IPL

Intermec Printer Language

Developer's Guide

Intermec Technologies Corporation

Worldwide Headquarters
6001 36th Ave.W.
Everett, WA 98203
U.S.A.

www.intermec.com

The information contained herein is provided solely for the purpose of allowing customers to operate and service Intermec-manufactured equipment and is not to be released, reproduced, or used for any other purpose without written permission of Intermec Technologies Corporation.

Information and specifications contained in this document are subject to change without prior notice and do not represent a commitment on the part of Intermec Technologies Corporation.

Document Change Record

This page records changes to this document. The document was originally released as version -001.

Version Number	Date	Description of Change
003	9/2009	Revised to add new printer support.
002	12/2007	Revised to add references to the PB50 and the communications commands.

Contents

Before You Begin

This section provides you with safety information, technical support information, and sources for additional product information.

Safety Information

Your safety is extremely important. Read and follow all warnings and cautions in this document before handling and operating Intermec equipment. You can be seriously injured, and equipment and data can be damaged if you do not follow the safety warnings and cautions.

This section explains how to identify and understand cautions and notes that are in this document.

⚠ Caution

A caution alerts you to an operating procedure, practice, condition, or statement that must be strictly observed to prevent equipment damage or destruction, or corruption or loss of data.

✎ **Note:** Notes either provide extra information about a topic or contain special instructions for handling a particular condition or set of circumstances.

Global Services and Support

Warranty Information

To understand the warranty for your Intermec product, visit the Intermec web site at **www.intermec.com** and click **Support** > **Returns and Repairs** > **Warranty**.

Disclaimer of warranties: The sample code included in this document is presented for reference only. The code does not necessarily represent complete, tested programs. The code is provided “as is with all faults.” All warranties are expressly disclaimed, including the implied warranties of merchantability and fitness for a particular purpose.

Web Support

Visit the Intermec web site at **www.intermec.com** to download our current manuals (in PDF). To order printed versions of the Intermec manuals, contact your local Intermec representative or distributor.

Visit the Intermec technical knowledge base (Knowledge Central) at **intermec.custhelp.com** to review technical information or to request technical support for your Intermec product.

Telephone Support

In the U.S.A. and Canada, call **1-800-755-5505**.

Outside the U.S.A. and Canada, contact your local Intermec representative. To search for your local representative, from the Intermec web site, click **About Us** > **Contact Us**.

Service Location Support

For the most current listing of service locations, click **Support** > **Returns and Repairs** > **Repair Locations**.

For technical support in South Korea, use the after service locations listed below:

AWOO Systems
102-1304 SK Ventium
522 Dangjung-dong
Gunpo-si, Gyeonggi-do Korea, South 435-776
Contact: Mr. Sinbum Kang
Telephone: +82-31-436-1191
E-mail: **mjyun@awoo.co.kr**

IN Information System PTD LTD
6th Floor
Daegu Venture Center Bldg 95,
Shinchun 3 Dong
Donggu, Daegu City, Korea
E-mail: **jmyou@idif.co.kr** or **korlim@gw.idif.co.kr**

Who Should Read This Manual

This document explains how to use the Intermec Printer Language (IPL) to program Intermec printers.

Before you use IPL, you should be familiar with your network, general networking terms, such as IP address, and your bar code label printers.

For information on specific IPL commands, see the ***Intermec Printer Language (IPL) Command Reference Manual***.

Related Documents

The Intermec web site at **www.intermec.com** contains our documents (as PDF files) that you can download for free.

To download documents

1. Visit the Intermec web site at **www.intermec.com**.
2. Click **Support** > **Manuals**.
3. Use the **Product Category** field, the **Product Family** field, and the **Product** field to help you locate the product whose documentation you want to download.

1

Introduction to IPL Commands

This chapter introduces the Intermec Printer Language (IPL) and includes these topics:

- **What Is IPL?**
- **Which Printers Support IPL?**
- **About IPL Commands**
- **Printing Labels with IPL Commands**
- **Learning the Structure of IPL Command Strings**
- **Sending IPL Commands to the Printer**
- **Switching Between Printer Operating Modes**

What Is IPL?

Intermec Printer Language (IPL) is one of the programming languages that has been developed for use with Intermec printers. IPL is an easy-to-use programming language that lets you:

- design formats (templates) for bar code labels, download formats to the printer, or modify formats stored in the printer.
- download data to fill in a bar code label (or RFID tag information) and print the label.
- enable or disable printer features and options, or abort print jobs and reset the printer.
- query the printer for hardware diagnostic data and status of print jobs.

This Developer's Guide includes information on how to use IPL to develop applications for your Intermec printer. For information on specific IPL commands, see the ***IPL Command Reference Manual***.

Which Printers Support IPL?

IPL is supported by these Intermec printers:

- 3240, 3400, 3440, 3600, 4100, 4400, 4420, 4440, 4630, 4830, 7421, PC41
- F2, F4
- PD/PF/PM/PX series: PD41, PD42, PF2i, PF4i, PM4i, PX4i, PX6i
- PB series: PB20, PB21, PB22, PB31, PB32, PB50, PB51

The PD series supports IPL v10.0 and later. The PF/PM/PX series support IPL v2.0 and later. The PB50 supports IPL v11.0 and later. Other Intermec printers support earlier versions of IPL. To locate the correct IPL version for your printer, visit **www.intermec.com**, choose **Support** > **Downloads**, and choose your printer from the list.

About IPL Commands

There are six types of IPL commands:

- **Communications commands** set printer communications parameters for 802.1x, Bluetooth, Serial, and Wired and Wireless LAN communications.
- **Print commands** download data to the printer, print bar code labels or RFID tag labels, or pass data to RFID tag formats. The printer must be placed in Print mode to use these commands.
- **Program commands** send label format information to the printer. RFID program commands create and define tag fields and data sources. The printer must be in Program mode to use these commands.
- **Configuration commands** enable or disable printer features and options. The printer must be in Print mode to use these commands.

- **Test and Service commands** query the printer for hardware diagnostic data and print job status. The printer must be in Test and Service mode to use these commands.
- **Immediate commands** abort print jobs, return the status of print jobs, and reset the printer. The printer can be in any mode to use these commands, and executes these commands immediately even if other IPL commands are waiting to be executed.

Printing Labels with IPL Commands

Follow this basic procedure to print bar code labels using IPL:

1. Use IPL commands to design the format for the bar code label.

   If you do not specify a format, the printer assumes you want to use the default format (format 0), which is stored permanently in the printer. The printer prints the label using that format.

2. Send the format to the printer as a series of IPL command strings. The printer stores the format in memory.

   You can store multiple formats on your printer. For more information, see the "Format, Create or Edit" command in the ***IPL Command Reference Manual***.

3. Send a command string to the printer that specifies the data for the fields of the format, and then prints the label.

Note: It is not always necessary to send the format and the data to the printer separately. You can send fixed data formats or even variable data formats that include the data in the same file.

Learning the Structure of IPL Command Strings

An IPL command string is a text string that you send to the printer. A command string is a combination of ASCII control codes (or their readable equivalents), the syntax for the specific command, and associated data.

For example, a simple command string can look like:

```
<STX>E2;F2;T4<ETX>
```

where:

IPL Command Structure

Syntax	Definition
`<STX>`	is the readable equivalent of the start of text character. Command strings must always begin with the start of text ASCII command code or its equivalent in readable characters. Readable characters must be enclosed in angle brackets. For more information on using ASCII command codes or their readable equivalents, see the next section, "Using ASCII Control Characters."
`E2`	is the syntax and data for the IPL command "Format, Erase."

IPL Command Structure (continued)

Syntax	Definition
;	is the command terminator character. Use ; to separate commands and associated data.
F2	is the syntax and data for the IPL command "Format, Create."
T4	is the syntax and data for the IPL command "Bitmap User-Defined Font, Clear or Define." Because this is the last command in the string, you do not need to follow it with ;.
<ETX>	is the readable equivalent of the end of text character. Command strings must always end with the end of text ASCII character or its equivalent in readable characters. Readable characters must be enclosed in angle brackets.

All command syntax is case sensitive.

Note: When the printer is in XON/XOFF mode, you can send data and print multiple labels without using the <ETX> character.

Using ASCII Control Characters

Command strings always include ASCII control characters. To use ASCII control characters, you can:

- enter a control code. Some word processing programs allow you to represent ASCII control characters by entering the appropriate control code.
- type the readable characters. Instead of control codes, you type the appropriate string of characters enclosed in angle brackets.

For example, an IPL command string must begin with the start of text control character. The control code for this character is 0x02 (hex notation, also represented by ^B in some text editing programs), and the readable characters for this control code are <STX>.

This example shows a command string written with readable characters and with control codes:

Command String

Type	Command String
Readable characters:	<STX><SI>g1,567<ETX>
Control codes:	^B^Og1,567^C

For a list of ASCII control characters, control codes, and definitions, see the **"Full ASCII Control Characters Table" on page 78**.

If the start of text character is the actual control code, then all other control characters in the same command string must be in control code format. The printer ignores any control characters represented by readable characters.

Conversely, if the start of text character is represented by readable characters (<STX>), then all other control characters in the same string must be represented by readable characters. The printer ignores any control characters represented by control codes.

For simplicity, this manual uses readable characters in all examples and syntax.

Note: Using readable characters consumes more memory space and takes more time to process, but each command string is more easily read and edited.

Creating Command Strings

IPL command strings can be as long as needed to perform a task. For example, this series of short command strings creates and defines a label format:

```
<STX><ESC>P<ETX>
<STX>E3;F3;<ETX>
<STX>H0;o81,100;f0;c0;d0,16;h1;w1;<ETX>
<STX>H1;o81,120;f0;c0;d0,16;h1;w1;<ETX>
<STX>H2;o81,150;f0;c2;d0,14;h1;w1;<ETX>
<STX>H3;o81,190;f0;c2;d0,16;h1;w1;<ETX>
<STX>B4;o81,0;f0;c0,1;h50;w1;d0,11;i0;p@;<ETX>
<STX>R<ETX>
```

This longer command string combines all eight of the previous command strings into a single string:

```
<STX><ESC>P;E3;F3;H0;o81,100;f0;c0;d0,16;h1;w1;H1;o81,120;f0;c0;
d0,16;h1;w1;H2;o81,150;f0;c2;d0,14;h1;w1;H3;o81,190;f0;c2;d0,16;
h1;w1;B4;o81,0;f0;c0,1;h50;w1;d0,11;i0;p@;
R<ETX>
```

The printer interprets each example exactly the same way.

For more information on creating and testing command strings, see Chapter 2, **"Designing Bar Code Labels."**

After you determine the IPL command strings you need to accomplish a task, save the strings as an ASCII text file for future use.

To create a text file of IPL command strings

1. Determine the IPL command strings you need to accomplish a task.
2. Open a text editor or word processor and type the command strings.
3. Save the file as an ASCII text file.

Sending IPL Commands to the Printer

You can send IPL commands to the printer by using a communications program such as Windows HyperTerminal.

Follow the next procedure to connect the printer to a PC and send a text file of IPL command strings to the printer.

To send a text file of IPL commands to the printer

1. Connect the printer to the serial port (COM1) on your desktop PC. For more information, see the user's manual for your printer.

2 On the desktop PC, start the HyperTerminal application.

3 Configure HyperTerminal for a serial connection using these parameters:

HyperTerminal Parameters

Parameters	Values
Baud rate	9600
Data bits	8
Parity	None
Stop bits	1
Flow control	XON/XOFF

These serial connection characteristics are the default for IPL printers. If you have changed the communication settings on your printer, change the HyperTerminal connection settings accordingly.

Note: If you receive the “write fault error” error message, it indicates that either you are sending the data to the wrong COM port or your cable does not support hardware flow control.

1 From the Main menu bar, select **Transfer**.

2 From the Transfer list box, select **Send Text File**.

3 In the Send Text File dialog box, locate your .txt file and click **Open**.

HyperTerminal sends the .txt file to your printer. Your command text file will either change a configuration setting in the printer or print your label format.

Switching Between Printer Operating Modes

When you send IPL commands to the printer, make sure you have placed the printer in the correct operating mode to interpret those commands:

- If you are sending print or configuration commands, the printer must be in Print operating mode to interpret the commands.
- If you are sending program commands, the printer must be in Program mode.
- If you are sending test and service commands, the printer must be in Test and Service mode.

Print mode and Program mode are two different operating modes of the printer. Before you download information to the printer, make sure that you are in the correct mode:

- Use Program mode to define formats, pages, fonts, and characters.
- Use Print mode to print labels, to download data to the printer, to download configuration commands to the printer, or to upload information from an RFID tag.

To place a printer in Program mode

- Send this command string:

```
<STX><ESC>P<ETX>
```

Send this command every time that you download formats, even if you think the printer is already in Program mode. If the printer is already in Program mode, it ignores this command.

To place a printer in Print mode

- Send this command string:

  ```
  <STX>R<ETX>
  ```

  Send this command before each set of data or as your last format command. If the printer is already in Print mode, it ignores this command.

Note: The R command may be treated as data if the data to the printer does not include a <CAN> or field pointer to clear the fields.

To place a printer in Test and Service mode

- Send this command string:

  ```
  <STX><ESC>T<ETX>
  ```

  Send this command before any test and service commands. Typically, you will not need to use test and service commands when defining label formats.

2

Designing Bar Code Labels

This chapter explains how to design and print your own labels using IPL commands. It covers the basic elements of label design and includes these topics:

Introduction to Label Formats

To print a label on an Intermec printer, you must create a label format, send the format to the printer, send data to fill in the fields in the format, and then print the label.

A format is a template that defines how the information prints on a label. You define a format by sending IPL command strings to the printer. The printer stores the format in its RAM or flash module. Formats can be edited with IPL commands if you place the printer in Program mode.

Note: If you intend to regularly reload a format, you can use the temporary format parameter (*) with some printers. The use of format * results is an optimal use of flash-based storage memory because the printer stores it in RAM and deletes it when the printer power is turned off.

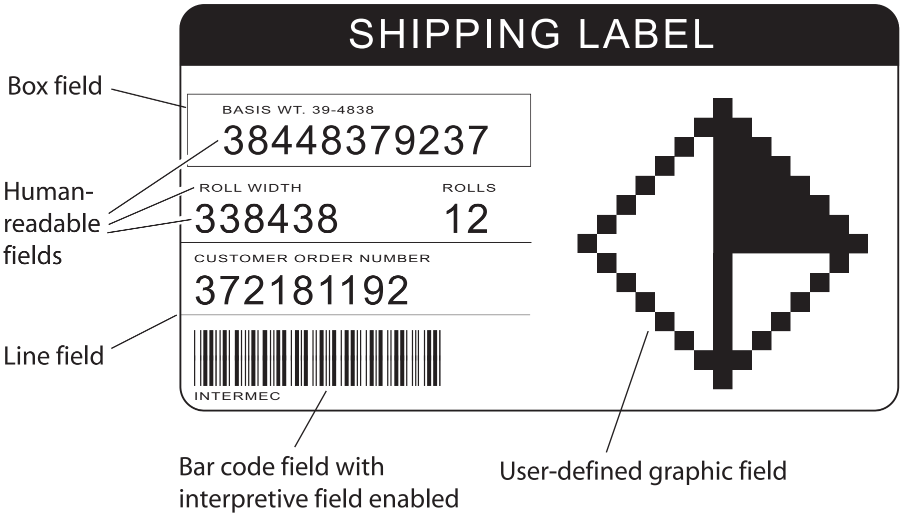
About Field Types

A bar code label format is composed of several different fields that hold different types of data. The fields may differ in size, location, and orientation, as well as data type. You define information to be printed on the label as a field in the format. After the fields are defined, you can pass data to the fields and print the label.

The IPL field types are:

- Human-readable fields
- Bar code fields
- Line and Box fields
- User-defined character (UDC) or graphics fields

This section describes each field type.



Sample Label: *This illustration shows a typical bar code label and how the different field types are used.*

Human-Readable Fields

Human-readable fields contain information or data that is printed in one of the printer's internal fonts, or in a user-defined font. Available fonts depend on your printer. Internal fonts on your printer include:

- Standard bitmap fonts measured in dots
- Bitmap fonts recognized by optical character recognition (OCR) applications
- Bitmap fonts measured in point sizes
- Bitmap monospaced fonts
- Outline fonts

Font character size can be changed by using the width and height magnification commands, or with the pitch and point-size commands. The fonts themselves remain unchanged. See Appendix B, "Character Sets," for the complete character set for each font.

Intermec printers support nine different international character sets for each command set mode. The printer may also contain character sets for IBM translation and code pages.

Note: In Emulation mode, international character substitution is compatible with Intermec 8336/8646 printers. In Advanced mode, the substitution complies with ISO standards.

Bar Code Fields

Bar code fields include data and information to be printed in a specific symbology. Intermec printers support these symbologies:

Intermec Supported Printer Symbologies

Printer Symbologies		
Aztec	Code 128	MicroPDF417
Codabar	Data Matrix	PDF417
Code 2 of 5	EAN.UCC Composite	Planet
Code 11	HIBC Code 128	Postnet
Code 16K	Interleaved 2 of 5	QR Code
Code 39	ISBT 128	RSS
Code 49	JIS-ITF	UPC/EAN Codes
Code 93	Maxicode	USPS4CB

Bar code fields can be accompanied by an interpretive field, which shows the bar code data in readable characters.

You can use IPL commands to define the height and width magnification of bar code fields, set the data source for each field, change the physical orientation of fields, and to enable or disable the interpretive field.

Line and Box Fields

You can use lines or boxes in a label format to separate fields and create borders on the label. Use IPL commands to define the location and size of line and box fields in a label format.

User-Defined Character or Graphic Fields

These fields contain graphics or bitmap characters. You can either download the graphic to the printer before printing labels, or you can create the graphic in a command string.

To download graphics or characters to the printer, you need to use:

- the Intermec PrintSet application. PrintSet converts the graphic into the appropriate format for your printer.
- a third-party label-generation application. These applications convert your graphics file to a format that your printer can interpret, and send the file to the printer.

To create the graphic in a command string, you use the IPL command "Graphic or UDC, Define." For help, see Chapter 3, **"Working with Fonts and Graphics."**

The maximum size of a user-defined graphic is limited by the capacity of your printer and can be increased if you install expansion RAM.

You can use Direct Graphics mode to reduce the time it takes to download and print an image. For more information, see **"Using Direct Graphics Mode" on page 64**.

Working With Fields

In addition to understanding the different types of printable fields on the printer, you need to know how to arrange them to define or change the format of a label. This section describes how you use IPL commands to position, size, rotate, and edit label fields.

Editing Field Zero (H0)

By default, the printer automatically creates a human-readable field numbered zero (H0) each time you create a format by using the "Format, Create" command. H0 is created with default parameters as defined in the next table.

Defaults for the H0 Field

Command	Definition
`o0,0;`	Field origin is 0,0.
`f0;`	Field direction is horizontal with respect to the label motion from printer.
`h2;`	Field height magnification is 2.
`w2;`	Field width magnification is 2.
`c0;`	Selects the 7x9 standard font.
`b0;`	Selects no border around human-readable field.

Defaults for the H0 Field (continued)

Command	Definition
`r0;`	Selects horizontal orientation of characters.
`d0,30;`	You enter variable data in Print mode. The maximum number of characters you can enter into this field is 30.

This automatically created field is always a human-readable field and is always assigned field number 0. Because a format must always have at least one defined field, you cannot delete field 0 if it is the only field in the format.

To change field 0 to a different type of field (for example, a bar code field), you have to create a temporary field of any type, delete field 0, create field 0 with a different type, and then delete the temporary field. The next example shows how this is done:

```
<STX><ESC>P;E1;F1;L39;D0;B0;D39;<ETX>
```

The next table describes each command in the string.

Command String Values

Command	Definition
`<ESC>P;`	Enters Program mode.
`E1;`	Erases format 1.
`F1;`	Creates format 1.
`L39;`	Creates line field (temporary field).
`D0;`	Deletes field 0.
`B0;`	Creates bar code field 0.
`D39;`	Deletes the temporary line field.

Numbering Fields in a Format

Be sure you assign a different field number to each field in a format. A format can have up to 400 fields, numbered 0 to 399.

Keep in mind that you enter data into fields according to their numeric value if you use a <CR> to separate the data. In other words, the first string of data you enter goes into the lowest numbered field, the second string of data you enter goes into the next lowest numbered field, and so on.

About Interpretive Fields and Field Locations

Each bar code field can have an interpretive field associated with it. For example, bar code field B33 would have an interpretive field I33.

Each interpretive field uses a field location from the available 400 fields. To avoid interference with other user-defined fields, interpretive fields are automatically assigned numbers beginning with the highest available field number. For example, if B33 is the first bar code field with interpretive text, then its interpretive field, I33, is assigned location 399. You can see how the printer uses the fields internally by uploading the format and examining where the interpretives appear. If you need to use more fields than IPL can accommodate in a single format (because of interpretive fields), you can use pages to combine several formats on one label.

Editing Existing Fields

You can change a specific field in a format by sending a command in Program mode. You can modify just the incorrect format fields instead of having to download the entire format again.

When the printer is in Program mode, it uses a field pointer to point to the field to be modified. The pointer continues to point to the most recently selected field until you select a different format or field.

For the next example, assume that you used the following format:

Sample Format

Command	Description
`<STX><ESC>C<ETX>`	Selects Advanced mode.
`<STX><ESC>P<ETX>`	Enters Program mode.
`<STX>E3;F3;<ETX>`	Erases format 3, Creates format 3.
`<STX>H0;o80,100;f0;c0;d0,16;h1;w1;<ETX>`	Creates field H0.
`<STX>H1;o80,120;f0;c0;d0,16;h1;w1;<ETX>`	Creates field H1.
`<STX>H2;o80,150;f0;c2;d0,14;h1;w1;<ETX>`	Creates field H2.
`<STX>H3;o80,190;f0;c2;d0,16;h1;w1;<ETX>`	Creates field H3.
`<STX>B4;o80,0;f0;c0,1;h50;w1;d0,11;i0;p@;<ETX>`	Creates field B3.
`<STX>R;<ETX>`	Saves and exits to Print mode.

To change the height and width of field 3 to 2 dots, download this command string:

```
<STX><ESC>P;F3;H3;h2;w2;R;<ETX>
```

The following table describes each command in this string:

Field Editing Commands in the Sample Format

Command	Description
`<ESC>P;`	Enters Program mode.
`F3;`	Accesses format number 3 from memory.
`H3;`	Accesses field 3.
`h2;`	Sets the height to 2 dots.
`w2;`	Sets the width to 2 dots.
`R;`	Returns to Print mode.

Deleting Fields

It is possible to delete any field from a format unless the format only has one field. For example, use the following command string to delete field 3 from format 4:

```
<STX><ESC>P;F4;D3;R<ETX>
```

The following table describes each command in this string:

Command String Descriptions

Command	Description
`<ESC>P`	Enters Program mode.
`F4`	Accesses format 4.
`D3`	Deletes field 3.
`R`	Returns to Print mode.

Positioning Fields

Using the IPL command language to position fields is the trickiest part of designing labels. Since you cannot tell exactly how the field looks until it prints, you may need to make several test prints before you get the field positioned correctly.

For all types of fields, determine the print position by defining the coordinates of the upper left corner of the unrotated field.

The field origin is the upper left corner of an unrotated field. To define the coordinates of the field origin, use the “Field Origin, Define” command (o*x*,*y*), where:

Field Origin, Define Command

Coordinates	Definition
o	is the command syntax.
x	is the distance (in dots) from the left edge of the label to the field origin.
y	is the distance (in dots) from the top edge of the label to the field origin.

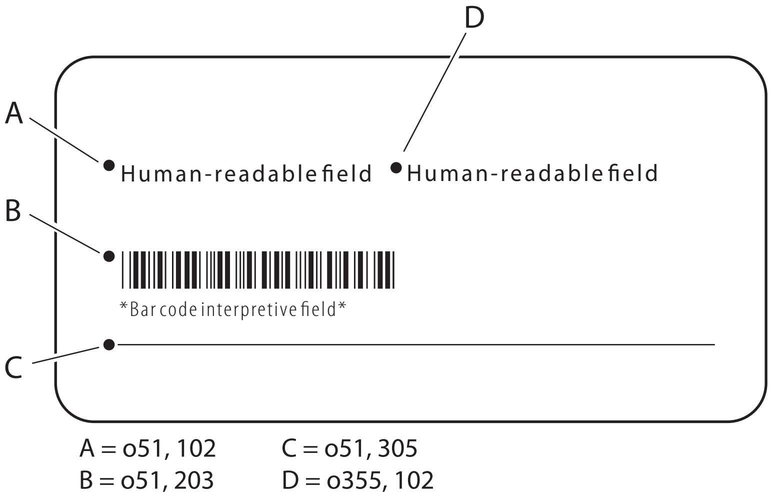
The actual measurements for *x* and *y* depend on the size of the printhead in your printer. The next table lists printhead sizes and the corresponding number of dots per mm.

Dots/mm by Printhead Size

Printhead Size	Dots/mm
203 dpi	8
300 dpi	11.8
406 dpi	16

Note: On the 4400, 7421, and PC41 printers, be sure to use the "Label Width, Set" command to set the printer for the correct label width, or the *y* setting will be incorrect.

For example, to position a field to print approximately 0.25 inch from the left side and 0.5 inch from the top of your label, the "Field Origin, Define" command value is o51,102.



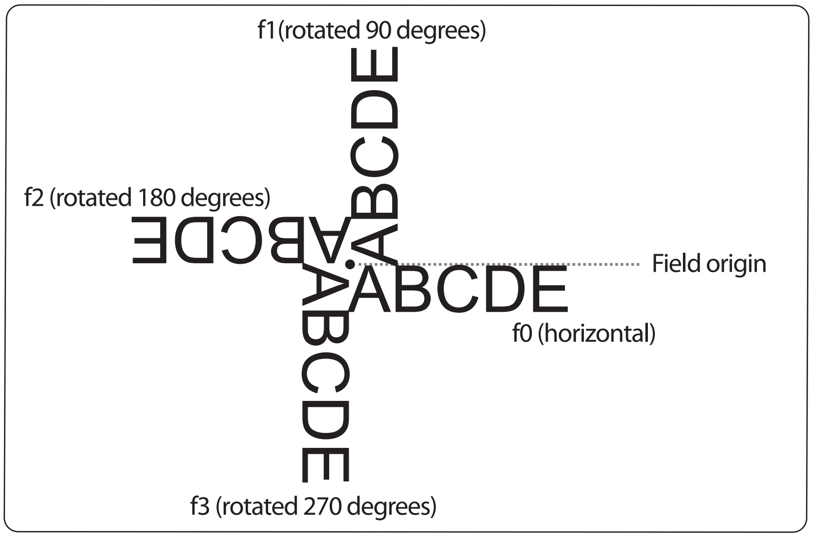
Field Positioning: *This illustration shows the relative positions of four fields with different origin points.*

Note: If you are operating your printer in Emulation mode, the dot sizes are doubled (101 dots per inch or 4 dots per mm), so the origin for 1 inch from the top of the label and 0.5 inch from the left side is o25,51.

Rotating Fields

You can rotate any type of printable field in increments of 90 degrees counterclockwise around the field origin. To position a rotated field, remember that the field origin remains on the corner where it was before you rotated the field. If you rotate a field 90 degrees counterclockwise, the origin that was at the upper left corner is now at the lower left corner. Use the "Field Direction" command (syntax f*n)* to define the field rotation. See the ***IPL Command Reference Manual*** for more information.

- To rotate a field 90 degrees, you must position the lower left corner of the rotated field.
- To rotate a field 180 degrees, you must position the lower right corner of the rotated field.
- To rotate a field 270 degrees, you must position the upper right corner of the rotated field.



Field Rotation: *This illustration shows the effects of rotating a field by using the Field Direction command.*

Scaling Fields

The size of a field is determined by the font or graphic you use and the field magnification factors you apply. You can use the height and width magnification commands to scale each of these fields even further.

Magnifying Fonts and Character Fields

Internal printer fonts already have specific sizes. For example, the letters in font c0 are 7 dots wide by 9 dots high, with a 1-dot gap between characters. If you design a field that prints 10 letters in font c0, the field will be 79 dots wide by 9 dots high.

By applying magnification factors (h for height and w for width), you can increase a field's height or width. If you increase the height to 2 (h2) for the field described above, the field height doubles, and the final field prints 79 dots long by 18 dots high. If you change the height magnification to h3, the field height triples, and the field prints 79 dots by 27 dots.

The default human-readable field H0 prints the 7 x 9 font as follows (assuming you enter the word "example" as data):

EXAMPLE

When you apply a magnification factor of 3 to human-readable field H0, the font now prints the image below:

EXAMPLE

Increasing the width of a text field to 2 makes each letter in the field twice as wide. If you did this to the example above, with field height h2, the final field would print 158 dots wide by 18 dots high.

When you magnify a bitmap font, the edges of the characters become jagged. If you want to print large text characters, use an outline font.

Magnifying Bar Code Fields

You can also use height and width commands to modify bar code fields, but the commands are defined differently for use with bar codes.

For bar code fields, the height magnification is the actual dot height of the bar code. If you set a height magnification of 20, the height of the bar code field will be 20 dots.

Printing narrow bar codes conserves space on each label as well as media; however, if you plan to scan bar codes from a distance, you may need to magnify the bar code widths.

The width magnification factor for bar code fields refers to the width of the narrowest element of the bar code. When you specify a narrow element width of 3, the width of the narrowest element in the symbology is 3 dots wide. The spaces and large element widths are magnified according to preset ratios for each symbology.

Note: You can only print a bar width of 1 if you are printing in drag mode (bars perpendicular to the print head). If you select a width of 1 in picket mode (bars parallel to the print head), the printer defaults to 2.

The default height for bar code fields is 50 dots, and the default width for narrow elements is 1 dot.

Note: If you are using the POSTNET symbology, follow the rules for magnifying fonts.

Specifying Data for Fields

After you design a format for a bar code label, you need to provide the data to be printed in human-readable or bar code fields.

For human-readable fields, you can specify data in two ways:

- You can add the data to the command string that defines that field. This is a good choice for text that is constant from label to label.
- You can add command strings after the label format that include the data to be printed. This is a good choice for text that may change from label to label, such as incremental numbering. You need to place the printer into Print mode before adding these command strings.

For bar code data, you place the printer into Print mode, and then add command strings that include the data to be printed.

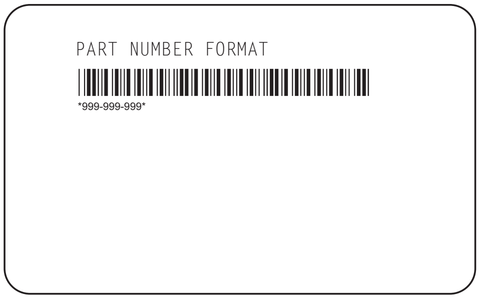
Example of Constant Data

This example defines a simple bar code label format that prints a part number in a bar code field with an interpretive field. The data (part number) remains constant for all labels printed using this format.

Label Format Example - Constant Data

Command String	Definition
`<STX><ESC>P<ETX>`	Places the printer in Program mode.
`<STX>E2;F2<ETX>`	Erases current format 2 and creates new format 2.
`<STX>H0;o200,50;d3,PART NUMBER FORMAT<ETX>`	Changes the default settings for field 0: places the origin at 200,50, and specifies the text "PART NUMBER FORMAT" for the field data.
`<STX>B1;o200,200;i1;d0,13<ETX>`	Defines a new bar code field at origin 200,200, enables an interpretive field, specifies that data for this field will be transmitted when the printer is in Print mode, and sets a data length of 13 characters. All other field settings are default values.
`<STX>R<ETX>`	Places the printer in Print mode.
`<STX><ESC>E2<CAN><ETX>`	Selects format 2 and clears all previous data.
`<STX><ESC>F1<LF>*999-999-999*<ETX>`	Selects field 1 (the new bar code field) and specifies the data to be printed in that field (*999-999-999*).
`<STX><ETB><ETX>`	Prints the bar code label.

This label format looks like this:



PART NUMBER FORMAT

999-999-999

Constant Data Example

Example of Changing Data

The next example shows how to define a simple bar code label format that prints a part number in a bar code field and an incremented order number in a human-readable field. The order number changes for every label printed in this format. You use the "Numeric Field Separator" command (syntax <FS>) to specify the numeric data to be incremented, and the "Field Increment, Set" command (syntax <ESC>I) to specify the increment value.

Label Format Example - Changing Data

Command String	Definition
`<STX><ESC>P;E2;F2<ETX>`	Places the printer in Program mode, erases current format 2, and creates a new format 2.
`<STX>H0;o200,50;d3,ORDER NUMBER FORMAT<ETX>`	Changes the default settings for field 0, placing the origin at 200,50, and specifying the text "ORDER NUMBER FORMAT" for the field data.
`<STX>B1;o200,100;i1;d0,13<ETX>`	Defines a bar code field at origin 200,200, enables an interpretive field, specifies that data for this field will be transmitted when the printer is in Print mode, and sets a data length of 13 characters. All other field settings are defaults.
`<STX>H2;o200,350;d3,ORDER NUMBER<ETX>`	Defines a human-readable field at origin 200,350, and specifies the text "ORDER NUMBER" for the field data. All other field settings are defaults.
`<STX>H3;o200,400;d0,30<ETX>`	Defines a human-readable field at origin 200,400, and specifies that data for this field will be transmitted when the printer is in Print mode. All other field settings are defaults.
`<STX>R<ESC>E2<CAN><ETX>`	Places the printer in Print mode, selects format 2, and clears all previous data.
`<STX><ESC>F1<LF>*999-999-999*<ETX>`	Selects field 1 (the bar code field) and specifies the data to be printed in that field (*999-999-999*).
`<STX><ESC>F3<FS>1000<FS><ESC>I5<ETX>`	Selects field 3 and specifies the value"1000" for the field data. **<FS>** is the syntax for the "Numeric Field Separator" command, and indicates that the value is to be incremented for each label. **<ESC>I** is the syntax for the "Increment Value, Set" command, and sets an incrementation value of 5.
`<STX><RS>3<ETB><ETX>`	**<RS>** is the syntax for the "Quantity Count, Set" command, and specifies that 3 labels are to be printed. **<ETB>** prints the labels. The order numbers on these labels should be 1000, 1005, and 1010.

Example Using the "Next Data Entry Field, Select" Command

This example shows how to define a label with four human-readable fields.

If you do not specify the field to which data is assigned, the printer automatically assigns that data to the lowest numbered field. In this example, the field is not specified, so the printer assigns the first data string to field 1, the next data string to field 2, and so on. The data strings are separated by the "Next Data Entry Field, Select" command (syntax <CR>).

Label Format Example Using <CR> Command

Command String	Definition
`<STX><ESC>P;E6;F6<ETX>`	Places the printer in Program mode, erases current format 6, and creates a new format 6.
`<STX>H0;o100,100;d0,20<ETX>`	Changes the default settings for field 0: places the origin at 200,50, and specifies that data for this field will be transmitted when the printer is in Print mode. All other field settings are defaults.

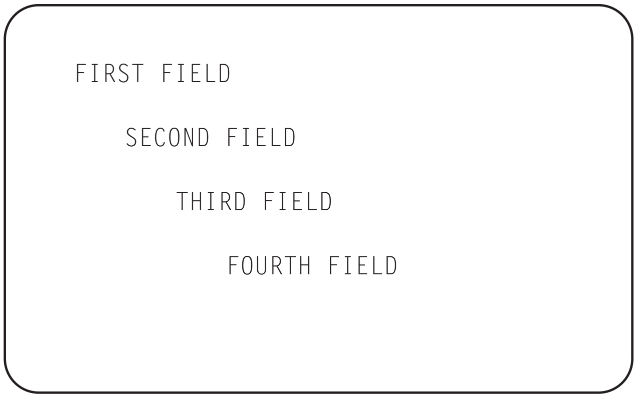
Label Format Example Using <CR> Command (continued)

Command String	Definition
`<STX>H1;o200,200;d0,20<ETX>`	Defines a human-readable field at origin 200,200. All other settings are identical to H0.
`<STX>H2;o300,300;d0,20<ETX>`	Defines a human-readable field at origin 300,300. All other settings are identical to H0.
`<STX>H3;o400,400;d0,20<ETX>`	Defines a human-readable field at origin 400,400. All other settings are identical to H0.
`<STX>R<ESC>E6<CAN><ETX>`	Places the printer in Print mode, selects format 6, and clears all previous data.

`<STX>FIRST FIELD<CR>`
`SECOND FIELD<CR>`
`THIRD FIELD<CR>`
`FOURTH FIELD<ETX>`
	Specifies the data to be assigned to the four fields requiring input. <CR> separates the individual data strings. The first data string (FIRST FIELD) is assigned to field 0, the second (SECOND FIELD) to field 1, and so on.

`<STX><ETB><ETX>`	Prints the label.

The label looks like this when printed:



```
FIRST FIELD
    SECOND FIELD
        THIRD FIELD
            FOURTH FIELD
```

Working With RFID Tags

If a printer has an RFID module installed, the RFID mode is automatically turned on. You need to configure the printer, declare a format that includes commands to read and write to the tag, and print the format to a label. The printer aligns the tag with the antenna, executes the RFID commands, repositions the label, prints the image buffer to the label, and then positions the next label for printing.

How to Write Data to an RFID Tag

To write data to an RFID tag you need to do two things: specify an area in the tag's memory and define the data that will be written to the tag. To specify an area in the tag's memory, you define an RFID tag write field. This field specifies which segment is written to, the start position, length of data (in bytes) and the format of the data.

How Data is Stored on an RFID Tag

Data written to the RFID tag is stored from the left starting with the byte that is defined as the start of field byte and then going the length of the defined field (in bytes).

Writing Hex or ASCII Formats

If the tag format is hex or ASCII and the field source data string is shorter than the allocated number of bytes, the data string’s least significant bytes (from the right) are padded with zeroes until they fill the length of the field.

If the field source data string is longer than the allocated number of bytes, the error <EOT> is sent and nothing is written to the tag. If auto-transmit level 3 is enabled, an <EOT> status response is returned to the host.

Writing Numerical Formats

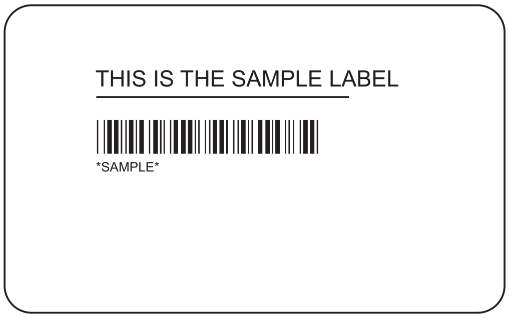
If the tag format is numerical (NUM), the data is stored as a numerical value. In this tag format, if the field source data string is shorter than the allocated number of bytes, the data string’s most significant bytes (from the left) are padded with zeroes until they fill the length of the field.

Note: The number of bytes needed to represent a numerical value is never greater than the number of bytes needed to represent the number as a data string.

Creating a Sample Label Format

In this example, you will design a simple label that includes a human-readable field, a line field, and a bar code field.

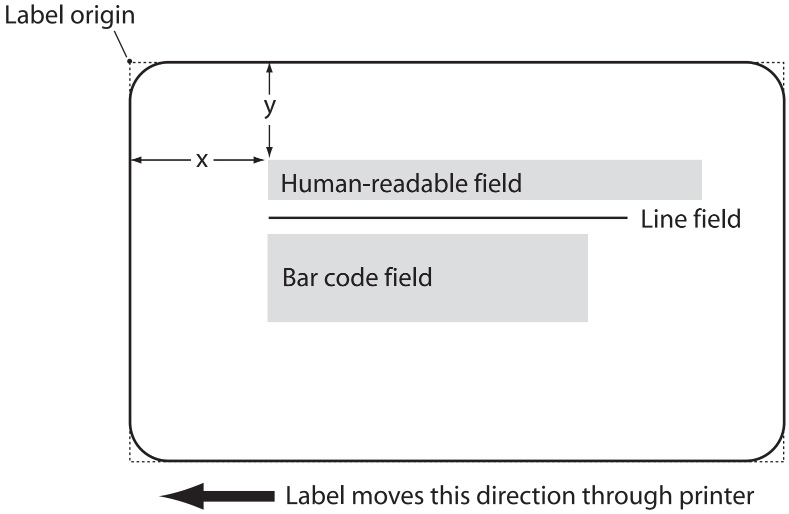
On a blank label, sketch a format that looks roughly like the sample. Use a human-readable field and a barcode field, and separate them with a line field.



Sample Label: *This sample uses a human-readable field, a line field, and a bar code field with interpretive field enabled.*

Positioning Fields in a Label Format

Each field in a label format is positioned relative to the label origin, which is the top left corner of the label. To set the origin of a field on a label, you assign x and y coordinates to the upper left corner of the field, where *x* is the distance (in dots) from the left edge of the label, and *y* is the distance (in dots) from the top edge of the label.



Positioning Fields: *This illustration shows the relationship of fields on a label to the label origin.*

To determine the field origin

1. Measure the distance (in mm) from the upper left corner of the human-readable field to the left edge of the label.
2. To determine the value for x, multiply the distance times the number of dots per mm for your printhead. See the next table for more information.

Dots/mm by Printhead Size

Printhead Size	Dots/mm
203 dpi	8
300 dpi	11.8
406 dpi	16

For example, if you are designing the label for a printer with a 203 dpi printhead and the distance is 25 mm:

25 mm x 8 dots/mm = 200. The value for *x* is 200.

3. Measure the distance (in mm) from the top left corner of the human-readable field to the top edge of the label.
4. To determine the value for *y*, multiply the distance times the number of dots for your printhead.

For example, if the distance is 12.5 mm:

12.5 mm x 8 dots/mm = 100. The value for *y* is 100.

The origin position for the human-readable field is 200, 100.

5 Repeat steps 1 to 4 to find the origin points for the line field and for the bar code field.

Note: On the 4400, 7421, and PC41 printers, be sure to use the label width command to set the actual width of your labels; otherwise, the *y* origins will be incorrect.

Creating Command Strings for the Label Format

Next, write the IPL command strings to create the label format.

You use IPL commands to create and define each field in the format. For example, to create a human-readable field, you use the "Human-Readable Field, Create or Edit" command.

Each type of field is defined by several parameters. For example, a human-readable field includes settings for field origin point, font, direction, height and width, data source, and number of characters. For more information, see the command in the *IPL Command Reference Manual*.

To create the command strings

1 Open a text editor or word processor application.

2 Type these three command strings to choose Advanced mode, place the printer in Program mode, and create and define the label format:

```
<STX><ESC>C<ETX>
<STX><ESC>P<ETX>
<STX>E4;F4<ETX>
```

The next table explains these values:

Values for Sample Label Command Strings

Text	Description
`<STX>`	ASCII start of text control code in readable characters. Each command string must begin with the start of text control code or its equivalent in readable characters.
`<ESC>C`	**<ESC>C** is the syntax for the "Advanced Mode, Select" command.
`<ETX>`	ASCII end of text control code in readable characters. Each command string must end with the end of text control code or its equivalent in readable characters.
`<ESC>P`	**<ESC>P** is the syntax for the "Program Mode, Enter" command. You need to place the printer in Program Mode to create and define a label format.
`E4`	**E** is the syntax for the "Format, Erase" command. This command clears format 4 from the printer memory.
`F4`	**F** is the syntax for the "Format, Create" command, which creates a new, blank format 4 in the printer memory.

3 Type the following text to create and define the human-readable field:

```
<STX>H0,o200,100;c25;f0;h20;w20;d0,30<ETX>
```

The next table explains these values.

Values for Human-Readable Field Command String

Text	Description
H0	**H** is the syntax for "Human-Readable Field, Create or Edit" command, with field number 0.
o200,100	**o** sets the origin point for this field, where x = 200 and y = 100. For more information, see **"Positioning Fields in a Label Format" on page 23**.
;	; is the command terminator character, which is used to separate each command and its associated data with the next command.
c25	**c** selects the font to be used in this field. 25 = outline font.
f0	**f** sets the field direction. 0 = horizontal.
h20	**h** sets the height magnification (in this case, 20).
w20	**w** sets the width magnification value (in this case, 20).
d0,30	**d** sets the data source and length for this field. 0 chooses data field 0, with a maximum length of 30 characters.

4 Press **Enter** and type the following text to create and define the line field:

```
<STX>L1;o200,200;f0;l575;w5;<ETX>
```

The next table explains these values:

Values for Line Field Command String

Text	Description
L1	**L** is the syntax for the "Line Field, Create or Edit" command, with field number 1.
o200,200	**o** sets the origin of the line field, where x = 200 and y = 200.
;	; is the command terminator character.
f0	**f** sets the direction of the line field (horizontal).
l575	**l** sets the length of the line in dots (575).
w5	**w** sets the width of the line in dots (5).

Note: The difference between the lowercase letter "l" and the numeral "1" is not very noticeable in the Courier font. Make sure that you enter the correct command.

5 Press **Enter** and type the following text to create and define the bar code field:

```
<STX>B2;o203,153;c0,0;h100;w2;i1;d0,10;<ETX>
```

The next table explains the bar code field command string values:

Values for Bar Code Field Command String

Text	Description
`B2`	**B** is the syntax for the “Bar Code Field, Create or Edit” command, with field number 2.
`o200,150`	**o** sets the origin of the bar code field, where x = 200 and y = 150.
`;`	**;** is the command terminator character.
`c0,0`	**c** is the syntax for the “Bar Code, Select Type” command, where 0,0 specifies Code 39, 8646 compatible, with no check digit.
`h100`	**h** is the syntax for the “Height Magnification of Bar Code, Define” command, where 100 indicates a magnification of 100 dots tall.
`w2`	**w** is the syntax for “Width of Bar Code, Define” command, where 2 indicates a magnification of 2 dots wide
`i1`	**i** is the syntax for the “Interpretive Field” command, where 1 enables the field.
`d0,10`	**d** is the syntax for the “Field Data, Define Source” command, where 0,10 indicates that the data is entered in Print mode and has a maximum length of 10 characters.

6 Press **Enter** and type the following text to define the interpretive field:

```
<STX>I2;h1;w1;c20<ETX>
```

The next table explains these values:

Values for Interpretive Field Command String

Text	Description
`I2`	**I** is the syntax for the “Interpretive Field, Define” command, with field number 2 (matches the accompanying bar code field number).
`h1`	**h** is the syntax for the “Height Magnification” command, with a magnification level of 1 dot vertically.
`w1`	**w** is the syntax for the “Width Magnification” command, with a magnification level of 1 dot horizontally.
`c20`	**c** is the syntax for the “Font Type, Select” command, where 20 specifies the 8-point monospace font.

7 Press **Enter** and type the following text to place the printer in Print mode, access format 4, and clear previous field data:

```
<STX>R<ETX>
<STX><ESC>E4<ETX>
<STX><CAN><ETX>
```

Values for Print Mode Command String

Text	Description
`R`	Places the printer in Print mode.
`<ESC>E4`	**<ESC>E** is the syntax for the “Format, Select” command. This command selects format 4.
`<CAN>`	**<CAN>** is the syntax for the “Clear All Data” command.

8 Press **Enter** and type the following text to create the data lines for the human-readable and bar code fields:

```
<STX>THIS IS THE SAMPLE LABEL<CR><ETX>
<STX>SAMPLE<ETX>
```

The next table explains these values:

Values for Data Line Command Strings

Text	Description
`THIS IS THE SAMPLE LABEL`	Data line for the human-readable field. The fields in the label format are populated in order by field number.
`<CR>`	**<CR>** is the syntax for the “Next Data Entry Field, Select” command.
`SAMPLE`	Data line for the bar code interpretive field.

9 Press **Enter** and type the following text:

```
<STX><ETB><ETX>
```

where <ETB> is the syntax for the “Print” command.

Note: The difference between the lowercase letter “l” and the numeral “1” is not very noticeable in the Courier font. Make sure that you enter the correct command.

10 Save the text file as “SampleLabel.txt”.

The label format command strings are complete and look like this:

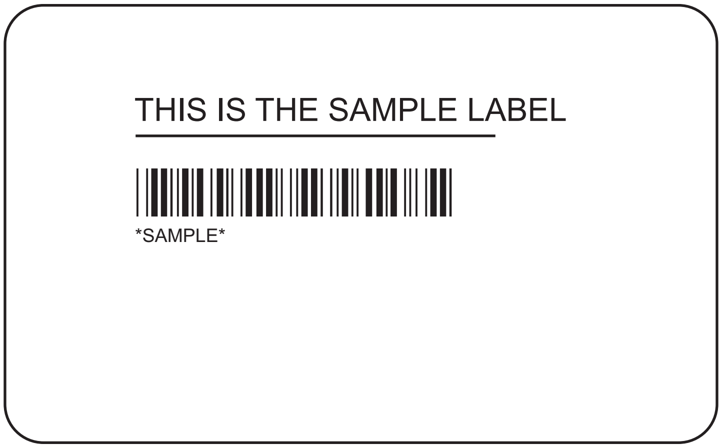
```
<STX><ESC>C<ETX>
<STX><ESC>P<ETX>
<STX>E4;F4<ETX>
<STX>H0;o200,100;f0;c25;h20;w20;d0,30<ETX>
<STX>L1;o200,150;f0;l575;w5<ETX>
<STX>B2;o200,200;c0,0;h100;w2;i1;d0,10<ETX>
<STX>I2;h1;w1;c20<ETX>
<STX>R<ETX>
<STX><ESC>E4<ETX>
<STX><CAN><ETX>
```

```
<STX>THIS IS THE SAMPLE LABEL<CR><ETX>
<STX>SAMPLE<ETX>
<STX><ETB><ETX>
```

Printing the Sample Label

To print the sample label, you can open a communications program (such as HyperTerminal) and send the text file to the printer. For more information on connecting to the printer, see your printer user's manual.

The label looks like this when printed:



THIS IS THE SAMPLE LABEL

SAMPLE

Sample Label

Complex Label Design Examples

This section includes several examples of complex bar code label designs. Use these examples as a source of ideas for your own label formats.

Example with Lines and Boxes

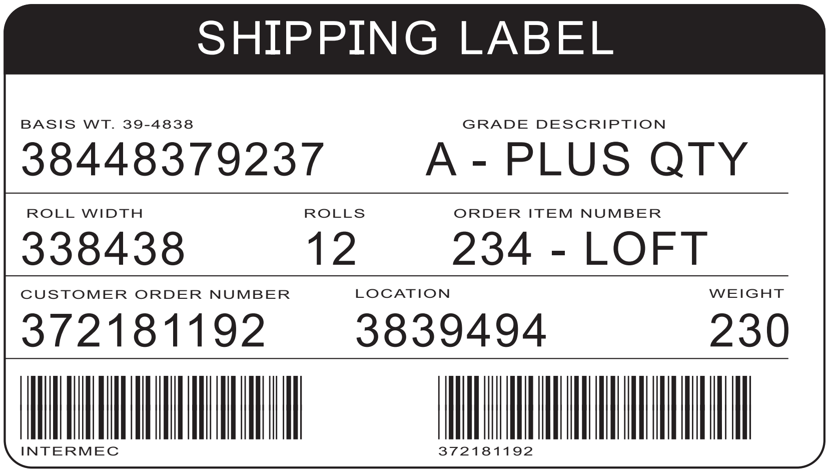
The following example uses horizontal lines to separate fields on a label and uses a box field to make a label border. This format uses bold text to highlight the lines that contain line or box fields.

Command Strings

```
<STX><ESC>C0<ETX>
<STX><ESC>P<ETX>
<STX>E4;F4,DEMO 4<ETX>
<STX>L1;o11,447;f0;l1207;w4<ETX>
<STX>L2;o11,285;f0;l1207;w4<ETX>
<STX>W3;o11,0;f0;l1207;h802;w4<ETX>
<STX>B4;o658,650;f0;h102;w2;c0,0;i1;r1;d0,11<ETX>
<STX>I4;o658,752;f0;h1;w1;c20;r0;b0<ETX>
<STX>B5;o87,650;f0;h102;w2;c0,0;i1;r1;d0,11<ETX>
<STX>I5;o87,752;f0;h1;w1;c20;r0;b0<ETX>
```

```
<STX>H6;o34,183;f0;h1;w1;c22;r0;b0;d0,17<ETX>
<STX>H7;o35,143;f0;h1;w1;c21;r0;b0;d3,BASIS WT. 39-4838<ETX>
<STX>H8;o389,305;f0;h1;w1;c21;r0;b0;d3,ROLLS<ETX>
<STX>H9;o40,305;f0;h1;w1;c21;r0;b0;d3,ROLL WIDTH<ETX>
<STX>L10;o11,609;f0;l1207;w4<ETX>
<STX>H12;o1022,508;f0;h1;w1;c22;r0;b0;d0,7<ETX>
<STX>H13;o1022,467;f0;h1;w1;c21;r0;b0;d3,WEIGHT<ETX>
<STX>H14;o539,508;f0;h1;w1;c22;r0;b0;d0,11<ETX>
<STX>H15;o539,467;f0;h1;w1;c21;r0;b0;d3,LOCATION<ETX>
<STX>H16;o42,508;f0;h1;w1;c22;r0;b0;d0,15<ETX>
<STX>H17;o43,467;f0;h1;w1;c21;r0;b0;d3,CUSTOMER ORDER
NUMBER<ETX>
<STX>H18;o840,346;f0;h1;w1;c22;r0;b0;d0,13<ETX>
<STX>H19;o840,305;f0;h1;w1;c21;r0;b0;d3,ORDER ITEM NUMBER<ETX>
<STX>H20;o389,346;f0;h1;w1;c22;r0;b0;d0,7<ETX>
<STX>H21;o34,346;f0;h1;w1;c22;r0;b0;d0,11<ETX>
<STX>H22;o747,183;f0;h1;w1;c22;r0;b0;d0,15<ETX>
<STX>H23;o743,143;f0;h1;w1;c21;r0;b0;d3,GRADE DESCRIPTION<ETX>
<STX>H24;o325,0;f0;h51;w34;c25;r0;b3;d3,SHIPPING LABEL<ETX>
<STX>R<ETX>
<STX><ESC>E4<CAN><ETX>
<STX><ESC>F4<LF>INTERMEC<ETX>
<STX><ESC>F5<LF>372181192<ETX>
<STX><ESC>F6<LF>38448379237<ETX>
<STX><ESC>F12<LF>230<ETX>
<STX><ESC>F14<LF>3839494<ETX>
<STX><ESC>F16<LF>372181192<ETX>
<STX><ESC>F18<LF>234-LOFT<ETX>
<STX><ESC>F20<LF>12<ETX>
<STX><ESC>F21<LF>338438<ETX>
<STX><ESC>F22<LF>A-PLUS QTY<ETX>
<STX><ETB><FF><ETX>
```

These command strings print this label:



SHIPPING LABEL

BASIS WT. 39-4838 GRADE DESCRIPTION

38448379237 A - PLUS QTY

ROLL WIDTH ROLLS ORDER ITEM NUMBER

338438 12 234 - LOFT

CUSTOMER ORDER NUMBER LOCATION WEIGHT

372181192 3839494 230

INTERMEC 372181192

Example Format with Lines and Boxes

Example with Graphics and Rotated Fields

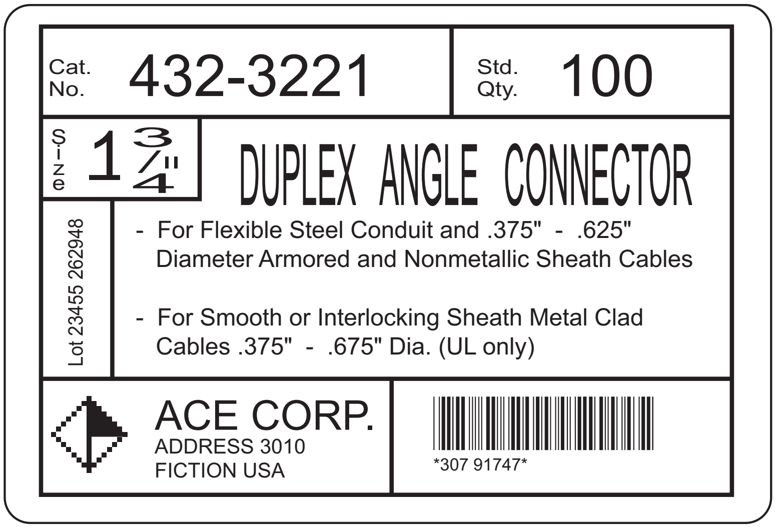
This example is a complex label that includes human-readable fields, a bar code field, line fields, a box field, and a graphic field.

Command Strings

```
<STX><ESC>C<ETX>
<STX><ESC>P<ETX>
<STX>E5;F5<ETX>
<STX>H0;o35,40;c25;d3,Cat.;k12<ETX>
<STX>H1;o35,70;c25;d3,No.;k12<ETX>
<STX>H2;o165,0;c25;d3,432-3221;k36<ETX>
<STX>H3;o785,40;c25;d3,Std.;k12<ETX>
<STX>H4;o785,70;c25;d3,Qty.;k12<ETX>
<STX>H5;o915,0;c25;d3,100;k36<ETX>
<STX>L6;o740,10;f3;l130;w8<ETX>
<STX>L7;o25,140;l1130;w8<ETX>
<STX>H8;o30,165;c25;f3;r1;d3,Size;k12<ETX>
<STX>H9;o80,170;c25;d3,1 ”;k30<ETX>
<STX>H10;o150,165;f3;r1;c25;d3,3/4;h3;w7<ETX>
<STX>L11;o300,140;f3;l130;w8<ETX>
<STX>L12;o25,270;l275;w8<ETX>
<STX>H13;o60,560;f1;c25;d0,20;h3;w2<ETX>
<STX>L14;o140,270;f3;l310;w8<ETX>
<STX>H15;o360,120;c22;d3,DUPLEX ANGLE CONNECTOR;h3;w1<ETX>
<STX>H16;o170,320;c25;d3,- For Flexible Steel Conduit and .375”
- .625”;k12<ETX>
<STX>H17;o212,375;c25;d3,Diameter Armored and Nonmetallic Sheath
Cables;k12<ETX>
<STX>H18;o170,450;c25;d3,- For Smooth or Interlocking Sheath
Metal Clad;k12<ETX>
```

```
<STX>H19;o212,505;c25;d3,Cables .375” - .675” Dia. (UL
only);k12<ETX>
<STX>L20;o25,580;l1130;w8<ETX>
<STX>U21;o40,610;c2;h9;w9<ETX>
<STX>H22;o210,600;c25;d3,ACE CORP.;k24<ETX>
<STX>H23;o210,670;c25;d3,ADDRESS 3010;k12<ETX>
<STX>H24;o210,710;c25;d3,FICTION USA;k12<ETX>
<STX>B25;o685,615;c0,0;d0,20;i1;h100;p@<ETX>
<STX>I25;h2;w2<ETX>
<STX>L26;o590,580;f3;l185;w8<ETX>
<STX>W27;o015,000;w10;l1150;h775<ETX>
<STX>R<ETX>
<STX><ESC>E5<CAN><ETX>
<STX>Lot 23455 262948<CR><FS>307 91747<FS><ESC>I2<ETX>
<STX><RS>5<ETB><ETX>
```

These command strings print this label:



Cat. No. 432-3221

Std. Qty. 100

Size 1 3/4"

DUPLEX ANGLE CONNECTOR

Lot 23455 262948

- For Flexible Steel Conduit and .375" - .625" Diameter Armored and Nonmetallic Sheath Cables

- For Smooth or Interlocking Sheath Metal Clad Cables .375" - .675" Dia. (UL only)

ACE CORP.
ADDRESS 3010
FICTION USA

307 91747

Example Format with Rotated Fields and Graphic

Note: This example uses the “diamond” graphic from the procedure in **“Creating User-Defined Characters for Advanced Mode” on page 48**. To print this label, you need to follow that procedure to create and download the graphic to your printer beforehand.

The next table explains the command strings for this example. For more information, see the ***IPL Command Reference Manual***.

Command String Descriptions for Example

Command	Description
`<ESC>C`	Places the printer in Advanced mode.
`<ESC>P`	Places the printer in Program mode.

Command String Descriptions for Example (continued)

Command	Description
`E5;F5`	Clears current format 5 and creates a new format 5.
`H0;`	Defines field 0 as a human-readable field.
`o35,40;`	Sets the origin of field 0 at 35,40.
`c25;`	Sets the font for field 0 to font 25.z
`d3,Cat.;`	Sets the data for field 0 to "Cat.".
`k12;`	Sets the font size to 12.
`L6;`	Defines field 6 as a line field.
`o740,10;`	Sets the origin for field 6 at 740,10.
`f3;`	Rotates field 6 by 270 degrees counterclockwise around the field origin.
`l130;`	Sets the length of the line in field 6 to 130 dots.
`w8;`	Sets the width of field 6 to 8 dots.
`H8;`	Defines field 8 as a human-readable field.
`o30,165;`	Sets the origin of field 8 at 30,165.
`c25;`	Defines the font for field 8 as font 25.
`f3;`	Rotates field 8 by 270 degrees counterclockwise around the origin.
`r1;`	Rotates the characters in field 8 by 90 degrees counterclockwise.
`d3,Size;`	Sets the data for field 8 to "Size".
`k12;`	Sets the Font size to 12.
`H10;`	Defines field 10 as a human-readable field.
`o150,165;`	Sets the origin for field 10 at 150,165.
`f3;`	Rotates field 10 by 270 degrees counterclockwise around the origin.
`r1;`	Rotates the characters in field 10 by 90 degrees counterclockwise.
`c25;`	Sets the font for field 10 to font 25.
`d3,3/4;`	Sets the data for field 10 to "3/4".
`k12;`	Sets the font size to 12.
`H13;`	Defines field 13 as a human-readable field.
`o60,560;`	Sets the origin of field 13 at 60,560.
`f1;`	Rotates field 13 by 90 degrees around the origin.
`c25;`	Sets the font for field 13 to font 25.
`d0,20;`	Specifies that data for field 13 is entered during Print mode with a maximum of 20 characters.
`k10;`	Sets the font size to 10.
`U21;`	Defines field 21 as a graphic field.
`o40,610;`	Sets the origin of field 21 at 40,610.
`c2;`	Specifies that field 21 prints in font 2. In this case, font 2 actually refers to the user-defined graphic ("diamond") previously downloaded to the printer.
`h9;`	Sets the height to nine times the original size.
`w9;`	Sets the width to 9 dots.
`B25;`	Defines field 25 as a bar code field.
`o685,615;`	Sets the origin of field 25 at 685,615.
`c0,0;`	Sets the bar code font to Code 39 with no check digit.

Command String Descriptions for Example (continued)

Command	Description
`d0,20;`	Determines that the data for field 25 is entered during Print mode with a maximum of 20 characters.
`i1;`	Enables the interpretive field for the bar code, to print with start and stop characters included.
`h100;`	Sets the bar code height to 100 dots.
`p@;`	Clears all prefixes from the bar code field.
`I25`	Edits the interpretive field for bar code field 25.
`h2`	Sets the height of the interpretive field to twice its original height.
`w2`	Sets the width of the interpretive field to 2 dots.
`W27;`	Defines field 27 as a box field.
`o015,000;`	Sets the origin of field 27 at 15,0.
`w10;`	Defines the line width of the box as 10 dots.
`l1150;`	Defines the length of the box as 1150.
`h775;`	Sets the height of the box to 775.
`R`	Places the printer in Print mode.
`<ESC>E5`	Accesses format 5 from the printer memory.
`<CAN>`	Clears all data for the current format and sets the field pointer to the lowest numbered data-entry field.
`Lot 23455 262948<CR>`	This is the data intended for the first data-entry field. <CR> instructs the printer to go to the next data-entry field.
`<FS>`	Specifies that the following data is to be incremented.
`307 91747`	This is the data intended for the next data-entry field (specified as data to be incremented).
`<FS>`	Specifies that the preceding data is to be incremented.
`<ESC>I2`	The data surrounded by <FS> commands is to be incremented by a value of 2 after each label is printed.
`<RS>5`	Sets the number of labels to print when the print command is executed.
`<ETB>`	Prints the label.

RFID Tag Example

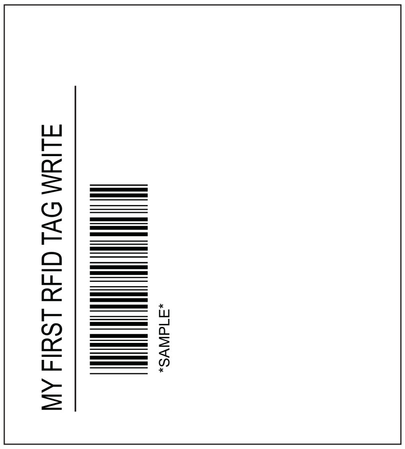
This example is a bar code label that contains an RFID tag.

Command Strings

```
<STX><ESC>C<ETX>
<STX><ESC>P<ETX>
<STX>E4;F4;<ETX>
<STX>H0;o102,51;f0;c25;h20;w20;d0,30;<ETX>
<STX>L1;o102,102;f0;l575;w5;<ETX>
<STX>B2;o203,153;c0,0;h100;w2;i1;d0,10;<ETX>
<STX>I2;h1;w1;c20;<ETX>
<STX>Q3;a2,2,0,23;d3,MY FIRST RFID TAG WRITE;<ETX>
<STX>R;<ETX>
<STX><ESC>E4<ETX>
```

```
<STX><CAN><ETX>
<STX>MY FIRST RFID TAG WRITE<CR><ETX>
<STX>SAMPLE<ETX>
<STX><ESC>J2,2,0,23<ETX>
<STX><ETB><ETX>
```

These commands print this label:



RFID Tag Example

The next table explains the command strings for the RFID tag example. For more information, see the ***IPL Command Reference Manual***.

Label Format – RFID Example Command Descriptions

Command	Description
`<ESC>C`	Places the printer in Advanced mode.
`<ESC>P`	Places the printer in Program mode.
`E4;F4`	Clears current format 4 and creates a new format 4.
`H0`	Defines field 0 as a human-readable field.
`o102,51`	Sets the origin of field 0 at coordinates 102,51.
`f0`	Rotates field 0 by 0 degrees.
`c25`	Sets the font to font 25.
`h20`	Sets the height to 20 times original size.
`w20`	Sets the width to 20 times original size.
`d0,30`	Specifies that data for this field is transmitted in Print mode with a maximum of 30 characters.
`L1`	Defines field 1 as a line field.
`o102,102`	Sets the origin of field 1 at coordinates 102,102.
`f0`	Rotates field 1 by 90 degrees counterclockwise around the origin.
`l575`	Defines the length of the line as 575.
`w5`	Sets the width of the line to 5 dots.
`B2`	Defines field 2 as a bar code field.
`o203,153`	Sets the origin of field 2 at coordinates 203,153.
`c0,0`	Sets the bar code font to Code 39 with no check digit.

Label Format – RFID Example Command Descriptions (continued)

Command	Description
`h100`	Sets the height to 100 dots.
`w2`	Sets the width to 2 dots.
`i1`	Enables the interpretive field, to print with start and stop characters included.
`d0,10`	Specifies that data for this field is transmitted in Print mode with a maximum of 10 characters.
`I2`	Edits the interpretive field for bar code field 2.
`h1`	Sets the height to original size.
`w1`	Sets the width to original size.
`c20`	Sets the font to font `20`.
`Q3`	Defines field 3 as an RFID field.
`a2,2,0,23`	Sets the format to ASCII, writes to the Data segment, starts on byte 0 and writes up to 23 bytes.
`d3,MY FIRST RFID TAG WRITE`	Defines constant data for field 3 to be "MY FIRST RFID TAG WRITE."
`R`	Places the printer in Print mode.
`<ESC>E4`	Accesses format 4 from the printer memory.
`<CAN>`	Clears all data for format 4 and sets the field pointer to the lowest numbered data-entry field.
`MY FIRST RFID TAG WRITE<CR>`	This is the data intended for the first data-entry field (human-readable field 0). <CR> instructs the printer to go to the next data-entry field.
`SAMPLE`	This is the data intended for the second data-entry field (bar code field 2).
`<ESC>J2,2,0,23`	Reads the data from the RFID tag. The data is in ASCII format on the Data segment, starting on byte 0 and having a length of 23 bytes.
`<ETB>`	Prints the label.

3

Working with Fonts and Graphics

This chapter explains fonts and how to choose them for your application, and describes how to create user-defined characters and fonts. Sections in this chapter include:

Learning About Fonts

In addition to the fonts permanently resident in your printer, you can download scalable outline fonts and user-defined bitmap fonts. This section explains the differences between bitmap and outline fonts and how to choose the best font for your application.

About Bitmap Fonts

Bitmap fonts can be used on any Intermec printer. These fonts typically print quickly, require less memory than outline fonts, and print at the highest quality in a specific size. Bitmap fonts are memory intensive when defined as large characters.

When you download bitmap characters to the printer, you must select the size of the characters. Make sure that you select the size you want to use in your formats. If you use IPL commands to magnify the character size in the printer, the edges of the characters will be jagged.

About Outline Fonts

Outline fonts can be scaled up to 10.16 cm (4 in) and will print in magnified sizes without jagged edges. These fonts also image more slowly than bitmap fonts.

Most Intermec printers include TrueDoc or Speedo outline fonts. You can also download TrueType outline fonts to some printers. The next table lists outline font compatibility by printer.

Outline Font Compatibility by Printer

Printer	TrueType	TrueDoc	Speedo
3240	·		·
3400A,B,C,D	·		·
3400e	·		·
3440	·		·
44X0	·		·
4X30			·
7421			·
F4	·	·	
PC41			·
PD series	·	·	
PF/PM/PX series	·	·	·

TrueType fonts are the most popular outline fonts, and several are available through Windows on your desktop PC. For more information on TrueType fonts, see **“Using TrueType Fonts” on page 41.**

Although some Intermec printers do not support TrueType outline fonts, you can use the PrintSet application to convert TrueType outline fonts to bitmaps for use on any Intermec printer. For more information, see **"Creating Bitmap Fonts From TrueType Fonts" on page 44.**

Choosing a Font

See the next table to decide which type of font best suits your needs.

Font Comparison Table

Font Type	Print Speed	Selection	Memory	Scalability
Bitmap	Fastest	Extensive	Varies with size of characters	Poor
Outline (TrueType)	Medium	Extensive in Windows	Typically 50-80KB; Asian fonts are much larger	Good
Outline (Speedo)	Fast	Limited	Typically 20-30KB	Good

Downloading Fonts to the Printer

You can download fonts to the printer and store them in the non-volatile memory. Although the printer reserves 16 font ID numbers (3 to 6, and 8 to 19) for downloaded fonts, the amount of available memory limits the number of fonts you can store.

There are two ways to download fonts to the printer:

- Use the PrintSet application. PrintSet is available on the Printer Companion CD that shipped with your printer. You can use PrintSet to select fonts and directly download them to your printer. You can also download the latest release of PrintSet from the Intermec web site at www.intermec.com.

  For more information on downloading fonts with PrintSet, see the next section.
- (Advanced users only) Use IPL commands. For help, see **"Using IPL Commands to Download Fonts" on page 40.**

Using PrintSet to Download Fonts

The easiest way to download fonts to the printer is with the PrintSet application (v2.0 or later).

Intermec printers require that bitmap fonts be either one bit per byte or six bits per byte. PrintSet automatically converts all bitmap fonts into the six bits per byte format. Because bitmap fonts can be memory intensive, you can use PrintSet to define and download only a subset of the font (for example, only the numbers), saving storage space on the printer.

PrintSet automatically converts scalable outline fonts into nybblized data that you can download to the printer. PrintSet can also convert fixed outline fonts into downloadable bitmap fonts.

For more information on using PrintSet to download fonts, see the PrintSet online documentation.

Using IPL Commands to Download Fonts

Although the easiest way to download fonts is with PrintSet, you can also use the IPL command set to create and download user-defined bitmap and outline fonts to your Intermec printer.

Downloading Bitmap Fonts

To learn more about bitmap font formats, see **“Creating User-Defined Font Characters for Advanced Mode” on page 53.**

Downloading Outline Fonts

If you need to download an outline font but you cannot use PrintSet, you can:

- create a program to convert the outline font to a format you can download to the printer.
- manually convert the outline font to a format you can download to the printer.

When you convert an outline font to a format you can download to the printer, you must change the font character data into nybblized data that the printer can interpret. When you nybblize data, you divide each byte of data into two bytes.

For example:

byte 0xAB becomes two bytes: 0x41, 0x42 or text string “AB”

To manually download an outline font to your printer

1. Nybblize the font data.

   Divide the nybblized data into separate lines preceded by the j command (for more information, see the **Outline Font, Download** command in the ***IPL Command Reference Manual***). Remember to wrap the entire line in <STX> and ;<ETX>. You need short lines for limited message length protocols and to make modifying easier by using an editor or word processing program.
2. Include IPL commands to instruct the printer what to do with the font data. For help, see the following example.
3. Send the commands to the printer using one of the methods described in Chapter 1.

This is an example of a nybblized outline font file that includes IPL commands to send it directly to a printer.

Nybblized Outline Font File Example

Command	Definition
`<STX>R<ESC>C<ESC>P<ETX>`	Enter Program mode.
`<STX>J03,Times,1;<ETX>`	Create TrueType font 3 and give it the name Times.
`<STX>j0001000000110100000400604c545348efe24cd00000;<ETX>`	TrueType data string.

Nybblized Outline Font File Example (continued)

Command	Definition
`<STX>j00ebf468646d78d956f5ab0001135800001508686561;<ETX>`	TrueType data string.
`<STX>j0d2000012f04000006a3706f7374d43c8176000135a8;<ETX>`	TrueType data string.
: : :	Several TrueType data strings not shown due to space constraints.
`<STX>j00b3008200b0008725ba0000;<ETX>`	Last TrueType data string.
`<STX>R<ETX>`	Exit Program mode.

Using TrueType Fonts

This section includes information to keep in mind if you are using TrueType fonts with your Intermec printer.

Since TrueType fonts may be very complex, they tend to image slower than bitmap fonts. You cannot subset a TrueType font. You must download the entire font to the printer.

Note: When using TrueType fonts, your printer must be configured to operate in 8 bit mode. Intermec also recommends the following printer configuration: highest supported baud rate, 8 data bits, no parity, 1 stop bit, XON/XOFF flow control.

Your TrueType fonts must be compatible with Microsoft Windows. To generate bitmap fonts from TrueType fonts for a particular language, you must run PrintSet under that language version of Microsoft Windows. You can run PrintSet Version 2.0 and later under Windows 95/98/NT 4.0/2000/XP.

Selecting the Printer Language

When you install a TrueType font, you must select the correct printer language. To choose the correct language, use the IPL command “Printer Language, Select.” If the language setting you have chosen does not work, use the default language setting.

TrueType Fonts and Memory Usage

To install a TrueType font in your printer, the printer must have enough memory available to contain the entire font file. The size of the TrueType font file, in bytes, is the minimum amount of memory you must have available in your printer. Bitmap fonts generated from TrueType files will vary in size, depending on the point size and number of characters you create. The larger the point size you choose, the larger the resulting file size.

Some TrueType fonts require large amounts of dynamic RAM to operate. If you receive an error code 37 when printing a label using TrueType fonts, you may need to purchase expanded dynamic RAM.

Installing Asian Fonts

For the PD/PF/PM/PX series printers, you can purchase a font card that includes scalable Chinese, Japanese, or Korean fonts. When you install the font card in your printer, the fonts can be used immediately.

Note: The EasyCoder F4 does not support bitmap or TrueType Chinese, Japanese, or Korean fonts.

For the 3240, 3400e, 3440, 4420, and 4440 printers, you can use PrintSet 2.1 or later to download Chinese, Japanese, or Korean scalable fonts to the printer. You may need to install more flash memory on the printer to use these fonts. The amount of memory you need depends on the size and number of fonts you want to install. For more information on downloading scalable fonts to your printer, see the PrintSet documentation.

When you download one of these scalable fonts, you must install the necessary code page tables. For help, see the next section.

Installing Code Page Tables for Asian Scalable Fonts

If you install a Japanese, Chinese, or Korean scalable font on a 3240, 3400e, 3440, 4420, or 4440 printer, you must also install the code page tables. You can download code pages from Intermec's web site at www.intermec.com. Follow the next procedure to download and install code pages on your printer.

Note: Before you begin, you need a serial connection between your desktop PC and your Intermec printer. For help, see your printer user's manual.

To download and install code pages

1. From the main Intermec web page, choose **Support** > **Downloads**. The Downloads page appears.
2. Choose your printer model from the drop-down list. A list of downloadable software appears.
3. Scroll down the list to Printer Fonts and click the code page you want to download. Follow the prompts to download the code page file to your PC.
4. Browse to the code page file and unzip it. After you unzip the file, the code page file is one of the following:

   SJIS.PCF - Japanese (Shift JIS)
   BIG5.PCF - Traditional Chinese (Big 5)
   GB.PCF - Simplified Chinese (GB 2312)
   KSC.PCF - Korean (KSC5601)

5 Open a HyperTerminal session to the printer.

Note: Configure the HyperTerminal session for these parameters:

- Baud rate: 19200
- Parity: None
- Data bits: 8
- Stop bits: 1
- Flow control: XON/XOFF

6 From the **Transfer** menu, select **Send File**. The Send File dialog box appears.

7 In the **Name** field, locate your file, and then click **Send**. The code page is downloaded to the printer.

Installing Asian Bitmap Fonts

Instead of using Asian scalable fonts, you can use PrintSet to convert the scalable fonts to bitmap fonts and download them to the printer. Asian bitmap fonts are supported by the 3240, 3400d, 3400e, 3440, 3600, 4420, and 4440 printers, generally require less memory, and do not require code page tables like the equivalent scalable fonts.

The next table lists the approximate amount of memory required to store Asian bitmap fonts. The memory required depends on the printhead installed in your printer.

Approximate Sizes – Asian Bitmap Fonts

TrueType Font	Point Size	Approximate Size (400 dpi printhead)	Approximate Size (200 dpi printhead)
Korean KSC-5601	12 pt	1.6MB	0.5MB
	16 pt	2.7MB	0.8MB
	24 pt	5.8MB	1.5MB
Traditional Chinese Big 5	12 pt	3.1MB	0.9MB
	16 pt	5.25MB	1.6MB
Simplified Chinese GB	12 pt	3.5MB	1MB
	16 pt	6.2MB	1.8MB
Japanese Shift-JIS	12 pt	3.9MB	1.1MB
	16 pt	6.75MB	2MB

You may need to install more flash memory on the printer to use these fonts. The amount of memory you need depends on the size and number of fonts you want to install. For more information on downloading bitmap fonts to your printer, see the PrintSet documentation.

Licensing Your Fonts

You are responsible for licensing the fonts that you purchase and install in your printers. Contact your font vendor for licensing information.

These companies provide TrueType fonts compatible with Intermec printers:

Dynalab Inc.
2055 Gateway Place
Suite 400
San Jose, CA 95110
Tel: 408-490-4224
Fax: 408-490-2233
www.dynalab.com

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399
Tel: 425-882-8080
www.microsoft.com

Note: Dynalab provides Chinese, Japanese, and Korean fonts. You can also purchase fonts from their offices in Taiwan and Hong Kong. See the Dynalab web page for the address of these offices.

Creating Bitmap Fonts From TrueType Fonts

To create bitmap fonts from TrueType fonts, you must run PrintSet under the correct language version of Windows. For example, to create a Traditional Chinese font, you must run PrintSet under Traditional Chinese Microsoft Windows. In addition, you must install the font using Microsoft Windows in order for PrintSet to correctly convert it to a bitmap font.

Note: You can create Traditional or Simplified Chinese bitmap fonts up to a size of 16 points at 400 dpi or 32 points at 200 dpi. The EasyCoder F4 printers do not support bitmap or TrueType Chinese, Japanese, or Korean fonts.

For more information, see the PrintSet online documentation.

Creating User-Defined Characters and Fonts

You can create user-defined characters and user-defined fonts for use in bar code labels.

A user-defined character (UDC) is a custom graphic you can use in graphic fields. For more information, see **"User-Defined Character or Graphic Fields" on page 12.**

A user-defined font (UDF) is a custom font you design yourself. UDFs can be used in human-readable fields or in the interpretive field that accompanies a bar code field. For more information, see **"Human-Readable Fields" on page 11** or **"Bar Code Fields" on page 11.**

UDCs and UDFs are created in a six bits per byte format for printers in Advanced mode, or in a one bit per byte format for printers in Emulation mode.

You use a variety of IPL commands to create and define user-defined characters and fonts. After you write the command strings that create UDCs and UDFs, you download the command strings to the printer and save them to the printer memory.

Creating User-Defined Characters for Emulation Mode

If your Intermec printer is running in Emulation mode (dot size of 10 or 15 mil), UDCs must be downloaded to the printer in one bit per byte format. A one bit per byte bitmap image is an arrangement of ones and zeros that looks similar to the following example.

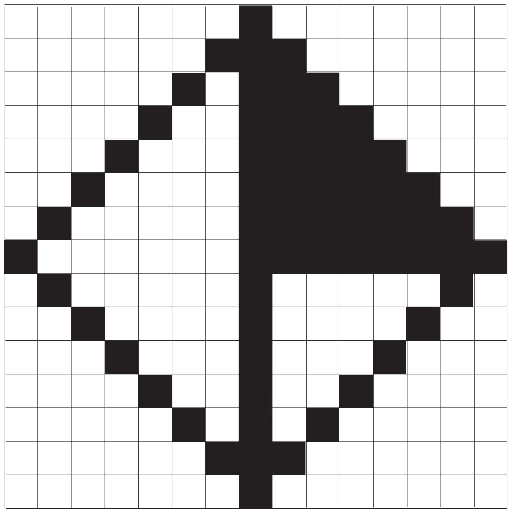
Example of One Bit Per Byte Bitmap Image

Bitmap Pattern	Row
000000010000000	Row 0
000000111000000	Row 1
000001011100000	Row 2
000010011110000	Row 3
000100011111000	Row 4
001000011111100	Row 5
010000011111110	Row 6
100000011111111	Row 7
010000010000010	Row 8
001000010000100	Row 9
000100010001000	Row 10
000010010010000	Row 11
000001010100000	Row 12
000000111000000	Row 13
000000010000000	Row 14

If you look closely at the previous table, you can see that it is the outline of a diamond with a line down the middle and the upper right corner blacked in.

To create a user-defined character for Emulation mode

1 Draw your design on a piece of graph paper. This procedure uses the next illustration as an example:



2 Convert each of the squares to either a one or a zero, where the blank boxes are zeros and the filled-in boxes are ones. Type these into a text file column by column, as shown here.

```
000000010000000
000000111000000
000001011100000
000010011110000
000100011111000
001000011111100
010000011111110
100000011111111
010000010000010
001000010000100
000100010001000
000010010010000
000001010100000
000000111000000
000000010000000
```

3 Read the pattern of ones and zeros down each column starting at the top left corner. The first column on the left becomes the data for the u0 command line:

```
u0,000000010000000
```

The second column becomes the data for the u1 command line:

```
u1,000000111000000
```

4 In a new text file, type the command line for all 15 columns, prefacing each line with u*n*:

```
u0,000000010000000
u1,000000111000000
u2,000001010100000
u3,000010010010000
u4,000100010001000
u5,001000010000100
u6,010000010000010
u7,111111110000001
u8,011111110000010
u9,001111110000100
u10,000111110001000
u11,000011110010000
u12,000001110100000
u13,000000111000000
u14,000000010000000
```

5 Before the u0 command line, create IPL command strings that place the printer in Emulation mode, in Program mode, and define the bitmap as a UDC with field number 3:

```
<STX><ESC>c<ETX>
<STX>P<ETX>
<STX>G3;x15;y15<ETX>
```

6 Add the ASCII start of text and end of text (<STX> and <ETX>) characters to the u*n* command lines.

7 Add a command string to place the printer in Print mode:

```
<STX>R<ETX>
```

8 Send the file to the printer. The UDC is created and assigned "font 3." When you specify data for a graphic field in a format, assign `c3` to the field to print this graphic in that field.

Example of UDC for Emulation Mode

Command Line	Description
<STX><ESC>c<ETX>	Places the printer in Emulation mode
<STX><ESC>P<ETX>	Places the printer in Program mode
<STX>G3;x15;y15;<ETX>	Creates UDC and assigns it “font 3.”
<STX>u0,000000010000000;<ETX>	Defines column 0
<STX>u1,000000101000000;<ETX>	Defines column 1
<STX>u2,000001000100000;<ETX>	Defines column 2
<STX>u3,000010000010000;<ETX>	Defines column 3
<STX>u4,000100000001000;<ETX>	Defines column 4
<STX>u5,001000000000100;<ETX>	Defines column 5
<STX>u6,010000000000010;<ETX>	Defines column 6
<STX>u7,111111111111111;<ETX>	Defines column 7
<STX>u8,011111110000010;<ETX>	Defines column 8
<STX>u9,001111110000100;<ETX>	Defines column 9
<STX>u10,000111110001000;<ETX>	Defines column 10
<STX>u11,000011110010000;<ETX>	Defines column 11
<STX>u12,000001110100000;<ETX>	Defines column 12
<STX>u13,000000111000000;<ETX>	Defines column 13
<STX>u14,000000010000000;<ETX>	Defines column 14
<STX>R;<ETX>	Saves the graphic and places the printer in Print mode

Creating User-Defined Fonts for Emulation Mode

Creating user-defined font characters for Emulation mode is almost identical to the procedure for creating user-defined characters. You need to define the format for each ASCII character you want to use. For help, see **“Creating User-Defined Characters for Emulation Mode” on page 45.**

Additionally, when you write the command strings that define the ASCII characters for the font, you need to include syntax for these commands:

- “Bitmap User-Defined Font, Clear or Define”: This command creates a new font and font number. Syntax is T*n*, where *n* is the font number.
- “Bitmap Cell Width for Graphic or UDF, Define”: This command defines the width for all cells that contain a character from this font. Syntax is x*n*, where *n* is the cell width in dots.
- “Bitmap Cell Height for Graphic or UDF, Define”: This command defines the height for all cells that contain a character from this font. Syntax is y*n*, where *n* is the cell height in dots.
- “User-Defined Font Character, Create”: This command creates the new character. Syntax is t*n*, where *n* is the decimal equivalent of the corresponding ASCII character. For help, see the **“Full ASCII Table” on page 76.** You need to add this syntax for each new character in your user-defined font.

"Intercharacter Space for UDF, Define": This command defines the amount of space added to the default intercharacter gap length for a bitmap font. Syntax is z*n*, where *n* is the number of dots to add to the default spacing of 2.

Example of User-Defined Font, One Bit Per Byte

Commands	Definition
`<STX><ESC>c<ETX>`	Selects 86XX Emulation mode
`<STX><ESC>P<ETX>`	Enters Program mode
`<STX>T11,FONT11<ETX>`	Creates bitmap font 11
`<STX>x10;y14;<ETX>`	Defines cell width and height
`<STX>t36;Z12;<ETX>`	Creates character 36 ($), character width is 12
`<STX>u0,00111110001100;<ETX>`	Defines column 0
`<STX>u1,01111111001110;<ETX>`	Defines column 1
`<STX>u2,01100011000110;<ETX>`	Defines column 2
`<STX>u3,01100011000110;<ETX>`	Defines column 3
`<STX>u4,11111111111111;<ETX>`	Defines column 4
`<STX>u5,11111111111111;<ETX>`	Defines column 5
`<STX>u6,01100011000110;<ETX>`	Defines column 6
`<STX>u7,01100011000110;<ETX>`	Defines column 7
`<STX>u8,01110011111110;<ETX>`	Defines column 8
`<STX>u9,00110001111100;<ETX>`	Defines column 9
`<STX>t105;Z4;<ETX>`	Creates character 105 (i), character width is 4
`<STX>u4,00110011111111;<ETX>`	Defines column 4
`<STX>u5,00110011111111;<ETX>`	Defines column 5
`<STX>R;<ETX>`	Saves and exits to Print mode

Creating User-Defined Characters for Advanced Mode

If your Intermec printer is running in Advanced mode (dot size of 2.5 or 5 mil), UDCs must be downloaded to the printer in the six bits per byte format.

The six bits per byte format is more compact than one bit per byte. When you use six bits per byte, you can download large graphics more quickly. The difference between 2.5 mil and 5.0 mil modes is the size of the image when it prints.

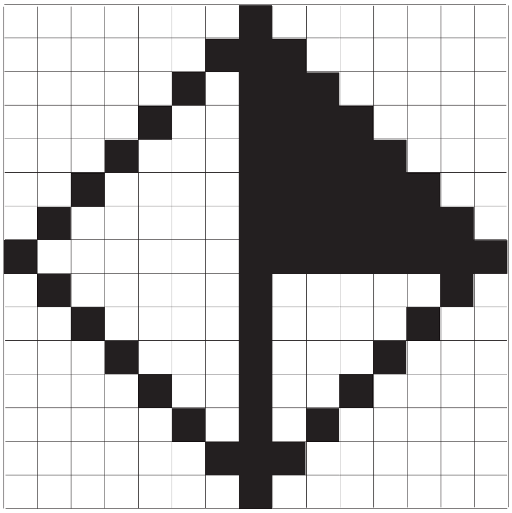
The arrangement of the bits is very important in this format. Eight bits (0 through 7) compose every byte, but the printer only uses bits 0 through 5 to map the image.

You must always set bit 6 (the seventh bit) to 1 so you can download data to the printer. 7-bit hosts reserve bit 7 (the eighth bit) for parity and compatibility, so the range of characters for any given UDC is 40 hex (@) to 127 hex (■ delete character).

The next procedure describes how to create a user-defined character in the six bits per byte format for use in Advanced mode.

To create a user-defined character for Advanced mode

1 Draw the graphic on graph paper. This procedure uses this graphic as an example:



2 Convert each of the squares to either a one or a zero, where the blank boxes are zeros and the filled-in boxes are ones. Type these into a text file column by column, as shown here:

```
000000010000000
000000111000000
000001011100000
000010011110000
000100011111000
001000011111100
010000011111110
100000011111111
010000010000010
001000010000100
000100010001000
000010010010000
000001010100000
000000111000000
000000010000000
```

3 Starting from the top row, divide each vertical column into groups of six digits. If the bottom group has less than six digits, add zeros to this group until it also has six.

```
000000010000000
000000111000000
000001011100000
000010011110000
000100011111000
001000011111100
- - - - - - - - - - -
010000011111110
100000011111111
010000010000010
001000010000100
000100010001000
000010010010000
- - - - - - - - - - -
000001010100000
000000111000000
000000010000000
000000000000000 ┐
000000000000000 │ Zeros added
000000000000000 ┘
```

The six digits in each group are the six bits that you download in a byte of data. The top digit of each group is bit 0, the bottom digit is bit 5.

4 To complete the bit, add ones in the bit 6 position, and then add zeros in the bit 7 position so that each group now has eight digits.

```
         000000010000000  Bit 0
         000000111000000  1
         000001011100000  2
Group 1  000010011110000  3
         000100011111000  4
         001000011111100  5
         111111111111111  6
         000000000000000  7
         - - - - - - - - - - - -
         010000011111110  Bit 0
         100000011111111  1
         010000010000010  2
Group 2  001000010000100  3
         000100010001000  4
         000010010010000  5
         111111111111111  6
         000000000000000  7
         - - - - - - - - - - - -
         000001010100000  Bit 0
         000000111000000  1
         000000010000000  2
Group 3  000000000000000  3
         000000000000000  4
         000000000000000  5
         111111111111111  6
         000000000000000  7
```

5 Starting with the first group of 8 bits in the first column, reverse the order of each group so that bit 0 is now last and bit 7 is first. One way to see this quickly is to imagine rotating each group 90° clockwise, as shown in the next illustration:

```
Group 1           Group 2           Group 3
01000000          01000010          01000000
01000000          01000101          01000000
01100000          01001000          01000000
01010000          01010000          01000000
01001000          01100000          01000000
01000100          01000000          01000001
01000010          01000000          01000010
01111111          01111111          01000111
01111110          01000011          01000010
01111100          01000011          01000001
01111000          01100011          01000000
01110000          01010011          01000000
01100000          01001011          01000000
01000000          01000111          01000000
01000000          01000010          01000000
|       |         |       |         |       |
Bit 7   Bit 0     Bit 7   Bit 0     Bit 7   Bit 0
```

Reading from left to right, each line is now made up of three eight-digit groups, and each group represents an ASCII character in binary form.

6 Translate each line into a string of three ASCII characters. For a list of ASCII characters and binary equivalents, see **"Full ASCII Table" on page 76.**

For example, reading from left to right, the first line is:

```
01000000       0100001001000000
```

The equivalent ASCII characters are @B@. The entire format should look like this:

```
@B@
@E@
`H@
PP@
H`@
D@A
B@B
<DEL><DEL>G
~CB
|CA
xc@
pS@
`K@
@G@
@B@
```

7 In a new text file, write command strings to place the printer in Advanced mode and then in Program mode:

```
<STX><ESC>C<ETX>
<STX><ESC>P<ETX>
```

8 Write a command string to create a user-defined character, and define its size as 15 x 15 dots:

```
<STX>G1;x15;y15<ETX>
```

9 Add command strings for the graphic format you created in step 6. Place the ASCII start of text character at the beginning of each line, followed by the

“Graphic or UDC, Define” command syntax (u). Number each line from 0 to 14, and then add the ASCII end of text character to the end of the line:

```
<STX>u0,@B@<ETX>
<STX>u1,@E@<ETX>
<STX>u2,`H@<ETX>
<STX>u3,PP@<ETX>
<STX>u4,H`@<ETX>
<STX>u5,D@A<ETX>
<STX>u6,B@B<ETX>
<STX>u7,<DEL><DEL>G<ETX>
<STX>u8,~CB<ETX>
<STX>u9,|CA<ETX>
<STX>u10,xc@<ETX>
<STX>u11,pS@<ETX>
<STX>u12,`K@<ETX>
<STX>u13,@G@<ETX>
<STX>u14,@B@<ETX>
```

10 Write a command string to place the printer in Print mode:

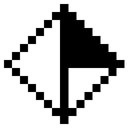
```
<STX>R<ETX>
```

11 Save the text file and send it to the printer. The graphic can now be used in a label format.

12 To print the graphic, send this set of command strings:

```
<STX><ESC>C<ETX>
<STX><ESC>P<ETX>
<STX>E4;F4<ETX>
<STX>U1;o100,100;c2;w20;h20;<ETX>
<STX>R;<ETX>
<STX><ESC>E4<ETX>
<STX><ETB><ETX>
```

This graphic prints:



The next table lists the complete command string set for the UDC example.

Command Strings for User-Defined Character Example

Command	Definition
`<STX><ESC>C<ETX>`	Places the printer in Advanced mode
`<STX><ESC>P<ETX>`	Places the printer in Program mode
`<STX>G1;x15;y15<ETX>`	Creates UDC 1, 15 rows by 15 columns
`<STX>u0,@B@<ETX>`	Defines column 0
`<STX>u1,@E@<ETX>`	Defines column 1
`` <STX>u2,`H@<ETX> ``	Defines column 2
`<STX>u3,PP@<ETX>`	Defines column 3
`` <STX>u4,H`@<ETX> ``	Defines column 4
`<STX>u5,D@A<ETX>`	Defines column 5
`<STX>u6,B@B<ETX>`	Defines column 6

Command Strings for User-Defined Character Example (continued)

Command	Definition
`<STX>u7,<DEL><DEL>G<ETX>`	Defines column 7
`<STX>u8,~CB<ETX>`	Defines column 8
`<STX>u9,|CA<ETX>`	Defines column 9
`<STX>u10,xc@<ETX>`	Defines column 10
`<STX>u11,pS@<ETX>`	Defines column 11
``<STX>u12,`K@<ETX>``	Defines column 12
`<STX>u13,@G@<ETX>`	Defines column 13
`<STX>u14,@B@<ETX>`	Defines column 14
`<STX>R<ETX>`	Saves and exits to Print mode
`<STX><ESC>C<ETX>`	Places the printer in Advanced mode
`<STX><ESC>P<ETX>`	Places the printer in Program mode
`<STX>E4;F4<ETX>`	Erases current format 4 and creates new format 4
`<STX>U1;o100,100;c2;w20;h20<ETX>`	Creates graphic field 1, origin of 100,100, uses graphic 1, and magnifies it by a factor of 20
`<STX>R<ETX>`	Saves and exits to Print mode
`<STX><ESC>E4<ETX>`	Selects format 4
`<STX><ETB><ETX>`	Prints the format

Creating User-Defined Font Characters for Advanced Mode

Creating user-defined font characters for Advanced mode is almost identical to the procedure for creating user-defined characters. You need to define the format for each ASCII character you want to use. For help, see **“Creating User-Defined Characters for Advanced Mode” on page 48.**

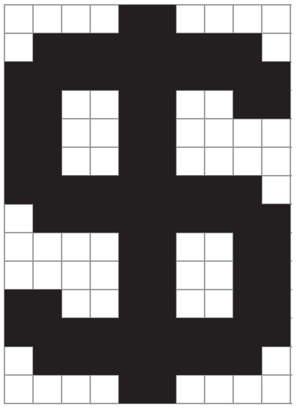
Additionally, when you write the command strings that define the ASCII characters for the font, you need to include syntax for these commands:

- “Bitmap User-Defined Font, Clear or Define”: This command creates a new font and font number. Syntax is `T`*n*, where *n* is the font number.
- “Bitmap Cell Width for Graphic or UDF, Define”: This command defines the width for all cells that contain a character from this font. Syntax is `x`*n*, where *n* is the cell width in dots.
- “Bitmap Cell Height for Graphic or UDF, Define”: This command defines the height for all cells that contain a character from this font. Syntax is `y`*n*, where *n* is the cell height in dots.
- “User-Defined Font Character, Create”: This command creates the new character. Syntax is `t`*n*, where *n* is the decimal equivalent of the corresponding ASCII character. For help, see the **“Full ASCII Table” on page 76.** You need to add this syntax for each new character in your user-defined font.

- "Intercharacter Space for UDF, Define": This command defines the amount of space added to the default intercharacter gap length for a bitmap font. Syntax is z*n*, where *n* is the number of dots to add to the default spacing of 2.

To create a user-defined font character for Advanced mode

1. Follow Steps 1 to 7 of **"To create a user-defined character for Advanced mode" on page 49.** This procedure uses the following graphic as an example:



2. Write a command string to create a user-defined font:

```
<STX>T3,FONT3<ETX>
```

3. Write a command string that defines the width and height for cells that contain characters in this font. For this example, the width and height are defined as 10 and 14 dots respectively:

```
<STX>x10;y14<ETX>
```

4. Write a command string that creates the specific ASCII character and defines the width of that character. For this example, the user-defined graphic corresponds to the $ symbol (ASCII character 36) and has a width of 12 dots:

```
<STX>t36;z12<ETX>
```

5. Add command strings for the character format you created earlier. Place the ASCII start of text character at the beginning of each line, followed by the "Graphic or UDC, Define" command syntax (u). Number each line from 0 to 14, and then add the ASCII end of text character to the end of the line:

```
<STX>u0,|q@;<ETX>
<STX>u1,~sA;<ETX>
<STX>u2,FcA;<ETX>
<STX>u3,FcA;<ETX>
<STX>u4,<DEL><DEL>C;<ETX>
<STX>u5,<DEL><DEL>C;<ETX>
<STX>u6,FcA;<ETX>
<STX>u7,FcA;<ETX>
<STX>u8,N<DEL>A;<ETX>
<STX>u9,L~@;<ETX>
```

6. Repeat steps 4 and 5 for each character in your user-defined font. You must create and download all characters in a user-defined font at the same time. If you try to add other user-defined characters to an already existing user-defined font, you erase previously created characters.

7. Write a command string to place the printer in Print mode:

```
<STX>R<ETX>
```

8 Save the text file and send it to the printer.

The user-defined font can now be used in a bar code label. You can specify the user-defined font for a human-readable or interpretive field. Characters in data you specify for those fields are mapped to the user-defined font.

The next table lists the complete command string set for the user-defined font in this example.

Command Strings for User-Defined Font Example

Commands	Definitions
`<STX><ESC>C<ETX>`	Selects Advanced mode
`<ESC>P;<ETX>`	Enters Program mode
`<STX>T3,FONT3<ETX>`	Create bitmap font 3
`<STX>x10;y14;<ETX>`	Define cell width (10) and cell height (14)
`<STX>t36;Z12;<ETX>`	Create ASCII character 36 ($), define character width as 12
`<STX>u0,|q@;<ETX>`	Defines column 0
`<STX>u1,~sA;<ETX>`	Defines column 1
`<STX>u2,FcA;<ETX>`	Defines column 2
`<STX>u3,FcA;<ETX>`	Defines column 3
`<STX>u4,<DEL><DEL>C;<ETX>`	Defines column 4
`<STX>u5,<DEL><DEL>C;<ETX>`	Defines column 5
`<STX>u6,FcA;<ETX>`	Defines column 6
`<STX>u7,FcA;<ETX>`	Defines column 7
`<STX>u8,N<DEL>A;<ETX>`	Defines column 8
`<STX>u9,L~@;<ETX>`	Defines column 9
`<STX>R;<ETX>`	Saves and exits to Print mode

4

Advanced Printer Programming

This chapter discusses topics for advanced IPL programmers and includes these sections:

- **Using the Printer Memory Efficiently**
- **Reimaging Modified Fields**
- **Using Emulation Mode**
- **Using Legacy Mode**
- **Using Direct Graphics Mode**
- **Designing Pages**

Using the Printer Memory Efficiently

To get the best performance from your printer, you need to use the printer RAM efficiently. This section includes tips for efficient use of printer memory.

Note: For the 3240, 3400C, 3400D, 3600, and 4100, using a significant amount of available memory for storage affects printer performance because there is less memory available for imaging.

How Is the Printer Storage Memory Used?

Although the printer contains enough static RAM or flash to store several different label formats, downloaded fonts, graphics, and data, you should be careful of how you use your printer memory.

Printers use either static RAM or flash memory for storing tables, pages, formats, fonts, and user-defined characters (UDCs). Some printers allow you to use the PrintSet application to adjust the amount of storage memory available for storage purposes. If you require additional storage memory, you can purchase a memory expansion option.

Making the Most of Your Storage Memory

There are limits to the number of formats, fonts, graphics, or pages that you can store in the printer. You can define up to 16 fonts, but there may not be enough room depending on the amount of memory being used for other purposes. The more formats, graphics, and fonts you store, the less memory is available.

When you encounter a memory usage problem, use PrintSet to see how much memory is available. You must upload the memory information from the printer first. For more information, see the PrintSet online help.

You can increase your available memory by following one of these suggestions:

- Delete any unneeded user-defined fonts, graphics, pages, or formats. For help, see the PrintSet online help or the specific commands in the ***IPL Command Reference Manual***.
- For the 3240, 3400 (except 3400e), 3600, and 4100 printers, you can adjust the amount of RAM allocated for storage purposes. For help, see the PrintSet software or the "Amount of Storage, Define" command in the ***IPL Command Reference Manual***.
- Purchase additional memory. For more information, contact your Intermec sales representative.

Reimaging Modified Fields

Note: This section applies only to the 3240, 3400C, 3400D, 3600, and 4100 printers.

You can reimage only the changed fields in a label format instead of the entire label. To update data in only changed fields, use the “Format, Select” command to specify that only changed fields should be reimaged. If used correctly, this command parameter can greatly increase the throughput of your printer; however, if you reimage a field that takes longer to erase and reimage than erasing and reimaging the entire label format, you will not be increasing throughput.

Follow the next procedure to use the “Format, Select” command as described.

To use the modified field reimaging command

1. Make sure that you select enough image bands to allow the printer to retain the entire label image.

   One image band is equal to 2.54 cm (1 in) of label length.

2. Select the fields to reimage by using the following command:

   ```
   <ESC>En,1
   ```

   where *n* is the format ID to be edited, and 1 specifies that only the changed fields should be reimaged.

   For more information, see the “Format, Select” command in the ***IPL Command Reference Manual***.

3. Print the label to see if throughput is improved.

Using Emulation Mode

Note: Emulation mode is not supported by the 4630 and 4830 printers, or by PD/PF/PM/PX series printers with 300 dpi printheads.

Emulation mode lets you print bar code labels that were designed on an 86XX printer in multiples of 10 or 15 mil. (“Emulation mode” is also called “86XX Emulation mode” in some printer manuals.)

Here is a summary of the features in Emulation mode:

- Pages are not available.
- Character size is specified by height and width magnification only.
- International characters are preceded by <SUB> or selectable by language.
- The bitmaps for user-defined characters (UDC) and user-defined fonts (UDF) are one bit per byte (instead of six bits per byte).
- Print resolution is in 10 mil dots.
- RFID commands are not available.

Two IPL commands place the printer in Emulation mode:

- Emulation or Advanced Mode on Power-Up
- Emulation Mode, Enter

The following table lists all the IPL commands that work when the printer is in Emulation mode.

IPL Commands and Emulation Mode

IPL Command Name	Syntax
12 Volt Supply Value, Transmit	`U`
Abort Print Job	`<EM>`
Advanced Mode, Select	`<ESC>C`
Alphanumeric Field Separator	`<GS>`
Auto-Transmit 1, Enable	`<ESC>j`
Auto-Transmit 2, Enable	`<ESC>d`
Auto-Transmit 3, Enable	`<ESC>e`
Auto-Transmit 1, 2, and 3, Disable	`<ESC>k`
Bar Code, Select Type	`c`
Bar Code Field, Create or Edit	`B`
Batch Count, Set	`<US>`
Bitmap Cell Height for Graphic or UDF, Define	`y`
Bitmap Cell Width for Graphic or UDF, Define	`x`
Bitmap User-Defined Font, Clear or Define	`T`
Border Around Human-Readable Text, Define	`b`
Box Field, Create or Edit	`W`
Character Bitmap Origin Offset, Define	`X`
Character Rotation or Bar Code Ratio, Define	`r`
Clear All Data	`<CAN>`
Clear Data From Current Field	`<DEL>`
Code 39 Prefix Character, Define	`p`
Command Tables, Load	`C`
Command Terminator	`;`
Command Terminator 1	`<NUL>`
Command Terminator 2	`<LF>`
Communication Port Configuration, Set	`<SI>P`
Configuration Parameters, Transmit	`<ESC>p`
Control Panel Access Permission, Set	`<SI>A`
Current Edit Session, Save	`N`
Cut	`<SO>`

IPL Commands and Emulation Mode (continued)

IPL Command Name	Syntax
Cutter, Enable or Disable	`<SI>c`
Dark Adjust	`K`
Dark Adjust, Set	`<SI>d`
Data Shift - International Characters	`<SUB>`
Data Source for Format in a Page, Define	`e`
Direct Graphics Mode, Select	`<ESC>g`
Emulation Mode, Enter	`<ESC>c`
Emulation or Advanced Mode on Power-Up	`<SI>C`
End-of-Print Skip Distance, Set	`<SI>D`
Error Code, Request	`<BEL>`
Factory Defaults, Reset	`D`
Field, Delete	`D`
Field, Select	`<ESC>F`
Field Data, Define Source	`d`
Field Decrement, Set	`<ESC>D`
Field Direction, Define	`f`
Field Increment, Set	`<ESC>I`
Field Origin, Define	`o`
First Data Entry Field, Select	`<ACK>`
Font, Transmit	`<ESC>v`
Font Character Width, Define	`Z`
Font Type, Select	`c`
Form Feed	`<FF>`
Format, Create or Edit	`A or F`
Format, Erase	`E`
Format, Select	`<ESC>E`
Format, Transmit	`<ESC>x`
Format Direction in a Page, Define	`q`
Format Offset Within a Page, Define	`O`
Format Position From Page, Delete	`m`
Format Position in a Page, Assign	`M`
Formats, Print	`f`
Graphic, Select	`c`
Graphic or UDC, Define	`u`
Hardware Configuration Label, Print	`h`
Height Magnification of Bar, Box, or UDC,	`h`
Human-Readable Field, Create or Edit	`H`
IBM Language Translation, Enable or Disable	`<SI>I`
Increment and Decrement, Disable	`<ESC>N`
Intercharacter Delay, Set	`<SYN>`
Intercharacter Space for UDF, Define	`z`
Interpretive Field, Edit	`I`

IPL Commands and Emulation Mode (continued)

IPL Command Name	Syntax
Interpretive Field, Enable or Disable	`I`
Label and Gap Length, Transmit	`<ESC> L`
Label Rest Point, Adjust	`<SI>f`
Label Retract, Enable or Disable	`<SI>R`
Label Retract Distance, Set	`<SI>r`
Label Stock Type, Select	`<SI>T`
Label Taken Sensor Value, Transmit	`T`
Length of Line or Box Field, Define	`l`
Line Field, Create or Edit	`L`
Maximum Label Length, Set	`<SI>L`
Media Fault Recovery Mode, Set	`<SI>e`
Media Sensitivity, Select	`<SI>g`
Memory Usage, Transmit	`<ESC>m`
Message Delay, Set	`<ESC><SYN>`
Next Data Entry Field, Select	`<CR>`
Number of Image Bands, Set	`<SI>I`
Numeric Field Separator	`<FS>`
Options Selected, Transmit	`<ESC>O`
Outline Font, Clear or Create	`J`
Outline Font, Download	`j`
Page, Create or Edit	`S`
Page, Delete	`s`
Page, Select	`<ESC>G`
Page, Transmit	`<ESC>y`
Pages, Print	`p`
Pin 11/20 Protocol, Set	`<SI>p`
Pitch Label, Print	`C`
Pitch Size, Set	`g`
Point Size, Set	`k`
Postamble, Set	`<EOT>`
Preamble, Set	`<SOH>`
Print	`<ETB>`
Print Quality Label, Print	`Q`
Print Speed, Set	`<SI>S`
Printer Language, Select	`<SI>l`
Printhead Loading Mode, Select	`<SI>h`
Printhead Parameters, Transmit	`<ESC>H`
Printhead Temperature Sensor Value, Transmit	`P`
Program Mode, Enter	`<ESC>P`
Program Mode, Exit	`R`
Program Number, Transmit	`<ESC>M`
Quantity Count, Set	`<RS>`

IPL Commands and Emulation Mode (continued)

IPL Command Name	Syntax
Reflective Sensor Value, Transmit	`M`
Remaining Quantity and Batch Count, Transmit	`<ESC>Q`
Reset	`<DLE>`
Self-Strip, Enable or Disable	`<SI>t`
Slash Zero, Enable or Disable	`<SI>z`
Software Configuration Label, Print	`s`
Start and Stop Codes (Code 39), Print	`<ESC><SP>`
Status Dump	`<VT>`
Status Enquiry	`<ENQ>`
Takeup Motor Torque, Increase	`<SI>b`
Test and Service Mode, Enter	`<ESC>T`
Test and Service Mode, Exit	`R`
Top of Form, Set	`<SI>F`
Transmissive Sensor Value, Transmit	`G`
User-Defined Character (UDC) and Graphics,	`g`
User-Defined Character, Clear or Create	`G`
User-Defined Character Field, Create or Edit	`U`
User-Defined Characters, Transmit	`<ESC>u`
User-Defined Font Character, Create	`t`
User-Defined Fonts, Print	`t`
User-Defined Tables, Transmit	`<ESC>Z`
Warm Boot	`<BS>`
Width of Line, Box, Bar, or Character, Define	`w`

Using Legacy Mode

Legacy mode lets you print bar code labels that were designed on legacy 3400/4420 printers.

Here is a summary of the features in Emulation mode:

- Fonts 23 and 24 will be bitmapped fonts instead of scalable fonts.
- Fonts 25, 26, and 28 will be Speedo fonts, and fonts 20 to 22, and 30 to 41 will be generated from the corresponding Speedo fonts.
- Code 39 will be 86XX-compatible. When the fields and formats are stored in Legacy Mode, they can be printed in any mode regardless of the mode the printer is in.
- When the Legacy mode setting is changed, an automatic reboot will be triggered upon exiting Setup.

Two IPL commands place the printer in Legacy mode:

- Emulation or Advanced Mode on Power-Up
- Emulation Mode, Enter

For help, see the “Emulation or Advanced Mode on Power-Up” command in the ***IPL Command Reference Manual***.

Using Direct Graphics Mode

You can significantly reduce the amount of time necessary to download and image a graphic by using Direct Graphics mode. Direct Graphics mode allows the printer to receive a compressed bitmap graphic and image it directly into the image bands without storing it in the printer.

Before you download the graphic, you must compress it into run-length encoded (RLE) data. The data compression greatly reduces the amount of data to download and the rasterized graphic requires minimal processing to image it into the image bands. You no longer need to store the graphic in Program mode and then set up a format in Print mode.

When you download a direct graphic to the printer, the printer stores the graphic in the image bands until you:

- clear the label data.
- set up another format.
- enter Program mode or Test and Service mode.

When printing a label with direct graphics, you must have enough dynamic RAM installed in your printer to contain the entire label. Because Intermec printers normally reuse image bands, you can print long labels with standard RAM; however, when you download direct graphics, the printer retains no information regarding the existence of the graphic in its image bands. Therefore, the printer cannot reuse those image bands when you download a direct graphic.

With standard memory, you should be able to print almost any label up to 15.2 cm (6 in) long. You may need to install more memory for longer labels.

What Is Run-Length Encoding?

Run-length encoding (RLE) is a method of compressing bitmap graphics. RLE compresses graphics that have repeated runs of white or black dots in a column, reducing the amount of time required to download the graphics to a printer.

RLE sends a series of commands that define each bitmap column of a graphic and takes advantage of a series of repeated dots within a column by encoding them as transition commands. Instead of sending the entire column of bitmap data, it sends commands telling the printer how many series of black and white dots to image.

If columns are identical, a command can instruct the printer to repeat the last column. RLE is ideal for bar code graphics or designs with simple patterns.

In cases where patterns do not exist, you can send uncompressed bitmap data to the printer. You can mix raw bitmap data and RLE commands to ensure the most efficient way to download a graphic.

The RLE file may contain five types of data, each of which is one byte long:

Immediate Commands

Recognized and executed as regular IPL commands or protocol commands. Syntax is removed from compressed data.

- Byte format (7-0): 000xxxx
- Range is 0 to 31

Compression Encoding Commands

Used as part of the compressed graphics file to change or set data modes, repeat lines, change the origin for the next lines of data, or end the compressed graphics file and return to IPL command printing.

- Byte format (7-0): 001xxxx
- Range is 32 to 63

Low Order Data

Can represent up to 7 bits of data (0 to 127). Must be preceded by a command byte so the printer knows how to interpret them.

- 7 bits long and may be combined with high order data. 8th bit is always set to 1.
- Byte format (7-0): 1xxxxxx.
- Range is 128 to 255

High Order Data

When combined with low order data, can represent up to 13 bits of data (0-8191). Must be preceded by a command byte so the printer knows how to interpret them. Printer ignores high order data followed by a command or more high order data.

- 6 bits long and always combined with low order data. 7th bit is always set to 1 and 8th bit is always set to 0.
- Byte format (7-0): 01xxxxx
- Range: 64 - 127
- Data represented: 0 to 63

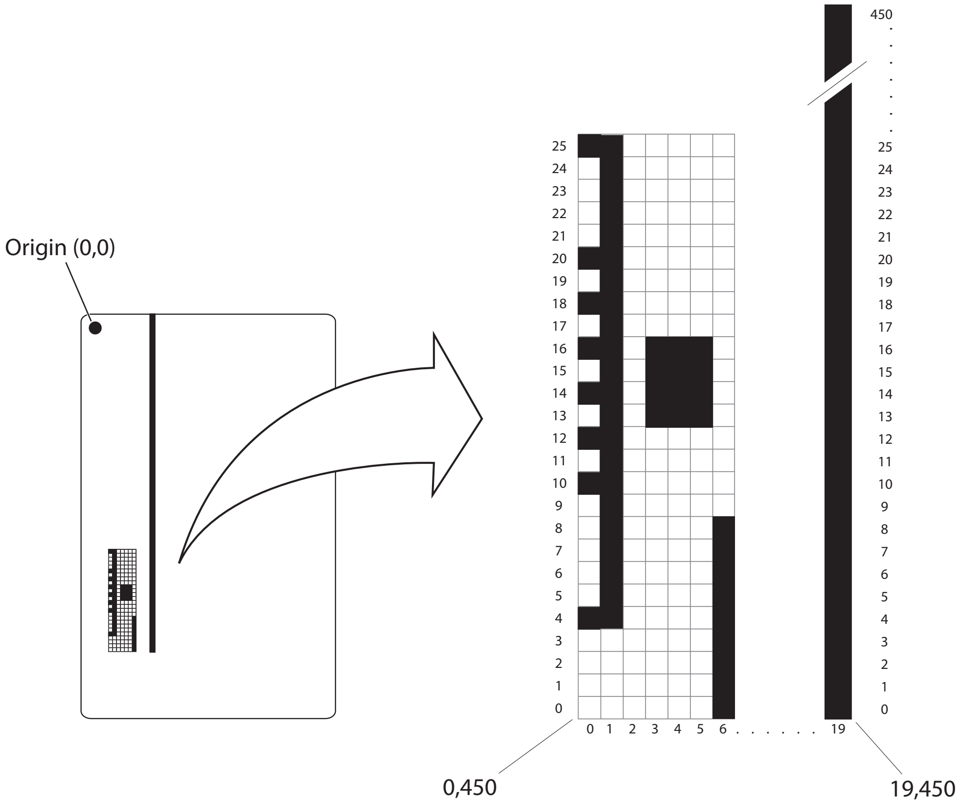
Bitmap Data

Composed of uncompressed bytes (7 data bits per byte) that represent columns of your graphic. 8th bit is always set to 1.

- Byte format (7-0): 1xxxxxx
- Range: 128 to 255
- Data represented: Raw data

Example of Direct Graphics Commands

This example consists of two parts: a graphic and a line. The origin of the direct graphic is defined the same as the origin of a normal graphic. In this example, the coordinates for the origin of the complex graphic are 0,450; however, once you enter Direct Graphics mode, your printer loads the information in the reverse y direction. Each column of the graphic loads from the bottom to the top. Y coordinates now start at 0 from the bottom left corner and increase in size as the data loads. So, the printer starts loading data for the complex graphic at 0,450 and loads up to 0,425. Likewise, the data for the line starts loading at 19,450 and loads up to 19,0.



Direct Graphics Commands: *This example shows how the printer loads information in Direct Graphics mode.*

This table shows a hex data file for the example:

Hex Data File

1B	67	30	21	80	43	C2	27	90	A8	D5
90	22	26	84	96	22	22	26	8C	84	22
24	82	25	88	22	21	93	43	C2	25	43
C2	28									

The next table explains the hex data file in the example.

Hex Data Commands in Example

Data	Command	Description
1B 67 30	<ESC>g0	Enter Direct Graphics mode
21 80 43 C2	0x21	Change origin
	80	80 - 80 (LO) = 0x00 -> X0
	43 C2	43 - 40 (HI) = 0x03 C2 - 80 (LO) = 0x42 (0x03 * 0x80) + 0x42 = 1C2 -> Y450
27 90 A8 D5 90 22	0x27	Raw bitmap data follows, starts at Y0
	90	90 - 80 (LO) = 0x10 -> 1 dot at Y4
	A8	A8 - 80 (LO) = 0x28 -> 2 dots at Y10 and Y12
	D5	D5 - 80 (LO) = 0x45 -> 3 dots at Y14, Y17, and Y20
	90	90 - 80 (LO) = 0x10 -> 1 dot at Y25
	0x22	End of line
26 84 96 22	0x26	Transition white
	84	84 - 80 (LO) = 4 white
	96	96 - 80 (LO) = 22 black
	0x22	End of line
22	0x22	End of line
26 8C 84 22	0x26	Transition white
	8D	8D - 80 (LO) = 13 white
	84	84 - 80 (LO) = 4 black
	0x22	End of line
24 82	0x24	Repeat last line
	82	82 - 80 (LO) = 2 times
25 88 22	0x25	Transition black
	88	89 - 80 (LO) = 9 black
	0x22	End of line
21 93 43 C2	0x21	Change origin
	93	93 - 80 (LO) = 0x13 -> X19
	43 C2	43 - 40 (HI) = 0x03 C2 - 80 (LO) = 0x42 (0x03 * 0x80) + 0x42 = 1C2 -> Y450
25 43 C2	0x25	Transition black
	43 C2	43 - 40 (HI) = 0x03 C2 - 80 (LO) = 0x42 (0x03 * 0x80) + 0x42 = 1C2 -> Y450
28	0x28	End of bitmap

Designing Pages

A page is a collection of one or more formats that you combine to print at the same time. This feature is helpful when you need to print several different labels for an application at once. For example, you may need to attach one type of label to a product and a different type of label to its container. With the page printing capability, you can print both labels at the same time. Because you can print pages of several formats at once, you can also print labels on media rolls that have different sizes and shapes of labels already precut.

When you group label formats into a page, you assign the formats to positions designated by the letters a through z. You can print the formats used in pages independent of each other.

5

Troubleshooting

This chapter describes the problems that may occur as a result of using IPL commands incorrectly. If you do not find your problem listed here, see the troubleshooting information in your printer user's manual.

Troubleshooting Checklist

If you receive an error message from the printer or have problems while operating the printer, follow the next procedure to troubleshoot the printer.

To troubleshoot your printer

1 Send the “Error Code, Request” command (syntax <BEL>) to the printer and see if the printer returns an error message.

2 If the printer returns an error message, find it in the section called **“Interpreting Error Codes and Solving Problems” on page 71**. Follow the instructions in the table to correct the problem.

If the printer does not return an error message, try to locate the symptom in the “Printer Operation Problems” and “Print Quality Problems” sections of the printer user’s manual. Follow the instructions in the manual to correct the problem.

3 Clean the printer components and check all connections. For more information, see the user’s manual for your printer.

4 If the problem persists, contact Intermec Product Support (1-800-755-5505) in North America. If you are an international customer, contact your local Intermec representative.

How the Printer Handles Error Conditions

This section describes how the printer handles error conditions that may occur while you use IPL commands.

Syntax Errors

The printer responds to syntax errors in the messages it receives from the host by attempting to execute the commands. It does not ignore a command with a syntax error; instead, the printer produces output, even if it is wrong. This output helps determine what went wrong and what should be done to correct the problem.

For more information on command syntax, see the ***IPL Command Reference Manual***.

Parameter Errors

Certain commands require optional parameters. If you do not supply these parameters, the printer substitutes default values. If a parameter is above its maximum range limit, the printer uses the maximum value. If it falls below the minimum range, the printer uses the minimum value. See the ***IPL Command Reference Manual*** for the range and default value for each command.

Image Overrun Errors

Image overrun occurs when a label is too complex to image for a given print speed. An overrun will cause the printer to abort the label being printed. This error is most common on labels over 12.7 cm (5 in) long.

The printer automatically attempts to correct for this error condition by resetting to the lowest print speed and to the highest number of image bands, then repeats printing the label. The printer remains at this setting until you reset it. If an image overrun still occurs, printing for that batch of labels stops and the printer executes any following commands.

Note: Installing more memory may decrease image overrun errors.

Invalid Numeric Character Errors

If you include non-numeric characters within a numeric data string in a command, the printer ignores them and continues to process the rest of the valid numeric characters. If a non-numeric character begins the numeric data string, however, the printer uses a default value for the affected command.

Here are two examples of valid numeric character strings and one example of an invalid string:

Valid: 12a
Valid: 1a2
Invalid: a12

Insufficient Storage Memory Errors

Before storing new formats, graphics, or user-defined fonts in the static RAM or flash, the printer ensures that it has sufficient memory to store them. If there is insufficient memory, the printer ignores the last editing session and preserves the existing data in the storage memory.

Interpreting Error Codes and Solving Problems

Most of the problems you may encounter cause the printer to send an error code to the host. To correct the error, find the error code in the following table and complete the instructions in the solution column.

Note: For information on specific commands, see the ***IPL Command Reference Manual***.

Error Codes and Possible Solutions

Error Code	Description of Problem	Solution
00	No error.	No action is necessary.
02	Invalid number of bar code characters (UPC/EAN).	Verify the number of bar code characters used. For more information, see the “Bar Code, Select Type” command in the ***IPL Command Reference Manual***.
11	Invalid bar code data.	Verify data in the label format.
12	Data count exceeded.	Data count should not exceed what is specified for the field.

Error Codes and Possible Solutions (continued)

Error Code	Description of Problem	Solution
21	Quantity or batch count out of range.	Quantity of labels or number of batches should be between 1 and 9999.
22	Field increment/decrement out of range.	Quantity should be between 1 and 9999.
25	Invalid format transmission syntax.	Check the "Format, Transmit" command syntax.
26	Invalid page transmission syntax.	Check the "Page, Transmit" command syntax.
27	Invalid font transmission syntax.	Check the "Font, Transmit" command syntax.
28	Invalid UDC transmission syntax.	Check the "User-Defined Characters, Transmit" command syntax.
33	Invalid field delimiters.	Check for all pairs of field delimiters and make sure both are numeric, or both are alphanumeric.
34	Invalid escape command.	Correct the escape command syntax.
35	Invalid data shift command.	Correct the shift command syntax.
36	Invalid or undefined format number.	Verify that the format numbers are between 0 and 19.
37	Insufficient room in RAM to print format.	Reduce the number of data fields in the format or add more DRAM.
38	Invalid or undefined field number.	Verify the field number in the label format.
41	Syntax error for program commands.	Check the program command for proper syntax.
42	Insufficient room in RAM to store format.	Empty the buffer contents. If the format still does not fit, delete some fields or other data from the format. You may have to remove or reduce the UDCs, formats, or fonts if necessary. To determine the amount of installed and available memory, send the "Memory Usage, Transmit" command.
43	Too many fields in label format.	You can use up to 200 fields in a format and each field can use up to 250 characters. Reduce field size or delete some fields.
46	Undefined statement.	Check the statement syntax.
52	Invalid UDC/UDF bitmap cell height/width or intercharacter space.	Verify that the UDC/UDF bitmap cell height/width or intercharacter space is within the specified values for *n*. For more information, see these commands in the ***IPL Command Reference Manual***: • "Bitmap Cell Height for Graphic or UDF, Define" • "User-Defined Character Field, Create or Edit" • "Intercharacter Space for UDF, Define"
53	Not enough room in RAM to store UDC or UDF.	Remove or reduce formats, fonts, or UDCs.
54	Invalid UDC command syntax.	Correct the UDC command syntax.
55	Not enough room in RAM to store downloaded font.	Add more printer RAM.
56	Flash file system error.	
57	Invalid parameter.	Correct the syntax.
60	No RFID support.	The printer is unable to access the RFID module. Contact your local Intermec representative.
61	No RFID tag found.	The RFID module has determined that the printer does not have RFID media loaded. Load RFID media. Verify that the TAGADJUST value aligns the tag over the antenna. For help, see the "RFID Parameters, Set" command in the ***IPL Command Reference Manual***.

Error Codes and Possible Solutions (continued)

Error Code	Description of Problem	Solution
62	Access outside of the RFID tag memory.	The defined RFID tag field specifies a position outside of the tag's memory or the tag read/write operation was not successful (bad tag). Verify that the field specifies a position within the tag's memory. For help, see the RFID Tag Field Setup command in the ***IPL Command Reference Manual***.
63	RFID number conversion error.	You have entered an invalid hex or numerical (NUM) format in the data string. Enter a valid value.
64	RFID inactive (off).	You have a printer with an RFID module and you have defined RFID commands, but you have turned RFID mode off. Turn RFID mode on. For help, see the RFID Parameters, Set command in the ***IPL Command Reference Manual***.
65	RFID tag type does not support the selected option.	An application has sent a command with an argument that is not supported by the printer's tag type. Verify that the application commands match the printer's tag type.
66	RFID module overheated.	The printer RFID module has exceeded its recommended operating temperature and shut down automatically. Wait until the module has cooled to the recommended operating temperature before you try to print tags again.
67	RFID module duty cycle exceeded.	The printer RFID module has exceeded its recommended duty cycle and shut down automatically. You may need to check the RFID module settings. For help, see the ***Basic Reader Interface (BRI) Programmer's Reference Manual***.
68	RFID lock error.	You are trying to write data to a locked RFID tag. Use the "RFID Tag Protect" command to unlock the tag and try again.
69	RFID Access error.	Check the syntax for the "RFID Tag Write Field, Create or Edit" command.
70	Illegal number of characters.	Check all parameter settings in your syntax.
71	Illegal characters.	Check all characters in your syntax and correct them if necessary.

Printing Labels with 86XX-Compatible Code 39

Note: This section applies to the PF/PM/PX series printers only.

If you are using a PF/PM/PX series printer, and are having trouble printing Code 39 labels designed for an 86XX printer, use the "Emulation or Advanced Mode on Power-Up" command to place the printer in Legacy Emulation mode. In this mode:

- Code 39 characters are interpreted as 86XX-compatible Code 39 characters. For example, the characters $, / +, and % are encoded as $, /, +, and % instead of as /D, /O, /K, and /E.
- Fonts 23 and 24 are bitmap fonts instead of TrueDoc fonts.
- Fonts 25, 26, and 28 are Speedo fonts instead of TrueDoc fonts.
- Fonts 20, 21, 22, and 30 through 41 are generated from the corresponding Speedo fonts.

For more information, see the “Emulation or Advanced Mode on Power-Up” command in the ***IPL Command Reference Manual***.

A

Full ASCII Tables and International Character Sets

This appendix contains the full ASCII chart, with binary, hexadecimal, and Code 39 equivalents, and an ASCII control character chart. The appendix also includes tables that show which hex codes to download for international characters not available in the U.S. character set.

Full ASCII Table

Full ASCII Table

Binary[0]	Hex[1]	Decimal	Code 39	ASCII[2]	Binary[0]	Hex[1]	Decimal	Code 39	ASCII[2]
00000000	00	00	%U	NUL	00100011	23	35	/C	#
00000001	01	01	$A	SOH	00100100	24	36	/D	$
00000010	02	02	$B	STX	00100101	25	37	/E	%
00000011	03	03	$C	ETX	00100110	26	38	/F	&
00000100	04	04	$D	EOT	00100111	27	39	/G	′
00000101	05	05	$E	ENQ	00101000	28	40	/H	(
00000110	06	06	$F	ACK	00101001	29	41	/I	)
00000111	07	07	$G	BEL	00101010	2A	42	/J	*
00001000	08	08	$H	BS	00101011	2B	43	/K	+
00001001	09	09	$I	HT	00101100	2C	44	/L	,
00001010	0A	10	$J	LF	00101101	2D	45	/M	-
00001011	0B	11	$K	VT	00101110	2E	46	/N	.
00001100	0C	12	$L	FF	00101111	2F	47	/O	/
00001101	0D	13	$M	CR	00110000	30	48	/P4	0
00001110	0E	14	$N	SO	00110001	31	49	/Q	1
00001111	0F	15	$O	SI	00110010	32	50	/R	2
00010000	10	16	$P	DLE	00110011	33	51	/S	3
00010001	11	17	$Q	DC1	00110100	34	52	/T	4
00010010	12	18	$R	DC2	00110101	35	53	/U	5
00010011	13	19	$S	DC3	00110110	36	54	/V	6
00010100	14	20	$T	DC4	00110111	37	55	/W	7
00010101	15	21	$U	NAK	00111000	38	56	/X	8
00010110	16	22	$V	SYN	00111001	39	57	/Y	9
00010111	17	23	$W	ETB	00111010	3A	58	/Z	:
00011000	18	24	$X	CAN	00111011	3B	59	%F	;
00011001	19	25	$Y	EM	00111100	3C	60	%G	<
00011010	1A	26	$Z	SUB	00111101	3D	61	%H	=
00011011	1B	27	%A	ESC	00111110	3E	62	%I	>
00011100	1C	28	%B	FS	00111111	3F	63	%J	?
00011101	1D	29	%C	GS	01000000	40	64	%V	@
00011110	1E	30	%D	RS	01000001	41	65	A	A
00011111	1F	31	%E	US	01000010	42	66	B	B
00100000	20	32	SP	SP[3]	01000011	43	67	C	C
00100001	21	33	/A	!	01000100	44	68	D	D
00100010	22	34	/B	″	01000101	45	69	E	E
01000110	46	70	F	F	01100011	63	99	+C	c
01000110	46	70	F	F	01100011	63	99	+C	c
01000111	47	71	G	G	01100100	64	100	+D	d

Full ASCII Table (continued)

Binary[0]	Hex[1]	Decimal	Code 39	ASCII[2]	Binary[0]	Hex[1]	Decimal	Code 39	ASCII[2]
01001000	48	72	H	H	01100101	65	101	+E	e
01001001	49	73	I	I	01100110	66	102	+F	f
01001010	4A	74	J	J	01100111	67	103	+G	g
01001011	4B	75	K	K	01101000	68	104	+H	h
01001100	4C	76	L	L	01101001	69	105	+I	i
01001101	4D	77	M	M	01101010	6A	106	+J	j
01001110	4E	78	N	N	01101011	6B	107	+K	k
01001111	4F	79	O	O	01101100	6C	108	+L	l
01010000	50	80	P	P	01101101	6D	109	+M	m
01010001	51	81	Q	Q	01101110	6E	110	+N	n
01010010	52	82	R	R	01101111	6F	111	+O	o
01010011	53	83	S	S	01110000	70	112	+P	p
01010100	54	84	T	T	01110001	71	113	+Q	q
01010101	55	85	U	U	01110010	72	114	+R	r
01010110	56	86	V	V	01110011	73	115	+S	s
01010111	57	87	W	W	01110100	74	116	+T	t
01011000	58	88	X	X	01110101	75	117	+U	u
01011001	59	89	Y	Y	01110110	76	118	+V	v
01011010	5A	90	Z	Z	01110111	77	119	+W	w
01011011	5B	91	%K	[	01111000	78	120	+X	x
01011100	5C	92	%L	\	01111001	79	121	+Y	y
01011101	5D	93	%M	]	01111010	7A	122	+Z	z
01011110	5E	94	%N	^	01111011	7B	123	%P	{
01011111	5F	95	%O	_	01111100	7C	124	%Q	|
01100000	60	96	%W	`	01111101	7D	125	%R	}
01100001	61	97	+A	a	01111110	7E	126	%S	˜
01100010	62	98	+B	b	01111111	7F	127	%T[5]	n[6]

Notes:
0 Bit positions are 76543210.
1 Hexadecimal value
2 ASCII character
3 SP is the SPACE character.
4 The Code 39 characters /P through /Y may be interchanged with the numbers 0 through 9.
5 May be interchanged with %X or %Y or %Z.
6 n is the DELETE character.

Full ASCII Control Characters Table

Control Character	Control Code	Definition
NUL	ˆ@	Null, or all zeroes
SOH	ˆA	Start of Heading
STX	ˆB	Start of Text
ETX	ˆC	End of Text
EOT	ˆD	End of Transmission
ENQ	ˆE	Enquiry
ACK	ˆF	Acknowledgment
BEL	ˆG	Bell
BS	ˆH	Backspace
HT	ˆI	Horizontal Tab
LF	ˆJ	Line Feed
VT	ˆK	Vertical Tab
FF	ˆL	Form Feed
CR	ˆM	Carriage Return
SO	ˆN	Shift Out
SI	ˆO	Shift In
DLE	ˆP	Data Link Escape
DC1	ˆQ	Device Control 1 (XON)
DC2	ˆR	Device Control 2
DC3	ˆS	Device Control 3 (XOFF)
DC4	ˆT	Device Control
NAK	ˆU	Negative Acknowledge
SYN	ˆV	Synchronous Idle
ETB	ˆW	End Transmission Block
CAN	ˆX	Cancel
EM	ˆY	End of Medium
SUB	ˆZ	Substitute
ESC	ˆ[	Escape
FS	ˆ\	File Separator
GS	ˆ]	Group Separator
RS	ˆˆ	Record Separator
US	ˆ_	Unit Separator
SP	None	Space
DEL	ˆ?	Delete

International Character Sets

The following tables show which hex codes to download for international characters not available in the U.S. character set. To use the tables, find the hex code for the U.S. character that corresponds with the character in your language.

Advanced Character Table

If you are running your printer in Advanced mode, use this table to find the right hex codes for the international character sets.

	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
U.S. ASCII	#	$	@	[	\	]	^	`	{	¦	}	˜
U.K. ASCII	£	$	@	[	\	]	^	`	{	¦	}	-
Germany	#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
France	£	$	à	°	ç	§	^	`	é	ù	è	¨
Norway/Denmark	#	$	@	Æ	Ø	Å	^	`	æ	ø	å	-
Sweden/Finland	#	≠	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Spain	£	$	§	¡	Ñ	¿	^	`	°	ñ	ç	˜
Switzerland	#	$	à	°	ç	é	^	ù	ä	ö	ü	è
Italy	£	$	§	°	ç	é	^	ù	à	ò	è	ì

86XX Character Table

This table shows the hex codes for the character sets that print if your printer is running under 86XX emulation mode.

	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
U.S. ASCII	#	$	@	[	\	]	^	`	{	¦	}	˜
U.K. ASCII	£	$	@	[	\	]	^	`	{	¦	}	˜
Germany	#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
France	£	$	à	°	ç	§	^	`	é	ù	è	¨
Norway/Denmark	#	$	@	Æ	Ø	Å	^	`	æ	ø	å	˜
Sweden/Finland	#	≠	É	Ä	Ö	Å	Ü	é	ä	ö	å	ü
Spain	₧	$	@	¡	Ñ	¿	^	`	¨	ñ	ç	˜
Switzerland	#	$	à	°	ç	é	^	ù	ä	ö	ü	è
Italy	#	$	§	°	ç	é	^	ù	à	ò	è	ì

IBM Translation Character Table

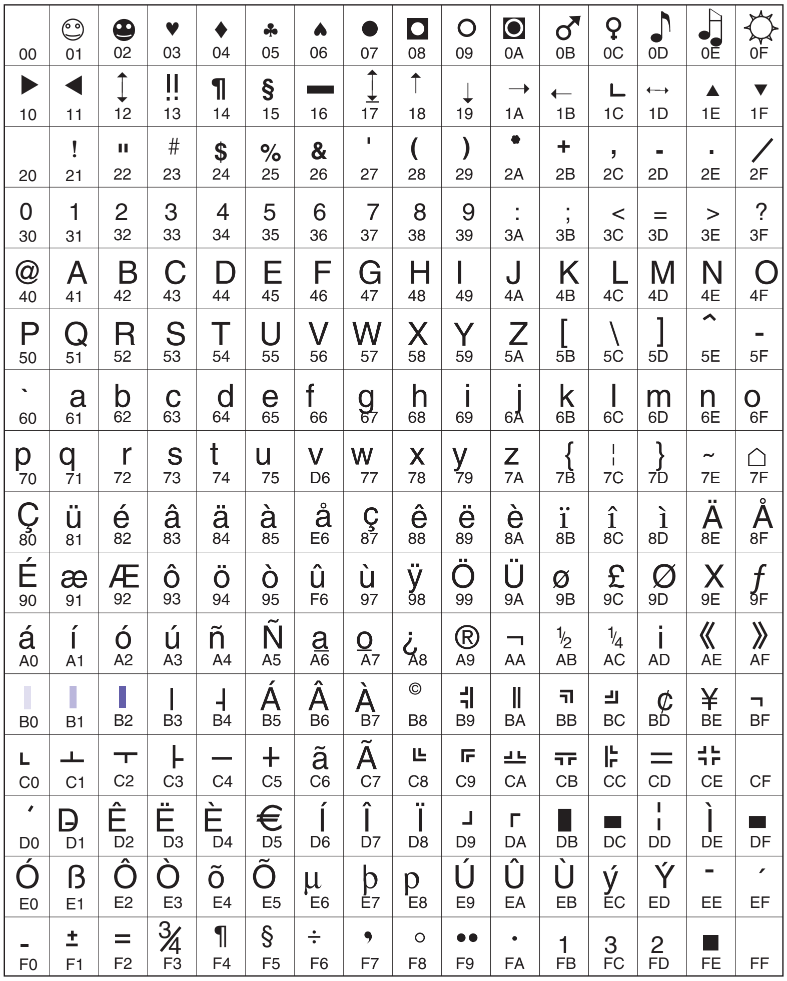
This table shows the hex codes for the international character sets that print if your printer is running with Translation enabled.

	21	23	24	40	5B	5C	5D	5E	60	7B	7C	7D	7E
U.S. ASCII	¦	#	$	@	¢	\	!	ÿ	`	{	¦	}	˜
U.K. ASCII	¦	#	£	@	$	\	!	ÿ	`	{	¦	}	-
Germany	!	#	$	§	Ä	Ö	Ü	^	`	ä	ö	ü	ß
France	!	£	$	à	°	ç	§	^	`	é	ù	è	¨
Norway/Denmark	!	Æ	Å	Ø	#	\	≠	^	`	æ	ø	å	ü
Sweden/Finland	!	Ä	Å	Ö	§	É	≠	^	é	ä	ö	å	ü
Spain	¦	Ñ	₧	@	[	\	]	ÿ	`	{	ñ	}	¨
Switzerland	!	#	$	à	°	ç	é	^	ù	ä	ö	ü	è
Italy	!	£	$	§	°	ç	é	^	ù	à	ò	è	ì

Code Page 850 Character Table

This table shows the character set that prints if your printer has Code Page 850 selected as the printer language.

Note: Some Intermec printers do not support Code Page 850. For more information, see the "Printer Language, Select" command in the ***IPL Command Reference Manual***.



00	☺ 01	☻ 02	♥ 03	♦ 04	♣ 05	♠ 06	● 07	◘ 08	○ 09	◙ 0A	♂ 0B	♀ 0C	♪ 0D	♫ 0E	☼ 0F
► 10	◄ 11	↕ 12	‼ 13	¶ 14	§ 15	▬ 16	↨ 17	↑ 18	↓ 19	→ 1A	← 1B	∟ 1C	↔ 1D	▲ 1E	▼ 1F
20	! 21	" 22	# 23	$ 24	% 25	& 26	' 27	(28	) 29	* 2A	+ 2B	, 2C	- 2D	. 2E	/ 2F
0 30	1 31	2 32	3 33	4 34	5 35	6 36	7 37	8 38	9 39	: 3A	; 3B	< 3C	= 3D	> 3E	? 3F
@ 40	A 41	B 42	C 43	D 44	E 45	F 46	G 47	H 48	I 49	J 4A	K 4B	L 4C	M 4D	N 4E	O 4F
P 50	Q 51	R 52	S 53	T 54	U 55	V 56	W 57	X 58	Y 59	Z 5A	[5B	\ 5C	] 5D	^ 5E	- 5F
` 60	a 61	b 62	c 63	d 64	e 65	f 66	g 67	h 68	i 69	j 6A	k 6B	l 6C	m 6D	n 6E	o 6F
p 70	q 71	r 72	s 73	t 74	u 75	v D6	w 77	x 78	y 79	z 7A	{ 7B	¦ 7C	} 7D	~ 7E	⌂ 7F
Ç 80	ü 81	é 82	â 83	ä 84	à 85	å E6	ç 87	ê 88	ë 89	è 8A	ï 8B	î 8C	ì 8D	Ä 8E	Å 8F
É 90	æ 91	Æ 92	ô 93	ö 94	ò 95	û F6	ù 97	ÿ 98	Ö 99	Ü 9A	ø 9B	£ 9C	Ø 9D	× 9E	ƒ 9F
á A0	í A1	ó A2	ú A3	ñ A4	Ñ A5	ª A6	º A7	¿ A8	® A9	¬ AA	½ AB	¼ AC	¡ AD	« AE	» AF
░ B0	▒ B1	▓ B2	│ B3	┤ B4	Á B5	Â B6	À B7	© B8	╣ B9	║ BA	╗ BB	╝ BC	¢ BD	¥ BE	┐ BF
└ C0	┴ C1	┬ C2	├ C3	─ C4	┼ C5	ã C6	Ã C7	╚ C8	╔ C9	╩ CA	╦ CB	╠ CC	═ CD	╬ CE	CF
ʹ D0	Đ D1	Ê D2	Ë D3	È D4	€ D5	Í D6	Î D7	Ï D8	┘ D9	┌ DA	█ DB	▄ DC	¦ DD	Ì DE	▀ DF
Ó E0	ß E1	Ô E2	Ò E3	õ E4	Õ E5	µ E6	þ E7	Þ E8	Ú E9	Û EA	Ù EB	ý EC	Ý ED	¯ EE	´ EF
- F0	± F1	‗ F2	¾ F3	¶ F4	§ F5	÷ F6	¸ F7	° F8	¨ F9	· FA	¹ FB	³ FC	² FD	■ FE	FF

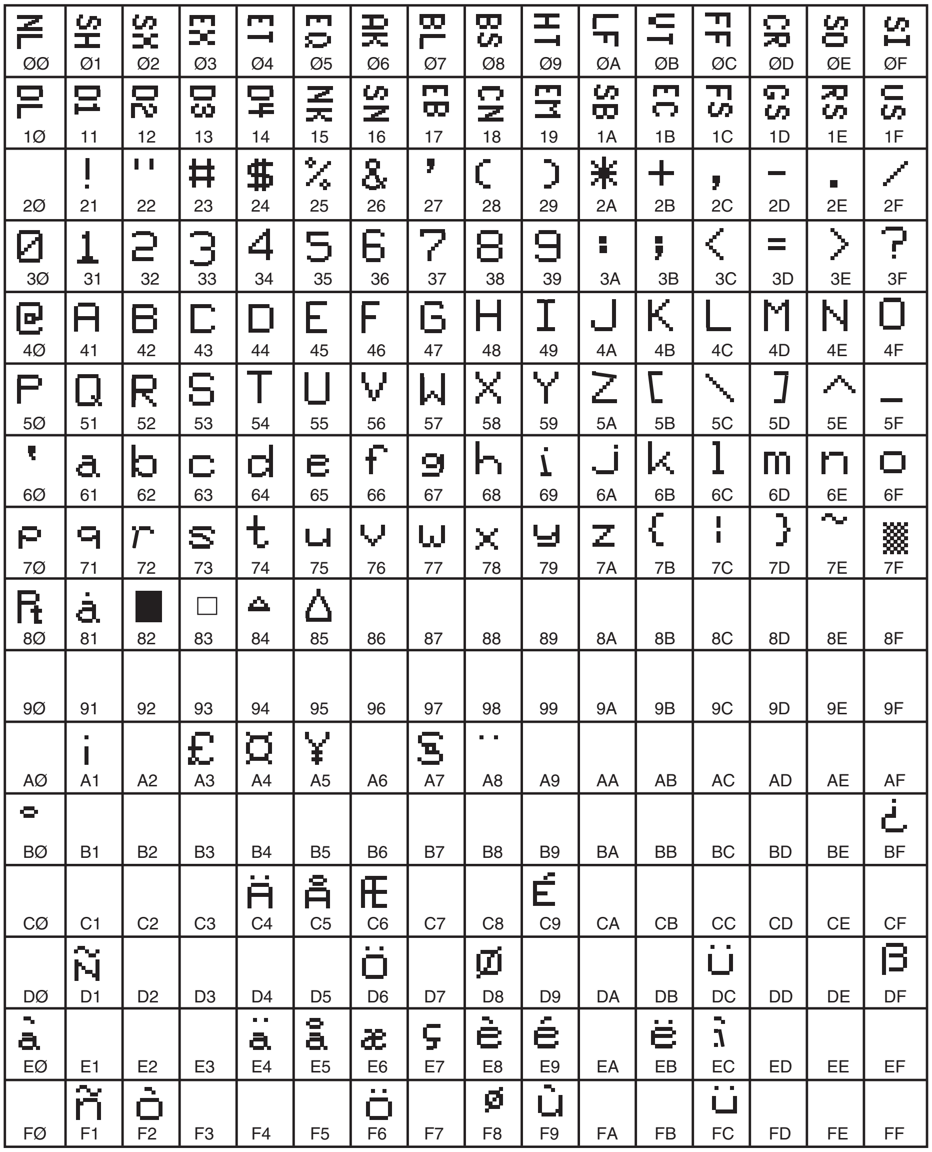
IPL019.eps

Extended Character Sets

Each internal printer font has a different character set as shown in the following tables. The hex codes accompany each character.

Note: You must set the serial port communication to 8 data bits to use the extended character sets.

Characters in Fonts c0, c1, c2, and c7



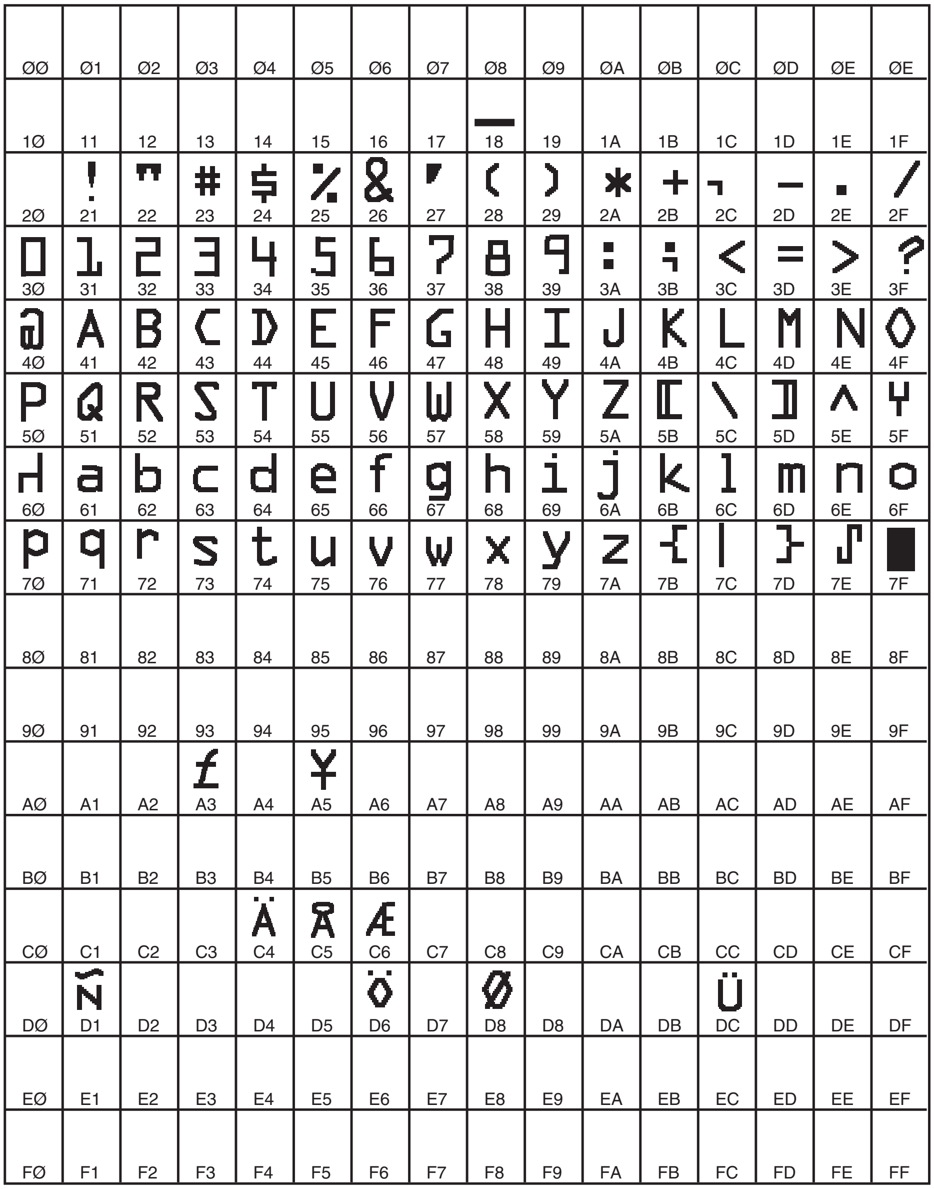
NL ØØ	SH Ø1	SX Ø2	EX Ø3	ET Ø4	EQ Ø5	AK Ø6	BL Ø7	BS Ø8	HT Ø9	LF ØA	VT ØB	FF ØC	CR ØD	SO ØE	SI ØF
DL 1Ø	D1 11	D2 12	D3 13	D4 14	NK 15	SN 16	EB 17	CN 18	EM 19	SB 1A	EC 1B	FS 1C	GS 1D	RS 1E	US 1F
2Ø	! 21	" 22	# 23	$ 24	% 25	& 26	' 27	(28	) 29	* 2A	+ 2B	, 2C	- 2D	. 2E	/ 2F
Ø 3Ø	1 31	2 32	3 33	4 34	5 35	6 36	7 37	8 38	9 39	: 3A	; 3B	< 3C	= 3D	> 3E	? 3F
@ 4Ø	A 41	B 42	C 43	D 44	E 45	F 46	G 47	H 48	I 49	J 4A	K 4B	L 4C	M 4D	N 4E	O 4F
P 5Ø	Q 51	R 52	S 53	T 54	U 55	V 56	W 57	X 58	Y 59	Z 5A	[5B	\ 5C	] 5D	^ 5E	_ 5F
` 6Ø	a 61	b 62	c 63	d 64	e 65	f 66	g 67	h 68	i 69	j 6A	k 6B	l 6C	m 6D	n 6E	o 6F
p 7Ø	q 71	r 72	s 73	t 74	u 75	v 76	w 77	x 78	y 79	z 7A	{ 7B	¦ 7C	} 7D	~ 7E	▒ 7F
₧ 8Ø	ȧ 81	█ 82	□ 83	▵ 84	△ 85	86	87	88	89	8A	8B	8C	8D	8E	8F
9Ø	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
AØ	¡ A1	A2	£ A3	¤ A4	¥ A5	A6	§ A7	¨ A8	A9	AA	AB	AC	AD	AE	AF
° BØ	B1	B2	B3	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	¿ BF
CØ	C1	C2	C3	Ä C4	Å C5	Æ C6	C7	C8	É C9	CA	CB	CC	CD	CE	CF
DØ	Ñ D1	D2	D3	D4	D5	Ö D6	D7	Ø D8	D9	DA	DB	Ü DC	DD	DE	ß DF
à EØ	E1	E2	E3	ä E4	å E5	æ E6	ç E7	è E8	é E9	EA	ë EB	ì EC	ED	EE	EF
FØ	ñ F1	ò F2	F3	F4	F5	ö F6	F7	ø F8	ù F9	FA	FB	ü FC	FD	FE	FF

IPL020.eps

Characters in Fonts c20, c21, and c22

ØØ	Ø1	Ø2	Ø3	Ø4	Ø5	Ø6	Ø7	Ø8	Ø9	ØA	ØB	ØC	ØD	ØE	ØE
1Ø	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
2Ø	! 21	" 22	# 23	$ 24	% 25	& 26	’ 27	(28	) 29	* 2A	+ 2B	, 2C	- 2D	. 2E	/ 2F
0 3Ø	1 31	2 32	3 33	4 34	5 35	6 36	7 37	8 38	9 39	: 3A	; 3B	< 3C	= 3D	> 3E	? 3F
@ 4Ø	A 41	B 42	C 43	D 44	E 45	F 46	G 47	H 48	I 49	J 4A	K 4B	L 4C	M 4D	N 4E	O 4F
P 5Ø	Q 51	R 52	S 53	T 54	U 55	V 56	W 57	X 58	Y 59	Z 5A	[5B	\ 5C	] 5D	^ 5E	_ 5F
` 6Ø	a 61	b 62	c 63	d 64	e 65	f 66	g 67	h 68	i 69	j 6A	k 6B	l 6C	m 6D	n 6E	o 6F
p 7Ø	q 71	r 72	s 73	t 74	u 75	v 76	w 77	x 78	y 79	z 7A	{ 7B	| 7C	} 7D	~ 7E	7F
€ 8Ø	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
9Ø	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
AØ	¡ A1	¢ A2	£ A3	¤ A4	¥ A5	¦ A6	§ A7	¨ A8	© A9	ª AA	« AB	¬ AC	- AD	® AE	¯ AF
° BØ	± B1	² B2	³ B3	´ B4	µ B5	¶ B6	• B7	¸ B8	¹ B9	º BA	» BB	¼ BC	½ BD	¾ BE	¿ BF
À CØ	Á C1	Â C2	Ã C3	Ä C4	Å C5	Æ C6	Ç C7	È C8	É C9	Ê CA	Ë CB	Ì CC	Í CD	Î CE	Ï CF
Đ DØ	Ñ D1	Ò D2	Ó D3	Ô D4	Õ D5	Ö D6	× D7	Ø D8	Ù D8	Ú DA	Û DB	Ü DC	Ý DD	Þ DE	ß DF
à EØ	á E1	â E2	ã E3	ä E4	å E5	æ E6	ç E7	è E8	é E9	ê EA	ë EB	ì EC	í ED	î EE	ï EF
õ FØ	ñ F1	ò F2	ó F3	ô F4	õ F5	ö F6	÷ F7	ø F8	ù F9	ú FA	û FB	ü FC	ý FD	þ FE	ÿ FF

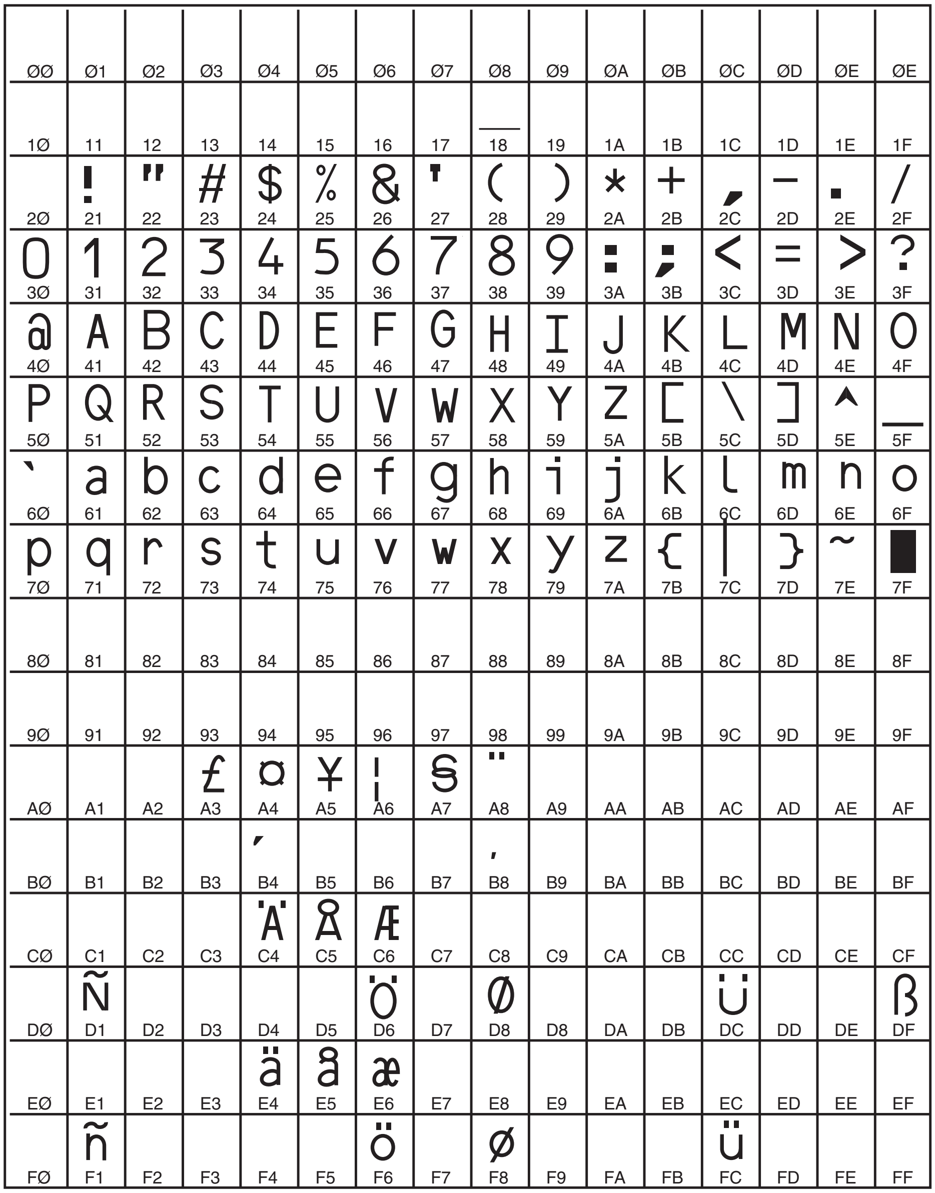
Characters in Font c23



ØØ	Ø1	Ø2	Ø3	Ø4	Ø5	Ø6	Ø7	Ø8	Ø9	ØA	ØB	ØC	ØD	ØE	ØE
1Ø	11	12	13	14	15	16	17	— 18	19	1A	1B	1C	1D	1E	1F
2Ø	! 21	" 22	# 23	$ 24	% 25	& 26	' 27	(28	) 29	* 2A	+ 2B	, 2C	- 2D	. 2E	/ 2F
0 3Ø	1 31	2 32	3 33	4 34	5 35	6 36	7 37	8 38	9 39	: 3A	; 3B	< 3C	= 3D	> 3E	? 3F
@ 4Ø	A 41	B 42	C 43	D 44	E 45	F 46	G 47	H 48	I 49	J 4A	K 4B	L 4C	M 4D	N 4E	O 4F
P 5Ø	Q 51	R 52	S 53	T 54	U 55	V 56	W 57	X 58	Y 59	Z 5A	[5B	\ 5C	] 5D	^ 5E	_ 5F
` 6Ø	a 61	b 62	c 63	d 64	e 65	f 66	g 67	h 68	i 69	j 6A	k 6B	l 6C	m 6D	n 6E	o 6F
p 7Ø	q 71	r 72	s 73	t 74	u 75	v 76	w 77	x 78	y 79	z 7A	{ 7B	| 7C	} 7D	~ 7E	█ 7F
8Ø	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
9Ø	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
AØ	A1	A2	£ A3	A4	¥ A5	A6	A7	A8	A9	AA	AB	AC	AD	AE	AF
BØ	B1	B2	B3	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	BF
CØ	C1	C2	C3	Ä C4	Å C5	Æ C6	C7	C8	C9	CA	CB	CC	CD	CE	CF
DØ	Ñ D1	D2	D3	D4	D5	Ö D6	D7	Ø D8	D8	DA	DB	Ü DC	DD	DE	DF
EØ	E1	E2	E3	E4	E5	E6	E7	E8	E9	EA	EB	EC	ED	EE	EF
FØ	F1	F2	F3	F4	F5	F6	F7	F8	F9	FA	FB	FC	FD	FE	FF

IPL024.eps

Characters in Font c24



ØØ	Ø1	Ø2	Ø3	Ø4	Ø5	Ø6	Ø7	Ø8	Ø9	ØA	ØB	ØC	ØD	ØE	ØE
1Ø	11	12	13	14	15	16	17	— 18	19	1A	1B	1C	1D	1E	1F
2Ø	! 21	" 22	# 23	$ 24	% 25	& 26	' 27	(28	) 29	* 2A	+ 2B	, 2C	- 2D	. 2E	/ 2F
0 3Ø	1 31	2 32	3 33	4 34	5 35	6 36	7 37	8 38	9 39	: 3A	; 3B	< 3C	= 3D	> 3E	? 3F
@ 4Ø	A 41	B 42	C 43	D 44	E 45	F 46	G 47	H 48	I 49	J 4A	K 4B	L 4C	M 4D	N 4E	O 4F
P 5Ø	Q 51	R 52	S 53	T 54	U 55	V 56	W 57	X 58	Y 59	Z 5A	[5B	\ 5C	] 5D	^ 5E	_ 5F
` 6Ø	a 61	b 62	c 63	d 64	e 65	f 66	g 67	h 68	i 69	j 6A	k 6B	l 6C	m 6D	n 6E	o 6F
p 7Ø	q 71	r 72	s 73	t 74	u 75	v 76	w 77	x 78	y 79	z 7A	{ 7B	| 7C	} 7D	~ 7E	█ 7F
8Ø	81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
9Ø	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
AØ	A1	A2	£ A3	¤ A4	¥ A5	¦ A6	§ A7	¨ A8	A9	AA	AB	AC	AD	AE	AF
BØ	B1	B2	B3	´ B4	B5	B6	B7	¸ B8	B9	BA	BB	BC	BD	BE	BF
CØ	C1	C2	C3	Ä C4	Å C5	Æ C6	C7	C8	C9	CA	CB	CC	CD	CE	CF
DØ	Ñ D1	D2	D3	D4	D5	Ö D6	D7	Ø D8	D8	DA	DB	Ü DC	DD	DE	ß DF
EØ	E1	E2	E3	ä E4	å E5	æ E6	E7	E8	E9	EA	EB	EC	ED	EE	EF
FØ	ñ F1	F2	F3	F4	F5	ö F6	F7	ø F8	F9	FA	FB	ü FC	FD	FE	FF

IPL025.eps

Characters in Fonts c25, c26 and c28

ØØ	Ø1	Ø2	Ø3	Ø4	Ø5	Ø6	Ø7	Ø8	Ø9	ØA	ØB	ØC	ØD	ØE	ØE
1Ø	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
2Ø	! 21	" 22	# 23	$ 24	% 25	& 26	’ 27	(28	) 29	* 2A	+ 2B	, 2C	- 2D	. 2E	/ 2F
0 3Ø	1 31	2 32	3 33	4 34	5 35	6 36	7 37	8 38	9 39	: 3A	; 3B	< 3C	= 3D	> 3E	? 3F
@ 4Ø	A 41	B 42	C 43	D 44	E 45	F 46	G 47	H 48	I 49	J 4A	K 4B	L 4C	M 4D	N 4E	O 4F
P 5Ø	Q 51	R 52	S 53	T 54	U 55	V 56	W 57	X 58	Y 59	Z 5A	[5B	\ 5C	] 5D	^ 5E	_ 5F
` 6Ø	a 61	b 62	c 63	d 64	e 65	f 66	g 67	h 68	i 69	j 6A	k 6B	l 6C	m 6D	n 6E	o 6F
p 7Ø	q 71	r 72	s 73	t 74	u 75	v 76	w 77	x 78	y 79	z 7A	{ 7B	| 7C	} 7D	~ 7E	7F
€ 8Ø	81	‚ 82	ƒ 83	„ 84	… 85	† 86	‡ 87	ˆ 88	‰ 89	Š 8A	‹ 8B	Œ 8C	8D	Ž 8E	8F
9Ø	‘ 91	’ 92	“ 93	” 94	• 95	– 96	— 97	˜ 98	™ 99	š 9A	› 9B	œ 9C	9D	ž 9E	Ÿ 9F
AØ	¡ A1	¢ A2	£ A3	¤ A4	¥ A5	¦ A6	§ A7	¨ A8	© A9	ª AA	« AB	¬ AC	- AD	® AE	¯ AF
° BØ	± B1	² B2	³ B3	´ B4	µ B5	¶ B6	· B7	¸ B8	¹ B9	º BA	» BB	¼ BC	½ BD	¾ BE	¿ BF
À CØ	Á C1	Â C2	Ã C3	Ä C4	Å C5	Æ C6	Ç C7	È C8	É C9	Ê CA	Ë CB	Ì CC	Í CD	Î CE	Ï CF
Ð DØ	Ñ D1	Ò D2	Ó D3	Ô D4	Õ D5	Ö D6	´ D7	Ø D8	Ù D8	Ú DA	Û DB	Ü DC	Ý DD	Þ DE	ß DF
à EØ	á E1	â E2	ã E3	ä E4	å E5	æ E6	ç E7	è E8	é E9	ê EA	ë EB	ì EC	í ED	î EE	ï EF
ð FØ	ñ F1	ò F2	ó F3	ô F4	õ F5	ö F6	÷ F7	ø F8	ù F9	ú FA	û FB	ü FC	ý FD	þ FE	ÿ FF

B

User-Defined Interface Tables

This chapter contains the user-defined interface tables, which you may need when programming with IPL. These tables show commands in the order that you must download them when you replace the User-Defined Command/Protocol characters. A table is shown for each type of command specified by a value for "t".

Print Commands (t = 0)

This list shows the Print Mode commands in the order you must download them when you are replacing the command codes.

Print Commands (t = 0)

Default Print Command	Hex Value	Print Command Description
NUL	00	Command Terminator 1
SOH	01	Set Preamble
EOT	04	Set Postamble
ENQ	05	Status Inquiry
ACK	06	Select First Data Entry Field
BEL	07	Transmit Error Code
BS	08	Warm Boot
LF	0A	Command Terminator 2
VT	0B	Status Dump
FF	0C	Form Feed
CR	0D	Select Next Data Entry Field
SO	0E	Label Cut Command
SI	0F	Go to Shift Command Table
DLE	10	Reset
SYN	16	Set Intercharacter Delay
ETB	17	Print
CAN	18	Clear All Data
EM	19	Abort Print Job
SUB	1A	Data Shift
ESC	1B	Go to Escape Command Table
FS	1C	Numeric Field Separator
GS	1D	Alphanumeric Field Separator
RS	1E	Set Quantity Count
US	1F	Set Batch Count
DEL	7F	Clear Data From Current Field

Escape Print Commands (t = 1)

This table lists the <ESC> commands in the order you must download them.

Escape Print Commands (t=1)

Default Escape Command	Hex Value	Escape Command Description
SYN	16	Set Message Delay
(space)	20	Enter Start/Stop Character
C	43	Select Advanced Mode
D	44	Set Field Decrement
E	45	Select Format
F	46	Select Field
G	47	Select Page
H	48	Transmit Printhead Parameters
I	49	Set Field Increment
L	4C	Transmit Label and Gap Length
M	4D	Transmit Software Version Number
N	4E	Disable Increment/Decrement
O	4F	Transmit Options Selected
P	50	Enter Program Mode
Q	51	Transmit Quantity and Batch Count
T	54	Enter Test and Service Mode
Z	5A	Transmit User-Defined Command Tables
c	63	Select 86XX Emulation Mode
d	64	Enable Auto-Transmit 2
e	65	Enable Auto-Transmit 3
g	67	Select Direct Graphics Mode
j	6A	Enable Auto-Transmit 1
k	6B	Disable Auto-Transmit 1, 2, and 3
m	6D	Transmit Static RAM Usage
p	70	Transmit Configuration Parameters
u	75	Transmit User-Defined Characters
v	76	Transmit Font
x	78	Transmit Format
y	79	Transmit Page

Shift Print Commands (t = 2)

This list shows the Shift commands in the order you must download them. You must precede these commands with the “Go to Shift Command Table” command (default value SI) listed in the Print Commands (*t* = 0) table on page 88.

Shift Print Commands (t = 2)

Default Shift Command	Hex Value	Shift Command Description
A	41	Control Panel Access
C	43	86XX or Advanced Mode on Power-Up
D	44	Set End-of-Print Skip Distance
F	46	Set Top of Form
H	48	Set Printhead Pressure
I	49	Set Number of Image Bands
L	4C	Set Maximum Label Length
N	4E	Define Amount of Storage
O	4F	Online or Offline on Power-Up
R	52	Enable or Disable Label Retract
S	53	Set Print Speed
T	54	Select Label Stock Type
U	55	Set Printhead Test Parameters
W	57	Set Label Width
Z	5A	Set Ribbon Save Zone
a	61	Audible Alarm
b	62	Increase Takeup Motor Torque
c	63	Enable or Disable Cutter
d	64	Set Dark Adjust
f	66	Adjust Label Rest Point
g	67	Select TTR or Direct Thermal
h	68	Select Printhead Loading Mode
i	69	IBM Language Translation
l	6C	Select Printer Language
p	70	Set Pin 11/20 Protocol
r	72	Set Label Retract Distance
t	74	Enable or Disable Self-Strip

Status Responses and Auto-Transmit Commands (t = 3)

This list contains the status responses and auto-transmit commands in the order you must download them.

Status Responses and Auto-Transmit Commands (t = 3)

Status Command	Hex Value	Status Description
GS	1D	Buffer Already Full
SO	0E	Printhead Test Fail
US	1F	Label Path Open
US	1F	Ribbon Fault
EM	19	No Label Stock
DC3	13	Buffer Now Full
EOT	04	RFID Tag Write Error
VT	01	RFID Tag Read Error
BS	08	Takeup Reel Full
SI*	0F	Printhead Hot
FS	1C	Label at Strip Pin
ACK	06	RFID Tag Write Ok
DC1**	11	Skipping
DC1**	11	Printing
DC1**	11	Ready
DC1 Auto-Transmit 1**	11	Clear
FS Auto-Transmit 1	1C	Label at Strip Pin
BS Auto-Transmit 1	08	Takeup Reel Full
EM Auto-Transmit 1	19	No Label Stock
US Auto-Transmit 1	1F	Ribbon Fault
DC1 Auto-Transmit 2	11	Room in Buffer
HT Auto-Transmit 3	09	Imager Overrun
SOH Auto-Transmit 3	01	Print Job Complete and Buffer Empty
RS Auto-Transmit 3	1E	Insufficient RAM

*Some older Intermec printers may not support this entry.

**The status responses in the above table are for standard protocol. In XON/XOFF protocol, most of the status responses are the same; however, instead of DC1, the status response is DC2, and instead of DC2, the status response is DC4.

See your printer user’s manual for more information about protocols and status responses.

Protocol Commands (t = 4)

This list contains the protocol codes in the order you must download them.

Protocol Commands (t = 4)

Command Characters	Hex Value	Command Description
GS	1D	SELECT IN
FS	1C	POLL IN
EOT	04	RES IN
ENQ	05	REQ IN
STX	02	SOM IN
ETX	03	EOM IN
ACK	06	AFF IN
NAK	15	NEG IN
DLE	10	DLE IN
DC1	11	XON IN
DC3	13	XOFF IN
GS	1D	SELECT OUT
FS	1C	POLL OUT
EOT	04	RES OUT
ENQ	05	REQ OUT
STX	02	SOM OUT
ETX	03	EOM OUT
ACK	06	AFF OUT
NAK	15	NEG OUT
DLE	10	DLE OUT
DC1	11	XON OUT
DC3	13	XOFF OUT
ENQ	05	Proto-Cmd 1
VT	0B	Proto-Cmd 2
20 (ms) (Range: 0 - 255)	14	Timeout on EOM ACK

Communications Protocol Characters

This table shows the characters available for different protocols. Refer to the protocol you are using for your system.

Communications Protocol Characters

Protocol Characters	Standard	XON/XOFF	Polling Mode D	Multi-Drop
Select In			GS	GS
Poll In			FS	FS
Reset In			EOT	EOT
Request for Acknowledgment In			ENQ	ENQ
Start of Message In	STX	STX	STX	STX
End of Message In	ETX	ETX	ETX	ETX
Acknowledgment In			ACK	ACK
Negative Acknowledgment In			NAK	NAK
Data Line Escape In	DLE	DLE	DLE	DLE
XON In		DC1		
XOFF In		DC3		
Select Out				GS
Poll Out				FS
Reset Out			EOT	EOT
Request for Acknowledgment Out			ENQ	ENQ
Start of Message Out			STX	STX
End of Message Out			ETX	ETX
Acknowledgment Out			ACK	ACK
Negative Acknowledgment Out	NAK		NAK	NAK
Data Line Escape Out	DLE	DLE	DLE	DLE
XON Out		DC1		
XOFF Out		DC3		
Status Enquiry In	ENQ	ENQ		
Status Dump In	VT	VT		
Timeout on EOM ACK			DC4	DC4

Index

Intermec

Worldwide Headquarters
6001 36th Avenue West
Everett, Washington 98203
U.S.A.

tel 425.348.2600

fax 425.355.9551

www.intermec.com

Intermec Printer Language (IPL) Developer’s Guide

P/N 934-013-003