split vertical bar, which we are representing as ¦ in this manual. The decimal value is 124. To enter this character, use the Shift key plus the Split Vertical Bar key on your computer’s keyboard. Depending on your 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 **200** fields in a format. Each ¦ indicates one field. However, options are not counted as fields.
- The field number (**0 - 999**) must be unique. We recommend starting at 1, instead of 0.
- Do not use a field number more than once per format.
- Define all fields in the order you want to image/print them.
- 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.
- 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 your 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.

Starting with a Design

Before you create a format packet, you must design your label. There are several steps to designing a custom label:

1. Decide which fields should appear on your label. See “Determining Format Contents” for more information.
2. Determine your label size. Labels are available from us in a wide variety of sizes. Your application and the amount of data you need to print determines the supply size. Contact your sales representative for more information.
3. Draw a rough sketch of your label. You may want to draw several variations to see what works best. See “Drawing Rough Sketches” for more information.
4. Identify the field types that appear on your label. See “Considering Field Types” for more information.
5. Decide which fonts you want to use. See “Considering Fonts” for more information.
6. Fill out your Format Worksheet. See “Using the Format Worksheet” for more information.

At this point, you are ready to use your design.

7. Create a format packet, based on how you filled out your worksheet. See Chapter 2, “Defining Fields,” for more information.
8. For your application to use the format, use the PCL library functions in the *Programmer Manual*.
9. Execute your application.

Determining Format Contents

Before you lay out your format, you need to make a few decisions. For example:

- How large is your supply?
- Which fonts do you want to use?
- Do you want to include a bar code?
- Do you want to include graphics?

Determining the Print Area

The print area varies, depending on the size of your supply. Below are the maximum and minimum print areas. Notice that the top edge of the supply exits the printer first. There is a non-print zone on the edges and top (0.235") and bottom (0.115") of the supply. For exact print area measurements of your supply, see the supply layout grids in Appendix D.

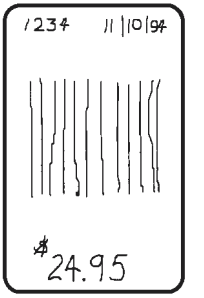
Unit of Measure	Maximum Supply Size	Maximum Print Area	Minimum Supply Size	Minimum Print Area
English (1/100")	205 x 400	189 x 365	120 x 55	109 x 20
Metric (1/10mm)	521 x 1016	480 x 927	305 x 140	277 x 51
Dots (1/203 dots)	416 x 812	384 x 741	244 x 112	221 x 41

The minimum supply length for peel mode is 0.785 inches with a printable area of 0.435 inches (11 mm or 88 dots).

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

```
Dots = inches x 203
Metric (1/10mm) = inches x 254
English (1/100 inch) = 100 x (dots/203)
Dots = Metric (1/10 mm) x .797
```

Drawing Rough Sketches



After you decide what information you want to print, sketch how you want the information to appear on the label. Note any areas that are preprinted on the label, such as a logo.

As soon as you know what information to include on the label, and you have a rough sketch, you can use a supply layout grid to help you layout and size your label. If you do not want to use a grid, go to "Considering Field Types" to choose what information you want on your label.

Using Supply Layout Grids

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

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

- English

  The English grid is measured in 1/100 inches.

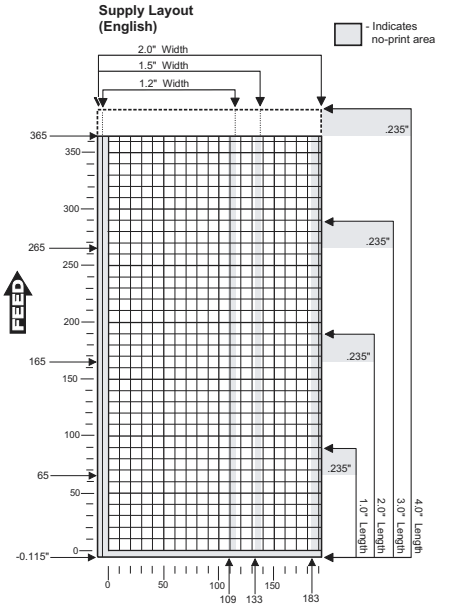
- Metric

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

- Graphic

  The printer uses dots to print images on a label. The printhead has 203 dots per inch (DPI).

If you want to use the supply layout grids, a copy of each is in Appendix D, "Format Design Tools."



Considering Field Types

After you select a supply size, the next step in designing a format is to decide what information you want to print on the label. For example, you may want to print your company name, price of an item, and a bar code that combines information from other places. Everything you want to print falls into one of the following categories.

Field Type	Description	Examples
Text	Contains letters, numbers, or symbols you want to print.	item number, item description, department number, price, date
Bar Code	Used for printing bar codes that can be scanned.	item or serial numbers, zip codes, information you don't want to have visible to customers
Constant Text	Prints fixed characters that print without changing.	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

Note: All of the above field types except graphics are discussed in Chapter 2. See Chapter 4, "Creating Graphics" for information on including graphics in your format.

Considering Fonts

When working with fonts, you have three considerations:

- font appearance
- font size (bitmapped)
- font spacing (monospaced or proportional)

See Appendix B, "Fonts," for samples of each font.

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 your 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 your label.

1. Make a copy of the Format Worksheet.
2. Define the Format Header.
3. Define options as you require them. See Chapter 3, “Defining Field Options” for more information.

DEFINING FIELDS

2

This chapter provides a reference for defining

- the format header
- text and constant text
- bar code fields
- line and box 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 **0 - 999** to identify the format.

F3. action Action. Enter **A** to add the format to the printer.

F4. device Format storage device. Options:

R RAM (default) When you turn off the printer, items saved in RAM are lost.
F Flash Memory. When you turn off the printer, items saved in Flash memory are saved.

F5. measure Unit of measure. Options:

E English, measured in 1/100 inches
M Metric, measured in 1/10 mm
G Graphic, measured in dots

F6. length Supply length, top to bottom, in selected units.

English	**55 - 400**
Metric	**140 - 1016**
Dots	**112 - 812**

Note: The minimum supply length for peel mode is 0.785 inches (20 mm or 159 dots).

F7. width Supply width, from left to right, in selected units.

English	**120 - 205**
Metric	**305 - 480**
Dots	**244 - 416**

F8. "name" Format name (optional), 0 - 8 characters, enclose within quotation marks.

Example

```
{F,1,A,R,E,300,100,"TEXTILES" ¦
```

Format 1 ("TEXTILES") uses a three inch long by one inch wide label.

Defining Text Fields

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

Syntax

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

T1. T Text Field.

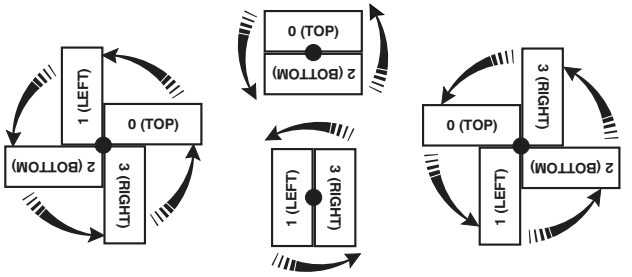
T2. field# Unique number from **0 - 999** to identify this field.

T3. # of char Maximum number of printed characters (**0 - 2710**) in the field.

T4. fix/var Fixed or variable length field. Options:

F Fixed length
V Variable length

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

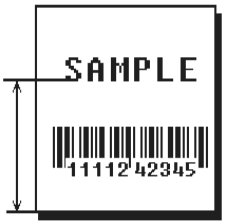


Left/Center/Right-Justified **Balanced** **End-Justified**

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

English	**0 - 365**
Metric	**0 - 927**
Dot	**0 741**

Note: The minimum printable length for peel mode is 0.435 inches (11 mm or 88 dots).



T6. column Distance from the left edge of the print area to the pivot point to find the column location.

English **0 - 189**
Metric **0 - 480**
Dots **0 - 383**

T7. gap Number of dots between characters (203 dots per inch). Range: **0 - 99**.

Note: For monospaced fonts, the additional spacing is added to the existing inter-character gap. This is also true for proportionally spaced fonts, but remember that the inter-character gap varies with character combinations.

Any number other than **0** or the default number affects your field width. Default spacing:

Reduced and Letter Gothic Bold 6 pt.	**1 dot**
HR2 and Letter Gothic Bold 9 pt.	**2 dots**
Standard, Bold, OCRA-like, and HR1	**3 dots**
All other fonts	**varies with each letter**

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

T8. font Style of font. Options:

1	Standard	**5**	HR1
2	Reduced	**6**	HR2
3	Bold	**10**	CG Triumvirate™ Typeface Bold 9 pt
4	OCRA-like	**11**	CG Triumvirate™ Typeface 6 pt

50 EFF Swiss Bold (scalable)

CG Triumvirate™ Typeface Bold		CG Triumvirate™ Typeface Bold Cond	
1000	6.5 pt	**1006**	6.5 pt
1001	8 pt	**1007**	8 pt
1002	10 pt	**1008**	10 pt
1003	12 pt	**1009**	12 pt
1004	18 pt	**1010**	18 pt
1005	22 pt	**1011**	22 pt

Letter Gothic Bold

1012	6 pt	**1013**	9 pt

Or a valid downloaded font selector number. Make sure the downloaded font is installed in the printer. Fonts 5 and 6 are for numeric data only.

Point sizes greater than 12 include only the following special characters: 0123456789#$%&(),./@DFKLMPS\kprö¢£¥. All other point sizes use the whole symbol set. This information may not apply to optional fonts.

T9. hgt mag Height magnifier, **1 - 7** (times- for bitmapped fonts). With TrueType/Scalable fonts, use **4 - 250**, for the font's point size.

T10. wid mag Width magnifier, **1 - 7** (times). With TrueType/Scalable fonts, use **4 - 250**, for the font's point size. Proportionally spaced fonts do not have a set width. To estimate the size of your field, use the letter "W" for the widest field or an "L" for an average width field. Find your selected font and the desired width in Appendix B, "Fonts."

T11. color Options for standard printer fonts:

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

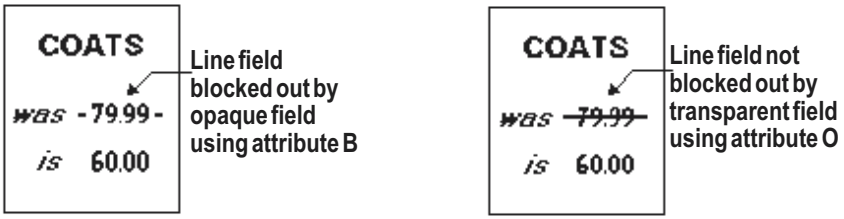
Options for scalable fonts:

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 30% on a given square inch of the label, or the printhead life may be decreased.

There are two types of field color overlay attributes:

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.



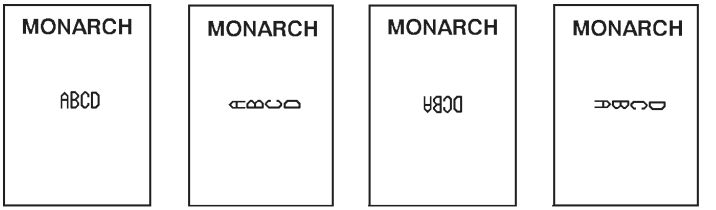
Field placement in the packet is an important consideration when using field color attributes. If a line field is defined before the overlay (text or constant text) field, the line field is blocked out by the overlay field, depending on the overlay field's color attribute. If a line field is defined 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 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.

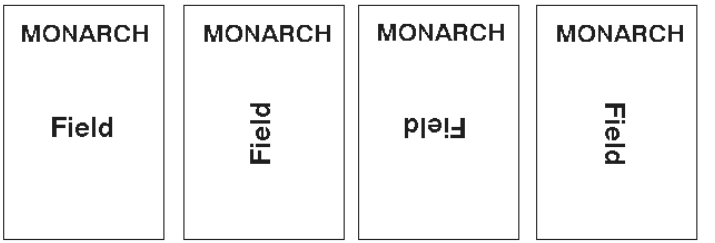
T13. char rot Character rotation. The field or supply does not rotate, only the characters do. Options:

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



T14. 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. 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. Use **0** for the Internal Symbol Set. For scalable fonts, use:

1	ANSI Symbol Set
100	Macintosh
101	Wingdings
102	Unicode (user input) for particular mapping
103	BIG5 (user input) for Unicode mapping
104	GB2312 (user input) for Unicode mapping
105	SJIS (user input) for Unicode mapping Code Page 932 (Japanese Shift-JIS)
106	GB2312 (user input) for GB2312 mapping Code Page 936 (Simplified Chinese)
107	BIG5 (user input) for BIG5 mapping Code Page 950 (Traditional Chinese)
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)

The CG Triumvirate™ typefaces support only the ANSI and DOS Code Page 437 and 850 Symbol Sets. The scalable font (font#50) does not support Code Page 1256 (Arabic). Code pages 852-860 and 1250-1258 are for downloaded TrueType fonts or the scalable font. Code pages 102-107 require the print engine memory expansion option and a downloaded International TrueType font. TrueType fonts are designed to be regionally specific; therefore, all code pages may not be supported in a given font. See Appendix C, “Symbol Sets/ Code Pages” for more information.

Note: Only the 6032 printer supports printing International fonts.

Example

```
T,2,10,V,250,80,0,1,1,1,B,C,0,0,0 ¦
```

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

Defining Bar Code Fields

Each bar code field requires a separate definition.

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 **0 - 999** to identify this field.

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 - 2710**.

For Quick Response bar codes, this number includes header information. The maximum depends on the type of characters entered for the batch data and differs for the two models of the bar 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. As you increase the error correction level, the maximum number of characters decreases.

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

Bar Code	Number of Characters	Fixed or Variable
UPCA	12	F
UPCA+2	14	F
UPCA+5	17	F
UPCA+Price CD	12	F
UPCE	7	F
UPCE+2	9	F
UPCE+5	12	F
EAN8	8	F
EAN8+2	10	F
EAN8+5	13	F
EAN13	13	F
EAN13+2	15	F
EAN13+5	18	F
EAN13+Price CD	13	F
POSTNET	9 or 11	F
Interleaved 2 of 5 or Interleaved I 2 of 5 with Barrier Bar	0 - 2710	F or V
Code 39 (w/ or w/o CD) or MOD43	0 - 2710	F or V
Codabar	0 - 2710	F or V
Code 128	0 - 2710	F or V
Code 93	0 - 2710	V
MSI	0 - 14	F or V
PDF 417	0 - 2710	F or V
Maxicode	15 - 99	F or V
Code 16K	0 - 2710	V
Quick Response	1167 - 2710 (numeric) 707 - 2710 (alphanumeric)	V

Note: The two-dimensional bar codes only apply to the 6037 printer.

B5. row

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. Pivot points:

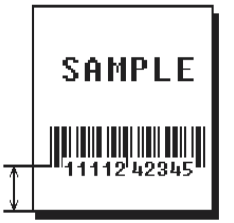
Remember to include text or numbers that may appear with the bar code for the row measurement.

Left/Center/Right-Justified Fields

Balanced Fields

End-Justified Fields

English **0 - 365**
Metric **0 - 927**
Dot **0 741**



B6. column

Distance from the lower left edge of the print area to the pivot point.

English **0 - 189**
Metric **0 - 480**
Dots **0 - 383**

Note: Allow a minimum of 1/10 inch between the scan edge of bar code and label edges or other data.

B7. font Bar code. Options:

1	UPCA	**15**	EAN8 +5
2	UPCE	**16**	EAN13 +2
3	Interleaved 2 of 5	**17**	EAN13 +5
4	Code 39 (no check digit)	**21**	LAC
5	Codabar	**22**	POSTNET
6	EAN8	**23**	Code 93
7	EAN13	**31**	Code 16K
8	Code 128	**32**	PDF 417
9	MSI	**33**	Maxicode
10	UPCA +2	**36**	Quick Response
11	UPCA +5	**40**	Code 39 (MOD 43 check digit)
12	UPCE +2	**41**	UPCA & Price CD
13	UPCE +5	**44**	EAN13 & Price CD
14	EAN8 +2	**50**	Interleaved 2of5 w/ Barrier Bar

Note: The two-dimensional bar codes only apply to the 6037 printer.

B8. density Bar code density. Use the following table for other bar codes.

Note: If the field contains an 11-digit UPC bar code, the printer automatically zero-suppresses it into a 6-digit UPCE bar code.

Bar Code Type	Density Selector	Density (% or cpi)	Narrow Element (dots/mils)	Narrow to Wide Ratio	Data Length	Appearance Codes Available	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
Interleaved 2 of 5 or I2of5 with Barrier Bar	1 2 3 4 5 6 7 8 9 10 11 **12** 13	1.1 2.1 3.2 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 8.7/11.5	5/24.6 4/19.7 3/14.8 2/9.9	N/A	0 to 2710	8	00H to 7FH

Note: For I 2of5 bar codes, pad field data to the left with a zero when an odd number of characters are used. Codabar uses “A” as the start and stop characters if not included with the data. All bar codes with

Bar Code Type	Density Selector	Density (% or cpi)	Narrow Element (dots/mils)	Narrow to Wide Ratio	Data Length	Appearance Codes Available	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 Available	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

Bar Code Type	Density Selector	Data Length
Quick Response 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.

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

English	**1**
Metric	**2**
Dots	**1**

For Quick Response bar codes, the value you enter is the symbol's maximum height. Small bar codes may not be scannable.

B10. text Appearance of text with bar code. For UPC and EAN, use **1** or **5 - 8**. For all others, use **8**. Options:

1 No check digit or number system or Quick Response Model 1
2 Quick Response Model 2
5 Number system at bottom, no check digit
6 Check digit at bottom, no number system
7 Check digit and number system at bottom
8 No text, bar code only

B11. alignment Choose **L**, **R**, **C**, **B** or **E** to align the bar code data correctly in the field. For I2 of 5, Code 39 (Mod 43), Codabar, and MSI, you can use **L**, **R**, **C**, **B** or **E**. For all other bar codes, use **L**.

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. 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

Note: Serial bar codes printed at speeds greater than 1.5 IPS may not scan properly.

Example

```
B,3,12,F,150,70,1,2,80,7,L,0 ¦
```

Defines a bar code field (field #3) with 12 characters of fixed length starting at row 150, column 70. 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 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 200 fields per format). The characters in this field cannot be changed by batch data. Field options do not apply to constant text fields.

Determine the height and the maximum width of the characters, using the tables in Appendix B, "Fonts." If you're using proportionally spaced fonts, use the average size of the characters. Mark the pivot point of your field. This will vary, depending on how your field is justified.

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 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 - 365**
Metric	**0 - 927**
Dot	**0 741**

C3. column Distance from the lower left edge of the print area to the pivot point.

English	**0 - 189**
Metric	**0 - 480**
Dots	**0 - 383**

C4. gap Number of dots between characters (203 dots per inch). Range: **0 - 99**.

Any number other than **0** or the default number affects your field width. Default spacing:

Reduced and Letter Gothic Bold 6 pt.	**1 dot**
HR2 and Letter Gothic Bold 9 pt.	**2 dots**
Standard, Bold, OCRA-like, and HR1	**3 dots**
All other fonts	**varies with each letter**

C5. font Style of font. Options:

1	Standard	**5**	HR1
2	Reduced	**6**	HR2
3	Bold	**10**	CG Triumvirate™ Typeface Bold 9 pt
4	OCRA-like	**11**	CG Triumvirate™ Typeface 6 pt
50	EFF Swiss Bold (scalable)		

CG Triumvirate™ Typeface Bold		CG Triumvirate™ Typeface Bold Cond	
1000	6.5 pt	**1006**	6.5 pt
1001	8 pt	**1007**	8 pt
1002	10 pt	**1008**	10 pt
1003	12 pt	**1009**	12 pt
1004	18 pt	**1010**	18 pt
1005	22 pt	**1011**	22 pt

Letter Gothic Bold

1012	6 pt	**1013**	9 pt

Or a valid downloaded font selector number. Make sure the downloaded font is installed in the printer. Fonts 5 and 6 are for numeric data only.

Point sizes greater than 12 include only the following special characters: 0123456789#$%&(),./@DFKLMPS\kprö¢£¥.
All other point sizes use the whole symbol set. This information may not apply to optional fonts.

C6. hgt mag Height magnifier, **1 - 7** (times). With TrueType/scalable fonts, use **4 - 250** for the font's point size.

C7. wid mag Width magnifier, **1 - 7** (times). With TrueType/scalable fonts, use **4 - 250** for the font's point size.

C8. color 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 30% on a given square inch of the label, or the printhead life may be decreased.

There are two types of field color overlay attributes:

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.

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. Options:

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

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

C10. char rot Character rotation. 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

C11. field rot Field rotation. Lower left corner of field is the pivot point. Options:

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

Note: Rotation is affected by the pivot point, which varies depending on how text is justified.

C12. "fixed char" Fixed characters to appear in the field. Maximum **2710** characters. Enclose in quotation marks.

C13. sym set Symbol set. Use **0** for the Internal Symbol Set. For scalable fonts, use:

1	ANSI Symbol Set
100	Macintosh
101	Wingdings
102	Unicode (user input) for particular mapping
103	BIG5 (user input) for Unicode mapping
104	GB2312 (user input) for Unicode mapping
105	SJIS (user input) for Unicode mapping Code Page 932 (Japanese Shift-JIS)
106	GB2312 (user input) for GB2312 mapping Code Page 936 (Simplified Chinese)
107	BIG5 (user input) for BIG5 mapping Code Page 950 (Traditional Chinese)
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)

The CG Triumvirate™ typefaces support only the ANSI and DOS Code Page 437 and 850 Symbol Sets. The scalable font (font#50) does not support Code Page 1256 (Arabic). Code pages 852-860 and 1250-1258 are for downloaded TrueType fonts or the scalable font. Code pages 102-107 require the print engine memory expansion option and a downloaded International TrueType font. TrueType fonts are designed to be regionally specific; therefore, all code pages may not be supported in a given font. See Appendix C, “Symbol Sets/ Code Pages” for more information.

Note: Only the 6032 printer supports printing International fonts.

Example

```
C,100,80,0,1,1,1,B,L,0,0,"MADE IN USA",0 ¦
```

Defines a constant text field starting at row 100, column 80. 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 **200** fields per format). You can define any line length and a thickness up to 99 dots, as long as the solid black print does not exceed 30 percent of any given square inch of the label.

Line Types

You can create horizontal and vertical lines. There are two ways to define lines.

Segments You choose the starting point and ending point.

Vectors You choose the starting point, the angle, and the length of the line.

Syntax

```
L,type,row,column,angle/end row,length/
end col,thickness,"pattern" ¦
```

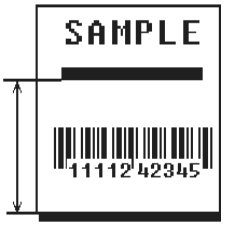
L1. L Line Field.

L2. type Type of line. Only vertical and horizontal lines are supported. Options:

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

L3. row Distance from bottom of print area to the starting point.

English **0 - 365**
Metric **0 - 927**
Dot **0 741**



L4. column Distance from left edge of the print area to line origin.

English **0 - 189**
Metric **0 - 480**
Dots **0 - 383**

L5. angle /end row

If Using Segments:

Row location of ending point. Measure from bottom of print area. Ranges same as row above. On horizontal lines, this value must match item **L3**.

If Using Vectors:
Angle of line. Options: **0**, **90**, **180**, or **270**.

L6. length/ end col

If Using Segments:

Column location of end point. Measure from left edge of print area. Ranges same as column above. On vertical lines, this value must match parameter **L4**.

If Using Vectors:
Length of the line in selected units.

Ranges for horizontal lines:

English	**0 - 189**
Metric	**0 - 480**
Dots	**0 - 383**

Ranges for vertical lines:

English	**0 - 365**
Metric	**0 - 927**
Dot	**0 - 741**

L7. thickness Using the chart below for reference, write the line thickness (**1 - 99**) in box **L7**. Measured in dots.

Dots	Thickness
1	
10	
24	
48	
96	

Note: Line thickness fills upward on horizontal lines, or to the right on vertical lines.

L8. "pattern" Line pattern. Enter "".

Example

```
L,S,110,70,110,350,10,"" ¦
```

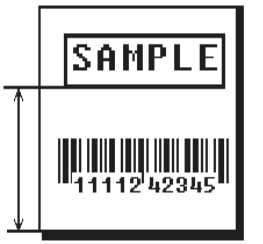
Defines a horizontal line field as a segment starting at row 110, column 70 and ending at row 110, column 350. The line thickness is 10 dots.

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 **200** fields per format). You can define any line length and a thickness up to 99 dots, as long as the solid black print does not exceed 30 percent of any given square inch of the label.

Syntax `Q,row,column,end row,end col,thickness,"pattern" ¦`

Q1. Q Box (**Q**uadrilateral) Field.

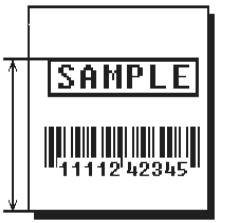


Q2. row Distance from bottom of print area to lower left corner of box.

English	**0 - 365**
Metric	**0 - 927**
Dot	**0 741**

Q3. column Distance from left edge of print area to lower left corner of box.

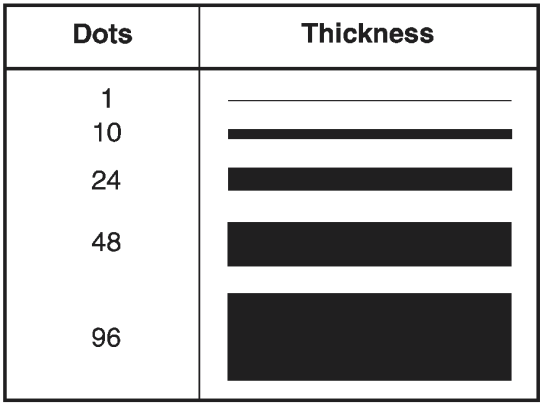
English	**0 - 189**
Metric	**0 - 480**
Dots	**0 - 383**



Q4. end row Distance from bottom of print area to upper right corner of box.
Ranges same as row.

Q5. end col Distance from left edge of print area to upper right corner of box.
Ranges same as column.

Q6. thickness Using the chart below for reference, write the desired line thickness (**1 - 99**) in box **Q6**. Measure in dots.



Note: Line thickness fills upward on horizontal lines, or to the right on vertical lines.

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

Example

```
Q,240,80,270,130,3,"" ¦
```

Defines a box field starting at row 240, column 80. It ends at row 270, column 130. It has a thickness of 3 dots.

DEFINING FIELD OPTIONS

3

This chapter provides a reference for defining

- field options in formats
- check digit packets.

Applying Field Options

Field options further define text and bar code fields. The text, constant text, or bar code field must be previously defined before you can apply any field option to it. Define options immediately after the field to which they apply.

Combining Field Options

You can use more than one option with most fields. When you use multiple options for the same field, you must place the options in the order you want to apply them to your format.

Note: Field options only apply to the 6037 printer.

Restrictions

Some options cannot be used together. See the following sections addressing individual options for specific combinations to avoid.

Example `R,1,3,1,3,1,1 ¦`

Syntax `R,option#,parameter...parameter ¦`

R1. R Indicates field option header.

R2. option# Option number:
- **1** Define Fixed Characters
- **4** Copy Data
- **30** Pad Characters
- **31** Calculate 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/Decrementing Fields
- **61** Re-Image Fields

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

Option 1 (Fixed Data)

Fixed data is information (a company name or store number) you want to print on all labels. You can 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 you are defining fixed characters for part of a field, place underscores(_) in non-fixed positions. Any spaces in the phrase are fixed characters. Range: **0 - 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 5. As an alternative, you can apply Option 4 to copy data into the non-fixed character positions.

Option 4 (Copy Data)

You can create a field that uses data from another field. This is useful for creating merged fields or sub-fields. You can copy the information from multiple fields into one field by applying the copy procedure more than once. Copy data is the only option you can apply to a field more than once.

The maximum number of characters defined in box **T3** or **B3** must allow for the number of characters you will place 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 Field Option Header.

R2. 4 Option **4**.

R3. src fld Field number from which data is copied. Range: **0 - 999**.

R4. src start Position number in the source field of the first character to be copied. Character positions are numbered **1 - 2710**, starting from the left.

R5. # to copy Number of characters to copy. Range: **1 - 2710**.

R6. dest start Position number where copied characters are to begin printing in the destination field. Range: **1 - 2710**.

R7. copy code Copy Method.

- **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

You can 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

You can 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.

Option 30 (Pad Data)

You can add characters to one side of a field to “pad” the field. Padding allows you to fill in the remaining spaces when the 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

L Pad field on left side
R Pad field on right side

R4. “character” Pad character must be within the **0 - 255** decimal range and enclosed inside quotation marks.

Note: 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

If you have a variable length bar code that you want to occupy a fixed amount of space on the supply, use pad characters. If the maximum number of characters in the bar code is 15, but the batch record only has 10 characters, the padding option fills the remainder of the field with pad characters.

Option 31 (Calculate Check Digit)

The printer generates a check digit if you apply Option 31 to the field. You cannot 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 - 10**.

Example `R,31,G,5 ¦`

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

Option 42 (Price Field)

You can apply options that will insert monetary symbols automatically. 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,appearance code ¦`

R1. R Option Header.

R2. 42 Option **42**.

R3. appearance code Enter **1** to print price field in standard notation, as defined by 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 you want to use 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.

Option 50 (Bar Code Density)

You can apply this option to bar code fields when you want to create custom densities. When you apply this option, it overrides the density value in the bar code field.

Do not use Option 50 with UPC or EAN bar codes. When using this option, set the density parameter in your bar code field to the default value. You can only use this option once for each bar code field. This option overrides the density selected in the bar code field.

Bar codes produced using Option 50 may not be scannable. The additional character gap, narrow space, and wide space parameters are valid **only** with Code 39 and Codabar. If these parameters are specified for any other bar codes, they will be ignored by the printer.

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 - 99**.

R4. wide Dot width of the wide element. Range: **1 - 99**.

R5. gap Additional dot space between characters. Enter a value of **1 - 99**. (Code 39 and Codabar only.)

R6. nar_space Additional dot width of the narrow bar code space. (Code 39 and Codabar only). Range: **1 - 99**.

R7. wide_space Additional dot width of the wide bar code space. (Code 39 and Codabar only). Range: **1 - 99**.

Example `R,50,4,8,4,4,8 ¦`

Creates a custom bar code density with a narrow element of 4 dots, a wide element of 8 dots, a gap of 4 dots, 4 additional dot widths for the narrow bar code space, and 8 additonal dot widths for the wide bar code space (if this is a Code 39 or Codabar bar code).

Option 51 (PDF417 Security/Truncation)

You can 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. You can 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 your PDF417 bar code. For each level increased, the bar code will double in size.

Syntax `R,51,security,stand/default ¦`

R1. R Option Header.

R2. 51 Indicates Option **51**.

R3. security Security level ranges from **0-8** (**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. Valid values:

S (default) a standard PDF417 bar code
T truncated

Example `R,51,2,S ¦`

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

Option 52 (PDF417 Width/Length)

This option defines the image width or length of a PDF417 bar code. If you define a fixed number of columns (width), the bar code expands in length. If you define 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. You can only 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 if you are defining the number of rows or columns.

R Row
C Column

If you specify rows, the bar code expands in columns, or vice versa.

R4. dimension The number of rows or columns defined for the bar code. The default is **4**. Valid values:

3-90 for rows
1-30 for columns

Example `R,52,C,10 ¦`

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

Option 60 (Incrementing/Decrementing Fields)

You may have an application, such as serial numbers, in which you need a numeric field to increment (increase in value) or decrement (decrease in value) on successive tickets within a single batch. Incrementing or decrementing can be applied to **numeric** data only. If you have a field that 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
I incrementing field
D decrementing field

R4. amount Amount to increase or decrease. Range: **0 - 999**.

R5. l pos Leftmost position in inc/dec portion of field. If this value is not entered, the default value 1 is used. Range: **0 - 2710**.

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 - 2710**.

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. You can use batch data or use 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**.

Option 61 (Re-image Field)

If you have a constant field that appears on top of a non-constant field, apply this option to the constant field. For example, you may create a tag for an item on sale. The tag shows both the old and the new prices, but has a line drawn through the old price.

The constant field can be a line, a box, a graphic, or constant text. This option redraws the constant field when the other field changes. If you do not use this option, the field may appear broken.

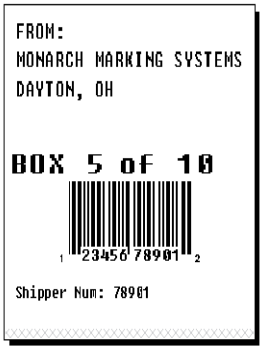
Syntax `R,61 ¦`

R1. R Option Header.

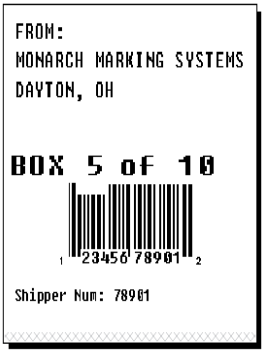
R2. 61 Option **61**.

Example `R,61`

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



Re-Image ON



Re-Image OFF

In the above example, Option 61 was applied to the bar code field to keep the incrementing field (Box #) from blocking out the bar code field.

Using Check Digits

Check digits are typically used to ensure that a text or bar code field scans correctly. If you apply Option 31, the printer calculates a check digit. A check digit scheme determines how the printer calculates a check digit. When you define a check digit scheme, you assign a number to identify it. This number is later entered in box **R4** when you apply Option 31 to a field. You can 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 - 10** to this check digit formula.

A3. action — The action to perform. Enter **A** to add the check digit scheme.

A4. device — Format storage device. Use **R**.

A5. modulus — Number from **2 - 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 - 2710**.

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 - 2710**.

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

4

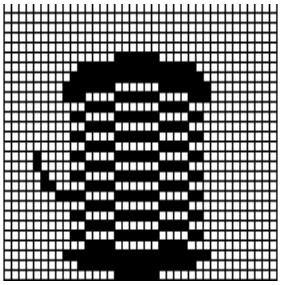
This chapter provides information on how to

- 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.

You can use graphic packets to create bitmapped images. To include a graphic packet within your format, your 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. You can print varying 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

You can use one of two methods to map out your 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.

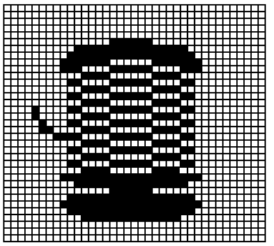
Run Length Encoding Method — The dot sequences are segmented into black and white strings within a row. The total count of each white string is converted to a lower-case letter, corresponding to numeric value. The total count of each black string is converted to an uppercase letter, corresponding to 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 your graphic image is and whether or not imaging time is a concern. You may want to experiment with both encoding methods to get optimal performance.

Designing Bitmapped Images

Once you determine the encoding method to use, you can begin mapping out your graphic image.

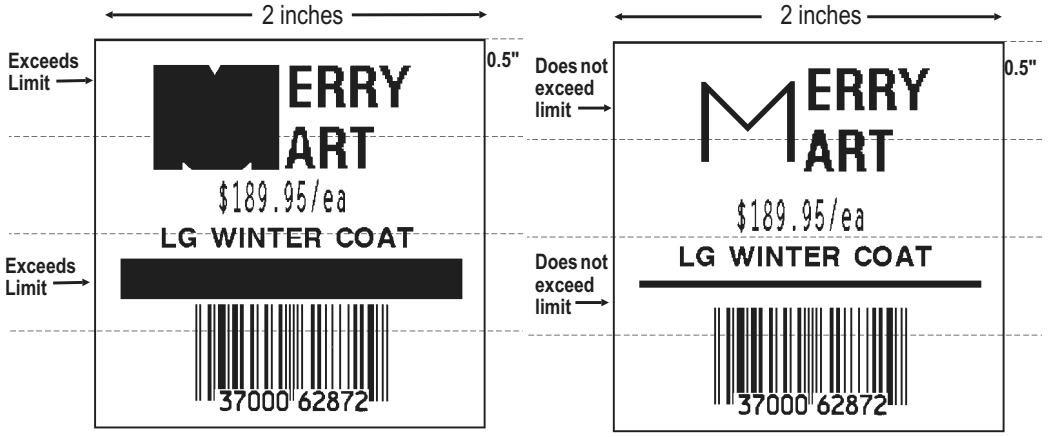
Note: The image that you map must be an upside down mirror image of the final result.



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.

1. Assign 1 to every black square and 0 to every white square.

```
00000000000000000000000000000000000000000000000000111111111111111111111111110000000000000000000000000000
00000000000000000000000000000000000000011111111111000000000000000000000000001111111110000000000000000000
00000000000000000000000000000000001111100000000000000000000000000000000000000000000001111110000000000000
00000000000000000000000000000011110000000000000000111111111111111111111111110000000000000000000111100000000
00000000000000000000000001111100001111111111111111111111111111111111111111111111111111111111111000011111110000
00000000000000000000000110000011111111111111111111111111111111111111111111111111111111111111111111000001100
00000000000000000000000110001111111111111111111111111111111111111111111111111111111111111111111111111111110
00000000000000000000000111111111111111111111111111111111111111111111111111111111111111111111111111111111110
00000000000000000000000111111111111111111111111111111000000000000000000111111111111111111111111111111110
00000000000000000000000111111111111111111111000000000000000000000000000000000000011111111111111111111100
00000000000000000000000000111111110000000000000000000111111111111111111000000000000000000011111111110000
00000000000000000000000000011111110000000000000000000111111111111111111000000000000000000001111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111111000000011100000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111111000011100000
00000000000000000000000000000011111111111111000000000000000000000000000000000000001111111111111100000000
00000000000000000000000000011111110000000000000000000111111111111111111000000000000000000001111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111111000000011100000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111111000011100000
00000000000000000000000000000011111111111111000000000000000000000000000000000000001111111111111100000000
00000000000000000000000000011111110000000000000000000111111111111111111000000000000000000001111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111111000000011100000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111111000011100000
00000000000000000000000000000011111111111111000000000000000000000000000000000000001111111111111100000000
00000000000000000000000000011111110000000000000000000111111111111111111000000000000000000001111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111111000000011100000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111111000011100000
00000000000000000000000000000011111111111111000000000000000000000000000000000000001111111111111100000000
00000000000000000000000000011111110000000000000000000111111111111111111000000000000000000001111111000000
00000000000000000000000000011100000000111111111111111111111111111111111111111111111111111000000011100000
00000000000000000000000000000000001111111111111111111111111111111111111111111111111111111111000011100000
00000000000000000000000000000011111111111111000000000000000000000000000000000000001111111111111100000000
00000010000000000000000000011111110000000000000000000111111111111111111000000000000000000001111111000000
00000011000000000000000000011100000000111111111111111111111111111111111111111111111111111000000011100000
00000001111000000000000000000000001111111111111111111111111111111111111111111111111111111111000011100000
00000000111000000000000000000011111111111111000000000000000000000000000000000000001111111111111100000000
00000000111111000000000000011100000000111111111111111111111111111111111111111111111111111000000111100000
00000000111111100000000000000000001111111111111111111111111111111111111111111111111111111111000111100000
```

2. Section off the grid in columns of eight. If any rows are not divisible by 8, add enough 0’s to complete a column.

```
00000000 00000000 00000000 00000000 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 00000111 11100000 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
```

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.

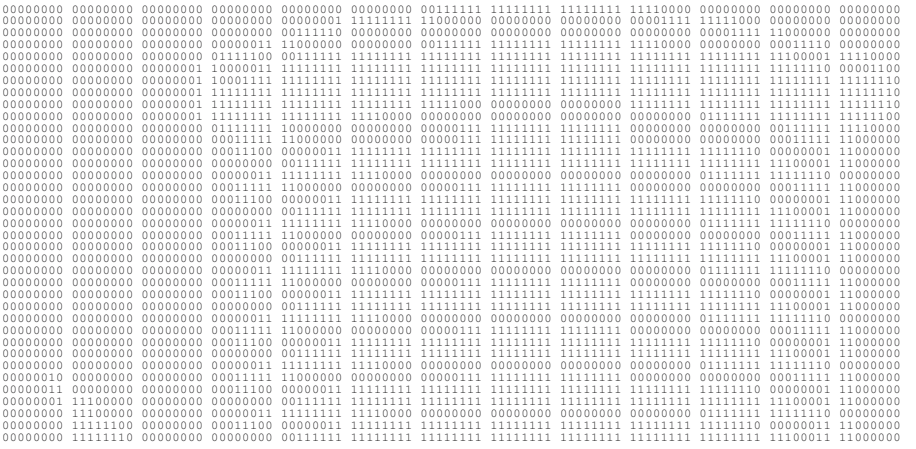
6. Insert the hex values in syntax format.

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: For visual clarity, the following example shows “1" to indicate when a square is ON, and ”0" to indicate when a square is OFF. You do not have to convert your dots when using the run length method.



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 you have written 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 you have mapped out your graphic image, determine how you want to store it. You have three options:

- RAM
- Temporary Storage
- Flash

Using RAM

You should use RAM when the graphic image is used by several formats, because you only have to send the graphic image 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 smaller than approximately 1/2 inch by 1/2 inch can be stored in printer RAM and referenced by the graphic ID number.

Note: Graphics are stored in the image buffer and remain there until another format is sent or the printer is turned off.

Using Temporary Storage

You should use temporary storage when the graphic image is used only in one format or your graphic image is very large. Graphic data in temporary storage is held in the image buffer until the graphic is printed. Then, it is cleared from memory when you send a new or updated batch. You can 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 causes an error. Instead, position the graphic image by using the row and column locations in the graphic packet header.

Image memory (temporary storage) accepts a graphic packet 2436 rows long with 384 dots per row.

Using Flash

You can use flash memory when the graphic image is used by several formats, because you only have to send the graphic image 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.

Creating a Graphic Packet

Your 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 your bitmapped image.

Once you design your graphic image, you are ready to define a graphic packet. This packet generates the graphic image you use in a format.

Positioning the Graphic Image

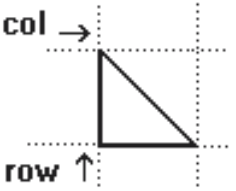
This section explains how to position the graphic image within a graphic packet header, a field of a graphic packet, or within a format.

Within the Graphic Packet Header

When you are using RAM, the row and column parameters in the graphic header are usually 0,0, because placement is controlled by the graphic field in your format.

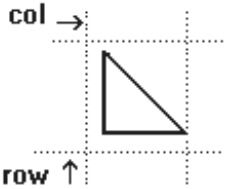
When you are using temporary storage, these parameters control the placement of the graphic image on the supply.

The area enclosed within the dotted lines represents the graphic image starting at **0,0** (as defined in the graphic header).



If you want a fixed amount of white space around your graphic image, use something other than 0 for row and/or column.

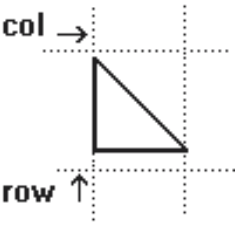
The area enclosed within the dotted lines represents the graphic image starting at **0,0** with a fixed amount of white space (**10,10**) around the graphic image.



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 you define the graphic field within your format, the row and column parameters represent where on the format to place the graphic image.

If you are placing a graphic (a logo, for example) within a certain area on your 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. This is the first thing you enter. 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 **0 - 999** to identify the graphic image.

G3. action The action to perform to the graphic. Options:
A Add the graphic.
C Clear the graphic from the printer.

G4. device Graphic storage device:

R Volatile RAM (format must contain a graphic field)
T Temporary Storage
F Flash Memory

Note: If you are using flash memory, send the graphic only once. Then, send only batches to reference it. If you re-send the graphic each time, it creates separate copies, and eventually fills up flash memory.

G5. units Unit of measure. Options:

E English, measured in 1/100 inches
M Metric, measured in 1/10 mm
G Graphic, measured in dots. For bitmapped graphics, G is the only valid option.

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. See “Positioning the Graphic Image,” for more information.

English	**0 - 365**
Metric	**0 - 927**
Dot	**0 741**

G7. column 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. See “Positioning the Graphic Image,” for more information.

English	**0 - 189**
Metric	**0 - 480**
Dots	**0 - 383**

G8. mode Imaging mode. Enter **0**.

G9. “name” Graphic name (optional), **0-8** characters, enclose in quotation marks.

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. A bitmap field can later be repeated by using a duplicate field.

Syntax `B,row,column,algorithm,"data" ¦`

B1. B Bitmap Field.

B2. row Distance (in dots) from the graphic image's bottom margin to the bitmap line.

English	**0 - 365**
Metric	**0 - 927**
Dot	**0 741**

B3. column Distance (in dots) from the graphic image's left edge to the bitmap line.

English	**0 - 189**
Metric	**0 - 480**
Dots	**0 - 383**

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 - 2710**.

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. It allows you to 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 Increments (inserts after)
1 Decrements (inserts 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 — Amount of row adjustment in dot rows. Using **0** overwrites the same line. Range: **0 - 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 - 2710**.

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 you to repeat the dot sequence without retyping the data. A duplicate field represents one row of dots on the image.

Note: Duplicate fields are useful when you have a graphic with 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 - 999**. 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 - 999**.

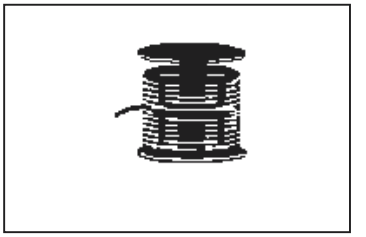
Example

```
B,117,24,H,"03FFFFFFFFFFFFFFFFFFC" ¦
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).

You can use constant text, line, or box fields in a graphic packet to create a compliance label overlay. See Chapter 2, “Defining Fields,” for more information about these fields.

Sample Hex Graphic Packet

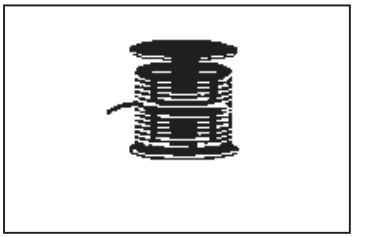


```
{G,99,A,R,G,0,0,0,"99WIRE"¦
B,39,48,H,"3FFFFFF0" ¦
B,40,32,H,"01FFC000000FF8"¦
B,41,32,H,"3E00000000000FC0" ¦
B,42,24,H,"03C0003FFFFFF0000F"¦
B,43,24,H,"7C3FFFFFFFFFFFFE1F0" ¦
B,44,16,H,"0183FFFFFFFFFFFFFF06" ¦
B,45,16,H,"018FFFFFFFFFFFFFFFFFE" ¦
B,46,16,H,"01FFFFFFFFFFFFFFFFFFE"¦
B,47,16,H,"01FFFFFF80001FFFFFFFE" ¦
B,48,16,H,"01FFFFF0000000007FFFFC" ¦
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,"FFDF000FFFFFFFFF8003C0" ¦
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,"03FFFFFFFFFFFFFF0" ¦
```

```
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,"0187FFFFFFFFFFFFFFC18" ¦
B,116,16,H,"027FFFFFFFFFFFFFFFFF2" ¦
B,117,16,H,"03FFFFFFFFFFFFFFFFFFC" ¦
D,0,1,2 ¦
B,120,16,H,"01FFFFFFFFFFFFFFFFFF8"¦
B,121,24,H,"FEFFFFFFFFFFFFFFFFE0"¦
B,122,24,H,"07FFFFFFFFFFFFFFC" ¦
B,123,32,H,"FFFFFFFFFFFFFFC0"  ¦
B,124,32,H,"01FFFFFFFFFF8" ¦ }
```

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,"TzkU" ¦
B,49,25,"HtRqJ" ¦
B,50,27,"GsSsG" ¦
D,0,4,4 ¦
B,51,27,"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 you are using RAM, place a graphic field in the format file to reference the graphic. See the following section, “Defining the Graphic Field,” for more information.

Note: If you are using temporary storage, you do not need a graphic field in your format to reference the graphic image.

3. Download all the necessary packets (check digit, format, etc.).
4. Send the graphic file to the printer, if you have not already done so. 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 **0 - 999** to identify the graphic image.

G3. row Distance between the bottom of the print area on the supply to the bottom of the graphic image. Measured in selected units.

English	**0 - 365**
Metric	**0 - 927**
Dot	**0 741**

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. Measured in selected units.

English	**0 - 189**
Metric	**0 - 480**
Dots	**0 - 383**

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.

G5. mode Imaging mode. Enter **0**.

G6. rotation The orientation of the graphic on the supply. Enter **0**.

Example

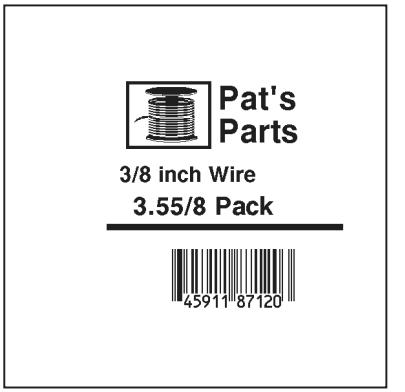
```
G,57,0,0,0,0 ¦
```

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 graphic packets (hex and run length) in a sample format.

```
{F,2,A,R,E,200,200,"FMT2"| ¦
G,99,132,30,0,0 ¦
Q,146,30,198,95,5,""  ¦
T,1,5,V,175,100,0,1004,1,1,B,L,0,0,0 ¦
T,2,5,V,150,100,0,1004,1,1,B,L,0,0,0 ¦
T,3,15,V,120,25,0,1003,1,1,B,L,0,0,0 ¦
T,4,15,V,95,35,0,1003,1,1,B,L,0,0,0 ¦
L,S,84,15,84,195,10,"" ¦
B,5,12,F,30,65,1,2,40,1,L,0 ¦ }
```



PRINTING

5

This chapter describes how to define the batch header, batch control, and batch data files.

Defining the Batch Header

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 contains the following three parts:

batch header identifies the format and how many labels to print.

batch control defines the print job.

batch data (optional) defines the actual information printed on the label.

A batch header begins the file. It tells which format the batch uses and how many labels to print. To record batch data, make a copy of the worksheet in Appendix D, "Format Design Tools."

Syntax `{B,format#,N/U,quantity ¦`

B1. B Batch Header.

B2. format# Format number (**0-999**) to use.

B3. N/U Controls how image is generated.

- **N** New (default). 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 (**1-999).**

Note: 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,1 ¦`

Defines a batch header that uses format #1 and reimages all fields using the online data. One label is printed with 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 ¦`

E1. E Batch Control Field.

E2. feed_mode Feed Mode. Options:

0 Continuous Feed (default)
1 On-Demand

E3. batch_sep Batch Separator. Use **0**.

E4. print_mult Number of tags (**1 - 24**) with the same image.

E5. multi_part Number of identical parts on one tag (**1 - 5**).

Example `E,0,0,1,1 ¦`

Defines a batch control field. Continuous feed mode is used and no separator prints between batches.

Defining Batch Data Fields

Batch data fields should be sent in field number order. Use continuation fields for large amounts of data. If you are using N (New) in the batch header, you must list all fields with your data in sequence. If you are using U, you need to 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: **0 - 999**.

"data string" Provides the actual information to appear in fields. Enclose in quotation marks. Length: **0 - 2710** characters.

C Optional. Identifies information to be appended to the data string.

"continuation" Optional. 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 - 2710** characters.

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, you can use “” or ~034 to print the “ character in your 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 data string.

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 - 999**, enter the starting number in the batch as **001**.

Downloading Methods

You can download the format and batch data using one of three methods: sequential, batch, and batch quantity zero.

Sequential Method

Using the sequential method, you send all your format and batch data at one time. Use this method when your 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, your labels begin to print.

Example

```
{Format}
{Batch Packet}
```

Batch Method

This is similar to the sequential method, but it is used when you want 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

You may use the batch quantity zero method when your 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 you send the batch header, 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 you want printed. When the printer receives input for the last field, it immediately prints the labels. See “Reducing Imaging Time” in Chapter 7 for an example using the batch quantity zero method.

Modifying Formats

The optional entry method is a quick way to modify your format fields, check digit fields and configuration packets.

Optional Entry Method

This method enables you to reset only the parameters you want to 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.

TROUBLESHOOTING

6

This chapter explains how to

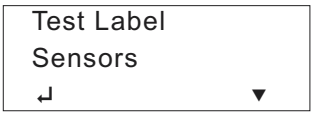
- print diagnostics labels
- reset the printer
- call Technical Support.

This chapter also provides explanations of your printer's errors. The errors are classified by type and are listed in order. If you have trouble loading supplies or performing maintenance, refer to your *Equipment Manual.*

Follow the directions provided with the error description to correct the problem. If you cannot clear an error, turn off the printer, wait several seconds and then turn on the printer. Call Technical Support if you receive any error message not listed in this chapter.

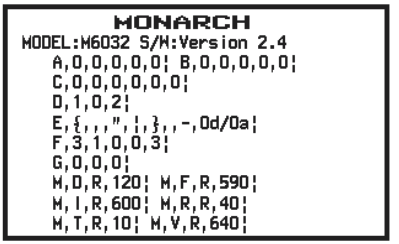
Printing Diagnostics Labels

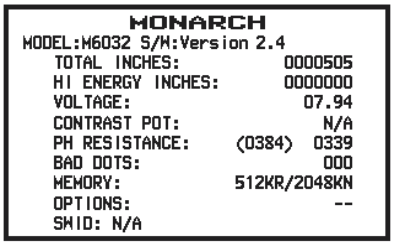
1. From the Diagnostics menu, select Printer. You see



2. From the Printer menu, select Test Label.
3. After you select Test Label, you see:

4. From the Test Label menu, select Diag Label.





Two diagnostic labels print and you return to the Test Label menu.

Refer to the *Equipment Manual* to print the checkerboard, grey scale, and other test labels.
With version 3.0 or greater software, the 6032 printer has International font capability.

Reading Diagnostics Labels

The first diagnostic (test) label shows the model number, software version, and the printer's configuration by packet.

The second diagnostic (test) label shows the model number, software version, total number of inches printed, voltage, print contrast, printhead resistance, number of bad dots, and memory. In the example above:

- The printer has printed 505 inches of regular supplies
- The printer has printed no high-energy (synthetic) supplies
- No options are set
- CONTRAST POT does not apply.

If You Receive an Error Message

Any time you receive a message that is not described in this manual, or the recommended action does not solve the problem, call Technical Support.

Calling Technical Support

Technical support representatives are available Monday through Friday during regular business hours. Their number is listed on the back cover of this manual. Follow these steps before you call:

1. Make sure your PC and printer are properly connected.
2. Record any error messages that occurred.
3. Recreate the problem, if you can.
4. Check your port settings. Your problem may corrected simply by changing the communication settings.
5. List any changes that have recently been made to the system. Try to record what you were doing when the problem occurred.
6. Reboot your computer. Refer to your computer documentation for specific instructions.
7. Print the Diagnostics labels. See "Printing Diagnostics Labels" for more information.

If these steps do not solve the problem, call Technical Support.

Have the following information ready before you call:

- computer brand name and type of software or terminal brand name and model
- Monarch printer model
- printer serial number
- support agreement, contract number, or invoice information
- customer number

Data Errors

A data error indicates that incorrect data was received from the host, causing the printer to ignore the entire print job. After checking the packet and correcting the problem, transmit the print job again.

The following is a list of data errors. These errors occur because data in the format, batch, check digit, font, or graphic packet is invalid.

Format Errors

Error Code	Description
001	Format ID number must be **1 - 999**.
002	Name must be **1 - 8** characters inside quotes or a printer-assigned name (“”).
003	Action must be **A** (add) or **C** (clear).
004	Supply length is invalid (maximum is **4**"). See “Defining the Format Header” in Chapter 2 for valid lengths.
005	Supply width is invalid (maximum is **2**"). See “Defining the Format Header” in Chapter 2 for valid widths.
006	Storage device must be **R** (volatile RAM), **T** (temporary storage), or **F** (flash memory).

007 Unit of measure must be **E** (English), **M** (Metric), or **G** (Dots). See "Defining the Format Header" in Chapter 2 for more information.

010 Field ID number is outside the range **0 - 999**.

011 Field length exceeds **200**.

012 Row field position is greater than the maximum stock dimension. See "Defining Text Fields" in Chapter 2 for valid row lengths.

013 Column field position is greater than the maximum stock dimension. See "Defining Text Fields" in Chapter 2 for valid column widths.

014 Font style is invalid. Must be **1, 2, 3, 4, 5, 6, 10, 11, 50, 1000, 1001, 1002, 1003, 1004, 1005, 1006, 1007, 1008, 1009, 1010, 1011, 1012,** or **1013**. See "Defining Text Fields" in Chapter 2 or Appendix B, "Fonts," for more information.

015 Character rotation must be **0** (0 degree), **1** (90 degree), **2** (180 degree), or **3** (270 degree). See "Defining Text Fields" in Chapter 2 for more information.

016 Field rotation must be **0** (0 degree), **1** (90 degree), **2** (180 degree), or **3** (270 degree). See "Defining Text Fields" in Chapter 2 for more information.

017 Field restriction must be **V** (variable) or **F** (fixed).

018 Text field symbol set selection defined in the field must be **0** (Internal), **1** (ANSI), **100** (Macintosh), **101** (Wingdings), **102** (Unicode), **103** (BIG5 for Unicode), **104** (GB2312 for Unicode), **105** (SJIS for Unicode), **106** (GB2312), **107** (BIG5), **437** (DOS Page 437), **850** (DOS Page 850), **852** (Latin 2), **855** (Russian), **857** (Turkish), **860** (Portuguese), **1250** (Latin 2), **1251** (Cyrillic), **1252** (Latin 1), **1253** (Greek), **1254** (Turkish), **1255** (Hebrew), **1256** (Arabic), **1257** (Baltic), or **1258** (Vietnamese). See "Defining Text Fields" in Chapter 2 for more information.

020 Vertical magnification must be **1 - 7** or for TrueType/Scalable fonts, use **4 - 250** (the point size).

021 Horizontal magnification must be **1 - 7** or for TrueType/Scalable fonts, use **4 - 250** (the point size).

022 Color must be **A, B, D, E, F, N, O, R, S, T,** or **W**. See “Defining Text Fields” in Chapter 2 for more information.

023 Intercharacter gap must be **0 - 99** dots.

024 Field justification must be **B** (balanced), **C** (centered), **E** (end), **L** (left), or **R** (right). See “Defining Text Fields” in Chapter 2 for more information.

025 Data length is too long.

030 Bar code height must be at least **1** (English), **2** (Metric), **1** (Dots), or is not within the supply dimensions.

031 Human readable option must be

1 no CD or NS
5 NS at bottom, no CD
6 CD at bottom, no NS
7 CD and NS at bottom
8 no text

032 Bar code type is invalid. See “Defining Bar Code Fields” in Chapter 2 for valid options.

033 Bar code density is invalid. See “Defining Bar Code Fields” in Chapter 2 for the bar code density chart.

040 Line thickness must be **0 - 99** dots.

041 Line direction must be **0, 90, 180,** or **270**.

042 End row is invalid. Line segment or box end row is defined outside of printable area.

043 End column is invalid. Line segment or box end column is defined outside of printable area.

044 Dot pattern for line or box must be “”.

045 Line length is defined beyond the maximum length. See “Defining Line Fields” in Chapter 2 for valid lengths.

046	Line type must be **S** (segment) or **V** (vector).
051	Imaging mode in the graphic header must be **0**.

Batch Errors

Error Code	Description
101	The format referenced by batch is not in memory.
102	Print quantity is outside the range **0 - 999**.
104	Batch mode must be **N** (new) or **U** (update).
105	Batch separator in a batch control field must be **0** (off).
106	Print multiple is outside the range **1 - 24**.
108	Multiple part supply is outside the range **1 - 5**.

Option Errors

Error Code	Description
200	Option number must be **1, 4, 30, 31, 42, 50, 51, 52, 60**, or **61**.
201	Copy length is outside the range **1 - 2710**.
202	Copy start position must be **1 - 255**.
203	Destination start position must be **1 - 255**.
204	Source field must be **0 - 999**.
205	Copy type must be **1** (copy after rules) or **2** (copy before rules).
206	Increment/Decrement selection must be **I** (increment) or **D** (decrement).

207 Incrementing start position must be **0 - 2710**.

208 Incrementing end position must be **0 - 2710**.

209 The incrementing amount must be **0 - 999**.

210 Security value for a PDF-417 bar code must be **0 - 8**. Correct the value and resend the format to the printer.

211 Narrow element value is less than **1** or greater than **99**. Correct the value and resend the format to the printer.

212 Wide element value is less than **1** or greater than **99**. Correct the value and resend the format to the printer.

214 Truncation code must be **S** (standard) or **T** (truncated bar code).

215 Aspect code must be **C** (columns) or **R** (rows).

216 Option definition must be **S** (set) or **T** (template).

217 Input device device must be **D** (Default), **H** (Host), **K** (Keyboard), **N** (None), or **S** (Scanner).

218 Pad direction must be **L** (from left) or **R** (from right).

219 Pad character is outside the range **0 - 255**.

220 Check digit selection must be **G** to generate check digit.

221 Primary or secondary price format is outside the range **1 - 16**.

222 Data type restriction is outside the range of **1 - 6**.

223 Option is not valid for the field.

224 Bar code intercharacter gap must be **0 - 99** dots. (This is also known as the additional character gap when using Option 50, Defining Bar Code Densities.)

Online Configuration Errors

Error Code	Description
251	Power up mode must be **0** (online).
252	Language selection must be **0** (English).
253	Batch separator code in a system setup packet must be **0** (off).
254	Slash zero selection must be **0** (standard zero) or **1** (slashed zero).
255	Supply type must be **0** (black mark), **1** (die cut), or **2** (non-indexed).
256	Printhead energy must be **0** (Standard supplies), **1** (Fax paper), or **2** (Synthetic supplies).
257	Feed mode must be **0** (continuous) or **1** (on-demand, if option is available).
258	Supply position is outside the range.
259	Contrast adjustment must be **28 - 40** dots.
260	Print adjustment must be **-99 - 99** dots.
261	Margin adjustment must be **-99 - 99** dots.
262	Speed adjustment must be **0**.
263	Primary monetary symbol is invalid.
264	Secondary symbol selection must be **0** (none) or **1** (print secondary sign).
265	Monetary decimal places must be **0 - 3**.
266	Character string length in the control characters packet must be **5** (MPCL control characters) or **7** (ENQ/IMD command character).
267	Baud rate selection must be **7** (115200).

268 Word length selection must be **1** (8 bits).

269 Stop bits selection must be **0** (1 bits).

270 Parity selection must be **0** (none).

271 Flow control selection must be **3** (XON/XOFF).

272 Symbol set (code page) in the System Setup packet must be **0** (Internal), **1** (ANSI), **2** (DOS 437), **3** (DOS 850), **4** (1250- Latin 2), **5** (1251- Cyrillic), **6** (1252- Latin 1), **7** (1253- Greek), **8** (1254- Turkish), **9** (1255- Hebrew), **10** (1256- Arabic), **11** (1257- Baltic), **12** (1258- Vietnamese), **13** (852- Latin 2), **14** (855- Russian), **15** (857- IBM Turkish), **16** (860- DOS Portuguese), **17** (Wingdings), **18** (Macintosh), **19** (Unicode), **20** (BIG5), **21** (GB), **22** (SJIS to SJIS), **23** (GB to GB), **24** (BIG5 to BIG5), or **25** (KSC to KSC).

282 RS232 Trailer string is too long. Use a maximum of **3** characters.

283 ENQ Trailer string is too long. Use a maximum of **3** characters.

284 The buffer type must be **T** (Transmit), **R** (Receive), **I** (Image), **F** (Format), or **D** (Downloadable Fonts).

285 The storage device type in the memory configuration packet must be **R** (volatile RAM).

286 The buffer size is invalid.

287 The printhead width must be **0**.

288 Invalid battery voltage packet. Must be **0** or **1**.

289 Invalid printer address in communication settings packet. Use **0** or none.

Check Digit Errors

Error Code	Description
310	Check digit scheme number must be **1 - 10**.
311	Modulus must be **2 - 11**.
314	Check digit algorithm must be **D** (sum of digits) or **P** (sum of products).

Graphic Errors

Error Code	Description
325	Duplicating direction must be **0** (insert after) or **1** (insert before) in duplicate fields for graphics.
327	Amount of row adjustment must be **0 - 999** dots in duplicate fields for graphics.
328	Duplicate count must be **0 - 999**.
340	Bitmap line encoding must be **H** (hex) or **R** (run length).
350	Font selector must be **1 - 9999**.
351	Font data length must be **68 - 16384**.
352	Insufficient font memory is available for the downloaded font.
380	Job request is outside the range **0 - 4**.
400	The character immediately following **{** is invalid.
401	Internal software error. Call Technical Support.
402	Field separator not in expected location.
403	Field separator not found.

404 The number or string that is currently being processed is too long.

405 Too many fields exist in the format. You cannot have more than **200** fields in the format. Lines and box fields count as fields.

Communication Errors

Error Code	Description
409	The printer memory is full. Delete unnecessary formats or graphics from memory. If you are using a graphic file that is very large, consider using another mapping method (such as run length encoding) to reduce the required memory.
410	Parity on the printer does not match the parity on the host. Check the parity setting under SETUP options.
411	Framing error. The printer cannot communicate with the host. Make sure the host is turned on, communication cables are connected correctly, port settings are correct, and communications are active. Check the baud rate, word length, and stop bits to make sure they match those at the host. Do not toggle between Microsoft Windows and MS-DOS while using the COPY command, or you will receive a framing error. Exit Windows **before** using the COPY command. Re-transmit the data.
412	There is a problem with flow control between the printer and the host. Make sure the printer and the host flow control settings match (both are DTR or both are XON/XOFF). If the error persists, call Technical Support.
413	Online receive queue is full. Check your printer's XON/XOFF or DTR SETUP values to be sure there isn't a flow control problem.
414	The internal keyboard buffer is full or you need a new keypad. Call Technical Support.
415	The buffer size you defined exceeds the total available in your machine.
423	Internal software error. Call Technical Support.

Data Formatting Failures

Formatting errors indicate that a field will print incorrectly. After you have checked the data stream and corrected the data, retransmit the format and batch.

Note: For these errors, the batch still prints, but the field, font, bar code, or density may be incomplete, missing or contain incorrect data.

571 UPC or EAN bar code data length is invalid. The bar code data length in the batch does not fit the format.

572 Invalid copy field, padded field, or incrementing field length. The field length in the batch does not fit the format or the field contains blanks. Or, the fixed length field does not contain the specified number of characters.

573 Invalid price length. The price field length in the batch does not fit the format or the field contains blanks.

574 No CD scheme or room for CD. The CD scheme in the batch does not fit the format or the field contains blanks.

575 The graphic included in your format could not be found. Resend the graphic packet.

601 An error occurred while the batch was imaging. Resend the format, batch, font, and/or graphic packet.

603 The batch was not found for imaging. Resend the batch packet.

611 Font, bar code or density is invalid. The font, bar code or density in the batch does not fit the format.

612 The data in this line of the batch is either missing or does not match the format.

613 Reference point off tag.

614 Portion of field off tag. There may be an invalid character in the packet. Make sure you did not enter **O** for **Ø**.

615 Bar code width is greater than 16 inches, or number of keywords for PDF 417 exceeds **928**. Decrease the density or shorten the amount of data to print the bar code.

616 Dot shifting failed. A bad dot falls on a bar code that cannot be shifted. Call Technical Support.

618 Magnification must be **1 - 7** or **4 - 250** for the scalable font.

621 An error occurred opening the TrueType font file. Select a different font to download. If the error message persists, call Technical Support.

622 Not enough memory to create the downloaded TrueType characters in the scalable (vector) fonts buffer. Reconfigure the printer's memory and increase the scalable (vector) fonts buffer. Resend the font, format, and batch packet after reconfiguring the memory. If the error message persists, call Technical Support.

Machine Faults

These errors occur when there is a problem with the printer.

703 The printer sensed a calibration of different-sized black marks. Make sure the correct supply is loaded.

704 Printer has not sensed a supply mark within the specified number of inches or is out of supplies. Check the supply tracking, supply marks, black mark sensor position, and supply roll for binding. Change supply.

706 The motor could not get up to speed within the allotted time. Call Technical Support.

707 The motor reached the desired speed, but was unable to maintain that speed. Call Technical Support.

708 The motor has stalled or is running too slow to measure the speed. Call Technical Support.

750 Printhead is overheated. Turn off the printer to let the printhead cool. If the error persists, call Technical Support.

751 Printer did not sense a black mark when expected. The supply may be jammed.

For these errors, Check the

- supply tracking
- supply marks
- black mark sensor position
- supply roll for binding.

If the error continues to appear, change the supply.

752 Printer sensed a mark in the wrong place.

753 Printer sensed a mark that is too long.

756 The printer is out of supplies. Load supplies.

757 Load supplies. The calibrated supply length differs by plus or minus 0.25 inches from the format.

758 Check supply. Either the supply is not seen, or the on-demand sensor is broken (purchase optional). Check for a label jam. Clear the supply path or reload supplies. This error may occur if you remove a label too quickly in on-demand mode. The printer does not recalibrate after this error.

762 Low battery. Recharge the battery.

763 Waiting to dispense label.

765 Printhead has less than **3** bad dots.

768 Printhead has more than **10** bad dots or is not connected. Make sure the printhead is connected.

790 The printer is busy. Turn off the printer. Wait two seconds and turn it back on. Resend the packets. If the problem continues, call Technical Support.

791 The printer has an error pending. Turn off the printer. Wait two seconds and turn it back on. Resend the packets. If the problem continues, call Technical Support.

792 Printer not initialized.

793 Printer job queue full.

Flash Memory Errors

These errors occur when there is a problem with the printer's flash memory.

800 A directory in flash memory is full.

801 Flash memory is full.

802 A directory in flash memory cannot be found.

803 There is no directory in flash memory.

810 A file in flash memory is not open.

811 A file in flash memory is already open.

812 The file in flash memory is full.

813 You cannot access flash memory.

820 An error occurred in the flash program.

821 An error occurred while erasing flash memory.

822 There is a flash ID error.

Memory Expansion Errors

These errors occur when there is a problem with the Print Engine memory expansion option.

860 There is an error opening a font file on the print engine memory expansion option.

861 There is an error when writing to a file on the print engine memory expansion option.

862 There is an invalid condition for the print engine memory expansion option.

Hard Printer Failure Errors

These errors are hard printer failures. Call Technical Support if you receive these messages.

904 No memory for native layer.

907 Low RAM error.

909 RAM corrupted.

911 Version string mismatch.

SYSTEM ERROR VECTOR ## ## can be:

- **2** Bus Error
- **3** Address Error
- **4** Illegal Instruction
- **5** Zero Division
- **6** CHK, CHK2 Instructions
- **7** TRAP Instructions
- **8** Privilege Violation
- **9** Trace
- **10** Line 1010 Emulator
- **48** User-defined vectors (48-255)

This error is fatal. If you receive it, reset and/or reflash the printer. If the error persists, call us at the number listed on the back cover of this manual for instructions. If they determine you should send the printer back, use the original packaging (box and packing material) and include the documentation. Use the following address:

Electronic Repair Center
200 Monarch Lane
Door 39
Miamisburg, OH 45342

The warranty does not apply if you do not follow these instructions.

PRINTER OPTIMIZATION

7

This chapter provides information on how to improve your 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.

Adjusting the Print Quality

Many factors affect print quality: type of supplies, print contrast, and the type of printer's application. The type of supply should match the printer's application.

- 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 Paxar Representative 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 your bar codes scan and how long the printhead lasts.

  Be sure to check the print quality of bar codes with a bar code verifier or scanner. If you do not have a verifier or scanner, check the bar code visually. A bar code that is **IN SPEC** will have complete bars and clear spaces. Small alphanumeric characters will look complete. A bar code that is **IN SPEC** may not look as good as one that is too dark, but it will have the highest scan rate.

Dark **IN SPEC** **Light**

Note: For highest scan rates, make sure there is adequate white space before and after the bar code. 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, you only need to 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 your 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 you send the batch header, 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 immediately 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 you want printed. 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. You can also 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.

General Format Tips and Hints

The following tips and hints are helpful to keep in mind when designing MPCLII formats.

With Formats

If you want to modify your format fields, check digit fields, and configuration commands, use the optional entry method. This method enables you to reset only the parameters you want to 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.

With Packets

Leave parameters blank that you do not need to change when sending online configuration packets.

You can 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: You should understand the basics of each field before using this method.

After you modify 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.

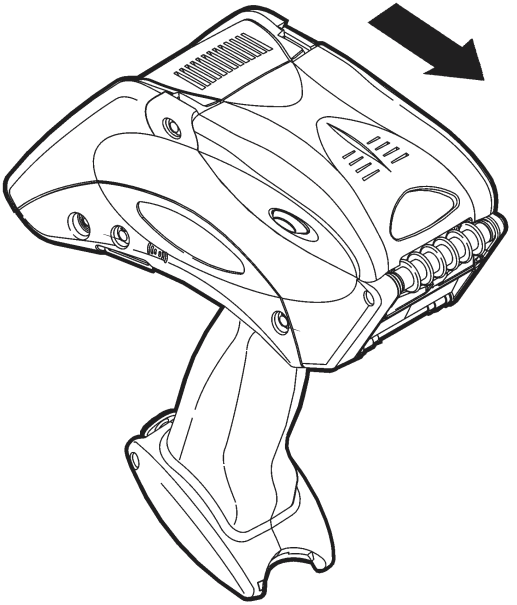
Check for trailing spaces in text or constant text fields if you receive a "field off tag" error. An easy way to see trailing spaces is to print the field in the reverse font.

Make sure if you magnify a field, it does not go off the label or cover another field. Magnifying a field increases the distance between the printed character and the edge of the cell.

SAMPLES

This appendix contains sample formats. You can customize any of these formats to meet your needs.

The samples pictured are oriented as if you are looking down at the display, watching labels come out of the front of the printer, as shown below.



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.

Note: Only the 6037 printer supports the Quick Response bar code.

- 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.

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 bar codes may still be scannable if the error correction is high enough. Options:

- **H** Ultra high reliability level
- **Q** High reliability level
- **M** Standard level
- **L** High density level

Note: As you increase 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. Options:

- **A** Automatic
- **M** Manual

char Type of characters. Options:

- **A** Alphanumeric
- **B** Binary
- **K** Kanji
- **N** Numeric

Note: In binary mode, the number of characters must be represented by the 4-digit number in decimal.

Example

```
1,"HM,N0123456789012345" ¦}
```

Defines the following batch data for the QR Code: The error correction level is **H**, which provides very high reliability. Leave the mask number blank. The data input mode is **M**anual. The type of characters are **N**umeric and the data is **0123456789012345**.

QR Code Packet

```
{F,1,A,R,E,200,200,"QRCODE"  ¦
B,1,200,V,50,50,36,0,100,2,B,0 ¦
{B,1,N,1 ¦
1,"HM,N0123456789012345" ¦ }
```

Structured Append Mode

QR Code offers a mode called structured append (or concatenated) that allows you to collect data from multiple QR Code symbols and use 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. You must use a two-digit number in decimal.

#of_div Total number of symbols in this concatenated set. You must use a two-digit number in decimal.

parity Parity byte. You must use a two-digit number in hexadecimal. There is no standard parity byte.

error_cor Level of error correction. Some damaged bar codes may still be scannable if the error correction is high enough. Options:

- **H** Ultra high reliability level
- **Q** High reliability level
- **M** Standard level
- **L** High density level

Note: As you increase 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. Options:

- **A** Automatic
- **M** Manual

char Type of characters. Options:

- **A** Alphanumeric
- **B** Binary
- **K** Kanji
- **N** Numeric

Note: In binary mode, the number of characters must be represented by the 4-digit number in decimal.

Example

```
1,"D0205E9,Q0A," ¦
C,"B006qrcode," ¦ }
```

Defines the structured append mode (**D**) for QR Code. This is symbol (**02**) of a concatenated set containing (**05**) symbols. The parity byte is **E9**. The error correction level is **Q**, which provides a high reliability. Use **0** for the mask number. The data input mode is **A**utomatic. 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" ¦ }
```

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 us for information about additional MaxiCode modes.

Note: Only the 6037 printer supports the Quick Response bar code.

Mode	Description
0	Obsolete
1	Obsolete
2	Structured Message
3	Structured Message
4	No Known Application
5	No Known Application
6	No Known Application

You can 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 your data. See “Defining a Bar Code Field” for more information. MaxiCode automatically pads data with the “!” character.

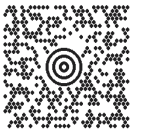
Note: MaxiCode does not support the NULL character.

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 you receive an error 612, check your MaxiCode data. You may have not correctly structured or left out 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

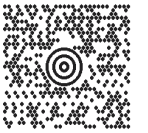


Mode 2 Sample

{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

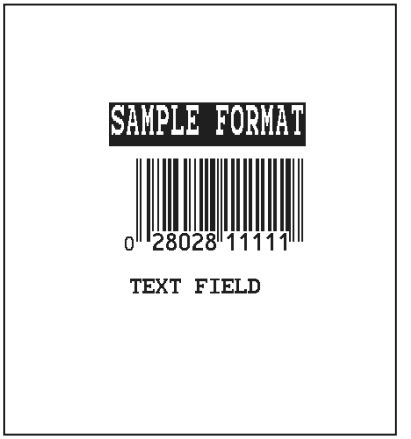
Mode 3 Sample

`{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 UPCA Format Packet

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

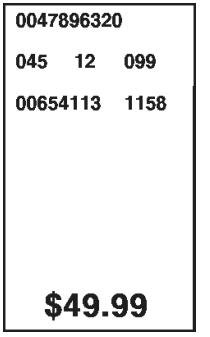


Sample Batch Packet

```
{B,25,N,1 ¦
1,"02802811111" ¦
2,"TEXT FIELD" ¦ }
```

HangTag Example

```
{F,1,A,R,E,275,125,"1TAG01"¦
C,228,20,0,50,8,8,A,L,0,0,"0047896320",1¦
C,203,20,0,50,8,8,A,L,0,0,"045",1¦
C,203,55,0,50,8,8,A,L,0,0,"12",1¦
C,203,85,0,50,8,8,A,L,0,0,"099",1¦
C,178,20,0,50,8,8,A,L,0,0,"00654113",1¦
C,178,85,0,50,8,8,A,L,0,0,"1158",1¦
C,54,37,0,50,14,14,A,L,0,0,"$49.99",1¦
}
```



Sample Batch Packet

```
{B,1,N,1¦E,0,0,1,1}
```

Tag Example

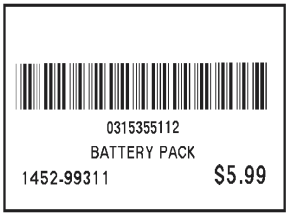
```
{F,1,A,R,E,200,150,"1LAB1520"¦
C,44,40,0,50,9,9,A,L,0,0,"PEANUTS",1¦
B,1,12,F,125,25,1,2,50,7,L,0¦
R,1,"028400067362"¦
C,20,34,0,50,8,8,A,L,0,0,"*SALT FREE*",1¦
C,84,45,0,50,14,14,A,L,0,0,"$1.19",1¦
}
```

Sample Batch Packet

```
{B,1,N,1¦E,0,0,1,1¦}
```

Label Example

```
{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,1,"031535512"¦
T,2,9,V,125,67,0,50,8,8,A,L,0,1,1¦
R,4,1,1,9,1,1¦
}
```



Sample Batch Packet

```
{B,1,N,1¦E,0,0,1,1}
```

Receipt Format Example

```
{F,1,A,R,E,300,175,"1Garage"¦
C,277,15,0,50,10,18,A,L,0,0,"KRAMER'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¦
}
```

Sample Batch Packet

```
{B,1,N,1¦E,0,0,1,¦}
```

Label Sample 2

```
{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¦
}
```

Sample Batch Packet

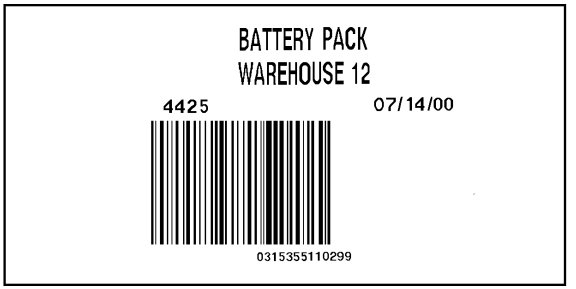
```
{B,1,N,1¦E,0,0,1,1¦}
```

Label Sample 3

```
{F,1,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¦
}
```

Sample Batch Packet

```
{B,1,N,1¦E,0,0,1,1¦}
```



FONTS

B

This appendix shows examples of the fonts loaded in your printer's memory.

Number	Font Size and Appearance	Type of Spacing	# of Dots Between Characters
1	Standard	monospaced	3 (203 dpi)
2	Reduced	monospaced	1 (203 dpi)
3	Bold	monospaced	3 (203 dpi)
4	OCRA-like	monospaced	3 (203 dpi)
5	HR1 - for numeric data	monospaced	2 (203 dpi)
6	HR2 - for numeric data	monospaced	1 (203 dpi)
10	CG Triumvirate™ Typeface Bold	proportional	varies w/ each letter
11	CG Triumvirate™ Typeface 6 pt.	proportional	varies w/ each letter
50	EFF Swiss Bold	scalable	varies w/ each letter
1000	CG Triumvirate™ Typeface Bold 6.5 pt	proportional	varies w/each letter
1001	CG Triumvirate™ Typeface Bold 8 pt	proportional	varies w/each letter
1002	CG Triumvirate™ Typeface Bold 10 pt	proportional	varies w/each letter
1003	CG Triumvirate™ Typeface Bold 12 pt	proportional	varies w/each letter
1004	CG Triumvirate™ Typeface Bold 18 pt	proportional	varies w/each letter
1005	CG Triumvirate™ Typeface Bold 22 pt	proportional	varies w/each letter
1006	CG Triumvirate™ Typeface Bold Condensed 6.5 pt	proportional	varies w/each letter
1007	CG Triumvirate™ Typeface Bold Condensed 8 pt	proportional	varies w/each letter
1008	CG Triumvirate™ Typeface Bold Condensed 10 pt	proportional	varies w/each letter
1009	CG Triumvirate™ Typeface Bold Condensed 12 pt	proportional	varies w/each letter
1010	CG Triumvirate™ Typeface Bold Condensed 18 pt	proportional	varies w/each letter
1011	CG Triumvirate™ Typeface Bold Condensed 22 pt	proportional	varies w/each letter
1012	Letter Gothic Bold 6 pt	monospaced	1
1013	Letter Gothic Bold 9 pt	monospaced	2

These samples were printed using the Internal Symbol set.

Standard Font

Reduced Font

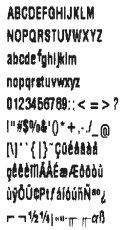
Bold Font

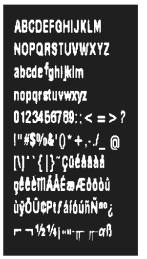
OCRA-like Font

These samples were printed using Code Page 437.

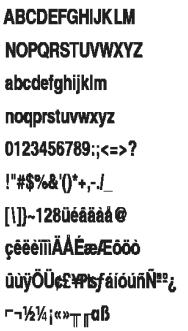
CG Triumvirate™ Typeface Bold

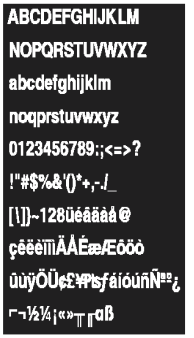
CG Triumvirate™ Typeface





EFF Swiss Bold Font





EFF Swiss Bold 9 pt

EFF Swiss Bold 15 pt

EFF Swiss Bold 20 pt

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 you want to list in a column. With proportionally spaced fonts, you may be able to place more characters on a line. However, you may need to experiment with these fonts and adjust field measurements in your format. The bitmapped fonts (either monospaced or proportional) appear jagged when magnified.
The magnification range is **1 - 7**.

Use the MONARCH® MPCL™ Toolbox (Soft 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 you are limited to the point size and characters you downloaded.

Monospaced Font Magnification

Monospaced characters occupy the same amount of space within a magnification. Use monospaced fonts for price fields and data you want to list in a column. Decide how wide and tall you want the characters to appear on the labels. The following two tables show the width and height of each of the monospaced fonts after magnification.

This table includes the default (3 dots for Standard, 1 dot for Reduced, 3 dots for Bold) spacing.

Width Mag.		Standard		Reduced		Bold	
	Units	Character Width	Sample	Character Width	Sample	Character Width	Sample
1x	1/100 in.	8.37		3.9		13.3	
	1/10 mm	21.26	A	9.9	A	33.78	A
	Dots	17		8		27	
7x	1/100 in.	49.75		24.63		84.24	
	1/10 mm	126.37		62.56		214	
	Dots	101		50		171	

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.

Standard	1x. A	7x 
1/100 in.	10.8	75.9
1/10 mm	27.4	192.8
dots	22	154

Reduced	1x. A	7x
1/100 in.	6.9	48.28
1/10 mm	17.5	122.6
dots	14	98

Bold	1x. A	7x 
1/100 in.	16.7	117.24
1/10 mm	42.4	297.8
dots	34	238

Proportional Font Magnification

Each character in a proportionally spaced font is a different height and width. You may be able to place more characters on a line using proportionally spaced fonts. You may want to experiment with these fonts and adjust field measurements in your format as needed. The bitmapped fonts (either monospaced or proportional) appear jagged when magnified. The magnification range is **1 - 7**.

The following tables provide height and width magnification of sample characters.

CG Triumvirate™ Typeface Bold (8 pt.)

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	1.56	5.73	10.94
	1/10 mm	3.97 I	14.55 L	27.78 W
	Dots	3	11	21
7x	1/100 in.	6.9	20.7	41.4
	1/10 mm	17.5 	52.6 	105.2 
	Dots	14	42	84

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.

CG Triumvirate™ Typeface Bold (6.5 pt.) Font #1000

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	1.56	4.69	9.90
	1/10 mm	3.97	11.91	25.14
	Dots	3	9	19
7x	1/100 in.	10.94	32.81	69.27
	1/10 mm	27.78	83.34	175.95
	Dots	21	63	133

Height Magnification

	1x	7x
1/100 in.	6.77	47.40
1/10 mm	17.20	120.39
dots	13	91

CG Triumvirate™ Typeface Bold (8 pt.) Font #1001

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	1.56	5.73	10.94
	1/10 mm	3.97	14.55	27.78
	Dots	3	11	21
7x	1/100 in.	6.9	20.7	41.4
	1/10 mm	17.5	52.6	105.2
	Dots	14	42	84

Height Magnification

	1x	7x
1/100 in.	8.33	58.33
1/10 mm	21.17	148.17
dots	16	112

CG Triumvirate™ Typeface Bold (10 pt.) Font #1002

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	1.56	6.77	13.02
	1/10 mm	3.97	17.20	33.07
	Dots	3	13	25
7x	1/100 in.	10.94	47.40	91.15
	1/10 mm	27.78	120.39	231.51
	Dots	21	91	175

Height Magnification

	1x	7x
1/100 in.	10.42	72.92
1/10 mm	26.46	185.21
dots	20	140

CG Triumvirate™ Typeface Bold (12 pt.) Font #1003

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	2.60	8.85	16.15
	1/10 mm	6.61	22.49	41.01
	Dots	5	17	31
7x	1/100 in.	18.23	67.94	113.02
	1/10 mm	46.30	157.43	287.07
	Dots	35	119	217

Height Magnification

	1x	7x
1/100 in.	12.50	87.50
1/10 mm	31.75	222.25
dots	24	168

CG Triumvirate™ Typeface Bold (18 pt.) Font #1004

Point sizes greater than 12 include only the following special characters: 0123456789#$%&(),./@DFKLMPS\kprö¢£¥. Since this font does not support the full ANSI character set, some characters above decimal value 127 are substituted. For example, decimal value 191 prints the Yen symbol (¥) instead of the upside-down question mark (¿).

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	7.81	12.50	22.92
	1/10 mm	19.84	31.75	58.21
	Dots	15	24	44
7x	1/100 in.	54.69	87.50	160.42
	1/10 mm	138.91	222.25	407.46
	Dots	105	168	308

Height Magnification

	1x	7x
1/100 in.	19.27	134.90
1/10 mm	48.95	342.64
dots	37	259

CG Triumvirate™ Typeface Bold (22 pt.) Font #1005

Point sizes greater than 12 include only the following special characters: 0123456789#$%&(),./@DFKLMPS\kprö¢£¥. Since this font does not support the full ANSI character set, some characters above decimal value 127 are substituted. For example, decimal value 191 prints the Yen symbol (¥) instead of the upside-down question mark (¿).

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	9.38	15.63	27.08
	1/10 mm	23.81	39.68	68.79
	Dots	18	30	52
7x	1/100 in.	65.63	109.38	189.58
	1/10 mm	166.69	277.81	481.54
	Dots	126	210	364

Height Magnification

1/100 in.	23.44	164.05
1/10 mm	59.53	416.72
dots	54	315

CG Triumvirate™ Typeface Bold Cond (6.5 pt.) Font #1006

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	1.04	3.65	6.25
	1/10 mm	2.65	9.26	15.88
	Dots	2	7	12
7x	1/100 in.	7.29	25.52	43.75
	1/10 mm	18.52	64.82	111.13
	Dots	14	49	84

Height Magnification

	1x	7x
1/100 in.	6.77	47.40
1/10 mm	17.20	120.39
dots	13	91

CG Triumvirate™ Typeface Bold Cond (8 pt.) Font #1007

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	1.56	4.69	8.85
	1/10 mm	3.97	11.91	22.49
	Dots	3	9	17
7x	1/100 in.	10.94	32.81	67.94
	1/10 mm	27.78	83.34	157.43
	Dots	21	63	119

Height Magnification

	1x	7x
1/100 in.	8.85	61.98
1/10 mm	22.49	157.43
dots	17	119

CG Triumvirate™ Typeface Bold Cond (10 pt.) Font #1008

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	2.08	5.73	11.98
	1/10 mm	5.29	14.55	30.43
	Dots	4	11	23
7x	1/100 in.	14.58	40.10	83.85
	1/10 mm	37.04	101.86	212.99
	Dots	28	77	161

Height Magnification

	1x	7x
1/100 in.	10.94	76.56
1/10 mm	27.78	194.47
dots	21	147

CG Triumvirate™ Typeface Bold Cond (12 pt.) Font #1009

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	2.60	7.29	14.58
	1/10 mm	6.61	18.52	38.04
	Dots	5	14	28
7x	1/100 in.	18.23	51.04	96.88
	1/10 mm	46.30	129.65	246.06
	Dots	35	98	186

Height Magnification

	1x	7x
1/100 in.	13.02	91.15
1/10 mm	33.07	231.51
dots	25	175

CG Triumvirate™ Typeface Bold Cond (18 pt.) Font #1010

Point sizes greater than 12 include only the following special characters: 0123456789#$%&(),./@DFKLMPS\kprö¢£¥. Since this font does not support the full ANSI character set, some characters above decimal value 127 are substituted. For example, decimal value 191 prints the Yen symbol (¥) instead of the upside-down question mark (¿).

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	7.81	10.94	21.88
	1/10 mm	19.84	27.78	55.56
	Dots	15	21	42
7x	1/100 in.	54.69	76.56	153.12
	1/10 mm	138.91	194.47	388.94
	Dots	105	147	294

Height Magnification

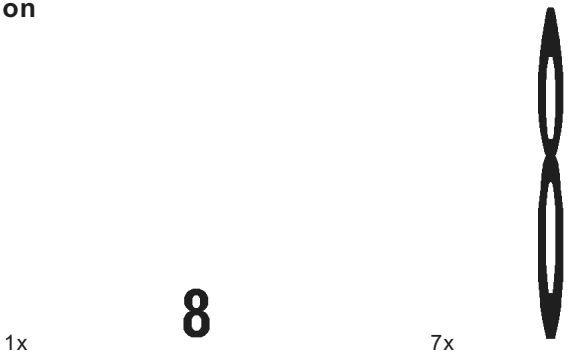
1/100 in.	20.31	142.19
1/10 mm	51.59	361.16
dots	39	273

CG Triumvirate™ Typeface Bold Cond (22 pt.) Font #1011

Point sizes greater than 12 include only the following special characters: 0123456789#$%&(),./@DFKLMPS\kprö¢£¥. Since this font does not support the full ANSI character set, some characters above decimal value 127 are substituted. For example, decimal value 191 prints the Yen symbol (¥) instead of the upside-down question mark (¿).

Width Mag.		Minimum	Average	Maximum
1x	1/100 in.	9.38	13.02	26.56
	1/10 mm	23.81	33.07	67.47
	Dots	18	25	51
7x	1/100 in.	65.63	91.15	185.94
	1/10 mm	166.69	231.51	472.28
	Dots	126	175	357

Height Magnification



1/100 in.	24.48	171.35
1/10 mm	62.18	435.24
dots	47	329

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 you define formats using scalable fonts, remember to set the character rotation to **0**, because 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.**

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 - 250**. 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.

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 72 point EFF Swiss Bold sample shows the one inch cell size.

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 - 250**. The field width varies with each letter. The printer can accept 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, you download the entire font, not particular characters or one point size. You can print a variety of symbol sets 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.

These font files are large and may image slower than bitmap fonts. The size of the font file, in bytes, is the minimum amount of memory you must have available for fonts in the printer's downloadable fonts buffer.

You may need to reallocate memory to use downloaded TrueType fonts. After reallocating memory, resend the font, format, and batch packets.

Using International Fonts

International fonts are available as bitmap or TrueType fonts. See "Bitmap Font Information" or "TrueType Font Information" for more details. To use these fonts in the 6032 printer, you must purchase the print engine memory expansion option and download the font to the printer.

Note: The 6037 printer does **not** support International fonts.

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 6032 printer supports several mapping standards:
    - Unicode (UCS-2)
    - BIG5
    - GB2312
    - 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. See "Selecting a Symbol Set" for more information.
- To enter batch data, use the method specified by the font's character mapping and a 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*	TrueType Font Character Mapping	use in text or constant text fields
Unicode	BIG5	102 - Unicode If you are unsure which character mapping to select, use this symbol set, because the printer automatically translates the character mappings.
Unicode	SJIS	
Unicode	KSC5601	
Unicode	GB2312	
Unicode	Unicode	
BIG5	Unicode	103 - BIG5
GB2312	Unicode	104 - GB2312
SJIS	SJIS	105 - SJIS (Code Page 932 - Japanese Shift-JIS)
GB2312	GB2312	106 - GB2312 (Code Page 936 - Simplified Chinese)
BIG5	BIG5	107 - BIG5 (Code Page 950 - Traditional Chinese)

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

Note: Symbol sets 102-107 require the 6032 print engine memory expansion option and 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

紀紂約紅

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

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 Your Fonts

We provide you with tools to create and download TrueType fonts. However, it is your responsibility to purchase and license any fonts you download to your printer. Contact your font supplier for licensing information. Additional fonts that are compatible with the 6032 printer can be purchased from:

The Electronic Font Foundry
11 Silwood Road; Ascot; SL5 OPY; England
(0)1344 875 201
www.eff.co.uk

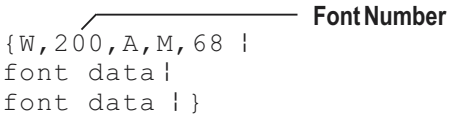
Korean, Chinese, and Japanese fonts can be purchased from:

Dynalab Inc.
2055 Gateway Place; Suite 400; San Jose, CA 95110
408-490-4224
www.dynalab.com

Locating the Font Number in a Font Packet

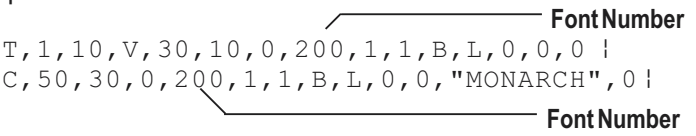
If you are creating font packets, the font number is the second parameter in the packet. Software is available to create the font data and packet. Call Service for more information.

Example



Use this number in **T8** or in **C5**. See “Defining Text Fields” or “Defining Constant Text Fields” in Chapter 2 for more information.

Example



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 printers support.

Use the charts in this appendix to convert dot sequences from the image dot pattern to codes you can use 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 printer supports 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 (double-byte) fonts.

The printer defaults to the internal symbol set.

Note: Fonts 1004, 1005, 1010, and 1011 are limited to the following special characters: 0123456789#$%&(),./@DFKLMPS\kprö¢£¥. These fonts do not support the full ANSI character set and some characters above decimal value 127 are substituted.

Selecting a Symbol Set or Code Page

The CG Triumvirate™ typefaces support only the ANSI and DOS Code Page 437 and 850 Symbol Sets. These fonts print a slashed zero when using the ANSI symbol set.

Internal	Use this symbol set to print international monetary symbols, the trademark (™) symbol, and for formats that may be used on other MPCLII printers.
ANSI	Use this symbol set with proportionally spaced fonts.
DOS Code Pages 437 or 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 your batch data file, type a tilde in front of the three-digit code. For example, if you want to include the character Ä in a text field using the Internal Symbol Set, type:

```
1,"~142" ¦
```

Using International Character Sets/Symbol Sets

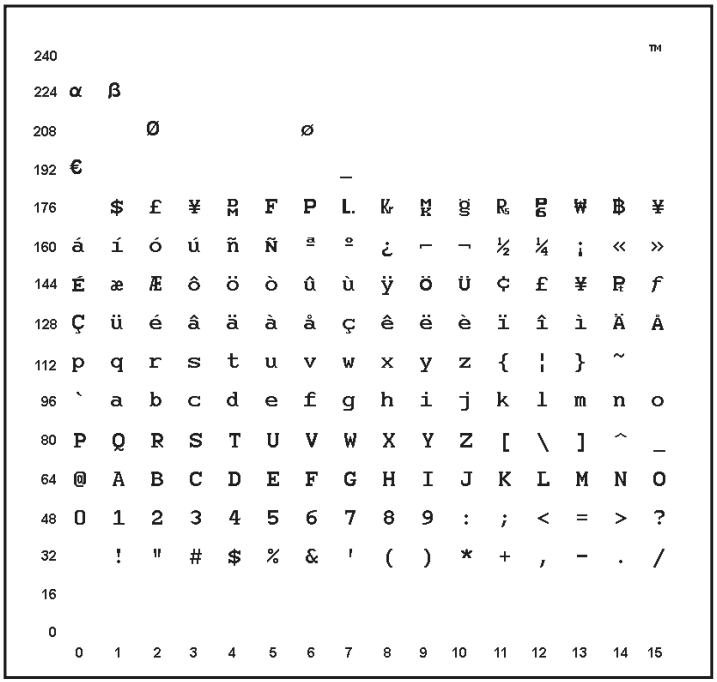
Symbol sets 100, 101, 852-860, and 1250-1258 may only be used with the scalable font (font#50) or downloaded TrueType fonts. TrueType fonts are designed to be regionally specific; therefore, all symbol sets may not be supported in a given font. For example, to print Hebrew characters, you need to find a font (such as Arial) that supports Hebrew characters; convert, and then download the font to your printer. Make sure the correct symbol set for Hebrew characters is selected.

Font 50 does not print the slashed zero or support Code Pages 101 (Wingdings) and 1256 (Arabic). The Euro-dollar symbol at position **~192** is only available in the Standard, Reduced, and Bold fonts.

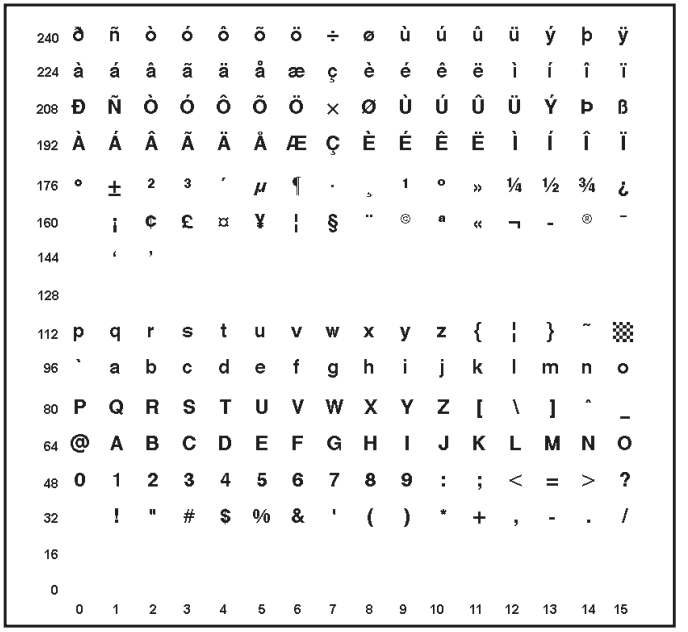
Code pages 102-107 contain thousands of characters, which are not represented in this manual. These code pages require the 6032 print engine memory expansion option and a downloaded International TrueType font.

The Code Pages (100 and greater) on the following pages were printed using Arial or a similar downloaded TrueType font. To determine the character code, add the column number and row number for the character. For example, to produce the # character with the ANSI character set, you would press Alt 255 (column 15 + row 240).

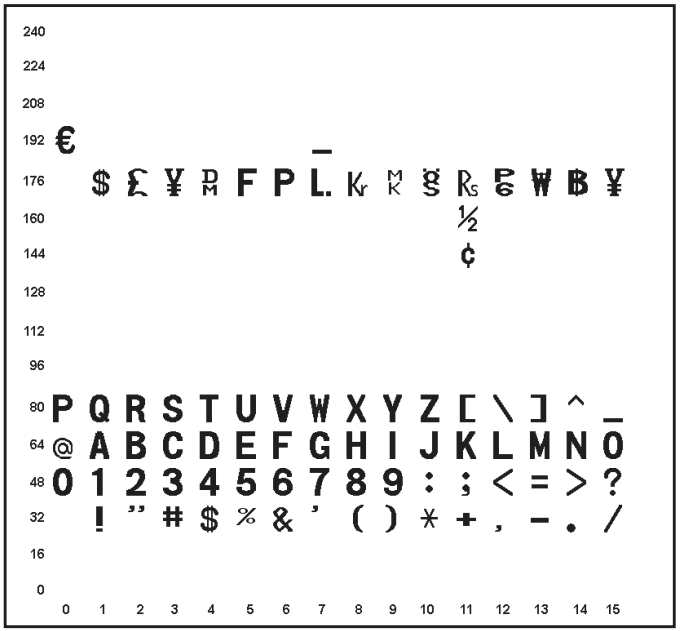
Internal Symbol Set



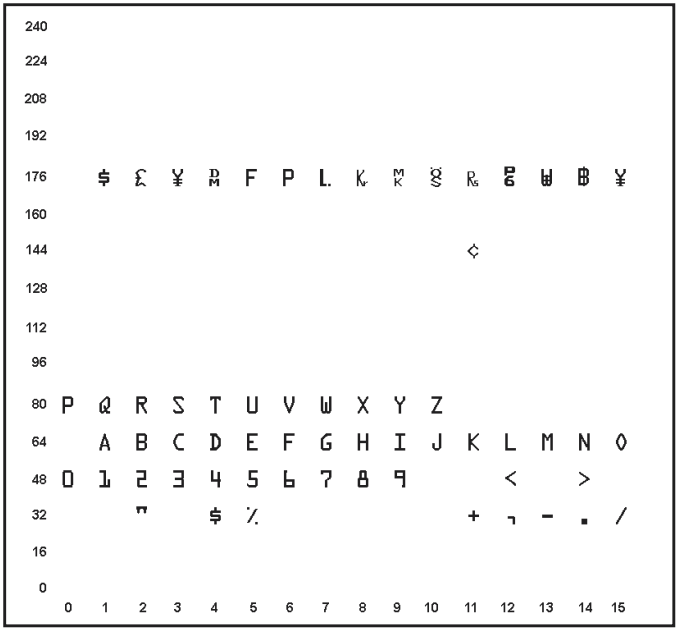
ANSI Symbol Set



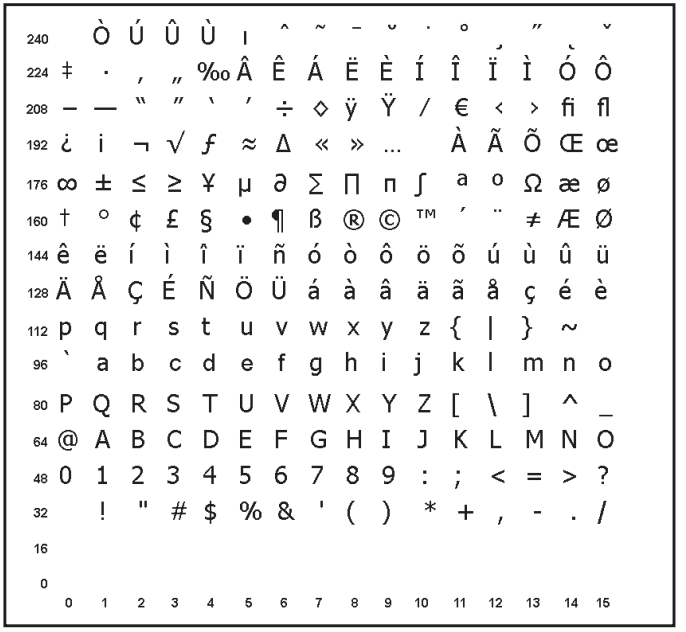
Bold Character Set



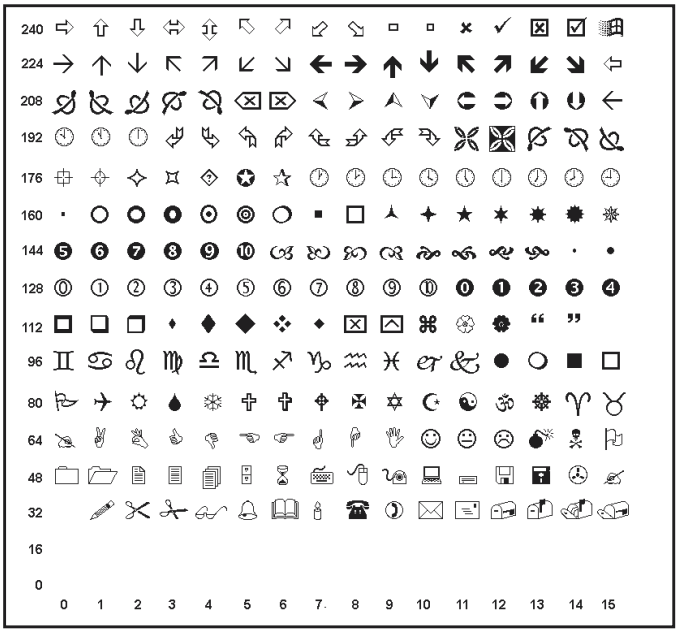
OCRA Character Set



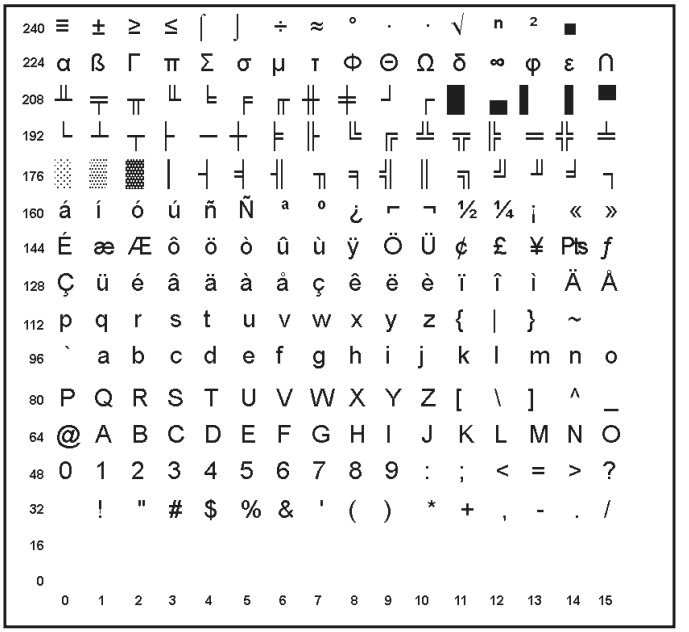
Code Page 100 (Macintosh)



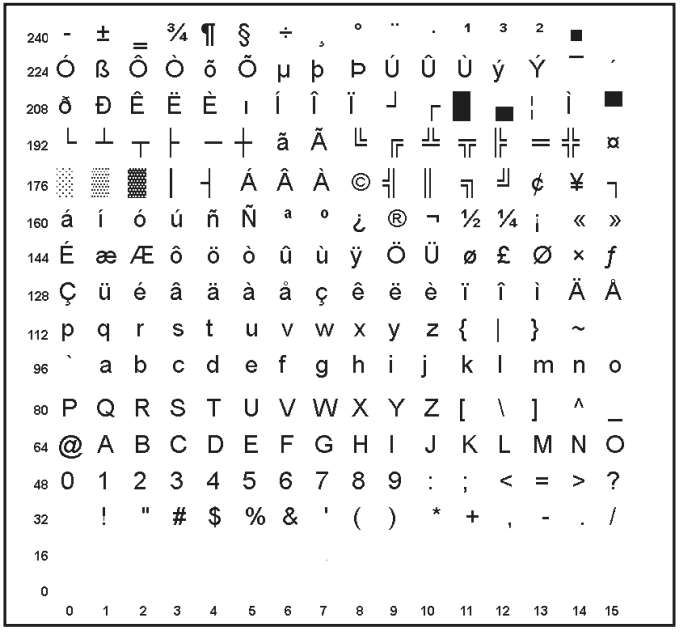
Code Page 101 (Wingdings)



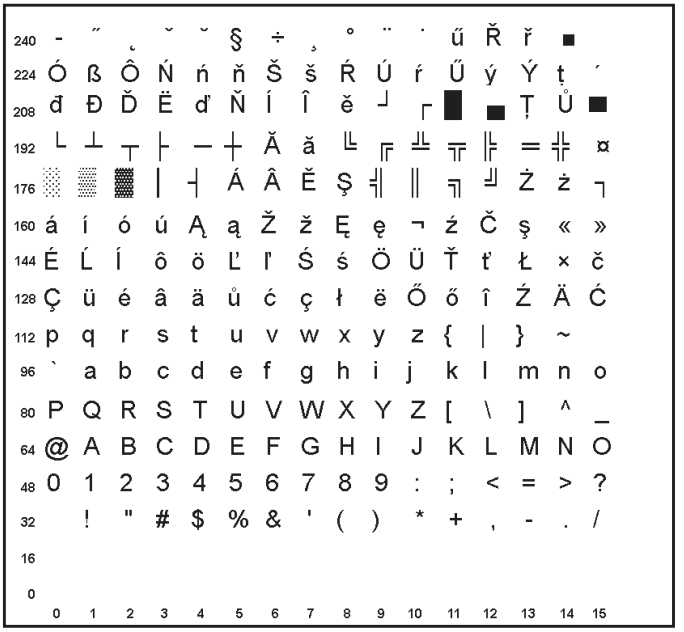
Code Page 437 (Latin U.S.)



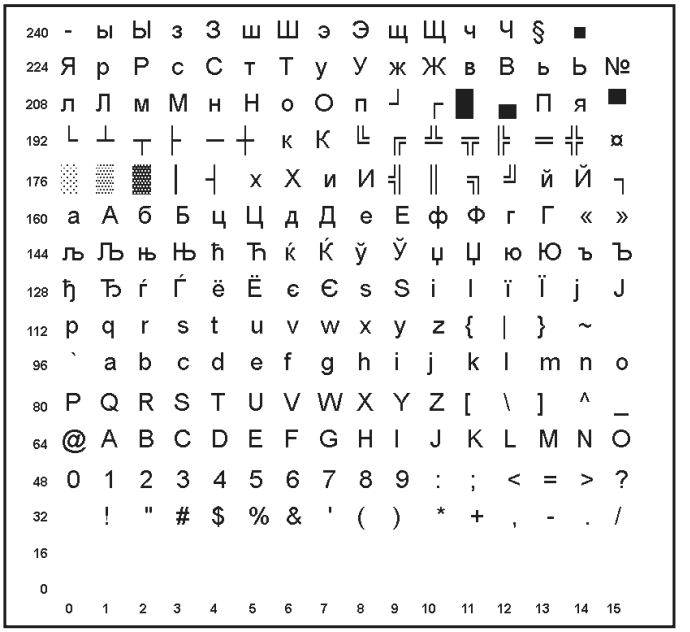
Code Page 850 (Latin 1)



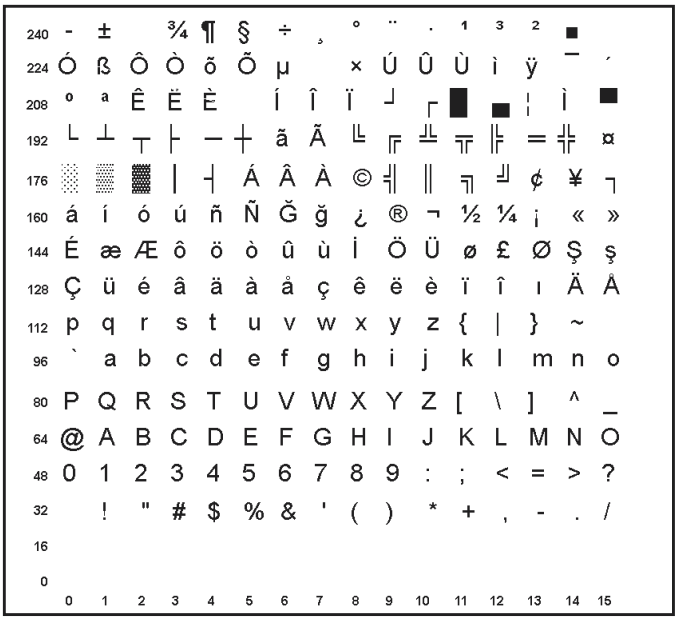
Code Page 852 (Latin 2)



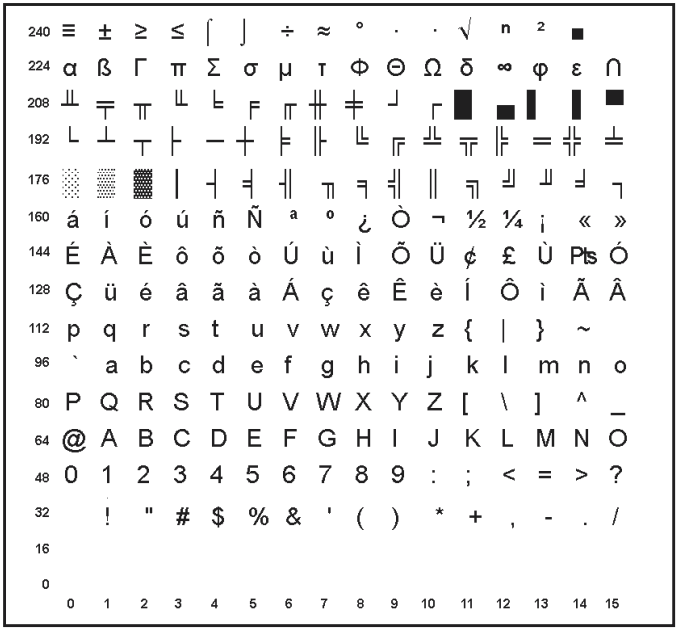
Code Page 855 (Russian)



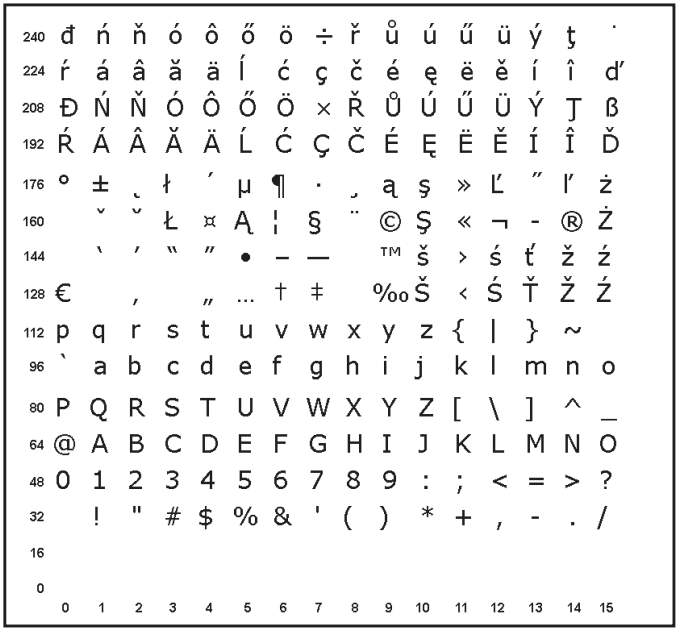
Code Page 857 (IBM Turkish)



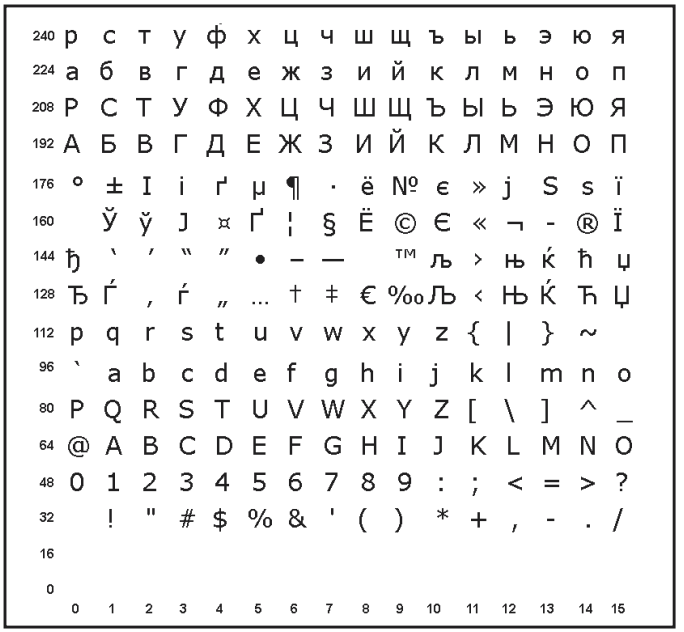
Code Page 860 (MS-DOS Portuguese)



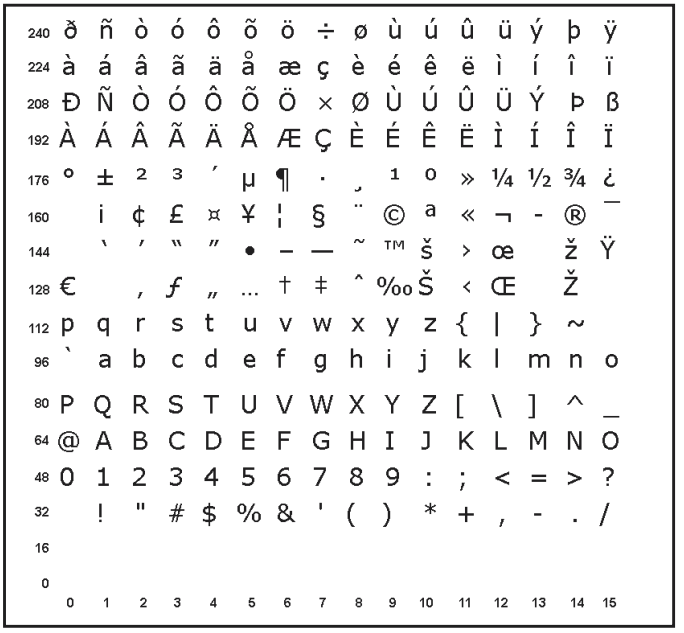
Code Page 1250 (Latin 2)



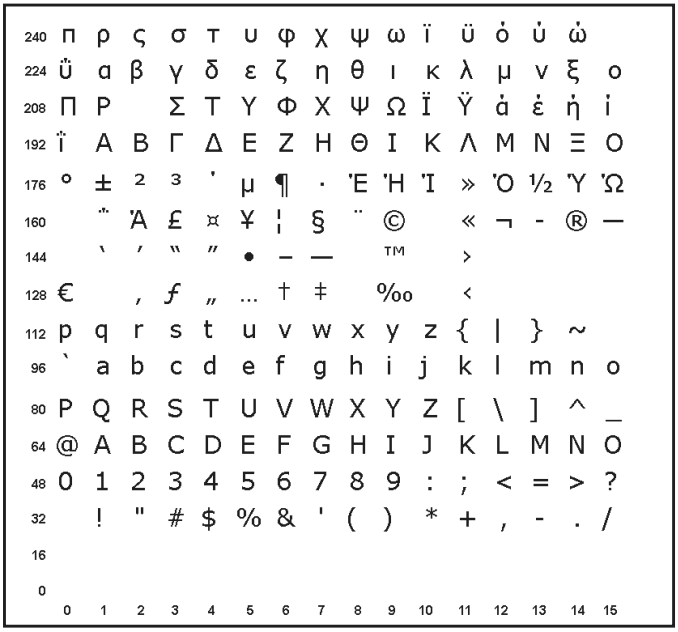
Code Page 1251 (Cyrillic)



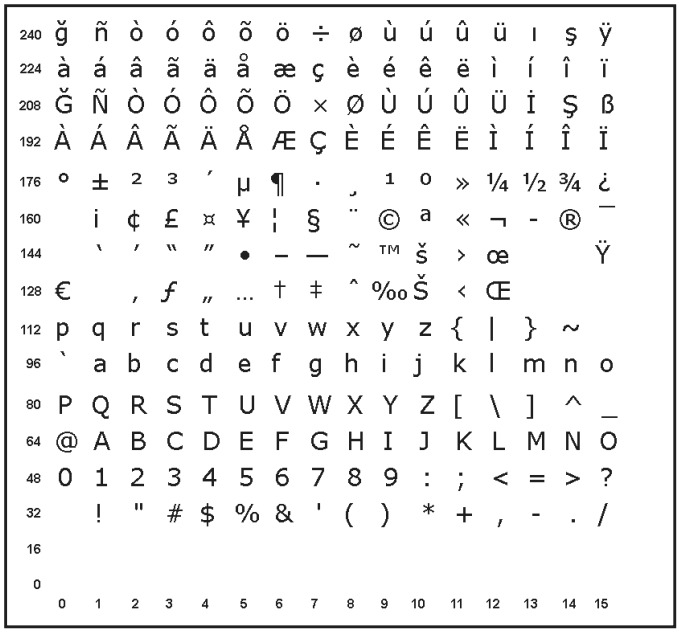
Code Page 1252 (Latin 1)



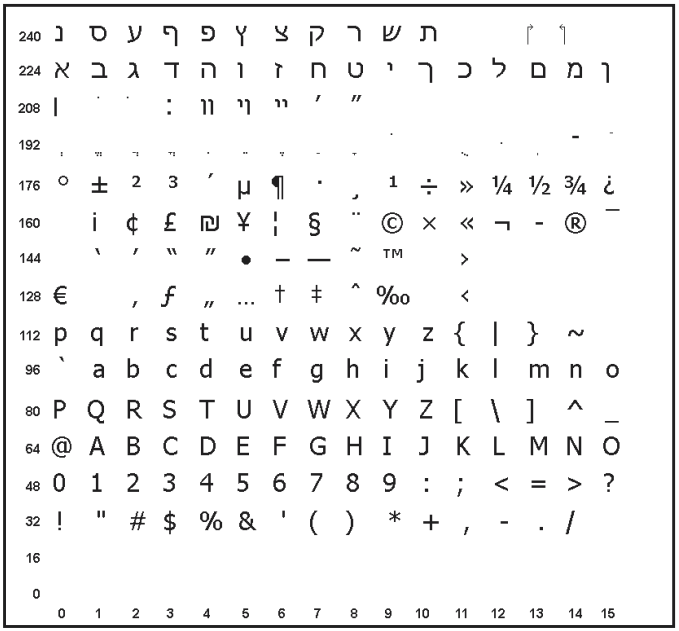
Code Page 1253 (Greek)



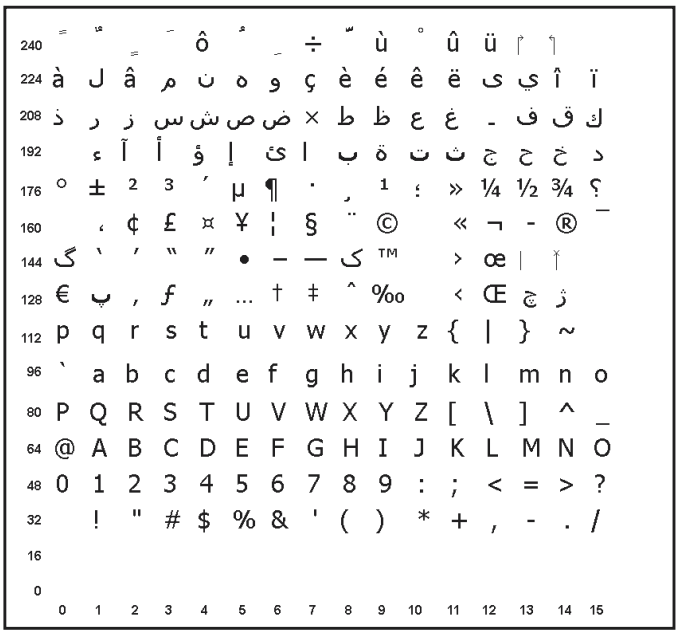
Code Page 1254 (Turkish)



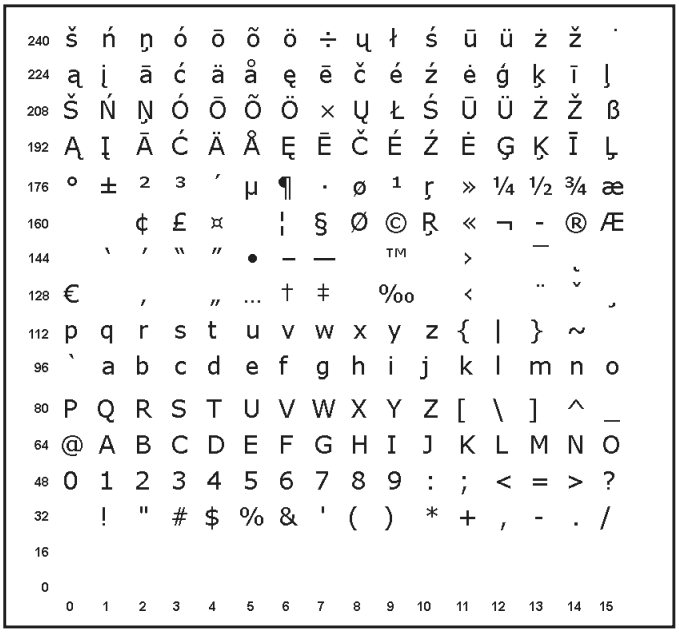
Code Page 1255 (Hebrew)



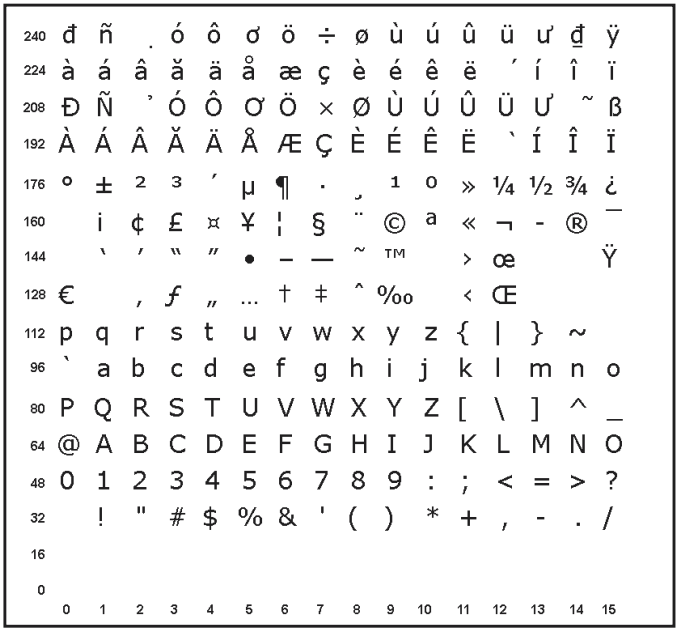
Code Page 1256 (Arabic)



Code Page 1257 (Baltic)



Code Page 1258 (Vietnamese)



ASCII to Hexadecimal Conversion Chart

Use the chart below to translate the characters printed on your 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
ENQ	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	cursor right	1C	28
home	0B	11	cursor left	1D	29
form feed	0C	12	cursor up	1E	30
carriage return	0D	13	cursor down	1F	31
SO	0E	14	space	20	32
SI	0F	15	!	21	33
DLE	10	16	“	22	34
DC1	11	17	#	23	35

ASCII to Hexadecimal Conversion Chart (continued)

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

ASCII to Hexadecimal Conversion Chart (continued)

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	Binary	Hex
00000000	00	00100000	20
00000001	01	00100001	21
00000010	02	00100010	22
00000011	03	00100011	23
00000100	04	00100100	24
00000101	05	00100101	25
00000110	06	00100110	26
00000111	07	00100111	27
00001000	08	00101000	28
00001001	09	00101001	29
00001010	0A	00101010	2A
00001011	0B	00101011	2B
00001100	0C	00101100	2C
00001101	0D	00101101	2D
00001110	0E	00101110	2E
00001111	0F	00101111	2F
00010000	10	00110000	30
00010001	11	00110001	31
00010010	12	00110010	32
00010011	13	00110011	33
00010100	14	00110100	34
00010101	15	00110101	35
00010110	16	00110110	36
00010111	17	00110111	37
00011000	18	00111000	38
00011001	19	00111001	39
00011010	1A	00111010	3A
00011011	1B	00111011	3B
00011100	1C	00111100	3C
00011101	1D	00111101	3D
00011110	1E	00111110	3E
00011111	1F	00111111	3F

Binary to Hexadecimal Conversion Chart (continued)

Binary	Hex	Binary	Hex
01000000	40	01100000	60
01000001	41	01100001	61
01000010	42	01100010	62
01000011	43	01100011	63
01000100	44	01100100	64
01000101	45	01100101	65
01000110	46	01100110	66
01000111	47	01100111	67
01001000	48	01101000	68
01001001	49	01101001	69
01001010	4A	01101010	6A
01001011	4B	01101011	6B
01001100	4C	01101100	6C
01001101	4D	01101101	6D
01001110	4E	01101110	6E
01001111	4F	01101111	6F
01010000	50	01110000	70
01010001	51	01110001	71
01010010	52	01110010	72
01010011	53	01110011	73
01010100	54	01110100	74
01010101	55	01110101	75
01010110	56	01110110	76
01010111	57	01110111	77
01011000	58	01111000	78
01011001	59	01111001	79
01011010	5A	01111010	7A
01011011	5B	01111011	7B
01011100	5C	01111100	7C
01011101	5D	01111101	7D
01011110	5E	01111110	7E
01011111	5F	01111111	7F

Binary to Hexadecimal Conversion Chart (continued)

Binary	Hex	Binary	Hex
10000000	80	10100000	A0
10000001	81	10100001	A1
10000010	82	10100010	A2
10000011	83	10100011	A3
10000100	84	10100100	A4
10000101	85	10100101	A5
10000110	86	10100110	A6
10000111	87	10100111	A7
10001000	88	10101000	A8
10001001	89	10101001	A9
10001010	8A	10101010	AA
10001011	8B	10101011	AB
10001100	8C	10101100	AC
10001101	8D	10101101	AD
10001110	8E	10101110	AE
10001111	8F	10101111	AF
10010000	90	10110000	B0
10010001	91	10110001	B1
10010010	92	10110010	B2
10010011	93	10110011	B3
10010100	94	10110100	B4
10010101	95	10110101	B5
10010110	96	10110110	B6
10010111	97	10110111	B7
10011000	98	10111000	B8
10011001	99	10111001	B9
10011010	9A	10111010	BA
10011011	9B	10111011	BB
10011100	9C	10111100	BC
10011101	9D	10111101	BD
10011110	9E	10111110	BC
10011111	9F	10111111	BF

Binary to Hexadecimal Conversion Chart (continued)

Binary	Hex	Binary	Hex
11000000	C0	11100000	E0
11000001	C1	11100001	E1
11000010	C2	11100010	E2
11000011	C3	11100011	E3
11000100	C4	11100100	E4
11000101	C5	11100101	E5
11000110	C6	11100110	E6
11000111	C7	11100111	E7
11001000	C8	11101000	E8
11001001	C9	11101001	E9
11001010	CA	11101010	EA
11001011	CB	11101011	EB
11001100	CC	11101100	EC
11001101	CD	11101101	ED
11001110	CE	11101110	EE
11001111	CF	11101111	EF
11010000	D0	11110000	F0
11010001	D1	11110001	F1
11010010	D2	11110010	F2
11010011	D3	11110011	F3
11010100	D4	11110100	F4
11010101	D5	11110101	F5
11010110	D6	11110110	F6
11010111	D7	11110111	F7
11011000	D8	11111000	F8
11011001	D9	11111001	F9
11011010	DA	11111010	FA
11011011	DB	11111011	FB
11011100	DC	11111100	FC
11011101	DD	11111101	FD
11011110	DE	11111110	FE
11011111	DF	11111111	FF

Dot to Run Length Encoding Chart

ON (Black) Dots

# of Dots	Code	# of Dots	Code
1	A	14	N
2	B	15	O
3	C	16	P
4	D	17	Q
5	E	18	R
6	F	19	S
7	G	20	T
8	H	21	U
9	I	22	V
10	J	23	W
11	K	24	X
12	L	25	Y
13	M	26	Z

OFF (White Dots)

# of Dots	Code	# of Dots	Code
1	a	14	n
2	b	15	o
3	c	16	p
4	d	17	q
5	e	18	r
6	f	19	s
7	g	20	t
8	h	21	u
9	i	22	v
10	j	23	w
11	k	24	x
12	l	25	y
13	m	26	z

FORMAT DESIGN TOOLS

D

Use copies of these worksheets and grids to create formats, batch data, and check digit schemes. You may want to keep copies of the completed forms for your records:

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

Batch Worksheet

BATCH HEADER

B	B1 HEADER	B2 FORMAT #	B3 NEW / UPDATE	B4 QUANTITY

BATCH CONTROL

E	E1 HEADER	E2 FEED-MODE	E3 BATCH-SEP	E4 PRINT-MULT	E5 MULTI-PART	E6 CUT-TYPE	E7 CUT-MULT

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	

PRINT CONTROL

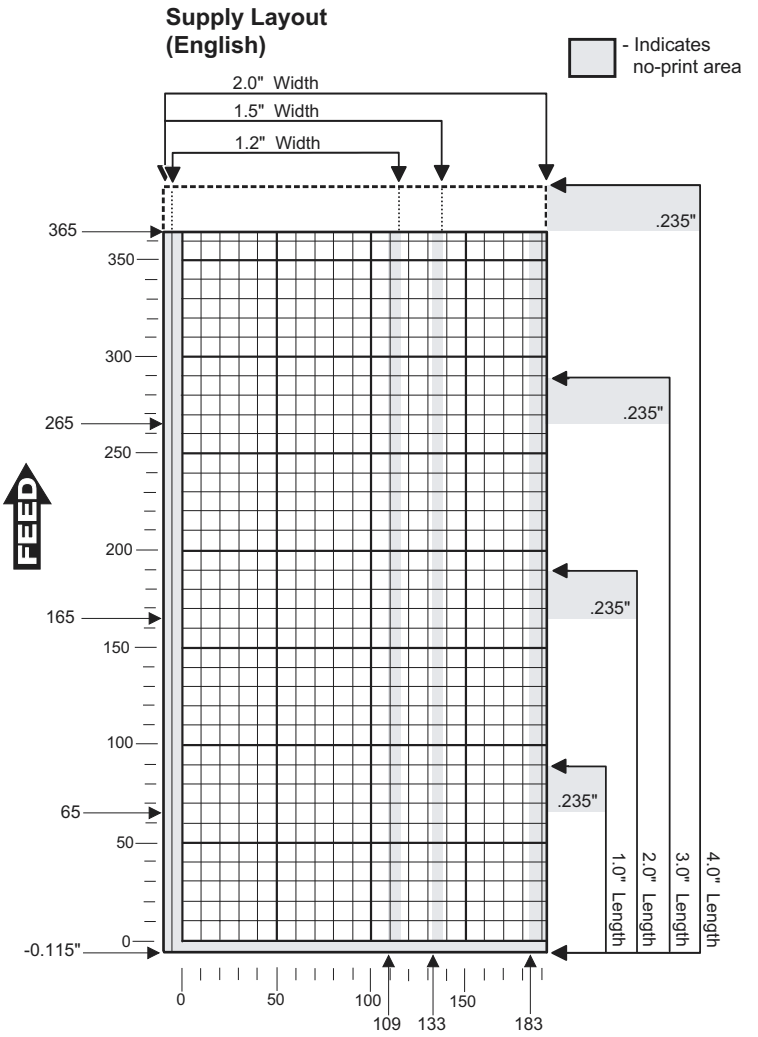
C	HEADER	CONTRAST	PRINT ADJUST	MARGIN ADJUST	SPEED ADJUST	PRINTHD WIDTH

Check Digit Worksheet

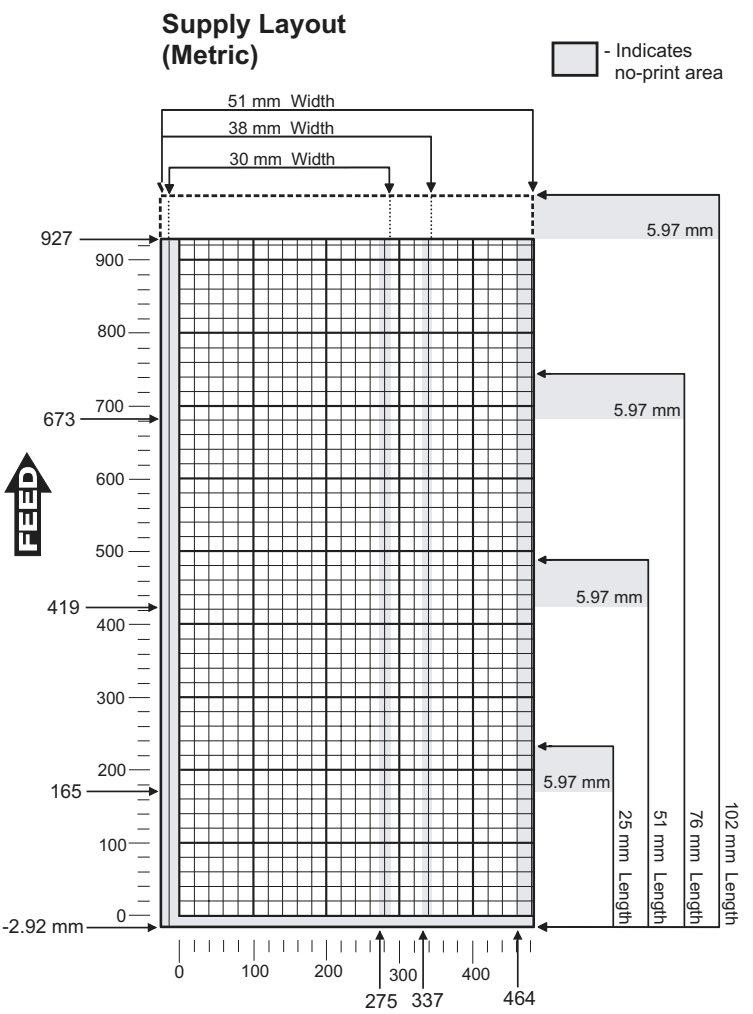
A1 HEADER	A2 SELECTOR #	A3 ACTION	A4 DEVICE	A5 MODULUS	A6 LENGTH	A7 ALGORITHM	WEIGHTS A8
A			**R**				

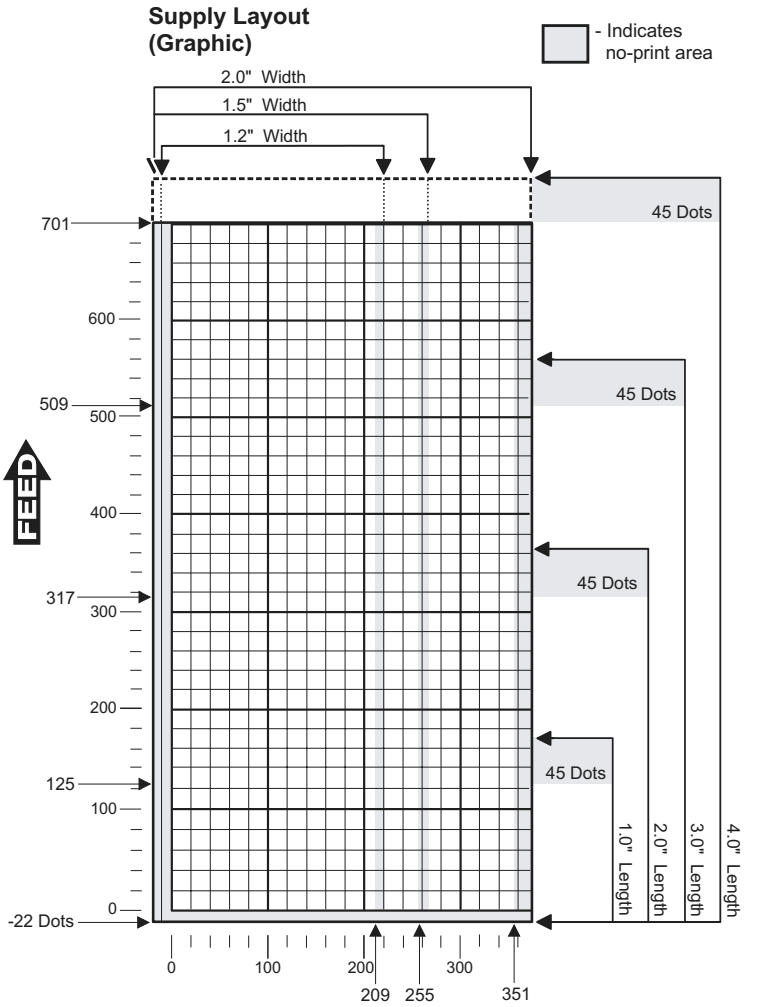
A1 HEADER	A2 SELECTOR #	A3 ACTION	A4 DEVICE	A5 MODULUS	A6 LENGTH	A7 ALGORITHM	WEIGHTS A8
A			**R**				

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


Supply Layout
(English)
- Indicates
no-print area
2.0" Width
1.5" Width
1.2" Width
.235"
.235"
.235"
.235"
1.0" Length
2.0" Length
3.0" Length
4.0" Length
FEED
365
350
300
265
250
200
165
150
100
65
50
0
-0.115"
0
50
100
150
109
133
183

Supply Layout (Metric)




Supply Layout
(Graphic)
- Indicates
no-print area
2.0" Width
1.5" Width
1.2" Width
45 Dots
45 Dots
45 Dots
45 Dots
701
600
509
500
400
317
300
200
125
100
0
-22 Dots
FEED
1.0" Length
2.0" Length
3.0" Length
4.0" Length
0
100
200
300
209
255
351

FORMAT HEADER

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

NON - PRINTABLE TEXT FIELDS

			OPTION #1 Fixed Characters		
D1 HEADER	D2 FIELD #	D3 # OF CHAR.	R1 HEADER	R2 CODE #	R3 FIXED CHAR. ("IN QUOTES")
D			R	1	
D			R	1	
D			R	1	
D			R	1	

TEXT FIELDS

															OPTION #1 Fixed Characters			OPTION #4 Copy Data from Previous 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 INPUT (√)
T															R	1		R	4						R	61	
T															R	1		R	4						R	61	
T															R	1		R	4						R	61	
T															R	1		R	4						R	61	
T															R	1		R	4						R	61	
T															R	1		R	4						R	61	
T															R	1		R	4						R	61	
T															R	1		R	4						R	61	
T															R	1		R	4						R	61	

BAR CODE FIELDS

												OPTION #1 Fixed Characters			OPTION #4 Copy Data from Previous Field							OPTION #31 Define Check Digit				OPTION #50 Bar Code Densities							OPTION #51 PDF417 Security / Truncation				OPTION #52 PDF417 Aspect Ratio				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 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 INPUT (√)
B												R	1		R	4						R	31			R	50						R	51			R	52			R	61	
B												R	1		R	4						R	31			R	50						R	51			R	52			R	61	
B												R	1		R	4						R	31			R	50						R	51			R	52			R	61	
B												R	1		R	4						R	31			R	50						R	51			R	52			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												

GRAPHICS FIELDS

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

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						

FORMAT WORKSHEET

Format Name ______________________

Format # ______________________

Date ______________________

Supply Size ______________________

Supply Type ______________________

Customer Name ______________________

Software Version ______________________

GLOSSARY

Batch Data
2,"Monarch"¦
Defines the actual information (as fields within **{ }**) printed on the label.

Batch Control
E,0,1,4,2¦
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. Enclosed within { }.

Bitmapped Fonts
Reside in the printer's memory. If you change the point size, you have changed the font. Magnifying these fonts causes some 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.

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 on power-down.

Format
Layout or design for your 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.)

Option
R,4,6,1,3,1 ¦ Any line within a format that applies special formatting to a field. This line always 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.

Soft (Downloaded) Fonts Reside in the printer's RAM. They can be erased or overwritten.

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 on power-down.

INDEX

C

D

E

F

G

H

P

R

S

T

U

V

W

Visit **www.paxar.com** for sales, service, supplies, information, and telephone numbers for our International locations.

TOLL FREE:
1-800-543-6650 (In the U.S.A.)
1-800-363-7525 (In Canada)