Epic 3000

Programmer's Guide

PN 100-14362, Rev A Dec 2015

Change History

Revision	*Description of Change*	*Date*	*ECR*
1	Initial Draft	Dec 2015	
A	Initial Release	Dec 2015	ER30301

Federal Communications Commission Radio Frequency Interference Statement
The EPIC 3000 Printer complies with the limits for a Class A computing device in accordance with the specifications in Part 15 of FCC rules. These regulations are designed to minimize radio frequency interference during installation; however, there is no guarantee that radio or television interference will not occur during any particular installation. Interference can be determined by turning the equipment off and on while the radio or television is on. If the printer causes interference to radio or television reception, try to correct the interference by one or more of the following measures:

1. Reorient the radio or television receiving antenna
2. Relocate the printer with respect to the receiver
3. Plug the printer and receiver into different circuits

If necessary, the user should consult their dealer or an experienced radio/television technician for additional suggestions. The user may find the following booklet prepared by the Federal Communications Commission helpful: How to Identify and Resolve Radio/TV Interference Problems. This booklet is available from the US Government Printing Office, Washington, DC 20402. Ask for stock number 004-000-00345-4.

Canadian Department of Communications Radio Interference Statement

The *EPIC 3000* Printer does not exceed Class A limits for radio noise emissions from digital apparatus set out in the Radio Interference Regulations of the Canadian Department of Communications.

Regulatory Compliance

FCC Class A
ULc
CE Mark
UL 1950
TUV

Disclaimer

NOTICE TO ALL PERSONS RECEIVING THIS DOCUMENT:
The information in this document is subject to change without notice. No part of this document may be reproduced, stored or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of TransAct Technologies, Inc. ("TransAct"). This document is the property of and contains information that is both confidential and proprietary to TransAct. Recipient shall not disclose any portion of this document to any third party.

TRANSACT DOES NOT ASSUME ANY LIABILITY FOR DAMAGES INCURRED, DIRECTLY OR INDIRECTLY, FROM ANY ERRORS, OMISSIONS OR DISCREPANCIES IN THE INFORMATION CONTAINED IN THIS DOCUMENT.

TransAct cannot guarantee that changes in software and equipment made by other manufacturers, and referred to in this publication, do not affect the applicability of information in this publication.

Copyright

Revision Level A
Dec 2015
Printed in USA

Trademarks

Some of the product names mentioned herein are used for identification purposes only and may be trademarks and/or registered trademarks of their respective companies.

BANKjet, 50Plus, Insta-Load, Ithaca, "Made to Order. Built to Last", Magnetec, PcOS, POSjet, PowerPocket, iTherm", "PRINT IT. STICK IT.", POWEROLL, "RECEIPTS THAT REGISTER", and TransAct are registered trademarks and Epic 950, Epicenteral, Flex-Zone, imPort, ithaColor, KITCHENjet, Momentum, QDT and TicketBurst are trademarks of TransAct Technologies, Inc.

QR Code is registered trademark of DENSO WAVE INCORPORATED in JAPAN and other countries.
EPSON and ESC/POS are registered trademarks of Seiko Epson Corporation in Japan and other countries.

Table of Contents

Figures

Tables

About the EPIC 3000 Printer

The Ithaca EPIC 3000 printer represents the very latest technology for use for thermal receipt printing for point-of-sale and retail environments. It builds upon the architecture of Ithaca's proven thermal printers, together with a host of features specifically designed to improve the performance of your receipt-printing applications, including:

- Crisp, clear receipt printing in either one or two colors
- Fast 11 inches per second print speed
- Rugged spill-resistant cover
- Large 4-inch paper roll capacity with drop-in loading
- Protected internal power supply
- Application-controllable buzzer

The EPIC 3000 also offers a wide range of programmable features, including color and font control, APA graphics support, bar codes, and support for multiple language character sets. These features let you quickly and easily integrate more layout and printing options than ever – while giving you the reliability, durability and uptime you have come to expect from Ithaca printers.

Who Should Read This Guide?

This document provides information and programming specifications for programmers and/or operators who will integrate the EPIC 3000 printer into their operations.

What Is Included in This Guide?

This Programmer's Guide includes information on the features and programming interface of the EPIC 3000 printer. It provides the following information to support your programming and implementation efforts:

- Warranty and technical support information.
- Specifications and functionality description.
- Programming information, including documentation of low-level and high-level command interfaces, as well as sample scripts to guide your own implementation efforts.

We want you to have a trouble-free implementation with your Transact printer. For any issues not covered in this guide, quality technical support is available on-line at **www.transact-tech.com**, or by telephone at (607) 257-8901 or (877) 7ithaca. Consult the following pages for more details about our support services.

Warranty Options

All EPIC 3000 printers come with a standard 24-month standard warranty covering both parts and labor that starts upon shipment from the factory. An optional extended warranty, covering both parts and labor for an additional 12 months, may be purchased separately. For more information concerning the warranty options, please contact the Sales Department at TransAct's Ithaca facility. You are responsible for insuring any product returned for service, and you assume the risk of loss during shipment to Ithaca. C.O.D. packages are not accepted and warranty repairs are subject to the terms and conditions as stated on the Ithaca warranty policy.

Technical and Sales support

Your Ithaca printer is backed by the resources of TransAct Technologies, a global technology firm with dedicated technical support and sales assistance. Here is how we can help you:

On-line Technical Support

Our web site at **www.transact-tech.com** is your on-line portal to obtaining technical assistance with your Ithaca printer. Click on Ithaca link and then the Technical Support link to find documentation for your EPIC 3000 printer, including a current copy of this Programmer's Guide.

Our on-line support site also includes a convenient e-mail assistance request form, where you can submit support requests 24 hours a day, and receive a return contact from a TransAct support technician during regular business hours.

Telephone Technical Support

Live telephone support is available Monday through Friday from 8 AM to 5 PM Eastern US time, excluding holidays. We can provide general information about programming for your EPIC 3000 printer, technical support, documentation, or assistance in sending a printer for service. To obtain telephone support, call TransAct's Ithaca Facility at (607) 257-8901 and ask for Technical Support. To help us serve you faster, please have the following information ready when you call:

- The Model Number and Serial Number of the printer.
- A list of any other peripheral devices attached to the same port as the printer.
- What application software, operating system, and network (if any) you are using.
- What happened and what you were doing when the problem occurred.
- How you tried to solve the problem.

Return Materials Authorization and Return Policies

If the technical support person determines that the printer should be serviced at our facility, and you want to return the printer for repair, we will issue you the Returned Materials Authorization (RMA) number that is required before returning the printer. Repairs are warranted for 90 days from the date of repair or for the balance of the original warranty period, whichever is greater. Please prepare the printer being returned for repair as follows:

- Pack the printer to be returned in the original packing material.
- Packing material may be purchased from TransAct's Ithaca Facility.
- Do not return any accessories unless asked to do so by a support technician.
- Write the RMA number clearly on the outside of the box.

Service Programs

TransAct Technologies Incorporated has a full service organization to meet your printer service and repair requirements. If your printer needs service, please contact your service provider first. If any problems still persist, you can directly contact the Ithaca facility's Technical Support Department at (607) 257-8901 or (877) 7ithaca for a return authorization. International customers should contact your distributor for services. TransAct offers the following service programs to meet your needs.

- Extended Warranty.
- Depot Repair.
- Maintenance Contract.
- Internet Support.

Sales Support

To order supplies, receive information about other Ithaca products, or obtain information about your warranty, contact our Sales Department at the contact telephone or fax numbers listed below. To receive information on International distribution, visit our web site at **www.transact-tech.com**.

Contact Information

TransAct Technologies Incorporated
Ithaca Facility
20 Bomax Drive
Ithaca, NY 14850 USA

Telephone	(877) 7ithaca or (607) 257-8901
Main fax	(607) 257-8922
Sales fax	(607) 257-3868
Technical Support fax	(607) 257-3911
Web site	**www.transact-tech.com**

EPIC 3000 Specifications and Requirements

Standard Features

The following features are common to the entire family of thermal printers:

- Print Speed for text is 11 inches per second (279 mm/sec)[1]
- 12.0 inches per second paper feed speed
- Built-in self-ranging External Power supply
- Clam-shell paper loading
- USB interface.
- Configurable receiver buffer
- Self diagnostics
- Set up and configuration utility program
- CPI selections from 8 to 30 CPI[2]
- Paper Out sensor
- Multiple printer emulations: Ithaca PcOS, Star, Citizen, and Epson
- APA and Epson graphics
- Over 25 Bar Codes[3] including 2D and Composite
- Resident Bitmap and True Type Fonts.
- UTF or ASCII with code page Character addressing
- WGL4.0 Character set.
- Metal receipt tear off
- 8 dots/mm. thermal print head resolution
- Diagnostics button located under the printer
- Cable routing strain relief
- Multi-Color Status LED
- Cover open button
- Spill resistant design vertical main PCB mounting
- 82 mm. paper width
- 7.8 inch (190 mm.) Paper roll diameter
- Portrait/landscape printing under Windows
- Page mode printing
- Cover Open sensor

[1] Monochrome printing. In some cases depending on the print density the print speed may be slower or faster. Print speed will be slower when using adhesive backed or color paper.

[2] Character spacing is adjustable from 1 to 30 CPI. Typical values will be between 8 and 20 CPI depending on the font selected. Values of 13.3, 14.86, or 17.3 are typical for each resident font.

[3] Barcodes include: EAN 8, EAN 13, EAN 14, GS1-128 (EAN128), Codabar, Code 2 of 5, Code-39, Code-39 Extended, Code-93, Code-128(A,B, and C), UPC A, UPC E, Code49, Code16K, PDF417, MicroPDF417, Maxicode, QRCode, Datamatrix, GS1-Databar-14, GS1-Databar-Truncated, GS1-Databar-Limited, GS1-Databar-Expanded, GS1-Databar-14 Stacked, GS1-Databar-14 Stacked-Omni, GS1-Databar-Expanded Stacked, Aztec, EANX Composite, EAN128 Composite, GS1-Databar-14 Composite, GS1-Databar-Truncated Composite, GS1-Databar-Limited Composite, GS1-Databar-Expanded Composite, UPC A Composite, UP CE Composite, GS1-Databar-14 Stacked Composite, GS1-Databar-14 Omni Composite, GS1-Databar-Expanded Stacked Composite, and EAN 2 and EAN 5 Add on barcodes.

- Internal counters for hours on, cuts, print lines and errors
- 100 km print head life
- 60 million print line printer MCBF (excluding knife)
- Buzzer

General Specifications

Printing Specifications

Printing method:	Thermal Sensitive Line Dot System
Vertical/Horizontal dot pitch:	0.125 mm.
Resolution:	8 dots per mm (203 DPI)
Line feed pitch:	3.2 mm. (.125 inches)
Print zone (maximum)	80 mm (3.15 inch)
Print Speed (monochrome):	11 inches per second
Print Speed (two color):	4-6 inches per second[4]
Number of print elements:	640 dots in-line

Note: This document is not the controlling document for print specifications, for print location, tolerances, or power requirements. The information specified here is to aid in program development.

Electrical Characteristics

Internal AC Powered

The EPIC 3000 Printer is designed to be AC self-powered in domestic and international markets. The printer is equipped with a universal input power supply that is designed to operate worldwide without modification.

Supply Voltage Rating (VAC)	Supply Voltage Range (VAC)	Frequency (Hz)	Rated Power (watts)	Idle Current (amps)	Printing Current (amps)
100-240	90-264	47 – 63	45	.1A @ 120VAC .05 A @ 240VAC	1.4 A @ 120VAC .7 A @ 240VAC

Table 1 Standard Power Input Requirements

Thermal Print Head

Thermal Print Head Overview:

Number of heat elements:	640
Heat element pitch:	0.125 mm (8 dots/mm.)
Print width (Max):	80 mm. +/- 0.2 mm. (640 dots)
Print width (80mm paper):	72 mm. +/- 0.2 mm. (576 dots)
Print width (58mm paper):	56 mm. +/- 0.2 mm. (448 dots)
Print width (40mm paper):	36 mm. +/- 0.2 mm. (288 dots)
Pulse Life:	100 million pulses
Abrasion Life:	100 km.
Vertical dot pitch	0.125 mm (0.0049 inch) or 203 DPI
Operating Temperature	5-45 degrees C
Humidity:	10-90 % RH (non-condensing)

Operation Precautions:

[4] Color paper print speed is paper dependent.

- Do not print without paper.
- Clean the head with ethyl-alcohol after power is removed from the printer. This will remove foreign particles or paper dust which may degrade print quality.

Media Specifications

Receipt Paper

Paper feed method	Friction feed
Paper feed pitch	Default - 1/8 inch

Monochrome

roll diameter	190 mm. (7.5 inches) Max.
paper thickness	0.05 to 0.09 mm. (.002 to .0035 inches)
Paper Width	81.5 +/- .5 mm (3.19 +/- .02 inches) wide

Paper Usage Precautions:

- The life of the thermal head, when two-color paper is used, is reduced to about half of the life when single-color thermal paper is used.
- Use only specified thermal paper. If other paper is used, print quality, head life, and cutter life may deteriorate.

Paper Low

Paper low is not supported by this printer.

Paper Out

A paper out sensor is provided as a standard feature. It senses when there is approximately .5 inches length of paper left on the paper roll.

Receipt Printing, Auto Cutter Position

A receipt auto-cutter is a standard feature with all EPIC 3000 Printers.

Cutter type	Rotary
Media width	3.19 +/- .02 inches (81.5 +/- .5 mm)
Media thickness range	0.0025 to 0.0035 inch
Cut to line of print	0.70 inch
Cutter life	1,000,000 cuts
Cut time:	Less than 350 milliseconds

Buzzer

A buzzer is provided as a standard feature. It is triggered upon command from the host terminal to make a sound loud enough to be heard under noisy conditions. It will produce a sound pressure level of at least 90 dBA, 1 foot from the front of the printer.

Cover Interlock

A paper cover interlock switch is provided as a standard feature. When the paper cover is open, the printer is off-line, and will not print.

Communications Interface

All EPIC 3000's are equipped with a USB 2.0 interface.

USB Interface

The USB interface is a Version 2.0 High or Full Speed implementation. The USB interface is standard on all printers and implemented through a Standard Series "B" Receptacle as defined in the USB Specification. The printer is self-powered and does not draw power from the standard type B USB interface cable.

The Standard USB Type B connector has the following pin functions:

Pin	**Signal**
1	Vbus (+5 V dc)
2	Minus data
3	Plus data
4	Ground

Note: The +5 power on the standard USB interface does not have enough power to run the printer.

Setup

Verifying the Configuration

Before you install an EPIC 3000 Printer into your system, you should verify that the printer is configured as required by your system. There are four parts to this verification process.

1. Verify that the power connection is correct.
2. Verify that the firmware in the printer is configured correctly.

Connecting Communications Cables

The EPIC 3000is USB only.

Verify the Firmware Configuration

An example receipt is Included in the box your printer shipped in that will show how the printer was configured before it shipped from our Ithaca facility. Compare this information to your system requirements. Pay attention to the emulation and the communications link. If they are wrong, the printer may appear inoperative. If the configuration is not correct, refer to the section on changing the EPIC 3000 configuration. If there are a number of printers to be installed and you want the identical configuration in each, you can use the universal configuration program to record the configuration on one printer, and replicate it over a group. The configuration program is available from Transact technical support.

Installing Paper

It is easy to install paper in the EPIC 3000.

1. Open the paper cover by grasping the cover release between your thumb and forefinger, and squeeze the release. This will pop the cover up.
2. Open the cover and install a roll of paper with the paper coming off of the bottom of the roll.
3. Lay the paper tail over the front of the printer and center it over the paper path.
4. Close the cover. When the cover is closed, the printer will feed several inches of paper to make sure the paper is aligned in the printer. If equipped with a cutter, the printer will automatically cut the paper tail and the printer is now ready to print. If the printer is not equipped with a cutter, the operator should remove the paper tail.

Indicator Light

Error Indication and Blink Patterns
The EPIC 3000 printer will blink the LED indicator to indicate various modes or faults.

The printer may be in normal operation, self test or in recovery modes. The multicolor Indicator LED will indicate which mode by a unique color and or blink pattern.

- Maintenance Mode
  In Transact boot load mode, the power LED will blink two or more times and then delay. It will repeat this cycle about every 2 seconds.
  If in an error condition, the indicator will be read and blink green an error code.

- Self Test Mode
  When the printer is in self test, the power indicator will blink slowly with a 50% duty cycle at a 2 second rate.

- Normal Operation
  During Normal operation, the Power LED will remain on unless an error is being indicated. There are two classes of errors, soft errors and hard errors. Soft errors are recoverable without power cycling the printer. Hard errors cannot be recovered from without removing the power from the printer, correcting the problem, and reapplying power.

  - Three colored Error Indicators
    - Power -> Green
    - Error -> Red
    - Paper -> Yellow

  - Soft Errors
    Soft errors may be recovered by the host, or by opening and closing the printers cover. All of these errors are indicated by a 5 second repeating blink patterns.

Error State	Indicators		
	Power	Error	Paper
Powering Down	Fast Blink	On	Off
Cover Open	ON	On	Off
Print Head Over Temp	4 Blinks.	Slow Blink	Off
Power Bad	2 Blinks.	Slow Blink	Off
Out of Paper	ON	On	On
Jam	ON	On	Fast blink

  - Hard Errors
    Hard errors have a similar pattern to Soft errors only they are slower and repeat every 10 seconds. In general they occur during level 0 diagnostics and are not recoverable.

No fonts	1 Blink	On	Off
EEPROM read fault	2 Blinks	On	Off
EEPROM write fault	3 Blinks	On	Off

Error Vector Taken	4 Blinks	On	Off
Knife Error	5 Blinks	On	Off
Flash File system Error	6 Blinks	On	Off
User Store Fault	7 Blinks	On	Off
Configuration Fault	8 Blinks	On	Off
Flash Read/Write Error	9 Blinks	On	Off
Dynamic Memory Allocation	10 Blinks	On	Off
Font system Fault	11 Blinks	On	Off
Static Memory Allocation	12 Blinks	On	Off
Communications Fault	14 Blinks	On	Off
Kernel Fault	15 Blinks	On	Off
Head Connection Fault	16 Blinks	On	Off
USB Fault	17 Blinks	On	Off

Power Saving Modes

Sleep

In Sleep mode, the EPIC 3000 printer enters a low power state where everything but the communications is disabled. In this mode, the printer may be reactivated by command or by pressing the Power Button. As the print head preheat is turned off, it may take a few seconds for the EPIC 3000 to warm up the print head in preparation for printing. This mode is activated by command only. (See page 238)

Green/Standby

In Green mode, the EPIC 3000 printer enters a lower power state where everything including the communications is disabled. The printer will enter and leave Green mode based on the Vbus signal on the USB link. This allows the printer to enter a low power state whenever the USB link is placed in a low power state.

The operational state of the EPIC 3000 can be determined by looking at the Power Indicator Light (LED). When the printer enters ON mode, the green power indicator light will be activated. When in Sleep mode, the Power Indicator Light (LED) will blink about every 3 seconds. In Green mode, the Power Indicator Light (LED) will blink about every 5 seconds.

Testing the Printer Overview

Using Self-Test, Configuration, and Hex Dump Mode

Self-Test Mode allows you to perform a series of tests to show if the printer is functioning correctly. Self-Test Mode also allows you to print a summary of how your EPIC 3000 is currently configured. Use this printout to compare your printer's settings to your system's requirements. Specific attention should be given to emulation and communications settings. For serial printers, the baud rate and other RS-232 interface settings are important. If there is a configuration problem, you should use Configuration Mode to make any changes necessary.

Entering Self-Test, and Configuration mode

To enter self test and or configuration mode, perform the following sequence of operations:

1. Press and release the Diagnostics button to enter self test
2. Press and release the Diagnostics button to run the indicated test.
3. Press and hold the Diagnostics button until the green paper LED is illuminated to select the next test.
4. Press and hold the Diagnostics button until the red error LED is l illuminated to exit self test mode.

The EPIC 3000 has a total of seven Self-Test and or configuration options. Two are designed to be useful when performing on-site print evaluations. One option allows the printers electronic journal to be maintained, one allows the printers configuration to be changed, and two are designed for factory setup by TransAct.

Testing the printer

Use the following two TEST options when verifying basic printer operation.

TEST-Receipt

The receipt test is the primary test option to use when determining if the printer is functioning correctly.

The receipt test is mostly used during the early stages of troubleshooting, to eliminate the possibility that the problem is occurring with the printer. If the printer experiences a failure, and the error indicator light is activated, call TransAct's Ithaca Facility's Technical Support Department.

TEST-Head Test

This test performs a test pattern that will print all the head print elements and verify that the drive roll is free from defects. The print head has two heating elements per dot position. A print element is not considered bad unless both elements are missing. If the head test shows that there is an inconsistency in the drive roll it may be debris or a void. If debris is indicated, cleaning the drive roll should correct the problem. If this does not correct the problem, contact TransAct's Ithaca Facility's Technical Support Department.

Configuring Options

The configuration option allows the configuration of the printer to be printed and, if necessary, changed. When configuration mode is entered, the current configuration, the

Current User store status, and the current totals are printed. If any printer errors have occurred, a hardware and software error log may also be printed. At the end of the print out are instructions on how to use configuration mode. Please read these instructions carefully, as they are not the same as Self-Test. You may exit configuration mode at any time, without affecting the printer configuration, by pressing the power button.

TEST- Configuration
The content of the configurable features will alter based on the hardware installed. There will be at least three groups of options:

1. Emulation
2. Communications
3. General Options

Under emulation the instruction set or emulation of the printer may be changed. Available options will depend on the model of printer, however Ithaca PcOS is always available. Other options may be Epson TM88, TM90, Microline, and Ithaca M50.

Under Communications, the way the printer deals with the communications port can be adjusted. The printer will only show communications options that deal with the communications adapter installed.

Under General Options, all other configurable features of the printer can be adjusted. The default language, paper options, electronic Journal features, and print defaults may be set.

Details of all printer options and features will de discussed later in this manual.

Factory Test
The printer is equipped with several factory test modes. These test options are only used for factory burn-in and testing.

TEST-Burn-in
TEST-Rolling ASCII

Level 0 Diagnostics

Level 0 diagnostics always and only run at power up, e.g. power being applied. These diagnostics perform the following tasks:

Cold Power On

1. Basic System Integrity
2. Vector Integrity
3. RAM Test
4. Flash Boot Loader Integrity
5. Flash Firmware Integrity (NOTE: If the firmware is corrupted, the printer will remain in boot load.)
6. Start Normal Firmware
7. Verify Configuration Integrity
8. Interface Card Configuration
9. User-store Integrity
10. Start Kernel, Verify Multitasking, Start Tasks

Once the kernel is running, the following tests must pass to allow operation. However, if any test fails (except the knife home test), the remaining tests will generate recoverable faults and normal operation will start as soon as the fault is cleared. These tests are also run when operation is resumed from OFF.

11. Cover Closed Check
12. Knife Home (if installed)
13. Paper Present
14. Place Printer On-line, Start Normal Operation

The first phase of testing consists of step 1-5, and determines that the boot loader is accurate and the printer firmware is correct. Tests 1 through 4 produce non-recoverable errors if they fail, in which case the power must be removed from the printer and the printer returned for service. If the boot loader is intact, but the main firmware is corrupted, the printer automatically enters boot loader mode, where the firmware can then be reloaded into the printer.

Boot Loader Maintenance Mode

Maintenance mode supports firmware updates and other maintenance and configuration operations.

Maintenance mode may be entered by a series of host commands or it may be entered manually.

To manually enter the maintenance, hold the paper sensor down while the paper cover is open and turn the power switch on. Level 0 diagnostics will be run and the power indicator will display a very slow double blink pattern. If the error Indicator comes on, some type of level 0 error was detected. In maintenance mode, the firmware may be loaded or reconfigured and fonts added or changed. For a complete list of operations available refer to the maintenance mode programmers guide.

The printer does not need to be functional for maintenance mode to function.

Configuring Your EPIC 3000 Printer

Configuration Mode Overview

There are two ways to configure the EPIC 3000 printer: the first is to use the manual configuration sequence by using the keypad controls, and the second is to use TransAct's remote configuration software. TransAct Technologies offers the use of a remote CONFIG program as a fast, easy way for system integrators to configure or reconfigure your EPIC 3000 printer. To obtain more information, or the latest version of the CONFIG program, call our Sales Department or Technical Support.

How to Change Configuration Settings

Entering into Configuration Mode

1) Press and release the Diagnostics button.

2) Press and hold the Diagnostics button until the Paper (Green) LED lights. Then release the Diagnostics button. Repeat this process until the "Operation – Configuration" Prompt.
3) Press and Release the Diagnostics button to enter Configuration mode.
4) You are now in Configuration Mode.

After you enter Configuration Mode, the printer will print the current configuration, the current totals and the error logs, if any. Save this printout as a guide to changing the configuration, and in case you wish to return the printer to the previous configuration.

Each emulation can have different configurable features. If you are changing the emulation, note that the printout that was printed at the beginning of the configuration process may be incorrect for the new emulation, and the configurable features may be different. If you are using this print out as a configuration guide, and you are changing the emulation, you may wish to save the new emulation and then re-enter Configuration Mode to change other options. This will print all the available features for the new emulation.

Using Configuration Mode

The Feed button is used to select and change configuration setting. By pressing and releasing the Feed button, the parameter to be changed can be selected. By pressing and holding the Feed button, the value of the selected parameter will change.

Note: There are a number of features that may be configurable. It is intended that if you need to change them, you will contact TransAct Technical support for assistance.

Remote Configuration

Remote configuration is provided for all printers, and is accessed through a series of extended diagnostic and configuration commands. The TransAct universal configuration program will allow the configuration to be read, edited, and written back to the printer. It will also allow the configuration of one printer to be recorded and replicated over a number of printers. The program is available from TransAct Technical Support or by downloading it from the Internet – consult the section On-line Technical Support for further details.

Programming Codes

Control Codes Overview

This Programmer's Guide is designed to help users of the EPIC 3000 printer develop applications. EPIC 3000 Printers are specialized point-of-sale printers that have several features not normally found on general-purpose printers. Because of these special features, EPIC 3000 Printers have unique control codes. This programmer's guide documents the control codes that are unique to the EPIC 3000 Printer.

Nomenclature

When describing control codes, confusion often occurs as to whether the description is decimal, hexadecimal, or ASCII. To minimize the problem, this programmer’s guide uses the following nomenclature when describing control code sequences.

[ ] Encloses a control character. [ ] represents a single, 8-bit value as defined in the standard ASCII tables. The ASCII Code Table in Appendix B lists the control codes. An example would be [ESC], which would represent a 1BH or 27 decimal.

< > Encloses an 8-bit value in decimal format. The value is from zero to 255. An example is <2>, which represents 02H or 2 decimal.

<n> Indicates a variable parameter. The variable parameter, <n>, can have a value from zero to 255. The meaning of <n> is described and defined in the description of the command.

<n$_1$> <n$_2$> Indicates that there are two parameters, <n$_1$> and <n$_2$>, where both can have values from zero to 255.

<m$_1$> <m$_2$> Is an Ithaca® Printer Control Language (IPCL) parameter consisting of two digits where <m$_1$> and <m$_2$> are ASCII characters from zero to nine. The parameter is combined to form a value from zero to 99. If <m$_3$> is included, the parameter is combined to be from zero to 999. If two values are specified, there must be two bytes added to the IPCL code. That is, if the command specifies <m$_1$> <m$_2$> and the desired value is five, it must be specified as 05.

x All other characters in control strings represent ASCII characters. For example, [ESC] 1 represents 1BH followed by 31H.

In many cases, applications require that control sequences be specified in hexadecimal or decimal codes. In most cases, commands are specified in ASCII, hexadecimal, and decimal. The ASCII Code Table in Appendix B (page 252) lists ASCII, hexadecimal, and decimal equivalents.

Standard Emulation

The standard control codes for the EPIC 3000 Printer are extensions and subsets of the PcOS IBM emulation provided on other Ithaca® products. In some cases, an application designed for a Series 50 Printer with IBM code sets will function with a EPIC 3000 Printer.

IPCL Codes

Ithaca® Printer Control Language (IPCL) codes are part of PcOS and designed to control a printer without using control characters (i.e. characters less than 20H). Only the standard PcOS emulation supports IPCL.

In rare cases, an IPCL code will interfere with the text that is to be printed. The IPCL translator can be disabled with an [ESC] y <4> command.

EPOS Emulation

ESC/POS[5] is referred to here as EPOS. The EPIC 3000 Printer supports two Epson emulations. One emulation is for the TM-T88, and the other is an enhanced EPOS emulation that may help the EPIC 3000 replace other printers.

Specific EPSON compatibility features, such as its Automated Status Back (ASB) feature, are available as an option from TransAct Technologies. Command codes pertaining to these features are patented by Epson and require added cost licensed firmware.

It is intended that the standard Ithaca® PcOS emulation be used for new applications. Not all features of EPIC 3000 Printers are supported by EPOS – specifically, the ability to print color horizontal graphics.

Application Development

To aid application development, several chapters in this manual are designed to help the programmer understand the EPIC 3000 Printer. The next chapter provides a detailed description of each of the commands. Subsequent chapters provide explanations of how the printer works, including a description of the internal print buffer, communications link, and interaction between the host computer and printer.

Ithaca Control Codes and Commands

Throughout this Programmer's Guide, charts and tables list commands and features. In most cases, the charts cross-reference the page that describes the command. Code summary charts, arranged by code and function, are provided to help quickly find commands.

PcOS Printer Control Codes

The following section defines the EPIC 3000 Ithaca® PcOS emulation. The native, Ithaca® PcOS emulation provides the most flexibility and control over the printer. It is consistent with most previous Ithaca® PcOS products, and should be used when the printer is placed in a new application.

One optional feature in the EPIC 3000 Printer is the ability to print color graphics. Due to the complexity of color graphics, TransAct provides several drivers to integrate into your

[5] ESC/POS is a registered trademark of the Seiko Epson Corporation.

application. TransAct does not recommend that you generate drivers. In addition, TransAct has created several tools that can be used to generate and maintain graphic images and files for print on the EPIC 3000. Information about drivers and tools are available on the TransAct web site. For additional information, contact Technical Support.

Quick PcOS Reference Chart

Description	ASCII	Hex	IPCL equivalent code	Page
Low Level paper Motion				
Line feed.	[LF]	0AH	&%LF	37
Carriage return.	[CR]	0DH	&%CR	37
Horizontal Motion				
Back space.	[BS]	08H	&%BS	38
Horizontal tab.	[HT]	09H	&%HT	37
Set horizontal tab stops.	[ESC] D <n1> <n2> ... <ni> <0>	1BH,44H	none	38
Set horizontal position.	[ESC] n <n1> <n2>	1BH,6EH	&%HP<m1> <m2> <m3>	39

Set justification.
n=0 Left
n=1 Center
n=2 Right
n=8 Left (No line feed)
n=9 Center (No line feed)
n=10 Right (No line feed)
	[ESC] a <n>	1BH,61H	&%JR {n=2}
			&%JC {n=1}
			&%JL {n=0}
				39

Reset horizontal and vertical tab stops.	[ESC] R	1BH,52H	&%HV	38
Set left/right print margins. n_1 = Left margin, n_2 = Right margin	[ESC] X <n_1><n_2>	1BH,58H	none	163
Select Minimum character Height and Width in points	[ESC] + P<w><h>	1BH 50H	none	213
Select Minimum character Height and Width in ¼ points	[ESC] + p<w><h>	1BH 70H	none	214
Vertical Motion				
Perform a fine line feed.	[ESC] J <n>	1BH,4AH	&%FM <m_1> <m_2> <m_3>	39
Set 1/8-inch line spacing.	[ESC] 0	1BH,30H	&%ST	40
Set 7/72-inch line spacing.	[ESC] 1	1BH,31H	&%SG	40
Begin variable line spacing. (Enable [ESC] A <n>).	[ESC] 2	1BH,32H	none	41
Set variable line spacing to n/216 inch.	[ESC] 3 <n>	1BH,33H	&%SV <m_1> <m_2> <m_3>	40
Set variable line spacing to n/72 inch.	[ESC] A <n>	1BH,41H	none	41
Feed <n> lines at current spacing.	[ESC] d <n>	1BH,64H	&%FL <m_1> <m_2>	41
Set vertical tab stops.	[ESC] B <n_1> <n_2> <n_3>...<n_i> 0	1BH,42H	none	42
Vertical tab.	[VT]	0BH	&%VT	42
Form feed.	[FF]	0CH	&%FF	43
Feed to black dot	[ESC][VT]	1BH,0BH	None	45
Set top of form.	[ESC] 4	1BH,34H	&%TF	43
Set form length in lines.	[ESC] C <n>	1BH,43H	&%SL <m_1> <m_2>	43
Select Minimum character Height and Width in points	[ESC] + P<w><h>	1BH 50H	none	213

Description	**ASCII**	**Hex**	**IPCL equivalent code**	**Page**
Select Minimum character Height and Width in ¼ points	[ESC] + p<w><h>	1BH 70H	none	214

Begin auto line feed. (n=0, end n=1)	[ESC] 5 <n>	1BH,35H	&%CA {n=0}
			&%MA {n=1}
				44

Set form length in inches.	[ESC] C [NUL] <n>	1BH,43H	&%SI <m_1> <m_2>	44
Reverse line feed.	[ESC]]	1BH,5DH	&%LR	44
Character Pitch				
Set character spacing in points	[ESC]+I<n>	1BH 2BH 49H	None	211
Set character spacing in ¼ points	[ESC]+i<n>	1BH 2BH 69H	None	212
Set character spacing in points with adjustment	[ESC]+J<n>	1BH 2BH 4AH	None	212
Set character spacing in ¼ points with adjustment	[ESC]+j<n>	1BH 2BH 6AH	None	213
Begin 10 CPI character pitch.	[DC2]	12H	&%F3	46
Begin 12 CPI character pitch.	[ESC] :	1BH,3AH	&%F2	46
Begin 17 CPI character pitch.	[SI]	0FH	&%F1	46
Set character pitch.	[ESC] [P <n>	1BH,5BH, 50H	&%F<n>	47
Set inter-character spacing.	[ESC] V <n>	1BH,56H	none	48
Begin 24 CPI character pitch	[ESC][SI]	1BH,0FH	&%F4	46
Select Minimum character Height and Width in ¼ points	[ESC] + p<w><h>	1BH 70H	none	214
Set left/right print margins. n_1 = Left margin, n_2 = Right margin	[ESC] X <n_1><n_2>	1BH,58H	none	163
Character Font				
Select font	[ESC] + 3 <ID>	1BH 2BH 31H	none	207
Select font by name	[ESC] + N Filename<0>	1BH 2BH 4EH	none	207
Define Stacked Font	[ESC] + S <ID_1> <ID_2> …	1BH 2BH 53H	none	207
Set stroke font Brush Size	[ESC] + B <w>	1BH 2BH 42H	none	214
Select Minimum character Height and Width in points	[ESC] + P<w><h>	1BH 50H	none	213
Select Minimum character Height and Width in ¼ points	[ESC] + p<w><h>	1BH 70H	none	214
Begin 12 x 12 draft print.	[ESC] # <0>	1BH,23H, 00H	&%QT	49

Set print quality mode.
n=0 Draft (12 x 12)
n=1 Large draft (12 x 14)
n=2 NLQ (24 x 16)
n=3 NLQ (24 x 16)
n=4-7 Repeats 0-3
	[ESC] I <n>	1BH,49H	&%QT {n=0}
			&%QU {n=1}
			&%QL {n=2}
			&%QS {n=3}
				49

Begin rotated font. (See command description).	[ESC] P <n>	1BH,50H	&%RI &%RF &%RN	50
Select character code page.	[ESC] [T <n_h> <n_l>	1BH,5BH, 54H	&%CP <m_1> <m_2> <m_3> <m_4>	54
Insert Euro character.	[ESC] [C <n>	1BH,5BH, 43H	&%EU	54
Print control character.	[ESC] ^ <n>	1BH,5EH	&%CC <m_1> <m_2> <m_3>	57
Redefine character set.	[ESC] [S …	1BH,5BH, 53H…		56

Description	ASCII	Hex	IPCL equivalent code	Page
Define user-defined characters.	[ESC] = <y> <c_1> <c_2>…	1BH,3DH	none	58
Enable user-defined characters.	[ESC] > <n>	1BH,3EH	none	58
Cancel user-defined characters.	[ESC] $	1BH,24H	none	58
Print control character.	[ESC] ^ <n>	1BH,5EH	&%CC <m_1> <m_2> <m_3>	57
Print Unicode character	[ESC] “ <n_L><n_H>	1BH,22H	&%PU <m_1> <m_2> <m_3> <m_4> <m_4>	57
Character Attributes				
Select color.	[ESC] c <n>	1BH,63H	&%CL <m_1>	59
Begin one-line double-wide print.	[SO]	0EH	&%MW	59
Cancel one-line double-wide print.	[DC4]	14H	&%MN	59

Multi-line double-wide double-high mode.
n=0 Standard mode
n=1 Double-wide
n=2 Double-high
3 = Both
	[ESC] W <n>	1BH,57H	&%FS {n=0}
			&%FD {n=1}
			&%FH {n=3}
				60

Enable/disable Strike Through.
n=0 End
n=1 Begin
	[ESC] _ <n>	1BH,5FH	&%CO {n=0}
			&%MO {n=1}
				60

Begin underline mode.
n = 0 End
n = 1 Begin
	[ESC] - <n>	1BH,2DH	&%CU {n=0}
			&%MU {n=1}
				62

Begin enhanced print.	[ESC] G	1BH,47H	&%ME	62
End enhanced print.	[ESC] H	1BH,48H	&%CE	62
Begin emphasized print.	[ESC] E	1BH,45H	&%MM	62
End emphasized print.	[ESC] F	1BH,46H	&%CM	63
Set print style. (See command description.)	[ESC] [@ ...	1BH,5BH, 40H…	&%DH &%SH	61
Select superscript.	[ESC] S <0>	1BH,53H, 00H	&%SP	63
Select subscript.	[ESC] S <1>	1BH,53H, 01H	&%SB	63
End superscript or subscript.	[ESC] T	1BH,54H	&%SE	63
Begin italics.	[ESC] % G	1BH,25H, 47H	&%MI	64
End italics.	[ESC] % H	1BH,25H, 48H	&%CI	64
Rotated Print				
Page Mode				
Select page mode	[ESC] t <n>	1BH,74H	&%PM <m_1>	66
Set page mode page position	[ESC] o <n_1> …	1BH,6FH		73
Exit page mode	[FF]	0CH	&%FF	73
APA Graphics				
Print single-density graphics. <n_1>=0...255 <n_2>=0…3 len=<n_1> + 256 * <n_2>	[ESC] K <n_1> <n_2>	1BH,4BH	none	74
Print half-speed double-density graphics.	[ESC] L <n_1> <n_2>	1BH,4CH	none	74
Print full-speed double-density graphics.	[ESC] Y <n_1> <n_2>	1BH,59H	none	74
Print quad-density graphics.	[ESC] Z <n_1> <n_2>	1BH,5AH	none	75

Select bidirectional or unidirectional print.
n=0 Bidirectional
n=1 Unidirectional
	[ESC] U <n>	1BH,55H	&%GU {n=1}
			&%GB {n=0}
				76

Print graphics in mode <m>.	[ESC] * <m> <n_1> <n_2>	1BH,2AH	none	75

Description	ASCII	Hex	IPCL equivalent code	Page
Reassign graphic mode.	[ESC] ? <m><n>	1BH,3FH	none	75
Two Color Graphics				
Process horizontal graphics data.	[ESC] h <color> <length> <format> <data>	1BH,68H	none	77
Set horizontal graphic mode.	[ESC] * <m> <0> <0>	1BH,2AH	none	78
User Store				
Begin named macro record.	[ESC] [US] b <Name..> <0>	1BH,1FH, 62H	&%UB <Name..><0>	85
Save macro data in user store.	[ESC] [US] m <Name..> <0>	1BH,1FH, 6DH	&%UM <Name..><0>	85
End named macro record.	[ESC] [US] e <Name..> <0>	1BH,1FH, 65H	&%UG <Name..><0>	85
Load item from user store.	[ESC] [US] l <Name..> <0>	1BH,1FH, 6CH	&%UL <Name..><0>	86
Save user-defined characters.	[ESC] [US] c <Name..> <0>	1BH,1FH, 63H	&%UC <Name..><0>	85
Run macro data from user store.	[ESC] [US] r <Name..> <0>	1BH,1FH, 72H	&%UR <Name..><0>	86
Flag item as a start-up macro.	[ESC] [US] s <Name..> <0>	1BH,1FH, 6DH	&%US <Name..><0>	86
Delete item from user store.	[ESC] [US] d <Name..> <0>	1BH,1FH, 64H	&%UD <Name..><0>	87
Flush information from user store.	[ESC] [US] f ALL <0>	1BH,1FH, 66H,00H	&%UF	87
Transfer user store to extended user store	[ESC] [US] t ALL <0>	1BH,1FH, 66H,00H	&%UF	87
Report on user store.	[ESC] [US] q <Name..> <0>	1BH,1FH, 72H	&%UQ <Name..><0>	87
Return a report on user store	[ESC] [US] ? <Name..> <0>	1BH,1FH, 72H	&%UQ <Name..><0>	87
Process user macro.	[ESC] g <0>	1BH,67H, 00H	&%GP	90
Start macro record.	[ESC] g <1>	1BH,67H, 01H	&%GS	90
Stop macro record.	[ESC] g <2>	1BH,67H, 02H	&%GE	90
Stop macro record and save.	[ESC] g <3>	1BH, 67H, 03H	&%GW	90
Bar Codes				
Print bar code.	[ESC] b <n> ... [NUL]	1BH,62H	See page 91	91
PDF 417 bar code control	[ESC] [EM] E <f> <v>	1BH 19H 45H	none	147

Set bar code height.
n=0 Restore defaults
n=1 - 9 Number of passes
(0.11 inch per pass)
	[ESC] [EM] B <n>	1BH,19H, 42H	&%BH <m$_1$><m$_2$>	159

Set bar code width	[ESC] [EM] W <n>	1BH 19H 57H	&%BW <m>	159
Set bar code justification, HRI print mode, and print direction.	[ESC] [EM] J <n>	1BH,19H, 4AH	&%BJ<m$_1$><m$_2$>	159
Unicode				
Initiate Unicode UTF-32BE encoding	[ESC] + h	1BH 2BH 68H	none	220
Initiate Unicode UTF-32LE encoding	[ESC] + l	1BH 2BH 6CH	none	221
Initiate Unicode UTF-16BE encoding	[ESC] + H	1BH 2BH 48H	none	220
Initiate Unicode UTF-16LE encoding	[ESC] + L	1BH 2BH 4CH	none	221
Initiate Unicode UTF-8 encoding (MBCS)	[ESC] + M	1BH 2BH 4DH	none	221
Initiate Unicode UTF-8 Text only encoding (MBCS)	[ESC] + T	1BH 2BH 54H	none	223
Initiate normal 8-bit ASCII character encoding	[ESC] + A	1BH 2BH 41H	none	223
File System Commands				

Description	ASCII	Hex	IPCL equivalent code	Page
Open File	[ESC][RS] O<Mod> FileName<0>	1BH 1EH 4FH	none	225
Return Free Space for Open File	[ESC][RS] S	1BH 1EH 53H	none	226
Return Free Space for Partition	[ESC][RS] s	1BH 1EH 73H	none	226
Return Last File Command Status	[ESC][RS] ?	1BH 1EH 3FH	none	228
Close File	[ESC][RS] C	1BH 1EH 43H	none	227
Close All Files	[ESC][RS] K	1BH 1EH 4BH	none	227
Open File	[ESC][RS] O FileName<0>	1BH 1EH 4FH	none	225
Set/Clear File Attributes	[ESC][RS] A <Atb> FileName<0>	1BH 1EH 41H	none	227
Write File Data	[ESC][RS] W <L_L> <L_H> <data>	1BH 1EH 57H	none	229
Read File Data	[ESC][RS] R <L_L> <L_H>	1BH 1EH 52H	none	229
File Directory File	[ESC][RS] I	1BH 1EH 49H	none	229
Delete all Files in partition	[ESC][RS] E <p>	1BH 1EH 45H	none	229
De-fragment File system	[ESC][RS] F	1BH 1EH 46H	none	230
Miscellaneous Commands				
Disable paper out sensor.	[ESC] 8	1BH,38H	&%PF	163
Enable paper out sensor.	[ESC] 9	1BH,39H	&%PO	163

Set left/right margins.
n1 = Left margin
n2 = Right margin
	[ESC] X <n_1><n_2>	1BH,58H	none	163

Clear print buffer.	[CAN]	18H	&%RP	164
Query marker.	[ESC] q <n>	1BH,71H	none	164
Perform Auto Cut	[ESC] v	1BH,76H	&%FC	165
Audio alert.	[BEL]	07H	&%BL	166
Configure audio alert.	[ESC] [BEL] <n_1> <n_2> <n_3>	1BH,07H	none	166
Initialize printer.	[ESC] @	1BH,40H	none	165
Enable paper error mode operation.	[ESC] p <n>	1BH,70H	&%PE <m_1> <m_2>	166
Inquire status. (Refer to command descriptions.)	[ENQ] <n>	05H	none	171
Control Periodic Status	[ESC][EM]P	1BH, 19H, 50H	none	185
Control Periodic Status	[ESC][EM]p	1BH, 19H, 70H	none	185
Extended Diagnostics				
Set control feature commands.	[ESC] y <n>	1BH,79H	&%Y0 - 9 or &%YX<m1><m2><m3> (for numbers >9)	168
Extended diagnostics.	[ESC] ~ <n>	1BH,7EH	none	169

Low Level Paper Motion Control

Print/Paper Motion

[CR] **Carriage return**

ASCII	[CR]
Hexadecimal	0DH
Decimal	<13>
IPCL	&%CR
EPOS	0DH

Description The [CR] command prints the contents of the print buffer (if any) and resets the next character print position to the left margin. A line feed is not performed unless auto-feed is active. The print rotation direction and the left margin command define the left margin.

[LF] **Line feed**

ASCII	[LF]
Hexadecimal	0AH
Decimal	<10>
IPCL	&%LF
EPOS	0AH

Description The [LF] command prints the contents of the buffer (if any) and advances paper one line at the current default line spacing. The next character print position is not reset to the left margin unless auto-CR is active.

Horizontal Motion Control

Several commands can be used to control the horizontal position of characters. Many applications use space control to position fields. However, the EPIC 3000 Printer has the ability to control character position with horizontal tab stops. This is done using the horizontal tab [HT] to move to those tab stops.

[HT] **Horizontal tab**

ASCII	[HT]
Hexadecimal	09H
Decimal	<9>
IPCL	&%HT
EPOS	[HT]

Description The [HT] command inserts spaces in the print buffer up to the next tab stop. The default tab locations are every eight spaces.

[BS] Back space

ASCII	[BS]
Hexadecimal	08H
Decimal	<8>
IPCL	&%BS
EPOS	[BS]

Description The [BS] command moves the print buffer one character width to the left. The pointer position cannot be moved to the left of the left margin. [BS] does not cause the buffer to be printed; rather, the following data is OR'ed with the previous data.

[ESC] D Set horizontal tab stops

ASCII	[ESC] D <n_1> <n_2> <n_3> ... <n_i> 0
Hexadecimal	1BH 44H <n_1> <n_2> <n_3> ... <n_i> 00H
Decimal	<27> <68> <n_1> <n_2> <n_3> ... <n_i> <0>
IPCL	none
EPOS	[ESC] D <n_1> <n_2> <n_3> ... <n_i> 0

Description The [ESC] D <n_1> <n_2> <n_3> ... <n_i> 0 command sets tab stops at the character columns specified by <n>. The end of the settings is specified by a <0>. All previously set tabs will be cleared. The restore-default procedure other than to re-specify the tabs. Column sizes are in accordance with the current character pitch. Setting tabs that are beyond the station width is possible. A [CR] is inserted when the tab is used. Printing begins at the home position. The power up default is every eight spaces, i.e., 9, 17, 25, and so on.

[ESC] R Reset horizontal and vertical tab stops

ASCII	[ESC] R
Hexadecimal	1BH 52H
Decimal	<27> <82>
IPCL	&%HV
EPOS	none

Description The [ESC] R command resets horizontal and vertical tab stops to the power up configuration. The power up horizontal default is every eight spaces, i.e., 9, 17, 25, and so on. The vertical default is every line.

[ESC] a Set Justification

ASCII	[ESC] a <n>
Hexadecimal	1BH 61H <n>
Decimal	<27> <97> <n>
IPCL	&%JL, &%JC, &%JR
EPOS	[ESC] a <n>

Description The [ESC] a <n> command sets the horizontal justification.

Where <n>	0 = Left justified	&%JL
	1 = Center justified	&%JC
	2 = Right justified	&%JR
	8 = Left justified (no line feed)	None
	9 = Center justified (no line feed)	None
	10 = Right justified (no line feed)	None

The print format of the printer can be right, center, or left justified. The value of <n> specifies the justification. The power on default is left justified.

Note: The justify commands do not affect graphics.
Note: For the no line feed option to function, the line buffer must be empty.

[ESC] n Set horizontal position

ASCII	[ESC] n <n_1> <n_2>
Hexadecimal	1BH 6EH <n_1> <n_2>
Decimal	<27> <110> <n_1> <n_2>
IPCL	&%HP <m_1> <m_2> <m_3>
EPOS	[ESC]$ <n_1> <n_2>

Description The [ESC] <n> <n_1> <n_2> command sets the print position to <n_1> + <n_2> * 256 in 1/196 inches.

Vertical Motion Control

[ESC] j Perform a fine line feed

ASCII	[ESC] J <n>
Hexadecimal	1BH 4AH <n>
Decimal	<27> <74> <n>
IPCL	&%FM <m_1> <m_2> <m_3>
EPOS	[ESC] J <n>

Description The [ESC] J <n> command prints the contents of the buffer and performs a line feed of n/216 inch. The default line spacing value is not changed. The next character print position is reset to the left margin if the Auto-CR mode is set.

Notes:

- In EPOS mode, the command performs feeds in n/144-inch increments.

- Immediately after APA graphics, the command is adjusted for the difference between 72 dpi graphics and 96 dpi print.

[ESC] 3 Set variable line spacing to n/216 inch

ASCII	[ESC] 3 <n>
Hexadecimal	1BH 33H <n>
Decimal	<27> <51> <n>
IPCL	&%SV <m1> <m2> <m3>
EPOS	[ESC] 3 <n>

Description The [ESC] 3 <n> command sets the default line spacing to n/216 inch. Set n = 1 to 255. The line feed spacing used by [LF] is set to values other than 1/8 or 7/72 inch. The command takes effect immediately.

Note: In EPOS mode, the command performs line feeds in n/144-inch increments.

[ESC] 0 Set line spacing to 1/8 inch

ASCII	[ESC] 0
Hexadecimal	1BH 30H
Decimal	<27> <48>
IPCL	&%ST
EPOS	[ESC] 2

Description The [ESC] 0 command sets the default line spacing to 1/8 inch (27/216 inch), which is the standard eight lines per inch line spacing at initial power-up.

Note: In EPOS mode, the command sets 1/6-inch spacing or six lines per inch.

[ESC] 1 Set line spacing to 21/216 inch or 7/72 inch

ASCII	[ESC] 1
Hexadecimal	1BH 31H
Decimal	<27> <49>
IPCL	&%SG
EPOS	none

Description The [ESC] 1 command sets the default line spacing to 21/216 inch. Use 21/216-inch line spacing for all-points-addressable (APA) graphics printing.

[ESC] A Set variable line spacing to n/72 inch

ASCII	[ESC] A <n>
Hexadecimal	1BH 41H <n>
Decimal	<27> <65> <n>
IPCL	none
EPOS	none

Description The [ESC] A <n> command sets the default line spacing to n/72. Set n = 1 to 85. Variable line spacing does not take effect until enabled by the [ESC] 2 command. The command is provided to maintain backward compatibility with the Ithaca® Series 50, OKIDATA, IBM, and other printers. It can also be used to print on preprinted forms.

[ESC] 2 Enable [ESC] A <n> variable line spacing

ASCII	[ESC] 2
Hexadecimal	1BH 32H
Decimal	<27> <50>
IPCL	none
EPOS	none

Description The [ESC] 2 command is a companion to the [ESC] A <n> command and puts the specified line spacing into effect. It remains in effect until another line spacing command is issued.

[ESC] d Feed <n> lines at the current spacing

ASCII	[ESC] d <n>
Hexadecimal	1BH 64H <n>
Decimal	<27> <100> <n>
IPCL	&%FL <m1> <m2>
EPOS	[ESC] d

Description The [ESC] d <n> command prints the contents of the buffer (if any) and performs <n> line feeds at the current line spacing. The command does not change the default line spacing value. The next character print position is reset to the left margin.

Note: The IPCL command prints from 00 to 99 lines. For example, if you wish to feed 12 lines, the IPCL command would be as follows: &%FL12.

Function	**Reverse feed <n> lines at the current spacing**
ASCII	[ESC] e <n>
Hexadecimal	1BH 65H <n>
Decimal	<27> <101> <n>
IPCL	&%FB <m1> <m2>
EPOS	[ESC] e

Description The [ESC] e <n> command prints the contents of the buffer (if any) and performs <n> reverse line feeds at the current line spacing. The command does not change the default line spacing value. The next character print position is reset to the left margin.

Note: The IPCL command prints from 00 to 99 lines. For example, if you wish to feed 12 lines, the IPCL command would be as follows: &%FL12.

[VT]	**Vertical tab**
ASCII	[VT]
Hexadecimal	0BH
Decimal	<11>
IPCL	&%VT
EPOS	(VT)

Description The printer sets a line counter to the top of the form at reset and when a set top of form command is issued. By setting vertical tab stops, various form positions can be reached with a [VT] operation.

[ESC] B	**Set vertical tab stops**
ASCII	[ESC] B <n_1> <n_2> <n_3> ... <n_i> 0
Hexadecimal	1BH 42H <n_1> <n_2> <n_3> ... <n_i> 00H
Decimal	<27> <66> <n_1> <n_2> <n_3> ... <n_i> <0>
IPCL	none
EPOS	[ESC] B <n_1> <n_2> <n_3> ... <n_i> 0

Description The [ESC] B <n_1> <n_2> <n_3> ... <n_i> 0 command sets tab stops at line positions specified by <n>. The end of the setting is specified by a <0>. All previously set tabs will be cleared. If n_n is less than n_{n-1}, then the command is in error, and all of the following information is printed. In other words, tab stops must be entered sequentially in order to be accepted. A total of 64 tab stops can be specified. (The power on default is a vertical tab on every line).

[ESC] R Reset horizontal and vertical tab stops

ASCII	[ESC] R
Hexadecimal	1BH 52H
Decimal	<27> <82>
IPCL	&%HV
EPOS	none

Description The [ESC] R command resets horizontal and vertical tab stops to power up configuration. The power up horizontal default is every eight spaces, i.e., 9, 17, 25, etc. The vertical default is every line.

[FF] Form feed

ASCII	[FF]
Hexadecimal	0CH
Decimal	<12>
IPCL	&%FF
EPOS	none

Description The [FF] command performs a form feed to the top of the form.

Note: The form feed command can be disabled. Set the form length to zero.

[ESC] 4 Set top of form

ASCII	[ESC] 4
Hexadecimal	1BH 34H
Decimal	<27> <52>
IPCL	&%TF
EPOS	[ESC] L

Description The [ESC] 4 command sets the top of form to the current position.

[ESC] C Set form length in lines

ASCII	[ESC] C <n>
Hexadecimal	1BH 43H <n>
Decimal	<27> <67> <n>
IPCL	&%SL <m_1> <m_2>
EPOS	[ESC] C <n>

Description The [ESC] C <n> command sets the form length to <n> lines at the current line spacing. If the current page position is greater than the new page length, the command also sets the current position as the top of form.

[ESC] C [NUL] Set form length in inches

ASCII	[ESC] C [NUL] <n>
Hexadecimal	1BH 43H <0> <n>
Decimal	<27> <67> <0> <n>
IPCL	&%SI <m_1> <m_2>
EPOS	none

Description The [ESC] C [NUL] <n> command sets the form length to <n> inches. If the current page position is greater than the new page length, the command also sets the current position as the top of form. If zero inches are specified, the form feed and vertical tab commands are ignored.

[ESC] 5 Begin auto line feed

ASCII	[ESC] 5 <n>
Hexadecimal	1BH 35H <n>
Decimal	<27> <53> <n>
IPCL	&%MA (Begin)
IPCL	&%CA (End)
EPOS	none

Description The [ESC] 5 <1> command sets auto line feed mode. [ESC] 5 <0> command ends auto line feed mode.

Note: The begin and end auto line feed command overrides the configuration setting.

[ESC]] Reverse line feed

ASCII	[ESC]]
Hexadecimal	1BH 5DH
Decimal	<27> <93>
IPCL	&%LR
EPOS	none

Description The [ESC]] command performs a reverse line feed at the current line spacing.

Note: The EPIC 3000 Printer can tolerate no more than 1/2 inch of reverse feed.

Feed to Black Dot

An option for the EPIC 3000 is a black dot sensor. This command is supported by a feed to black dot command.

Function	**Feed to Black Dot**
ASCII	[ESC][VT]<n>
Hexadecimal	!BH,0BH,<n>
Decimal	<27><11><n>
IPCL	None
EPOS	[GS][VT]<n>
Where n	<n> is the number of inches to feed while looking for the black marker.
Description	The [ESC][VT] command performs feeds the paper until the black dot is positioned for cut. The EPIC 3000 keeps track of the location of the black dot and will feed paper until the black dot is positioned so as to allow the auto cutter to cut within the black dot.

Character Pitch

Character pitch commands that set specific characters per inch (cpi) disable any right-side spacing set by the [ESC] V <n> command. In addition, when font changes are made, the character pitch is maintained.

[DC2]	**Begin 10 cpi character pitch**
ASCII	[DC2]
Hexadecimal	12H
Decimal	<18>
IPCL	&%F3
EPOS	[ESC][SP] <n>
Description	The [DC2] command sets 9.905 characters per inch print pitch.

[ESC] :	**Begin 12 cpi character pitch**
ASCII	[ESC] :
Hexadecimal	1BH 3AH
Decimal	<27> <58>
IPCL	&%F2
EPOS	[ESC] [SP] <n>
Description	The [ESC] : command sets 12.235 characters per inch print pitch.

[SI]	**Begin 17 cpi character pitch**
ASCII	[SI]
Hexadecimal	0FH
Decimal	<15>
IPCL	&%F1
EPOS	[ESC] [SP] <n>
Description	The [SI] command sets 17.333 characters per inch print pitch.

[ESC] [SI]	**Begin 24 cpi character pitch**
ASCII	[ESC] [SI]
Hexadecimal	1BH 0FH
Decimal	<27> <15>
IPCL	&%F4
EPOS	[ESC] [SP] <n>
Description	The [ESC] [SP] command sets 23.111 characters per inch print pitch.

[ESC] [P Set character pitch

ASCII [ESC] [P <n>
Hexadecimal 1BH 5BH 50H <n>
Decimal <27> <91> <80> <n>
IPCL &%F1, &%F2, &%F3, &%F4, &%F5, &%F6, &%F7
EPOS [ESC] [SP] <n>

Description The [ESC] [P <n> command sets character per inch print pitch to <n>. The printer resolution limits the exact print pitch. The following table lists the exact pitch for various values on <n>.

<n>	Resulting Characters per Inch	IPCL
1	1.00	
2	2.00	
3	3.01	
4	4.00	
5	4.95	
6	5.94	
7	6.93	
8	8.00	&%F7
9	9.04	
10	9.90	&%F3
11	10.95	
12	12.23	&%F2
13	13.00	
14	13.87	
15	14.86	&%F6
16	16.00	
17	17.33	&%F1
18	17.33	
19	18.91	
20	20.8	&%F5
21	20.8	
22	23.11	
23	23.11	
24	23.11	&%F4
25	23.11	
26	26	
27	26	
28	26	
29	29.71	
30	29.71	

Table 2 Character Pitch

This command disables any right-side spacing set by the [ESC] V command. In addition when font changes are made, the character pitch is maintained.

[ESC] V Set inter-character spacing

Mode	Global
ASCII	[ESC] V <n>
Hexadecimal	1BH 56H <n>
Decimal	<27> <86> <n>
IPCL	none
EPOS	[ESC] [SP] <n>

Description The [ESC] V <n> command sets inter-character spacing by adding white space between characters. The value of <n> sets the spacing in 216ths of an inch. The printer can only set the spacing in 208ths of an inch and converts 216ths to the nearest 208th of an inch. Each font has a basic size, and the inter-character spacing value is added to the basic size. Therefore, the affect of this command on characters per inch (cpi) will depend on the font selected.

Note 1: The [ESC] V <n> command disables any pitch settings established by pitch set commands that establish a cpi (like [ESC] [P <n>). After a set right-side spacing command is issued, the pitch will vary with font selection. Font selections use the current, active, right-side spacing.

Note 2: With the inter-character spacing command, the pitch cannot be set less than the font size. Therefore, it is not as effective as the pitch command, [ESC] [P <n>. The following table lists the cpi equivalent for several values of <n>.

The following table lists the cpi equivalent for several values of <n>.

<n>	Small Draft (cpi)	Large Draft (cpi)	NLQ (cpi)
0	17.33	14.86	13
1	16	13.9	12.24
2	14.86	13	11.6
3	13.9	12.24	10.9
4	13	11.6	10.4
5	12.24	10.9	9.90

Table 3 Inter-character Spacing

Character Font

[ESC] # Begin 12 x 12 draft print mode

ASCII	[ESC] # <0>
Hexadecimal	1BH 23H 00H
Decimal	<27> <35> <0>
IPCL	&%QT
EPOS	[ESC] ! <n>

Description The [ESC] # <0> command begins 12 x 12 draft print mode. Draft print is provided to maintain compatibility with other Ithaca® products.

[ESC] I Set print quality mode

ASCII	[ESC] I <n>
Hexadecimal	1BH 49H <n>
Decimal	<27> <73> <n>
IPCL	&%QT 24 x 10 Small
	&%QU 24 x 14 Medium
	&%QL 24 x 18 Large
	&%QS 24 x 20 Larger
EPOS	[ESC] x <n> and/or [ESC] ! <n>

Description The [ESC] I <n> (e.g. capital “i”) command begins draft, large draft or near letter quality print mode.

Where n	0 = 24 x 10 Small
	1 = 24 x 14 Medium
	2 = 24 x 18 Large
	3 = 24 x 20 Larger
	4 - 7 repeats 0 - 3

Note 1: In EPOS mode, [ESC] x <n> is similar to [ESC] I <n>. [ESC] ! <n> performs a similar function.

[ESC] P Begin rotated font/Right to left entry

ASCII	[ESC] P <n>
Hexadecimal	1BH 50H <n>
Decimal	<27> <80> <n>
IPCL	&%RI {n=2}
&%RF {n=1}	
&%RN {n=0}	
EPOS	[ESC] V <n>

Description The [ESC] P <n> command sets the print font to a rotated 90° or 270° font.

Where n

n = 0	Normal
n = 1	Rotate 90°
n = 2	Rotate 270°
n = 8	Enter Text right to left
n = 9	Enter Text right to left and Rotate 90°
n = 10	Enter Text right to left and Rotate 270°

Character Sets and Code Pages

The EPIC 3000 Printer is primarily intended to be used in Unicode based systems. However to provide legacy support, the printer supports 8 bit and double byte ASCII encoding with code pages.

When not using Unicode or double byte encoding, the printer is restricted to the 8 bit ASCII character set. To support international languages, the characters that are assigned to each of the 256 possible locations can be remapped to any character in the Unicode standard. Typically, the first 32 characters are reserved for control characters. The next 72 are typically fixed to alpha numeric and punctuation. The upper 128 characters are typically redefined to characters that support the specific language or country. These mappings are generally referred to as codepages.

The EPIC 3000 provides several legacy commands to select a country code or codepage maps. However, they all simply select an ASCII to Unicode translation map. Several predefined mappings to provide legacy support are internal to the printer, but the majority of the maps are placed in the printer's file system. This allows any preexisting codepage mapping to be defined as well as the ability to define custom mappings unique to your application.

Note: Note: If UTF encoding is active code pages are not meaningful. Code page commands will have no affect on the character addressing.

Codepage description files

The format of the codepage description file is somewhat flexible. The basic format is that each line will specify an ASCII character ID and the Unicode character that is to appear in that ASCII ID location.

The file format is one character per line with the first value being the ASCII ID and the second value being the Unicode address. The file should be something like this:

```
0x00  0x0000          # NULL
0x01  0x0001          # START OF HEADING
0x02  0x0002          # START OF TEXT
0x03  0x0003          # END OF TEXT
...
or
\Language = USA
\Code Page = 437
0x00  0x0000          ; 0      NULL
0x01  0x263A          ; 9786  WHITE SMILE
0x02  0x263B          ; 9787  BLACK SMILE
0x03  0x2665          ; 9829  BLACK HEART SUIT
0x04  0x2666          ; 9830  BLACK DIAMOND SUIT
0x05  0x2663          ; 9827  BLACK CLUB SUIT
...
or
0       0          ; NULL
1       9786 ;  WHITE SMILE
2       9787 ;  BLACK SMILE
3       9829 ;  BLACK HEART SUIT
4       9830 ;  BLACK DIAMOND SUIT
5       9827 ;  BLACK CLUB SUIT
...
or
SYMBOL SET = WE
/name = Windows 3.1 Latin 2
/pcl char = E
...
/symbols =
32  0x0020 ; Space Code, Prntabl Thin Space
33  0x0021 ; Exclamation
34  0x0022 ; Neutral Double Quote
5   0x0023 ; Number
36  0x0024 ; Dollar
...
or
\Language = USA
\Code Page = 437
0x00  -         0x0000          ; 0      NULL
0x01  -         0x263A          ; 9786  WHITE SMILE
0x02  -         0x263B          ; 9787  BLACK SMILE
0x03  -         0x2665          ; 9829  BLACK HART SUIT
0x04  -         0x2666          ; 9830  BLACK DIAMOND SUIT
0x05  -         0x2663          ; 9827  BLACK CLUB SUIT
...
or
\Language = USA
\Code Page = 437
```

```
0x00 =      0x0000      ; 0     NULL
0x01 =      0x263A      ; 9786  WHITE SMILE
0x02 =      0x263B      ; 9787  BLACK SMILE
0x03 =      0x2665      ; 9829  BLACK HART SUIT
0x04 =      0x2666      ; 9830  BLACK DIAMOND SUIT
0x05 =      0x2663      ; 9827  BLACK CLUB SUIT
```

Numbers beginning with 0x… are treated as hexadecimal, all other as decimal. Any line beginning with a non-numeric value is ignored. Any information after the Unicode value is ignored. Not all of the ASCII ID's need to be present, however, only ID's present will be affected.

There are two ways to select a codepage file. The first is by using the standard code page select command. If this command is used, the file name is critical; it must follow the format of Cpxyz.CPM. The xyz is the code page number that is being selected in the command. For example CP850.CPM would be referred to as 850. If the legacy commands are to be used to select file based code page mapping, the selection mode must be selected in the printer's configuration.

The second form is by name. This command is free form and will select any file present which will then be to use it as a code page definition. If the file is not a code page file, you will get unexpected results. If the file does not exist, the command is ignored. Any extension may be used for a codepage map, however, it is best to use the .CPM extension, with a descriptive filename recommended. For example ISO8859-1.CPM would be a good choice for the ISO8859-1 code page.

Double-Byte and Multi-Byte Code Page Description Files

In ASCII mode the EPIC 3000 Printer supports double byte code pages. Currently there are 4 double-byte codepages available and one multi-byte.
Double Byte code pages available are: Code page 932, 936, 949 and 950
Multi-byte code page available are GB18030-2000.

These code pages require an appropriate Unicode font be loaded that support all the characters in the requested code page.

Due to the large number of characters in these code pages. These files are not supported in a customer definable ASCII form like the Code page descriptions files described above. They are compiled and compressed into a 2 or 3 file code description set. One file is like the Code page description file in that it may be referenced just like the normal code page description files. The other two files are double byte and multi-byte decode files and are referenced in the master code page description. These files not distributed with the standard printer but are available from Transact upon request.

Code page selection

In ASCII mode the EPIC 3000 Printer supports over 40 different international character sets by default. In IBM and EPOS printers, there are two ways of selecting a character set.

One way substitute's international characters in the upper 128 characters of a standard character set. The substitution technique supports a few different countries. However, as more and more countries were added, too many characters were being replaced, and it became a problem for the application to match the characters displayed and printed.

To solve this problem, a second method of selecting a character set was developed – code pages. The printer and display use the same code page, and the application displays and prints the same characters. IBM and EPOS defined new commands to select code pages, and left the old commands in effect.

The EPIC 3000 Printer supports international character sets as well as code pages. To allow the most flexibility for the application programmer, both methods are extended in the EPIC 3000 Printer.

The EPIC 3000 Printer allows the IBM code page selection command to choose character sets as well as normal IBM code pages. The EPOS character set select command has been extended to allow additional character sets over and above the 11 defined by EPOS. The EPOS[6] code page select command has not been extended because there is no EPOS definition beyond the first six ID's.

All characters in code pages as well as character sets are addressed as zero through 255. (Characters below 32 must be addressed with the [ESC] ^ <n> command.) Code pages may be changed at any time and are active for all features including rotated print. To allow other code pages to be created by an application, a redefine character set command is provided.

As discussed above, there are two commands for language selection in IBM mode. The first is [ESC] !, which selects one of 19 international character sets. The [ESC] ! command does not allow all of the possible character sets to be selected, it is provided for compatibility with older programs only. The second is [ESC] [T, which selects any code page. In EPOS mode, the [ESC] R command has been expanded to select any code page.

[ESC] ! Select international character set

ASCII [ESC] ! <n>
Hexadecimal 1BH 21H <n>
Decimal <27> <33> <n>
IPCL &%CS <n>
EPOS [ESC] R <n>

Description The [ESC] ! <n> command selects international character set <n>. In standard mode, the value of <n> is as follows.

<n>	Language	<n>	Language
64-'@'	ASCII (slashed zero)	73-'I'	Italian
65-'A'	ASCII (unslashed zero)	74-'J'	French Canadian

[6] Epson provides limited code page support through ID to code page translation. Only six translations are defined.

66-'B'	British	75-'K'	Spanish
67-'C'	German	76-'L'	Swedish II
68-'D'	French	77 -'M'	Swedish III
69-'E'	Swedish	78-'N'	Swedish IV
70-'F'	Danish	79-'O'	Turkish
71-'G'	Norwegian	80-'P'	Swiss I
72-'H'	Dutch	81-'Q'	Swiss II

Table 4 Language Table ID's

[ESC] [T   Select character code page

ASCII [ESC] [T <n_h> <n_l>
Hexadecimal 1BH 5BH 54H <n_h> <n_l>
Decimal <27> <91> <84> <n_h> <n_l>
IPCL &%CP <m_1> <m_2> <m_3> <m_4>
EPOS [ESC] t <n>

Description The [ESC] [T <n_h> <n_l> command selects character code page <n_h> <n_l>. The EPIC 3000 Printer supports many code pages. The following code pages are supported.
Refer to Appendix A for a list of supported code page.

Note: The code page field is a 16-bit field that is a function of the code page numbers <n_h> and <n_l>, e.g. (<n_h> * 256) +<n_l>. For example, 1 * 256 + 181 = 437. For the IPCL command, the page is specified in ASCII as a 4-byte field.
Note: If the code page is not found in the internal translation, the code page requested is translated into a code page file name and if a user defined file is found, it will be used as the code page definition.
Note: If UTF encoding is active code pages are not meaningful. This command will have no affect on the character addressing.

Function **Select character code page by name.**
ASCII [ESC] + C Codepage.CPM <0>
Hexadecimal 1BH 2BH 43H
Decimal <27> <43> <67>
CodePage File name from 5 to 30 characters.
Description The [ESC] + C command will select and read a code page encoding file. If the file does not exist, the current code page will be selected. If the code page definition is not complete, only the character locations defined by the file will be effected.

Note: This command functions by redefining the code page translation table. if Unicode UTF encoding is active, all code page selection and modification commands do not functional and have no effect.

[ESC] [C   Insert Euro character

ASCII ESC] [C <n>
Hexadecimal 1BH 5BH 43H <n>
Decimal <27> <91> <67> <n>

IPCL &%EU

Description The [ESC] [C <n> command allows an application to replace any character in the currently active character set with the Euro character. The character to be replaced is defined by <n>. For example, if the currently active character set is CP 850 (multi-lingual) and 0D5H character is to be the Euro character, "1BH 5BH 43H 0D5H" replaces the character at 0D5H with the Euro symbol.

Euro Character Substitution Matrix

Name	Epson	IBM	Code Page Insertion Point (hex)
850	26	850	0xD5
Turkey 857	57	857	0XD5
Win Cyrillic	52	1022	0X88
Win Turkish	51	1021	0X80
Win Greek	50	1020	0X80
Win Hebrew	62	1032	0X80
Win Baltic	68	1034	0X80

Table 5 Euro Character Substitution Matrix

Note: This command functions by redefining the code page translation table. if Unicode UTF encoding is active, all code page selection and modification commands do not functional and have no effect.

[ESC] [S Redefine character set

ASCII [ESC] [S <L_L> <L_H> <B_C> <T_{1H}><T_{1L}> <T_{2H}><T_{2L}> <T_{3H}><T_{3L}> …<T_{nL}><T_{nH}>
Hexadecimal 1BH 5BH 53H …
Decimal <27> <91> <83> …
IPCL none

Description The [ESC] [S <L_L> <L_H> <B_C> <T_{1H}><T_{1L}> <T_{2H}><T_{2L}> <T_{3H}><T_{3L}> … <T_{nH}><T_{nL}> command allows an application to replace or redefine the active character set mapping in the printer, where <L_H> <L_L> defines the total length of the following data:

<L_L> + 256 * <L_H> = 1 + 2 * the total number of characters to be replaced;
<B_C> is the first character in the active map to be replaced
<T_{1H}> <T_{1L}>[7] is the internal address of the replacement character image.

The mapping of a print pattern to each character address is referred to a code page or character set. At any given time, the printer character set is comprised of 256 characters. Each character is addressed by an 8-bit value generally referred to as a character code. For example, if you want to print an 'A', it would be addressed by sending a <65> decimal to the printer. Sixty-five predefined code pages or character maps assign characters to a particular address built into the printer. Occasionally, an application needs to redefine a character or group of characters in a code page. The EPIC 3000 Printer allows the map for any code page to be redefined or replaced. The define character set command allows any character or group of characters to be replaced with any other printable character. Unicode addressing is used. The redefine character set command is used as follows:

```
[ESC] [ S <3> <0> <35> <90> <1>
      ^^^^^   ^^   ^^^^^^
      |       |     +- 346th Character in the Master Set
      |       |           [(1 * 256) + 90]
      |       +------- 35th Character
      +--------------- 3 bytes to follow [(0 * 256) + 3]
```

The new map remains until the printer is power cycled or the character set is redefined. The code page and character set commands completely redefine the table.

Note: This command functions by redefining the code page translation table. if Unicode UTF encoding is active, all code page selection and modification commands do not functional and have no effect.

[7] The internal character map is provided in the *Master Character Set Definitions Guide*, PN 100-9785.

Print Control Characters

[ESC] ^ Print control character

ASCII	[ESC] ^ <n>
Hexadecimal	1BH 5EH <n>
Decimal	<27> <94> <n>
IPCL	&%CC <m1> <m2> <m3>
EPOS	[ESC] ^ <n>

Description The [ESC] ^ <n> command allows characters from zero to 31 codes to be printed. During normal operation, characters from zero to 31 are control characters. The command turns off control code translation for the following character. <n> can be from zero to 255.

[ESC] " Print Unicode character

ASCII	[ESC] " <n_L><n_H>
Hexadecimal	1BH 22H <n_L><n_H>
Decimal	<27> <34> <n_L><n_H>
IPCL	&%PU<m1> <m2> <m3><m4> <m5>
EPOS	[ESC] " <n_L><n_H>

Description The [ESC] "<n_L><n_H>command allows any Unicode character to be directly addressed and inserted into the print data. <n_L><n_H> can range in value from zero to 65535. (Does not support Extended 24 bit addressing)

User Defined characters

[ESC] = **Define user-defined characters**

ASCII [ESC] = <y> <c_1> <c_2> [<x_1> <d_1> … d(y *x* x_1)] … [<x_k> <d_1> … d (y *x* x_k)]
Hexadecimal 1BH 3DH <y> …
Decimal <27> <51> <y> …
Range y = 2 or 3
$32 \le c_1 \le c_2 \le 255$
$0 \le x \le 24$ font character width
$0 \le d_1$ … d (y *x* x) ≤ 255
IPCL none

Description: The [ESC] = <y> <c_1> <c_2> [<x_1> <d_1> … d(y *x* x_1)] … [<x_k> <d_1> … d (y *x* x_k)] command defines user-defined characters from character code <c_1> to <c_2>. <y> and <x> are the configurations of a user-defined character. <y> specifies the number of bytes in the vertical direction. <x> specifies the number of bytes in the horizontal direction. Character code ranges from 32 (20H) to 255 (FFH) can be defined by <c_1> and <c_2>*.* Up to 223 user-defined characters can be defined. Data (<d>) specifies a bit printed to one and not printed to zero. At the default, user-defined characters are not defined and the internal character set is printed. Once the user-defined characters have been defined, they are available until [ESC] $ is executed; the user-defined characters are redefined; the power is turned off; or the printer is reset.

Note: User defined are bit ages and are not scalable. It is intended that user defined characters be defined using a custom true type font. That font may then be selected by the user's application.

[ESC] $ **Cancel user-defined characters**

ASCII [ESC] $
Hexadecimal 1BH 24H
Decimal <27> <36>
IPCL none

Description The [ESC] $ command removes all user-defined characters from the printer's memory. After the user-defined characters are canceled, the internal character set is printed.

[ESC] > **Enable user-defined characters**

ASCII [ESC] > <n>
Hexadecimal 1BH 3EH <n>
Decimal <27> <62> <n>
Range <0>, <1>, 0, or 1
1 Enables the characters
0 Disables the characters
IPCL none

Description The [ESC] > <n> command enables or disables the user-defined characters. The internal character set is printed.

Character Attributes

[ESC] c Select color

ASCII	[ESC] c <n>
Hexadecimal	1BH 63H <n>
Decimal	<27> <99> <n>
IPCL	&%CL <m_1>
EPOS	none

Description The [ESC] c <n> command selects the print color, and should match the color of the paper installed.

Where <n>	0 = Black
	1 = Red
	2 = Green
	3 = Blue

[SO] Begin one-line double-wide print

ASCII	[SO]
Hexadecimal	0EH
Decimal	<14>
IPCL	&%MW
EPOS	none

Description The [SO] command causes subsequent characters to be printed at twice the currently selected character width. For example, ten cpi becomes five cpi, 17 cpi becomes 8.5 cpi, etc. The [SO] command remains in effect until: A valid line terminator is received ([CR], [LF], or [ESC] J <n> (fine line feed)); The command is canceled; or the maximum number of characters per line is reached and the printer performs an auto-print.

[DC4] Cancel one-line double-wide print

ASCII	[DC4]
Hexadecimal	14H
Decimal	<20>
IPCL	&%MN
EPOS	none

Description The [DC4] command cancels one-line double-wide mode set by the [SO] command and allows single- and double-wide characters to be printed on the same line.

[ESC] _ Enable/Disable Strike Through

ASCII	[ESC] _ <n>
Hexadecimal	1BH 5FH 01H
Decimal	<27> <95> <n>
IPCL	&%MO (Begin)
IPCL	&%CO (End)
EPOS	[ESC] ! <n>

Description The [ESC] _ <1> command begins strike through print mode. All subsequent text, leading spaces, and trailing spaces are over-scored. [ESC] _ <0> ends the mode.

[ESC] W Multi-line double-wide and double-high mode

ASCII	[ESC] W <n>
Hexadecimal	1BH 57H <n>
Decimal	<27> <87> <n>

IPCL	&%FD {n = 1}
	&%FS {n = 0}
	&%FHA {n = 3}
	(Note: Single-wide, double-high mode is not available in IPCL mode.)

EPOS	[ESC] ! <n>

Description The [ESC] W <n> command controls multi-line double-wide or double-high mode, where n specifies the mode:

n = 0 is standard single-wide and single-high;
n = 1 begins double-wide;
n = 2 begins double-high; and
n = 3 begins double-wide double-high.

Note: The [ESC] W <n> command does not affect line spacing.

Where n Bits	**76543210**	**Function**
	1-------	Underline
	--1-----	Double-wide
	---1----	Double-high
	-------X	Font: 0 = draft; 1 = large draft

[ESC] [@ Set print style: double-wide, double-high, italic control

ASCII	[ESC] [@ [EOT] [NUL] <k> [NUL] <n> <m>	
Hexadecimal	1BH 5BH 40H 04H 00H <k> 00H <n> <m>	
Decimal	<27> <91> <64> <04> <0> <K> <0> <n> <m>	
IPCL	&%DH	Double-high, double-wide, and double-space
	&%SH	Single-high, single-wide, and single-space
	Also, see [ESC] W above.	

Description The [ESC] [@ [EOT] [NUL] <k> [NUL] <n> <m> command sets double-wide, double-high, and italic print mode.

Where k bits	**k**	**76543210**	
		`----xxxx`	Italic control
	0	`----0000`	No change
	1	`----0001`	Italics On
	2	`----0010`	Italics Off
Where n bits	**n**	**76543210**	
		`----0nnn`	Height multiplier (Maximum 4)
	0	`----0000`	No change
		`xxxx----`	Line spacing
	0	`0000----`	No change
Where m bits	**m**	**76543210**	
		`----0nnn`	Width multiplier (Maximum 4)
	0	`----0000`	No change

Note: The maximum height and width multiplier is four.

[ESC] - Begin underline

ASCII	[ESC] - <n>
Hexadecimal	1BH 2DH 01H
Decimal	<27> <45> <n>
IPCL	&%MU (Begin)
IPCL	&%CU (End)
EPOS	[ESC] ! <n>

Description The [ESC] - <1> command begins underline print mode. All subsequent text, leading spaces, and trailing spaces are underlined. [ESC] - <0> ends the mode.

Note: In EPOS mode, [ESC] ! <n> performs a similar function; however, near letter quality (NLQ) is not available.

[ESC] G Begin enhanced print

ASCII	[ESC] G
Hexadecimal	1BH 47H
Decimal	<27> <71>
IPCL	&%ME
EPOS	[ESC] G <1>

Description All subsequent text is printed in enhanced print mode (two passes with a vertical offset). Enhanced printing provides a deeper resolution of each character and may enhance multiple part forms printing.

[ESC] H End enhanced print mode

ASCII	[ESC] H
Hexadecimal	1BH 48H
Decimal	<27> <72>
IPCL	&%CE
EPOS	[ESC] G <0>

Description The [ESC] H command cancels enhanced print mode and returns to the currently selected font.

[ESC] E Begin emphasized print mode

ASCII	[ESC] E
Hexadecimal	1BH 45H
Decimal	<27> <69>
IPCL	&%MM
EPOS	[ESC] E <1>

Description The [ESC] E command begins emphasized print mode (one pass with horizontal offset). Emphasized print is bolder than normal print.

[ESC] F End emphasized print mode

ASCII	[ESC] F
Hexadecimal	1BH 46H
Decimal	<27> <70>
IPCL	&%CM
EPOS	[ESC] E <0>

Description The [ESC] F command cancels emphasized print mode.

[ESC] S <0> Select superscript

ASCII	[ESC] S <0>
Hexadecimal	1BH 53H 00H
Decimal	<27> <83> <0>
IPCL	&%SP
EPOS	none

Description The [ESC] S <0> command selects superscript. The following characters are printed half size on the upper side of the print line.

Note: Superscript is not available in all print modes.

[ESC] S <1> Select subscript

ASCII	[ESC] S <1>
Hexadecimal	1BH 53H 01H
Decimal	<27> <83> <1>
IPCL	&%SB
EPOS	none

Description The [ESC] S <1> command selects subscript. The following characters are printed half size on the bottom side of the print line.

Note: Subscript is not available in all print modes.

[ESC] T End superscript or subscript

ASCII	[ESC] T
Hexadecimal	1BH 54H
Decimal	<27> <84>
IPCL	&%SE
EPOS	none

Description The [ESC] T command ends superscript or subscript.

[ESC] % G Begin italics

ASCII	[ESC] % G
Hexadecimal	1BH 25H 47H
Decimal	<27> <37> <71>
IPCL	&%MI
EPOS	[ESC] 4

Description The [ESC] % G command begins italic print mode.

Note: Italics are not available in all print modes.

[ESC] % H End italics

ASCII	[ESC] % H
Hexadecimal	1BH 25H 48H
Decimal	<27> <37> <72>
IPCL	&%CI
EPOS	[ESC] 5

Description The [ESC] % H command ends italic print mode.

Page Mode

The EPIC 3000i supports two operational modes, standard and page mode. In standard mode, as a line of text is received it is buffered and printed when the line feed is received. In page mode the printer waits for a complete "page" (a number of lines) to be received before printing the complete page. The advantage of page mode is that text and/or graphics can be placed anywhere on the page, in any order, and in any of 4 orientations.

How to use page mode

Page mode requires two phases to operate correctly.

1. Page definition
    a. Define the master page size either just before or just after entering page mode.
    b. Optionally define a sub page.
       The master page defines the maximum page size, all sub pages must be smaller and contained within the master page. Master and sub page definitions are always done base on the 0 degree orientation not the current rotation.
    c. Optionally set an orientation. This may be 0, 90, 180 or 270 degrees.
    d. Optionally set the entry position. This is based on the current sub page and the current rotation.
    e. Enter text or graphics.
    f. Go to step b to define additional sub pages or step c to change the orientation.
2. Print the page.

Page Definition

The [ESC]t command will start page definition and define the initial orientation. An [ESC]t command during page definition will change the orientation and reset the entry location back to the top left corner of that orientation.

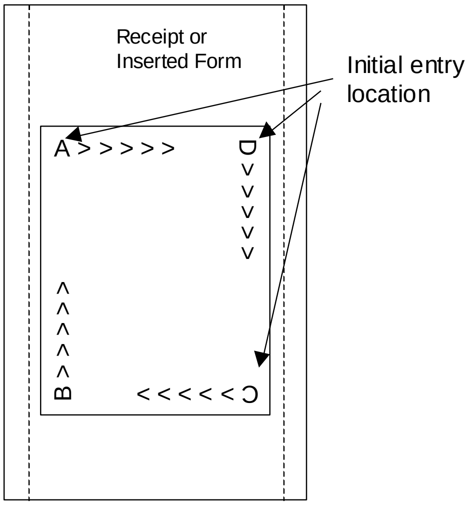


Figure 1 Page Mode Entry Orientations

The [ESC] u command wet the maximum page dimensions. Note that these dimensions are always based on 0 degree rotation.
Enter text and/or graphics as required.

Auto-cutter and page mode
You may embed an auto-cutter command with in a page definition. The auto-cut command may be placed anywhere in the page definition, however, it will be processed after the page is printed. To prevent the cut from occurring in the page, it will be preceded with a feed that will place the end of the page about 0.125 inches above the auto-cut position.

Mechanism commands in page mode
In general, mechanism commands received during page mode will be processed if the result will not affect the printed result.

Stopping page mode definition
The following operations will stop a page mode definition:

1) [ESC]@ Printer initialize command.
2) Real time reset request [ENQ]<10>
3) Turning the printer off.

Printing the page
The [FF] command starts the printing process.

Printing starts at the current paper position. The complete page definition is printed excluding any blank information at the bottom of the page.

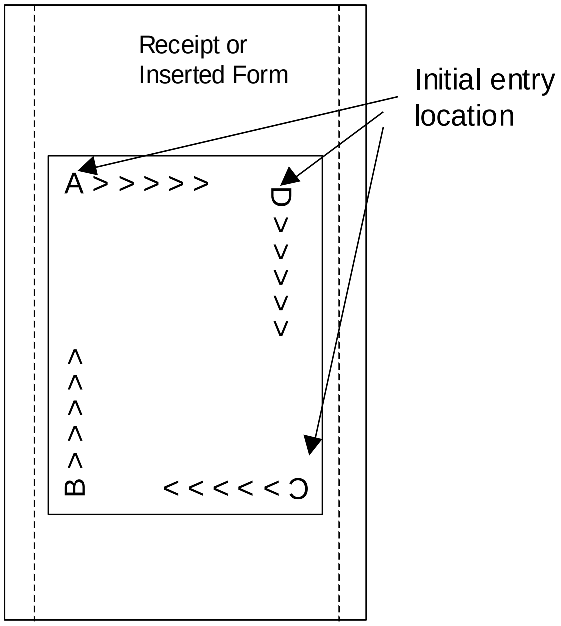
If the [FF] command is used to print the page, the memory used to store the page image is not maintained and is released to be used by other functions.

Page mode commands

Function	**Select page mode**	
ASCII	[ESC] t<x>	or [ESC] [SUB] t
Hexadecimal	1BH 74H<x>	or 1BH 1AH 74H
Decimal	<27><116><x>	or <27><26><116>
IPCL	&%PM<x_1>	
EPOS	[ESC] L	
Description	This command activates page mode and sets the orientation.	

This command may also be issued during page definition. When issued in page mode it resets the orientation and entry position, but does not cause the currently defined image to be erased or printed.

Where x = 0 for standard orientation (Direction A).
x = 1 for 270° Rotation (Direction B)
x = 2 for 180° Rotation (Direction C)
x = 3 for 90° Rotation (Direction D)



Note 1: This command saves the current right and left margin and sets them to the maximum values for the orientation currently defined.
Note 2: During page mode definition almost all printer commands are active. The following table lists the exceptions.

Command	Active	Action
Cash drawer commands	Yes	Immediate action
Bell command	Yes	Immediate action
Auto cut commands	Delayed	The printer will perform a feed to cut and then operate the auto cutter after the page is printed. The cut command may be anywhere in the definition. If the page is printed twice, the Auto cut command must be reissued after the first print to generate additional cuts.
Electronic journal station select	No	This command is like a station select and is not active. You cannot store page mode images in the electronic journal. A page mode command in journal station mode will exit journal mode.
Electronic journal entries	Yes	You can make journal entries as part of a page description. They are saved as text and not part of the page.
Status requests	Yes	The status is returned during definition.
Printer initialize command	Yes	Cancels page mode definition and returns the printer to standard mode.
Set top of form	No	You must set the top of form outside of page mode. The form position is maintained after the page is printed.
Vertical tab	No	The definition of a vertical tab is ambiguous in page mode and is ignored.
Macro definitions	No	You can not define or delete a macro while in page mode. You can, however, invoke a macro or stored graphic. You can not enter page mode while in a macro definition.

Note 3: Unless specified by a page mode set page size command, the default page size is the full paper width for about 6 inches. Printing starts at the current paper position. The complete page definition is printed

excluding any blank information at the bottom of the page.

Function **Set Print Area in Page Mode** **Enhanced**

ASCII [ESC] [SUB] S <XO$_L$><XO$_H$><<YO$_L$><YO$_H$><W$_L$><W$_H$>< H$_L$><H$_H$>
Hexadecimal 1BH 1AH 53H <XO$_L$><XO$_H$><<YO$_L$><YO$_H$><W$_L$><W$_H$>< H$_L$><H$_H$>
Decimal <27><26><83><XO$_L$><XO$_H$><<YO$_L$><YO$_H$><W$_L$><W$_H$>< H$_L$><H$_H$>
IPCL **None**
EPOS [ESC] W
Description This command Sets the position and size of the initial area in page mode and sub pages.
Where:

<XO$_L$><XO$_H$> the x direction offset	Min 0
<YO$_L$><YO$_H$> the y direction offset	Min 0
<W$_L$><W$_H$> the width in dots	Max 540
< H$_L$><H$_H$> the height in dots	Max 6000

This command should always be sent before or immediately after page mode is entered to define the initial page size.

Once in page mode (after the [ESC]t command) the command can be sent multiple times so that several different print areas, aligned in different print directions, can be developed in the printer's page buffer before being printed using the Print Page Mode commands
(FF or ESC FF).

The starting position of the print area is the upper left of the area to be printed. The Yoffset is in the y direction and is YO dots and the Xoffset is in the x direction and is XO dots in. The length of the area to be printed in the y direction is set to H dots. The length of the area to be printed in the x direction is set to W dots.

The set print area command may be invoked multiple times while in page mode. The first invocation specifies the initial master page size. Following invocations will define smaller windows within the initially defined page. If the Set page size command is not used, the page size will default to the maximum size.

The orientation of the set print area command is always based on 0 degrees regardless of the current orientation setting. This includes if the command is sent before the start page mode command, after the start page mode command, or as a sub-page.

Note: The maximum printable area in the x direction is 576/203 or 3.15 inches.
Note: The maximum printable area in the y direction is 2999/203 or 14.78 inches.
Note: Only the used portion of the page is printed. That is the page length will only include what actually has print data. See illustration below.

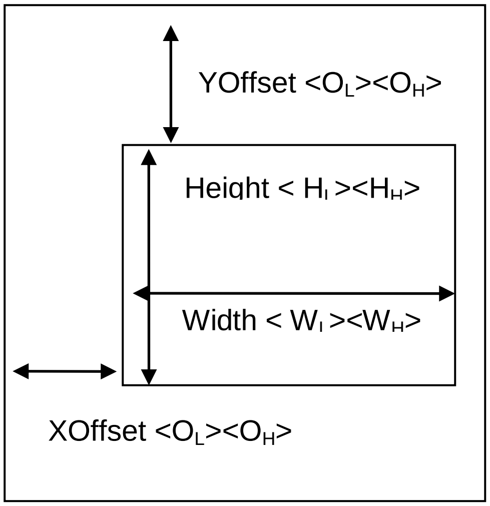


Figure 2 Page mode set printable area

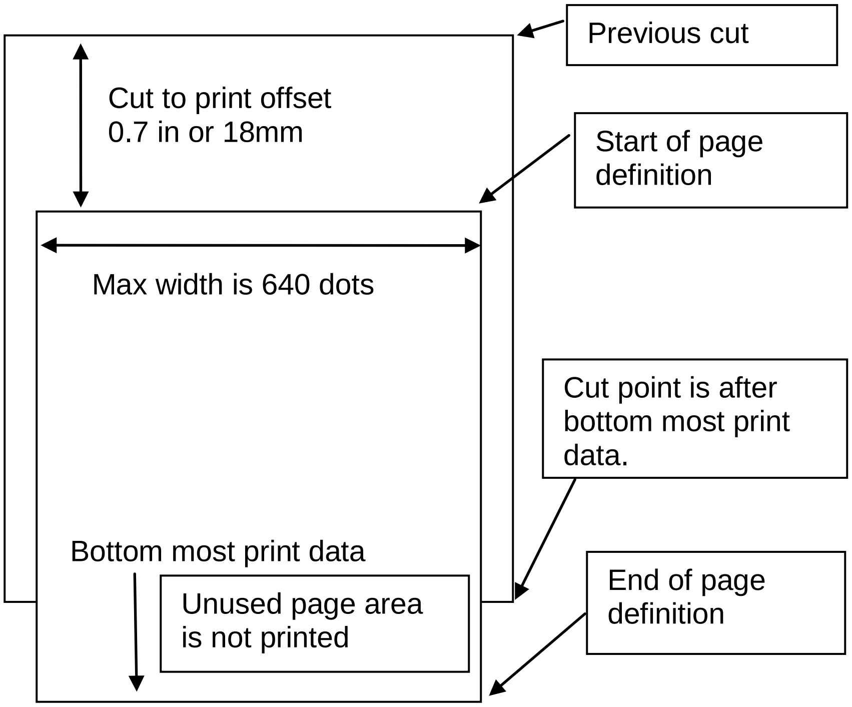


Figure 3 Default Page mode printed area

Function **Set Printed Area in Page Mode** **Enhanced**

ASCII [ESC] [SUB] W <XO$_L$><XO$_H$><<YO$_L$><YO$_H$><W$_L$><W$_H$>< H$_L$><H$_H$>
Hexadecimal 1BH 1AH 57H <XO$_L$><XO$_H$><<YO$_L$><YO$_H$><W$_L$><W$_H$>< H$_L$><H$_H$>
Decimal <27><26><87><XO$_L$><XO$_H$><<YO$_L$><YO$_H$><W$_L$><W$_H$>< H$_L$><H$_H$>
IPCL **None**
EPOS [ESC] W

Description This command is similar Set Print Area in Page Mode command above, however it will force the complete page to be printer rather than only what is used. It will define sub pages, however is intended to set the initial page size.
Where:

<XO$_L$><XO$_H$> the x direction offset	Min 0
<YO$_L$><YO$_H$> the y direction offset	Min 0
<W$_L$><W$_H$> the width in dots	Max(640)
< H$_L$><H$_H$> the height in dots	Max(3000)

This command should always be sent before or immediately after select page mode command and will define the initial page size. This command differs from the Set Print Area in Page Mode command in that it does not allow the page to be shortened. The complete page is printed even if it is not used.

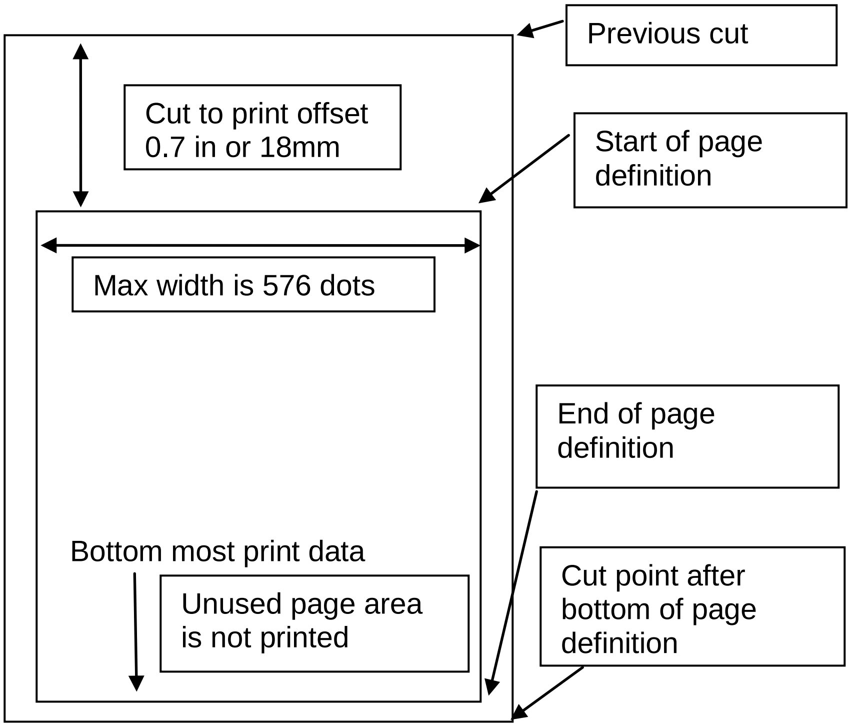

Previous cut
Cut to print offset 0.7 in or 18mm
Start of page definition
Max width is 576 dots
End of page definition
Bottom most print data
Unused page area is not printed
Cut point after bottom of page definition

Figure 4 Defined Page mode printed area

Function	**Set Print Area in Page Mode**	**Legacy Support Command**
ASCII	[ESC] u <O_L><O_H><W_L><W_H>< H_L><H_H>	
Hexadecimal	1BH 75H <O_L><O_H><W_L><W_H>< H_L><H_H>	
Decimal	<27><117><O_L><O_H><W_L><W_H>< H_L><H_H>	
IPCL	&%PS<0000><WWWW><HHHH>	
EPOS	[ESC] W	

Description This command Sets the position and size of the printing initial area in page mode and sub pages.

Where:

<O_L><O_H> the y direction offset	Default(0)
<W_L><W_H> the width in dots	Default(640)
< H_L><H_H> the height in dots	Default(6000)

This command should always be sent before or immediately after page mode is entered to define the initial page size.

Once in page mode (after the [ESC]t command) the command can be sent multiple times so that several different print areas, aligned in different print directions, can be developed in the printer's page buffer before being printed using the Print Page Mode commands
(FF or ESC FF).

The starting position of the print area is the upper left of the area to be printed. The offset is in the x direction and is O dots. The length of the area to be printed in the y direction is set to H dots. The length of the area to be printed in the x direction is set to W dots. (Note that the Y offset is always 0)

The set print area command may be invoked multiple times while in page mode. The first invocation specifies the final page height. Following invocations will define smaller windows within the initially defined page. If the Set page size command is not used, the page size will default to the maximum size.

The orientation of the set print area command is always based on 0 degrees regardless of the current orientation setting. This includes if the command is sent before the start page mode command, after the start page mode command, or as a sub-page.

Note: The maximum printable area in the x direction is 576/203 or 3.15 inches.
Note: The maximum printable area in the y direction is 5999/203 or 29 inches.
Note: The printed page length will only include what actually has print data.

Function **Set Page Mode Entry Position**

ASCII [ESC] [SUB] A <X_L><X_H><Y_L><Y_H>

Hexadecimal 1BH 1AH 41H<X_L><X_H><Y_L><Y_H>

Decimal <27><26><65><X_L><X_H> <Y_L><Y_H>

IPCL &%PY<XXXX><YYYY>

EPOS [ESC] W

Description This command sets the horizontal and vertical entry position to anywhere on the page. It is only valid in page mode. If the value specified is beyond the page boundary, the command is ignored. (X and Y refers to the current active orientation specified by the [ESC]t command, and is not same as the page definition of X and Y.)

The X and Y positions are in dots.
Y = Y_h* 256 + Y_l dots from the top
X = X_h* 256 + X_l dots from the left

Note: You can also use the [ESC]J, [ESC]M. [ESC]d and [ESC]e commands to position the print on the page.

Function **Set Page Mode Entry Position Relative**

ASCII [ESC] [SUB] R <X_L><X_H><Y_L><Y_H>

Hexadecimal 1BH 1AH 52H <X_L><X_H><Y_L><Y_H>

Decimal <27><26><82><X_L><X_H> <Y_L><Y_H>

IPCL None

EPOS [ESC] W

Description This command sets the horizontal and vertical entry position to anywhere on the page. It is only valid in page mode. If the value specified is beyond the page boundary, the command is ignored. (X and Y refers to the current active orientation specified by the [ESC]t command, and is not same as the page definition of X and Y.)

The X and Y positions are in dots.
Y = Y_h* 256 + Y_l dots from the current Y
X = X_h* 256 + X_l dots from the current X

Note: You can also use the [ESC]J, [ESC]M. [ESC]d and [ESC]e commands to position the print on the page.

Function	**Set Page Mode Entry Position**	**Legacy Support**
ASCII	[ESC] o <X_L><X_H><Y_L><Y_H><F>	
Hexadecimal	1BH 6FH <X_L><X_H><Y_L><Y_H><F>	
Decimal	<27><111><X_L><X_H> <Y_L><Y_H><F>	
IPCL	None	
EPOS	[ESC] W	

Description This command sets the horizontal and vertical entry position to anywhere on the page. It is only valid in page mode. If the value specified is beyond the page boundary, the command is ignored. (X and Y refers to the current active orientation specified by the [ESC]t command, and is not same as the page definition of X and Y.)

The X and Y positions are in dots.
Y = Y_h* 256 + Y_l dots from the top
X = X_h* 256 + X_l dots from the left

The F parameter is a flag that specifies if this command is an absolute position command or relative to the current position. If its value is <1> the command is processed as a relative position command, and if its value is <0> the command is interpreted as an absolute position command.

Note: You can also use the [ESC]J, [ESC]M. [ESC]d and [ESC]e commands to position the print on the page.

Function	**Exit Page Mode**	
ASCII	[FF]	or [ESC][SUB] P
Hexadecimal	0CH	or 1BH 1AH 50H
Decimal	<12>	or <27><26><80>
IPCL	&%FF	
EPOS	[FF]	

Description This command exits page mode definition and starts the print process. If the printer is not in page mode, this command is treated as a normal form feed command.

Note: When page mode finishes printing, the left and right margins are restored to the values before the select page mode command. All other format changes are preserved.

Graphic Modes

The EPIC 3000 Printer conforms to the basic definition of IBM all-points-addressable (APA) graphic commands. It is not designed to print large quantities of graphical data. The printer only prints graphics that are 2.5 inches wide. At this time, there is not a graphics mode for Epson emulation.

The EPIC 3000 Printer always prints in one of the native resolutions of 104 x 96, 208 x 96, 104 x 192, or 208 x 192 dpi. To provide compatibility with the standard IBM APA resolutions, the printer internally modifies the graphics to print as expected. The printer converts the vertical resolution by altering the [ESC] J command (which is typically used for vertical spacing) and adjusting it so that horizontal passes touch as expected. The requested horizontal resolution is converted by data scaling. Because the vertical dpi of the printer is always greater, the resulting APA graphics printed on the EPIC 3000 Printer are slightly smaller than the same graphic printed on an impact printer. All of this is done transparently to the application; however, loss of resolution may result in some modes. If desired the [ESC] * <m> command can be used to select the native resolution.

Note 1: If the EPIC 3000 Printer is used with programs that convert text to graphics, the printer is slower than if the printer is sent ASCII text. The EPIC 3000 Printer is supported by a Windows' print driver that allows applications to select supported fonts.

Note 2: Generally, the horizontal graphic commands provide faster print than the APA graphic commands.

Standard APA Graphics

[ESC] K **Print single-density graphics (60h x 72v dpi)**

ASCII	[ESC] K <n_1> <n_2>
Hexadecimal	1BH 4BH <n_1> <n_2>
Decimal	<27> <75> <n_1> <n_2>
IPCL	none

Description The [ESC] K <n_1> <n_2> command prints <n_1> + 256 * <n_2> bytes of single-density graphics (60 dpi).

[ESC] L **Print half-speed double-density graphics (120h x 72v dpi)**

ASCII	[ESC] L <n_1> <n_2>
Hexadecimal	1BH 4CH <n_1> <n_2>
Decimal	<27> <76> <n_1> <n_2>
IPCL	none

Description The [ESC] L <n_1> <n_2> command prints <n_1> + 256 * <n_2> bytes of double-density graphics (120 dpi) at half speed, allowing full and half dots to be printed.

[ESC] Y **Print full-speed double-density graphics (120h x 72v dpi)**

ASCII [ESC] Y <n_1> <n_2>
Hexadecimal 1BH 59H <n_1> <n_2>
Decimal <27> <89> <n_1> <n_2>
IPCL none

Description The [ESC] Y <n_1> <n_2> command prints <n_1> + 256 * <n_2> bytes of double-density graphics (120 dpi) at full speed with no consecutive dots. (The mode is generally used to print 120h by 144v dpi resolutions in two passes).

[ESC] Z Print quad-density graphics (240h x 72v dpi)

ASCII [ESC] Z <n_1> <n_2>
Hexadecimal 1BH 5AH <n_1> <n_2>
Decimal <27> <90> <n_1> <n_2>
IPCL none

Description The [ESC] Z <n_1> <n_2> command prints <n_1> + 256 * <n_2> bytes of quad-density graphics (240 dpi) at half speed with no consecutive dots. (The mode is generally used to print 240h by 144v dpi resolutions in two passes).

Extended APA Graphics

[ESC] * Print graphics in mode <m> (60h/ 120h/ 240h x 72v dpi)

ASCII [ESC] * <m> <n_1> <n_2>
Hexadecimal 1BH 2AH <m> <n_1> <n_2>
Decimal <27> <42> <m> <n_1> <n_2>
IPCL none

Description The [ESC] * <m> <n_1> <n_2> command selects one of three graphic modes as specified by <m>.

Where <m>	0	60 dpi	Full speed	8-bit slices
	1	120 dpi	Half speed	8-bit slices
	2	120 dpi	Full speed	8-bit slices
	3	240 dpi	Full speed	8-bit slices
	4	80 dpi	Full speed	8-bit slices
	5	72 dpi	Full speed	8-bit slices
	6	90 dpi	Full speed	8-bit slices
	7	144 dpi	Full speed	8-bit slices
	8,9	Not supported		
	10	104 x 96 dpi	1 horizontal 1 vertical pass	
	11	208 x 96 dpi	2 horizontal 1 vertical pass	
	12	104 x 192 dpi	1 horizontal 2 vertical passes	
	13	208 x 192 dpi	2 horizontal 2 vertical passes	
	15,16	Not supported		

[ESC] ? Reassign graphic mode

ASCII [ESC] ? <m> <n>
Hexadecimal 1BH 3FH <m> <n>
Decimal <27> <63> <m> <n>

IPCL none

Description The [ESC] ? <m> <n> command reassigns graphic mode <m> to resolution <n>. Possible values for <m> are K, L, Y, or Z. Resolutions, <n>, are zero to seven as follows:

Where <m>	0	60 dpi	Full speed	8-bit slices	Default for K
	1	120 dpi	Half speed	8-bit slices	Default for L
	2	120 dpi	Full speed	8-bit slices	Default for Y
	3	240 dpi	Full speed	8-bit slices	Default for Z
	4	80 dpi	Full speed	8-bit slices	
	5	72 dpi	Full speed	8-bit slices	
	6	90 dpi	Full speed	8-bit slices	
	7	144 dpi	Full speed	8-bit slices	
	10	104 x 96 dpi	1 horizontal, 1 vertical pass		
	11	208 x 96 dpi	2 horizontal, 1 vertical pass		
	12	104 x 192 dpi	1 horizontal, 2 vertical passes		
	13	208 x 192 dpi	2 horizontal, 2 vertical passes		

Note: Modes 11 through 13 are designed to support horizontal graphics and not intended for APA graphics.

[ESC] U <1> Select unidirectional print

ASCII [ESC] U <1>
Hexadecimal 1BH 55H 01H
Decimal <27> <85> <1>
IPCL &%GU
EPOS ESC] U <1>

Description The [ESC] U <1> command prints all data in unidirectional print mode to improve line to line registration for graphical data.

Note: Unidirectional print should be canceled before normal text is printed. Print time is slowed if it is not canceled.

[ESC] U <0> Select bidirectional print

ASCII [ESC] U <0>
Hexadecimal 1BH 55H 00H
Decimal <27> <85> <0>
IPCL &%GB
EPOS [ESC] U <0>

Description The [ESC] U <0> command prints all data in bi-directional, logic-seeking print mode. (This command has no affect on the printer but is provided for compatibility with impact and inkjet printers.

Horizontal Graphics

The EPIC 3000 Printer supports graphics sent as horizontal scan lines. Individual scan lines of graphic data are sent to the printer one line at a time. Although the EPIC 3000 Printer only supports two colors (black and white), the horizontal graphic command interface gives color or gray scale support for printer graphics. Color support is provided to establish a full color standard for future printers. Color data is sent in one of three color planes. Typically, a red plane or scan line is sent, then green and blue. Gray scale support is provided by using the red, green and blue as gray levels. The sequence of lines defines one row of dots that is printed on the paper.

The horizontal graphic commands do not include resolution information. Therefore, only once before sending graphics data, set the graphics resolution by sending the [ESC] * command with a zero length (no data). The graphic resolution sets the internal graphic mode of the printer. The printer stays in graphic mode until it is changed by another command. Note that the bar code generation and other graphic commands change graphics mode. The format of the horizontal graphic command follows.

[ESC] h Process horizontal graphics data

ASCII	[ESC] h <color> <length> <format> <data>
Hexadecimal	1BH 68H
Decimal	<27> <104>
IPCL	None
EPOS	None

Description The [ESC] h <color> <length> <format> <data> command processes horizontal graphic data, where <color> is a byte that specifies the color of the data being sent.

<color> =	0 Use Previously Selected Color		
	1 Red		Gray
	2 Green	or	Light Gray
	4 Blue		Dark Gray

Note 1: If the printer is configured for Gray scale paper, this command is used for gray scale graphics.
Note 1: Red, green, and blue pixels set to one at the same location result in a white dot, while red, green, and blue pixels set to zero form a black dot. For black print, one represents a black dot and zero represents a white dot.
Note 2: More than one color may be set at a time, for example, setting the color to six would set green and blue simultaneously

<length> = byte specifying the length of the data including the format byte, ranging from 0 to 254 (255 is reserved for future use.)
<format> = byte specifying the format of the graphics data.
0 for raw data
1 for bit wise RLE compression
8 for byte wise RLE compression

254 for difference compression
255 for same as previous scan line data
<data> = the data bytes that define the graphics to be printed.

[ESC] * **Set horizontal graphic mode**

ASCII	[ESC] * <m> <0> <0>
Hexadecimal	1BH 2AH <m> <0> <0>
Decimal	<27> <42> <m> <0> <0>
IPCL	none

Description The [ESC] * <m> <0> <0> command selects one of the three graphic modes specified by <m>. The two bytes after the mode must be zero.

Where <m>			
0,2,3,4,5,6,7	Standard Graphic Modes (See ESC * command documentation in previous section.)		
8,9	Not supported		
10	102 x 102dpi	1 horizontal, 1 vertical pass	
11	203 x 102 dpi	2 horizontal, 1 vertical pass	
12	102 x 203 dpi	1 horizontal, 2 vertical passes	
13	203 x 203 dpi	2 horizontal, 2 vertical passes	
14,15,16	Not supported		

Note 1: Only modes 10 thorough 13 should be selected for horizontal graphics.

Example ***Command***	***Comment***
[ESC] * <10> <0> <0>	Set resolution to 100x96 dpi.
[ESC] h <1> <9> <0> <eight data bytes>	Send 8-bytes red pixels.
[ESC] h <2> <9> <0> <eight data bytes>	Send 8-bytes green pixels.
[ESC] h <3> <9> <0> <eight data bytes>	Send 8-bytes blue pixels.
LF	Send line feed to force print of any buffered data not yet printed.

Note 2: Graphic data is committed to paper when more than 12 dot rows have been transmitted to the printer. If less than 12 dot rows have been sent, they are not printed until the line is terminated (i.e. a line feed command is sent). To make graphics faster to send and smaller to store, several algorithms are included with the graphic command to compress the data.

Graphics Compression
Although the printer compression algorithms are documented, it is recommended that our Windows' printer driver be used to generate a graphic image. Our Windows' printer driver selects the best compression method to use on a scan line by scan line basis. The print driver can be directed to print to file, creating a .prn file. When creating a .prn file, it is recommended that the Start/End Doc settings be cleared in the Start/End Doc tab of the printer properties page. After the .prn file is created, it can be read and sent to the printer by the host application.

Bit wise RLE. In bit wise RLE compression, the Most Significant Bit (MSB) compression of each data byte denotes if the compressed data represents one or zero bits. Bits zero through six indicate how many bits are represented as a one or zero. A 34 Hex (34H) represents 34H bits set to zero. A 97H represents 17H bits set to one.

[ESC] h <1> <5> <1> <34H> <97H> <8fH> <09H>

Byte wise RLE. In byte wise RLE compression, data is represented in byte Compression pairs. The first byte is a count, and the second is the graphics data. The graphics data byte is repeated the number of times represented by the count byte.

[ESC] h <1> <5> <8> <09H> <ffH> <02H> <55H>
Where <09H> <ffH> means repeat ffH nine times and <02H> <55H> means repeat 55H two times.

Difference Compression. In difference compression, data is represented in byte pairs. In compression, the first byte is an index into the byte stream, as it would exist if sent in an uncompressed format. The second byte is the data that is different in the new scan line data. Think of compression mode as, "The scan line is the same as the previous except for the byte at a specific position."

[ESC] h <1> <5> <254> <03H> <d5H> <0bH> <51H>
Where <03H> <d5H> means use the previously transmitted scan line data but change byte 3 to a d5H and change byte 11 (0bH) to a 51H.

Same-as-previous Compression. In same-as-previous compression, the command does not contain any graphics data. The command specifies that the printer is to use the previous scan line data for the current scan line.

[ESC] h <1> <1> <255>.

Bitmap Graphics File Support

The EPIC 3000 printer supports direct printing of Monochrome, 4 bit-16 color, 8bit-256 color, 24bit True color bitmap files. The image may be directly printed or saved temporarily and scaled at 1 to 1 or 2 to 1. Printing in Monochrome, two color or gray scale is supported based on paper selection.

Color bitmaps are converted to a grayscale representation of its luminance by adding together 30% of the red value, 60% of the green value, and 10% of the blue value. These weights are predefined in the printer and are close to the industry standard 30%, 59% and 11%. Two color print is based on the luminance with color weighting of the selected paper color. Grayscale print is based only on the luminance value.

Note: Where as the printer will process and print an 8 bit or 24 bit color image, the actual print will be Monochrome, two color or grayscale. The printer will translate the color image based on its own rules so the resulting image may not be as you intended. In addition, the amount of data in an 8 or 24 bit color bitmap is extensive. The time required to transfer the image will be much longer than the same monochrome image. You are much better off converting the image to Monochrome or 16 colors within your application.

[ESC] [FS] Print Bitmap File Record

ASCII	[ESC] [FS] <Bitmap file data>
Hexadecimal	1BH 1CH
Decimal	<27> <28>
IPCL	None
EPOS	None

Description The [ESC][FS] command is actually a group of commands intended to print graphics. All bit map files begin with "BM" so when the bitmap data is sent after the [ESC][FS], the command is really [ESC][FS] B, This command accepts Monochrome and 16 color bitmap files and saves it as a temporary RAM file. The image may then be printed with the [ESC][FS]p command or saved in the file system with and [ESC][FS]S command.

Note: A Bitmap graphic file may also be written to the EPIC 3000's file system using the file system commands. It can then be printed by the [ESC][FS]P command. In this case, it is not saved in the temporary file so the [ESC][FS]p does not work.

[ESC] [FS] p Print bitmap image buffer.

ASCII	[ESC] [FS] p<Scale>
Hexadecimal	1BH 1CH 70H
Decimal	<27> <28><112>
IPCL	None

EPOS	None

Description The [ESC][FS] p command prints a bitmap image in the temporary buffer.

Where Scale 0 = one to one.
1 = twice the width
2 = Twice the height
3 = Twice the height and width.

The intent of this command is to allow a bitmap file to be loaded into the EPIC 3000 and printed scaled up to 2 to 1. Use the [ESC] [FS] <Bitmap file data> command to load the bitmap image and the [ESC][FS] p to print it.

Note: The temporary buffer is also used for various other commands. If the data in the buffer is not a bitmap graphic, it won't be printed by this command.

[ESC] [FS] P Print Bitmap File Print

ASCII	[ESC] [FS] P<Bitmap file data>
Hexadecimal	1BH 1CH 50H
Decimal	<27> <28><80>
IPCL	None
EPOS	None

Description The [ESC][FS] P command accepts Monochrome, 16 color, 256 color and 24bit color bitmap files and prints them immediately bit for bit with no scaling.

[ESC] [FS] P Print Saved Bitmap File Print

ASCII	[ESC] [FS] P <Scale><File Name><0>
Hexadecimal	1BH 1CH 50H <Scale><File Name><0>
Decimal	<27> <28><80><Scale><File Name><0>
IPCL	None
EPOS	None

Description The [ESC][FS] P command followed by a value that is not 'B' selects a scale, retrieves a graphic file named in the File Name field and prints it at the selected scale. This graphic file must previously have been defined and saved by the [ESC][FS] command and the [ESC][FS]S command or by writing a bitmap file to the file system with file system commands.

Where Scale 0 = one to one.
1 = twice the width
2 = Twice the height
3 = Twice the height and width.

Note: The [ESC][FS] P command looks for graphic files defined by the [ESC][FS] S command first. If the EPIC 3000 cannot find a .bgp command, it will search for a .bmp file. If there are two files with the same root name, the .bgp file will be processed.

[ESC] [FS] S Save Bitmap File Print

ASCII	[ESC] [FS] S <File Name><0>
Hexadecimal	1BH 1CH 53H <File Name><0>
Decimal	<27> <28><83> ><File Name><0>
IPCL	None
EPOS	None

Description The [ESC][FS] S accepts a file name and saves the previously defined bitmap file in the RAM file system to the Flash file system. If the file already exists, the existing file will be erased.

Note: The [ESC][FS] S command erases the RAM file so the [ESC][FS] p will no longer print the saved image.
Note: This command followed by a zero length file name will flush the stored graphic image.

User Store (Graphic Save)

The EPIC 3000 Printer maintains a 16K (16384 bytes) section of flash memory and up to 192K of extended flash memory to save user information. The information can be either macros or user-defined characters. These groups of data are indexed by name, and may be called up at any time after they are stored. See the sections on Macros and User-defined Characters for definitions of these functions.

To allow the host application to maintain these groups of data, a series of user store maintenance commands are available. As referenced earlier in this manual, the user can define a limited number of custom characters and define a macro. These character/macro definitions can also be saved in user store. However, only one character definition and one macro can be active at any time. One macro and one user-defined character definition can be flagged to load and run at startup. If a flag is set, the printer will automatically process the macro and/or load the user-defined character set at initialization.

Because user store is intended to be loaded only a few times and then printed as part of normal operation, the programmer must take some care during the definition phase. The programmer must assume the responsibility to assure the 16K buffer size is not exceeded. User store can save macros and user-defined character sets.

When the basic user store is full, it can be moved to extended user store. Individual items in the extended user store can not be erased. The entire extended user store must be erased all at once. You may place two items in user store with the same name and the last defined item will be used.

Defining Macros

Macros can be defined two ways. The first is by using the begin and end named macro commands. These commands start the recording process and automatically save the macro when it is complete. The macro data is not processed, as it is sent to the printer.

Function	**Begin named macro record**
ASCII	[ESC] [US] b <Name..> <0> Then send the data to be recorded. The printer does not process the data. The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function	**End name macro record**
ASCII	[ESC] [US] e <Name..> <0> The second way to define macros is to use [ESC] g commands to define the macro, and then the save macro data command to save the data. The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function	**Start macro record**
ASCII	[ESC] g <1>Then send the data to be recorded. (The data is processed and printed).

Function	**Stop macro record**
ASCII	[ESC] g <2>Then save the macro.

Function	**Save macro data**
ASCII	[ESC] [US] m <Name..><0> Saving User-defined Characters. To save user-defined characters, first define the character set.

Function	**Define user-defined characters**
ASCII	[ESC] = <y> <c_1> <c_2> [x_1 d_1 … d(y *x* x_1)] … [xk d_1 … d(y *x* xk)] Second, save the definition in the nonvolatile flash memory with the appropriate command. Save the definition. Note the "Save user-defined characters" command saves all three character definitions.

Function	**Save user-defined characters**
ASCII	[ESC] [US] c <Name..> <0> Third, load the character set or load and run the macro. To restore the character definitions, issue a load item command with the name of the character set to be loaded. The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function **Load item from user store**

ASCII [ESC] [US] l <Name..> <0>

If the item referenced is a user-defined character set, it is loaded into the current definition. If it is a macro, it is loaded into the macro buffer. It is not processed or printed.

To help maintain the user-store area, the following commands can be used.

The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function **Flag as a start-up macro**

ASCII [ESC] [US] s <Name..> <0>

The [ESC] [US] s <Name..> <0> command flags the referenced item to be processed at startup. No more than one user character definition and user data item may be flagged.

The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function **Remove item from user store**

ASCII [ESC] [US] e <Name..> <0>

The [ESC] [US] e <Name..> <0> command removes an item from user store and frees up its space. The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function **Flush information from user store**

ASCII [ESC] [US] f ALL <0> Base User Store

or [ESC] [US] f EXT <0> Extended User Store.

Description The [ESC] [US] f ALL <0> command clears all of the information to the user store and frees the data space. The [ESC] [US] f EXT <0> command clears all of the information in the extended the user store.

The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function **Report on user store**

ASCII ESC] [US] q <0>

Description The [ESC] [US] q <0> prints or returns information about the contents of and available space in user store.

Note: A configuration option is available that locks the user store data. The configuration option prevents the occurrence of new user store data operation until the lock is manually reset and accidental deletion of the saved information. The user-defined character buffer and/or user data buffer may be redefined and used but cannot be stored.

The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

User-Store Commands

Function	**Begin named macro record**
ASCII	[ESC] [US] b <Name..> <0>
Hexadecimal	1BH 1FH 62H
Decimal	<27> <31> <98>
IPCL	&%UB <Name..> <0>
EPOS	none

Description The [ESC] [US] b <Name..> <0> command erases the current macro, initializes the macro buffer structure, and redirects the following data to the macro buffer. It uses the <Name..> field as a reference. If the name already exists in the flash user store, the command is ignored. The command must be followed by the "End name macro record" command with the same name. If the data that follows is larger than the macro buffer (about 16K), the macro definition is terminated without saving any data.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function	**End named macro record**
ASCII	[ESC] [US] e <Name..> <0>
Hexadecimal	1BH 1FH 65H
Decimal	<27> <31> <101>
IPCL	&%UG <Name..> <0>
EPOS	none

Description The [ESC] [US] e <Name..> <0> command ends the macro record operation and saves the macro to flash. It uses the <Name..> field to verify the command end and must match the "Begin named macro record" command. If the name already exists in the flash user store or the macro memory is exceeded, the command is valid, and the <Name..> field prints. If there is not enough room in the flash user store for the macro, the save is not performed, but the macro buffer is valid.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function	**Save macro data in user store**
ASCII	[ESC] [US] m <Name..> <0>
Hexadecimal	1BH 1FH 6DH
Decimal	<27> <31> <109>
IPCL	&%UM <Name..> <0>
EPOS	[GS] -…<Name..> <0> is from one to 15 characters and must be null terminated.

Description The [ESC] [US] m <Name..> <0> command saves the current macro buffer structure into the flash user-store area. It uses the <Name..> field as a reference name. If the name already exists in the flash user store, the command does not store the data.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function	**Save user-defined characters**
ASCII	[ESC] [US] c <Name..> <0>
Hexadecimal	1BH 1FH 63H
Decimal	<27> <31> <99>

IPCL &%UC <Name..><0>
EPOS [GS] 6**<**Name..> <0> is from one to 15 characters and must be null terminated.

Description The [ESC] [US] c <Name..> <0> command saves the current user-defined character structure in the flash user-save storage area. It uses the<Name..> field as a reference. If the name already exists in the flash user store, the command will not store the data.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function **Load item from user store**

ASCII [ESC] [US] l <Name..> <0>
Hexadecimal 1BH 1FH 6CH
Decimal <27> <31> <108>
IPCL &%UL <Name..> <0>
EPOS [GS] 0<Name..> <0> is from one to 15 characters and must be null terminated.

Description The [ESC] [US] l <Name..> <0> command loads the referenced item into the appropriate structure. If the item referenced is a user-defined character set, it is loaded into the current user-character definition, which does not affect the active state of user-defined characters. If it is a macro, it is loaded into the macro buffer, but it is not inserted into the data stream. [ESC] g <0> inserts the macro buffer into the data stream. If the named item does not exist, the command does nothing.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT>n> on page 88.

Function **Run macro data from user store**

ASCII [ESC] [US] r <Name..> <0>
Hexadecimal 1BH 1FH 72H
Decimal <27> <31> <114>
IPCL &%UR <Name..> <0>
EPOS [GS] 0**<**Name..> <0> is from one to 15 characters and must be null terminated.

Description The [ESC] [US] r <Name..> <0> command loads the referenced macro into the macro buffer. The macro buffer is then inserted into the data stream. If the named item does not exist or is not a macro, nothing happens.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Function **Flag item as a start-up macro**

ASCII [ESC] [US] s <Name..> <0>
Hexadecimal 1BH 1FH 73H
Decimal <27> <31> <115>
IPCL &%US <Name..> <0>
EPOS [GS] 0**<**Name..> <0> is from one to 15 characters and must be null terminated.

Description The [ESC] [US] s <Name..> <0> command flags the referenced item to be processed at startup. Only one user character definition and one macro may be flagged to run at startup.

Note: If a character definition is loaded at startup, it is automatically made active.

The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page88.

Function **Delete item from user store**
ASCII [ESC] [US] d <Name..> <0>
Hexadecimal 1BH 1FH 64H
Decimal <27> <31> <100>
IPCL &%UD <Name..> <0>
EPOS [GS] 1 <Name..> <0> is from one to 15 characters and must be null terminated.

Description The [ESC] [US] d <Name..> <0> command removes an item from user store and frees up space. If the item does not exist, the command does nothing.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT>n> on page 88.

Function **Transfer all items from user store to extended user store.**
ASCII [ESC] [US] t <0>
Hexadecimal 1BH 1FH 74H
Decimal <27> <31> <116>
IPCL &%UX <0>
EPOS None

Description This command transfers the information in the base 16K user store to the extended user store. The base user store is erased if the transfer was successful.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT>n> on page 88.

Function **Flush information from user store or extended user store**
ASCII [ESC] [US] f ALL <0> User Store.
Or [ESC] [US] f EXT <0> Extended User Store.
Hexadecimal 1BH 1FH 66H 00H
Decimal <27> <31> <102> <0>
IPCL &%UF
EPOS [GS] 5

Description The [ESC] [US] f ALL <0> command clears all entries in user store and frees the data space. It must have the name, "ALL" (in uppercase) attached. If "EXT" is substituted for "ALL", extended user store (If any) is cleared.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT>n> on page 88.

Function **Report on user store**
ASCII [ESC] [US] q <0> Print a user store report
Or [ESC] [US] ? <0> Return a formatted user store report
Hexadecimal 1BH 1FH 71H

Decimal <27> <31> <113>
IPCL &%UQ <Name..> <0>
EPOS [GS] 3
Description The [ESC] [US] q <Name..> <0> command prints a status report. The file name is ignored and may be omitted. The NUL must be present. The intention of the command is to aid in macro development.
The terminating <0> may be replaced with an & or redefined. See [ESC] [EM]T<n> or &%UT<n> on page 88.

Note: The report is also printed as part of the configuration report.

Function **Redefine User Store Termination Character**
ASCII [ESC] [EM] T <n>
Hexadecimal 1BH 19H 54H <n>
Decimal <27> <25> <84> <n>
IPCL &%UT <n>
EPOS None
Description This command allows the terminator used to signal the end of the name field in User Store commands to be modified. The value of <n> is used (in addition to the <0>) for the terminator. The value of n may be from 0 to 255.
The default value for the second terminator is &. If this command redefines the terminator to something other than &, the & will no longer function.
Example If &%UT% were sent to the printer, the user store command to run macro "Demo" would be &%URDemo%.

User Macros

The user macro feature works by inserting the macro data buffer into the printer data stream when the print user-store data command is encountered.

Macros can be any data normally sent to the printer including graphics. (Note: user-store maintenance and inquire commands may not be included in the macro definitions.) The printer stores macro data in a RAM-based storage buffer as it is received and processed. The storage buffer may then be saved to a flash-based user store or inserted into the print data stream. If a macro is recalled from user store, it is expanded into the macro buffer and replaces whatever is currently there.

Programming Considerations

The flash (nonvolatile) memory has a limited number of write cycle operations. Consequently, the number of saves should be limited. The buffer should not be saved on a transaction by transaction basis, but rather a maximum of once per day.

The buffer is initially about 16K[8] bytes long. All commands[9] and print data are placed in the buffer and must be included in the size limits. The printer does not indicate when the buffer is full. The application must make sure that the buffer is not overfilled. The printer

[8] The actual buffer is smaller because of the overhead.

[9] IPCL commands are converted by the printer into an equivalent [ESC] code and then placed in the save buffer. The equivalent [ESC] code should be used to calculate the size of the save buffer data.

simply stops saving information when it is full. As the buffer fills, the input data is printed normally. The effect of the macro start command is to clear the buffer and to start to save the input data. The macro stop command stops saving data and initializes internal pointers for the next print. To store the data in the nonvolatile flash, it must be named and saved by one of the user-store save commands.

When the macro buffer is inserted into the data stream, configuration commands (like font or pitch changes) remain in effect after the macro is processed. Illegal commands are placed in the buffer and take up space.

Horizontal color graphics should be sent to the printer compressed. If the data is not compressed, it is saved in the macro buffer. If the buffer is saved into the user-store nonvolatile flash, there must be enough room in the user store for all of the data. As user-store space is used, the macro buffer will be larger than the available space in user store. Only the used space is saved, but it is possible to define a macro that does not fit in the remaining user-store space.

[ESC] g <0> Process user macro

ASCII	[ESC] g <0>
Hexadecimal	1BH 67H 00H
Decimal	<27> <103> <0>
IPCL	&%GP
EPOS	[ESC] g <0>

Description The [ESC] g <0> command prints the user-store data buffer.

[ESC] g <1> Start macro record

ASCII	[ESC] g <1>
Hexadecimal	1BH 67H 01H
Decimal	<27> <103> <1>
IPCL	&%GS
EPOS	ESC] g <1>

Description The [ESC] g <1> command clears the user-store data buffer and begins recording data. The next 2000 bytes (including characters and commands) are recorded.

[ESC] g <2> Stop macro record

ASCII	[ESC] g <2>
Hexadecimal	1BH 67H 02H
Decimal	<27> <103> <2>
IPCL	&%GE
EPOS	[ESC] g <2>

Description The [ESC] g <2> command stops recording user-store data information. The buffer is not saved into the nonvolatile memory.

[ESC] g <3> Stop macro record and save

ASCII	[ESC] g <3>
Hexadecimal	1BH 67H 03H
Decimal	<27> <103> <3>
IPCL	&%GW
EPOS	[ESC] g <3>

Description The [ESC] g <3> command stops recording graphic save information. The buffer is saved into the user-store nonvolatile memory under the name, "ESCg3_Save"

Note: The [ESC] g <3> command is supplied for compatibility with the Series 80PLUS and 90PLUS printers.

Bar Codes

The EPIC 3000 Printer supports the ability to print bar codes. The printer offers a number of formats as defined below. The host does not need to form the graphic image for these bar codes. The host need only send the printer the information to be bar coded and a graphic is generated by the printer. In some cases, a check character is required by the format. In most cases, the printer generates the check character and inserts it correctly in the format. The printer uses internal graphic modes to form bar code images, and the images are adjusted for ink bleed. In general, the bar codes generated by sending graphic data to the printer are not as readable as the bar codes the printer generates. Bar codes are printed at a 203 x 203 resolution.

Barcodes may be printed horizontally or vertically (using page mode).

Function	**Print bar code**	
ASCII	[ESC] b <n> {information} [NUL]	
Hexadecimal	1BH 62H <n> ... 03H	
Decimal	<27> <98> <n> ... <3>	
IPCL	&%25 ... [CR]	Interleaved 2 of 5
	&%39 ... [CR]	Code 39
	&%12 ... [CR]	Code 128
	&%28 .<m$_1$> <m$_2$>	Code 128 (allows a two character length, preceding the information)
	&%UP ... [CR]	UPC A
	&%UE ... [CR]	UPC E
	&%EA … [CR]	EAN-13
	&%E8 ... [CR]	EAN-8
	&%93 … [CR]	Code 93
	&%CB .. [CR]	Codabar

Description The [ESC] b <n> {Bar Data} [NUL] command prints information as a bar code. The bar data is terminated with an [ETX], [CR], [LF] or [NUL].

An alternate Format is provided if control characters can be part of the barcode data.
The [ESC] b <n><L$_L$> <L$_H$>{ Bar Data } command prints information as a bar code.
The <L$_L$> <L$_H$>parameters make up a 16 bit length of bar data as follows:
Length = (L$_H$ * 256) + L$_L$.

An alternate format is provided that allows only a single character length for some bar codes.
The [ESC] b <n><L$_L$>{ Bar Data } command prints information as a bar code.
The <L$_L$> parameters make up a 8 bit length of bar data as follows:
Length = L$_L$.

If a start code is to be specified , The following format may be required.
The [ESC] b <n><S>{ Bar Data } [NUL] command prints information as a bar code.
<S> indicates the start code and data up to the [NUL] is the bar data.

Where n =

0	Interleaved 2 of 5	Numeric (0-9) only; must be even number of digits	
1	Code 39	26 uppercase letters (A-Z); 10 digits (0-9)	
2	Code 128	Three sets of 106 different characters	
3	UPC A	Numeric (0-9) only; 11 digits	
4	EAN-13	Numeric (0-9) only; 12 digits	
5	UPC E	Numeric (0-9) only; 11 digits	
6	EAN-8	Numeric (0-9) only; 7 digits	
7	Code 93	26 letters; 10 digits (0-9); and 7 special Characters	
8	Codabar	10 digits (0-9); 4 start/stop characters, A, B, C, and D; and 6 special characters.	
9	PDF-417	2D alpha Numeric (16bit length) (See notes below)	
10	PDF-417	2D alpha Numeric (NUL term.) (See notes below)	
11	EAN-128	GS-1 code 128 Alpha Numeric	
12	EAN-14	Numeric (0-9) only 13 digits	
13	ITF-14	GS-1 Numeric (0-9) only 13 digits	
14	Code 49	2D alpha Numeric (NUL term.)	
15	Maxicode	2D alpha Numeric (16bit length) (See notes below)	
16	Maxicode	2D alpha Numeric (NUL term.) (See notes below)	
17	Code16K	2D alpha Numeric (NUL term.)	
18	RSS -14	See GS1 Databar barcode definitions below	
19	RSS -14 Truncated	“	
20	RSS Limited	“	
21	RSS Stacked	“	No HRI will be printed
22	RSS Stacked Omni	“	No HRI will be printed
23	RSS Expanded	“	
24	RSS Expanded Stacked	“	No HRI will be printed
25	QRCode	2D alpha Numeric (16bit length) (See notes below)	
26	QRCode	2D alpha Numeric (NUL term.) (See notes below)	
27	Data Matrix	2D alpha Numeric (16bit length) (See notes below)	
28	Data Matrix	2D alpha Numeric (NUL term.) (See notes below)	
29	Aztec	2D alpha Numeric (16bit length) (See notes below)	
30	Aztec	2D alpha Numeric (NUL term.) (See notes below)	
31	Aztec Rune	2D Numeric (NUL term.) (3 digits representing 0-255)	
32	Code One	2D alpha Numeric (NUL term.)	
33	MicroPDF-417	2D alpha Numeric (16bit length) (See notes below)	
34	MicroPDF-417	2D alpha Numeric (NUL term.) (See notes below)	
36	Micro QRCode	2D alpha Numeric (16bit length) (See notes below)	
37	Micro QRCode	2D alpha Numeric (NUL term.) (See notes below)	

COMPOSITE Barcodes

	Composite data	
64	Secondary Data	GS-1 Alpha Numeric (NUL term.)
	Linear Barcode data	
65	UPC A	Numeric (0-9) only; 11 digits
66	UPC E	Numeric (0-9) only; 11 digits
67	EAN-8,13,14	Numeric (0-9) only; 7,12, or 13 digits
68	EAN 128	GS-1 code 128 Alpha Numeric
69	RSS -14	See GS1 Databar barcode definitions below
70	RSS -14 Truncated	“
71	RSS Limited	“

72	RSS Expanded	"	
73	RSS Stacked	"	No HRI will be printed
74	RSS Stacked Omni	"	No HRI will be printed
75	RSS Expanded Stacked	"	No HRI will be printed

Note1: You may print barcodes in page mode. If you rotate these barcodes 90° or 180° you can get significantly longer barcodes.
Note2: A [CR], [LF] or [NUL] may also be used in place of the [NUL] to end the bar code data field.
Note 3: Only information that is usable in a particular bar code will be printed.

Barcodes and Code Pages

In ASCII mode the barcode data is not translated by the active code page. In general barcode data is numeric (0-9) or alpha numeric (0-9 and A-Z). Some barcodes support additional characters and are defined by the barcode specification.

Barcodes and Unicode

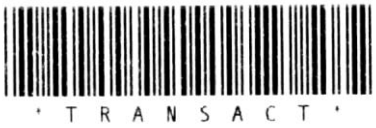
In Unicode mode, the printer can accept character values from 0 to 65535. In general barcode data is generally numeric (0-9) or alpha numeric (0-9 and A-Z) and are limited to values from 0-127. Some barcodes support additional characters as defined by the barcode specification, however, only character values from 0 to 255 will be used for barcodes. Any barcode character translations must be done by the host application. Depending on the Unicode mode, values from 128 to 255 may be required to be encoded in UTF before they are sent to the printer.

Note: Any values greater than 255 will be converted into a space (0x20).
Note: In UTF8 or UTF16 modes all data is assumed to be UTF encoded.

Linear Barcodes

Code-39 and Code-39 Extended

Function	**Code-39**
ASCII	[ESC] b <1> {information} [NUL]
Hexadecimal	1BH 62H <1> ... 03H
Decimal	<27> <98> <1> ... <3>

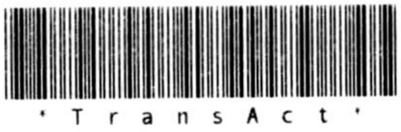


[ESC] b <1> TransAct[NUL]
Note: Case conversion to upper case.

Figure 5 Code 39 Example

Code 39 is an alphanumeric bar code. It is a discrete, self-checking, variable-length code. The printer prints the complete data field. The number of characters that can be printed depends on the bar width scaling. There are two modes of operation for the Code 39 barcodes. The first is in a variable length format. In this mode all characters sent to the printer will be printed up to the termination character. Only 0-9, A-Z -, period, and space may be printed. $, %, +, and / Characters may be used as escape characters for full 128 character support. If illegal characters are passed to the printer, they are converted to legal codes. (For example, a would be converted to A).

Function	**Code-39 Extended**
ASCII	[ESC] b <1><L>{information}
Hexadecimal	1BH 62H <1><L>{ ...
Decimal	<27> <98> <1><L>{ ...



[ESC] b <1><8>TransAct
Note: Extended character support however the barcode is much denser and harder to read.

Figure 6 Code 39 Extended Example

In full 128 character mode, the printer will encode the full 128 character set. In this mode, the first character received must be the length. IE. [ESC]b<1><L>… where L specifies the number of characters to follow and must be less than 32. The characters following the length may be from 0 to 127. Values greater than 127 are converted to printable characters by removing the 8th bit.

Note: In extended mode the printer will automatically convert extended characters to the character pair equivalent. The HRI will print the extended character not the character pair.

The following table specifies the Code 39 character set.

ASCII	Code	ASCII	Code	ASCII	Code	ASCII	Code
NUL	%U	SP	Space	@	%V	`	%W
SOH	$A	!	/A	A	A	a	+A
STX	$B	"	/B	B	B	b	+B
ETH	$C	#	/C	C	C	c	+C
EOT	$D	$	/D	D	D	d	+D
ENQ	$E	%	/E	E	E	e	+E
ACK	$F	&	/F	F	F	f	+F
BEL	$G	'	/G	G	G	g	+G
BS	$H	(	/H	H	H	h	+H
HT	$I	)	/I	I	I	i	+I
LF	$J	*	/J	J	J	j	+J
VT	$K	+	/K	K	K	k	+K
FF	$L	,	/L	L	L	l	+L
CR	$M	-	-	M	M	m	+M
SO	$N	.	.	N	N	n	+N
SI	$O	/	/O	O	O	o	+O
DLE	$P	0	0	P	P	p	+P
DC1	$Q	1	1	Q	Q	q	+Q
DC2	$R	2	2	R	R	r	+R
DC3	$S	3	3	S	S	s	+S
DC4	$T	4	4	T	T	t	+T
NAK	$U	5	5	U	U	u	+U
SYN	$V	6	6	V	V	v	+V
ETB	$W	7	7	W	W	w	+W
CAN	$X	8	8	X	X	x	+X
EM	$Y	9	9	Y	Y	y	+Y
SUB	$Z	:	/Z	Z	Z	z	+Z
ESC	%A	;	%F	[	%K	{	%P
FS	%B	<	%G	\	%L	|	%Q
GS	%C	=	%H	]	%M	}	%R
RS	%D	>	%I	^	%N	~	%S
US	%E	?	%J	_	%O	DEL	%T

Figure 7 Code 39 full 128 character encoding

Code 128 (Code-128(A,B, and C))

Code 128 is an alphanumeric bar code. It is a high-density, variable-length, continuous code, which employs multiple element widths. Code 128 has three possible start codes. The start code defines the code set, Code A, B, or C that will be used to generate the barcode. The EPIC 3000 allows the code set to be specified, or it can be select by the printer based on the information in the data field.

To specify code set: [ESC] b <2> <Code>{information} [NUL]

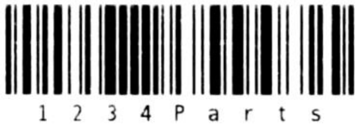
Function	Code 128 Manual Encoding
ASCII	[ESC] b <2> <Code>{information} [NUL]
Hexadecimal	1BH 62H <2> <Code>{information} [NUL]
Decimal	<27> <98> <2> <Code>{information} [NUL]

If the first character in the data field <Code> is a start code as shown in Figure 11 below, the printer will print the complete data field from the selected set. Due to space limitations, only ten characters can be printed. The check digit is generated and printed by the printer. Characters are also specified as shown in Figure 11.

Space is defined as a <0>, which makes programming difficult and causes control character conflicts for the printer. To solve this problem, the EPIC 3000 Printer subtracts 32 from all characters that are to be included in the bar code. In the Code 128 definition, an 'A' is <33>; however, the printer converts an ASCII 'A' (<65>) to a <33> internally. This sets Code 128C and the start codes off by 32.

Barcode 128 consists of 107 unique symbols. 101 of the symbols take on different meanings based on the start code or an embedded shift code sequence. Code stick A consists of alphanumeric characters and ASCII control codes (see the table below). Code stick B consists of Alpha numeric with lower case alpha, Code stick C consists of numeric pairs.

In most cases, the EPIC 3000 will generate the most compact barcode for you. However, if it is desirable to have complete control, the programmer should use manual mode.



[ESC]b<2><137><44><66><132>Parts[NUL]
Start with Code C, <44> is for character pair 12, <66> is for character pair 34, <132> shifts to Code B, then it's followed with the characters "Parts" and the ending [NUL].

Figure 8 Code 128 Manual Encoding Example

Code 128 Encoding				Accutherm Manual Encoding	
Code Stick			Code 128 Value	Decimal Value	Hex Value
Code A	Code B	Code C			
Space	Space	00	00	32	20
!	!	01	01	33	21
"	"	02	02	34	22
#	#	03	03	35	23
$	$	04	04	36	24
%	%	05	05	37	25
&	&	06	06	38	26
'	'	07	07	39	27
(	(	08	08	40	28
)	)	09	09	41	29
*	*	10	10	42	2A
+	+	11	11	43	2B
,	,	12	12	44	2C
-	-	13	13	45	2D
.	.	14	14	46	2E
/	/	15	15	47	2F
0	0	16	16	48	30
1	1	17	17	49	31
2	2	18	18	50	32
3	3	19	19	51	33
4	4	20	20	52	34
5	5	21	21	53	35
6	6	22	22	54	36
7	7	23	23	55	37
8	8	24	24	56	38
9	9	25	25	57	39
:	:	26	26	58	3A
;	;	27	27	59	3B
<	<	28	28	60	3C
=	=	29	29	61	3D
>	>	30	30	62	3E
?	?	31	31	63	3F
@	@	32	32	64	40
A	A	33	33	65	41
B	B	34	34	66	42
C	C	35	35	67	43
D	D	36	36	68	44
E	E	37	37	69	45
F	F	38	38	70	46
G	G	39	39	71	47
H	H	40	40	72	48
I	I	41	41	73	49
J	J	42	42	74	4A
K	K	43	43	75	4B
L	L	44	44	76	4C
M	M	45	45	77	4D
N	N	46	46	78	4E
O	O	47	47	79	4F
P	P	48	48	80	50
Q	Q	49	49	81	51
R	R	50	50	82	52
S	S	51	51	83	53
T	T	52	52	84	54
U	U	53	53	85	55
V	V	54	54	86	56
W	W	55	55	87	57
X	X	56	56	88	58
Y	Y	57	57	89	59
Z	Z	58	58	90	5A
[	[	59	59	91	5B

Code 128 Encoding				Accutherm Manual Encoding	
Code Stick			Code 128 Value	Decimal Value	Hex Value
Code A	Code B	Code C			
\	\	60	60	92	5C
]	]	61	61	93	5D
^	^	62	62	94	5E
_	_	63	63	95	5F
NUL	`	64	64	96	60
SOH	a	65	65	97	61
STX	b	66	66	98	62
ETH	c	67	67	99	63
EOT	d	68	68	100	64
ENQ	e	69	69	101	65
ACK	f	70	70	102	66
BEL	g	71	71	103	67
BS	h	72	72	104	68
HT	i	73	73	105	69
LF	j	74	74	106	6A
VT	k	75	75	107	6B
FF	l	76	76	108	6C
CR	m	77	77	109	6D
SO	n	78	78	110	6E
SI	o	79	79	111	6F
DLE	p	80	80	112	70
DC1	q	81	81	113	71
DC2	r	82	82	114	72
DC3	s	83	83	115	73
DC4	t	84	84	116	74
NAK	u	85	85	117	75
SYN	v	86	86	118	76
ETB	w	87	87	119	77
CAN	x	88	88	120	78
EM	y	89	89	121	79
SUB	z	90	90	122	7A
ESC	{	91	91	123	7B
FS	|	92	92	124	7C
GS	}	93	93	125	7D
RS	~	94	94	126	7E
US	DEL	95	95	127	7F
FNC3	FNC3	96	96	128	80
FNC2	FNC2	97	97	129	81
Shift	Shift	98	98	130	82
Code C	Code C	99	99	131	83
Code B	FNC4	Code B	100	132	84
FNC4	Code A	Code A	101	133	85
FNC1			102	134	86
Start Code A			103	135	87
Start Code B			104	136	88
Start Code C			105	137	89
Stop			-	-	

Figure 9 Code 128 encoding values

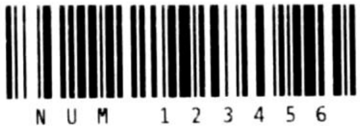
Code 128 Auto Encoding

To have the printer selected code set and automatically generate an optimal barcode, the value of Code should be the length.

Function	**Code 128 Automatic Encoding**
ASCII	[ESC] b <2> <Length>{information}
Hexadecimal	1BH 62H <2> <Length>{information}
Decimal	<27> <98> <2> <Length>{information}

If the first character <Length> is from 1 to 31, the printer will automatically select Code A, B, or C depending on the data present. If the data is all numeric, the data can be printed as pairs. This effectively doubles the amount of data that can be printed. The check digit is generated and printed by the printer. In this mode the exact data sent to the printer is encoded. There is no offset, function code, or shift code requirements.

For example the following is printed in a mixture of code A and C encoding:



[ESC] b <2> <10>NUM 123456

Figure 10 Automatic Encoding Example

Note: If the first character is greater than <31> and not <135> through <137>, the printer will discard the first character and print the data as defined in Code A.

In automatic mode, any ASCII data from 0 to 127 could be entered. Values less than 32 will be encoded as Code stick A NUL- US, values from 96 through 127 will be encoded from Code stick B. Where ever possible numeric pairs will be encoded from Code stick C.

FNC1, FNC2, FNC3, and FNC4 may be encoded based on the table below. All other values will result in a barcode data error and the barcode will not be generated.

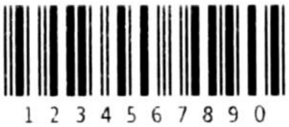
128 Code	Value in Decimal	Value in Hex
FNC3	128	80
FNC2	129	81
Not Valid	130	82
	131	83
	132	84
FNC4	133	85
FNC1	134	86

Figure 11 Code 128 FNC encoding

Interleaved 2 of 5 (Code 2 of 5)

Function	**Interleaved 2 of 5**
ASCII	[ESC] b <0> {information} [NUL]
Hexadecimal	1BH 62H <0> ... 03H
Decimal	<27> <98> <0> ... <3>

Interleaved 2 of 5 is a high-density, self-checking, continuous, numeric bar code. It is mainly used where fixed-length numeric fields are required. The data field must be an even number of characters. If an odd data field is sent to the EPIC 3000 Printer , it will be zero padded.



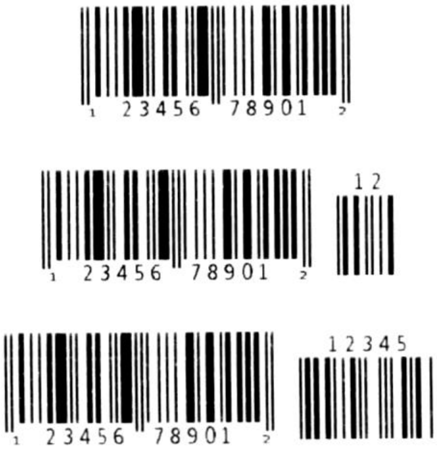
[ESC] b <0> 1234567890 [NUL]

Figure 12 Interleaved 2 of 5 Example

UPC A

UPC A is a fixed-length, numeric, continuous code that employs four element widths. The printer supports Universal Product Code Version A, E, EAN-8, and EAN-13. Version A encodes 11 digits. Typically, the UPC A format starts with a number system digit, five-digit manufacturer's code, five-digit product code, and a check digit. The printer makes no assumptions about any of the codes except the check digit. The printer prints an UPC bar code with the 11 digits sent to it and generates the check digit. If fewer than 11 digits are sent, the remaining digits will be zeros. UPC A may include an EAN 2 or EAN 5 Addenda.

Function	**UPC A**
ASCII	[ESC] b <3> {information} [NUL]
Hexadecimal	1BH 62H <3> information} [NUL]
Decimal	<27> <98> <3> information} [NUL]



[ESC] b <3> 12345678901[NUL]

UPC A with an EAN 2 Addenda

[ESC] b <3> 12345678901+12[NUL]

UPC A with an EAN 5 Addenda

[ESC] b <3> 12345678901+12345[NUL]

Figure 13 UPC A Examples

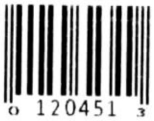
UPC E

UPC E is a zero suppression version of UPC. To allow the use of UPC barcodes on smaller packages where a full 12-digit barcode may not fit, a 'zero-compressed' version of UPC was called UPC-E. This barcode differs from UPC-A in that it only a 6-digit code is used, it does not use middle guard bars, and the end bit pattern is altered. UPC E requires that the first digit is zero for number system zero however; the printer does not enforce it. The printer does the compression based on the compression rules for UPC E, prints an UPC bar code based on the 11 digits sent to it, and generates the check digit. If fewer than 11 digits are sent leading zeros are added to form an 11 digit code. If the barcode does not meet the compression rules, invalid zero suppression digits are removed. UPC E may include an EAN 2 or EAN 5 Addenda.

Last digit	UPC-E equivalent is	UPC-A equivalent is
0	XXNNN0	0XX000-00NNN + check
1	XXNNN1	0XX100-00NNN + check
2	XXNNN2	0XX200-00NNN + check
3	XXXNN3	0XXX00-000NN + check
4	XXXXN4	0XXXX0-0000N + check
5	XXXXX5	0XXXXX-00005 + check
6	XXXXX6	0XXXXX-00006 + check
7	XXXXX7	0XXXXX-00007 + check
8	XXXXX8	0XXXXX-00008 + check
9	XXXXX9	0XXXXX-00009 + check

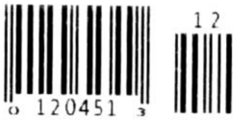
Figure 14 UPC E Zero Suppression Formats

Function	**UPC E**
ASCII	[ESC] b <5> {information} [NUL]
Hexadecimal	1BH 62H <5> information} [NUL]
Decimal	<27> <98> <5> information} [NUL]



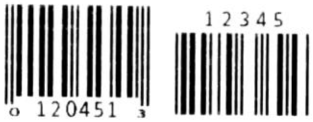
[ESC] b <5>01210000345[NUL]

UPC E with an EAN 2 Addenda



[ESC] b <5> 01210000345+12[NUL]

UPC E with an EAN 5 Addenda



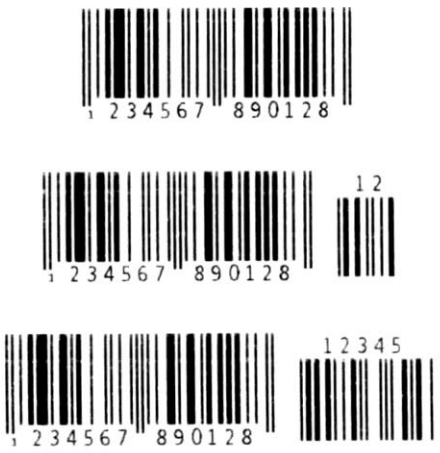
[ESC] b <5> 01210000345+12345[NUL]

Figure 15 UPC E Examples

EAN-13

EAN-13 is a fixed-length, numeric, continuous code that employs four element widths. The printer supports EAN-13, which is a superset of UPC that encodes 12 digits. Typically, the format starts with a number set digit, which defines how the next six digits are encoded. The next five digits have fixed encoding. The last is a check digit. The printer prints an EAN-13 bar code with the 12 digits sent to it and generates the check digit. If fewer than 12 digits are sent, the remaining digits will be zeros. EAN 13 may include an EAN 2 or EAN 5 Addenda.

Function	**EAN-13**
ASCII	[ESC] b <5> {information} [NUL]
Hexadecimal	1BH 62H <5> information} [NUL]
Decimal	<27> <98> <5> information} [NUL]



[ESC] b <3> 12345678901[NUL]

UPC A with an EAN 2 Addenda

[ESC] b <3> 12345678901+12[NUL]

UPC A with an EAN 5 Addenda

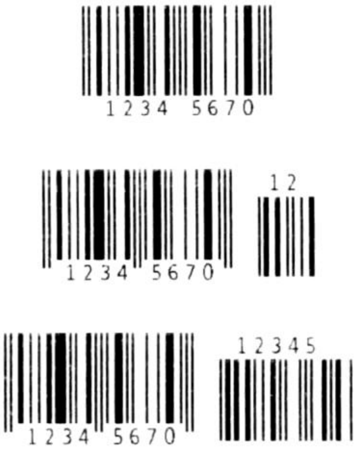
[ESC] b <3> 12345678901+12345[NUL]

Figure 16 EAN 13 Examples

EAN-8

EAN-8 is a fixed-length, numeric, continuous code that employs four element widths. The printer supports EAN-8, which is a superset of UPC that encodes seven digits. The printer prints an EAN-8 bar code with the seven digits sent to it and generates the check digit. If fewer than seven digits are sent, the remaining digits will be zeros. EAN 8 may include an EAN 2 or EAN 5 Addenda.

Function	**EAN-8**
ASCII	[ESC] b <6> {information} [NUL]
Hexadecimal	1BH 62H <6> information} [NUL]
Decimal	<27> <98> <6> information} [NUL]



[ESC] b <6> 1234567[NUL]

EAN-8 with an EAN 2 Addenda

[ESC] b <6> 1234567+12[NUL]

EAN-8 with an EAN 5 Addenda

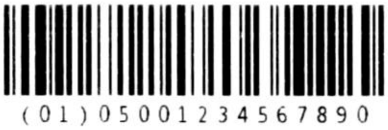
[ESC] b <6> 1234567+12345[NUL]

Figure 17 EAN 8 Examples

EAN-14

EAN-14 It is a high-density, fixed-length, numeric, continuous code, which employs multiple element widths. EAN-14, is a subset of Code 128 that encodes FNC1 and 14 digit pairs. If fewer than 14 digits are sent, leading zeros will be added to complete the code.

Function	**EAN-14**
ASCII	[ESC] b <12> {information} [NUL]
Hexadecimal	1BH 62H <12> information} [NUL]
Decimal	<27> <98> <12> information} [NUL]



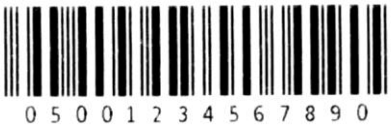
[ESC] b <12>0500123456789[NUL]

Figure 18 EAN 14 Example

ITF-14

ITF-14 is the GS1 formatted Interleaved 2 of 5 barcode. It is intended to encode a Global Trade Item Number. The ITF-14 will always encode 13 input digits and adds a 14th check digit.

Function	**ITF-14**
ASCII	[ESC] b <13> {information} [NUL]
Hexadecimal	1BH 62H <13> information} [NUL]
Decimal	<27> <98> <13> information} [NUL]



[ESC] b <13>0500123456789[NUL]

Figure 19 ITF-14 Example

NOTE: ITF-14 is generally printed with a surrounding box, however it is not mandatory and the EPIC 3000 does not add it.

EAN 2 and EAN 5 Addenda barcodes

EAN2 and EAN 5 Addenda barcodes are used to add additional information to several barcodes. The Addenda field is either 2 or 5 characters. The size of the Addenda will automatically be generated based on the size of the data supplied.

Add on data will only be accepted by barcodes that support it. To add Addenda data, append the linear barcode data with a "+" and the numeric Addenda data field. The top image below shows an EAN 2 added to a standard UPC A barcode and the bottom image is an EAN 5 added to the same UPC A barcode.

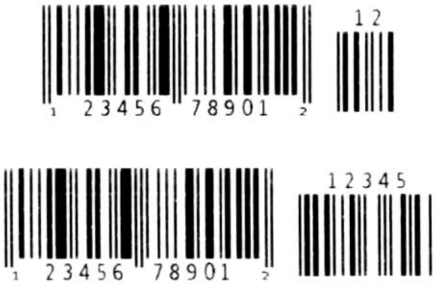
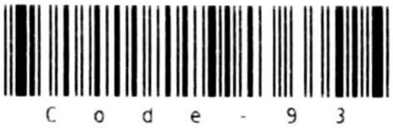


Figure 20 EAN 2 and EAN 5 Addendas

Code 93

Code 93 is a variable-length, alphanumeric bar code. The complete data field is printed by the printer. Due to space limitations, only 10 or 11 characters can be printed.

Function	**Code 93**
ASCII	[ESC] b <7> {information} [NUL]
Hexadecimal	1BH 62H <7> information} [NUL]
Decimal	<27> <98> <7> information} [NUL]



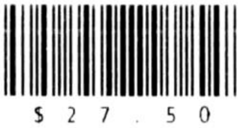
[ESC] b <7> Code-93[NUL]

Figure 21 Code 93 Example

Codabar

Codabar is a variable-length format, primarily used for numeric symbols. It offers 16 data characters, including the numeric digits zero through nine, and -, $, :, /, ., and +. Four unique start/stop characters, designated A, B, C, and D, are also available. Due to space limitations, only 12 characters can be printed. If the first character is 1 to 31 it will be used as the length. Note: If the first and last characters are not a start or stop code, Code A will be added.

Function	**Codabar**
ASCII	[ESC] b <8> {information} [NUL]
Hexadecimal	1BH 62H <8> {information} [NUL]
Decimal	<27> <98> <8> {information} [NUL]



[ESC] b <8> $27.50[NUL]
As Start and Stop are not specified, start and stop A has been added.

Figure 22 Codabar Example

DataBar (RSS) GS-1 barcodes

The Reduced Space Symbology (RSS) barcodes are intended to encode 14 digit fields and is intended to replace UPC barcodes in the Retail industry. In February 2007 the GS1 organization took over control of the RSS barcodes and renamed them DataBar. RSS barcodes use 6 variable bar and space widths to encode the value. Because of the width encoding, a single scaling factor is used to adjust the barcode. The wide and narrow bar settings are not used. Non-stacked barcodes will use the height setting, however, stacked versions have specific height to width ratio requirements and will adjust the ratios to fit in the selected height. The scale factor will also be used but will only affect the width.

All RSS barcodes are encoded with awareness of the GS-1 General specification for AI and AI field requirements. In some cases not all possible character sequences are allowed in an AI field and many are fixed length. In some cases AI fields contain check digits and during the encoding process they are removed. When specifying AI fields with check digits the check digit must be passed to the EPIC 3000 printer. In some cases it is used and in others it is removed and replaced by the reader. The EPIC 3000 will make some attempt to verify the format, however only critical data that would prevent encoding the data is actually checked by the printer. It's up to the application programmer to format the data in accordance with the GS-1 specification.

RSS barcodes may get excessively large. The printer will attempt to generate a barcode that is the requested size but will automatically shrink the barcode scale if it will not fit in the print zone. This may result in elements that are too small to print reliably. If that happens the printer will not fault but will slow the print process and print the barcode anyway. In general barcodes with a single dot wide element are unreadable.

There are several versions or formats of RSS bar codes and each has specific options and requirements.

GS1-Databar-14 (GS1-Databar-Omni-directional)

RSS-14 (GS-1 DataBar Omni-directional) encodes the full 14 digit EAN.UCC[10] item identification in a symbol that can be omni-directionally read. For example:

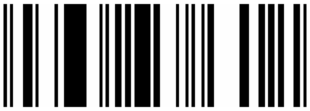


Figure 23 RSS-14 symbol representing (01)20012345678909

Where the leading (01) is the implied application identifier and is not encoded in the symbol. The last digit, 9, is not directly encoded in the symbol, but is a calculated mod 10 check digit. The (01) and the last digit are not sent as part of the command so the data field would therefore be "2001234567890"

Function	**GS1-Databar 14**
ASCII	[ESC] b <18> {information} [NUL]
Hexadecimal	1BH 62H <18> {information} [NUL]
Decimal	<27> <98> <18> {information} [NUL]

(01)12345678901231

[ESC] b <18> 1234567890123 [NUL]

Figure 24 GS1-Databar 14 Example

Note: Note that the data to be encoded does not include the GS1 format [01] as it is implied. You only need to send the 13 digit item number. The check digit will be generated by the printer.

.

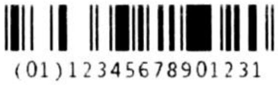
RSS-14 is capable of encoding $2x10^{13}$ values. These values are expressed as 14 digits. The first digit is a linkage flag, followed by 13 data digits. The 13 data characters plus the check digit form the 14 digit identification number including the leading indicator digit. Values 10,000,000,000,000 and above indicate that the linkage flag is set and therefore a 2D component is present. That is 10,001,234,567,890 encodes as 00012345678905 with a linkage flag of 1.

[10] The 14 digit field is not a simple sequence of digits but must follow EAN.UCC conventions and the GS-1 General specification.

GS1-Databar-Truncated

GS1-Databar truncated is a more compact version of the GS1-Databar and will encode the full 14 digit EAN.UCC item identification but using less vertical height. The specification defines the truncated version to be 13 times the X dimension, however, the height of the barcodes can be set by the configuration commands. The GS1-Databar truncated version printed by the EPIC 3000 is simply ½ the height of the standard GS-1 DataBar symbol.

Function	**GS1-Databar 14 Truncated**
ASCII	[ESC] b <19> {information} [NUL]
Hexadecimal	1BH 62H <19> {information} [NUL]
Decimal	<27> <98> <19> {information} [NUL]



[ESC] b <19> 1234567890123 [NUL]

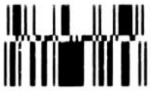
Figure 25 GS1-Databar 14 Truncated Example

Note: Note that the data to be encoded does not include the GS1 format [01] as it is implied. You only need to send the 13 digit item number. The check digit will be generated by the printer.

GS1-Databar-14 Stacked and GS1-Databar-14 Stacked-Omni

GS1-Databar-14 Stacked and GS1-Databar-14 Stacked Omni-directional are RSS-14 barcodes printed in a stacked format. The ratios between the top and bottom of this symbol are fixed and are not adjustable. There is also a separator between the barcodes that has a specific relationship to the top and bottom bars. This barcode must be printed within a controlled aspect ratio. GS1-Databar Stacked symbols are typically printed without HRI. To allow HRI to be printed if required the Stacked HRI may be enabled with a special configuration command.

Function	**GS1-Databar 14 Stacked**
ASCII	[ESC] b <21> {information} [NUL]
Hexadecimal	1BH 62H <21> {information} [NUL]
Decimal	<27> <98> <21> {information} [NUL]

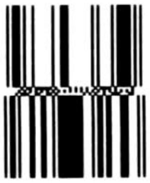


[ESC] b <21> 1234567890123 [NUL]

Figure 26 GS1-Databar 14 Stacked Example

GS1-Databar-14 Stacked Omni-directional is spececfied to be printer with a greater height than the GS1-Databar-14 Stacked by a factor of 3. As the height of the symbol may be specified by command, the GS1-Databar-14 Stacked Omni-directional symbol will be printed 3 times the GS1-Databar-14 Stacked symbol.

Function	**GS1-Databar 14 Stacked-Omni**
ASCII	[ESC] b <22> {information} [NUL]
Hexadecimal	1BH 62H <22> {information} [NUL]
Decimal	<27> <98> <22> {information} [NUL]



[ESC] b <22> 1234567890123 [NUL]

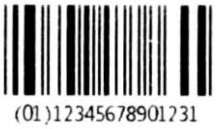
Figure 27 GS1-Databar 14 Stacked Omni Example

Note: Note that the data to be encoded does not include the GS1 format [01] as it is implied. You only need to send the 13 digit item number. The check digit will be generated by the printer.

GS1-Databar-Limited

RSS Limited (GS-1 DataBar Limited) encodes a 14 digit EAN.UCC item identification with an indicator digit of 0 or 1 in a small symbol that is not intended to be scanned. As the indicator can only be 0 or 1, the barcode must start with 0 or 1 or it is invalid.

Function	**GS1-Databar Limited**
ASCII	[ESC] b <20> {information} [NUL]
Hexadecimal	1BH 62H <20> {information} [NUL]
Decimal	<27> <98> <20> {information} [NUL]



[ESC] b <20> 1234567890123 [NUL]

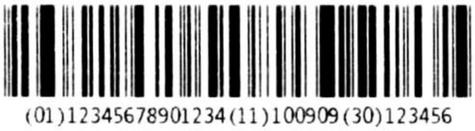
Figure 28 GS1-Databar Limited Example

Note: Note that the data to be encoded does not include the GS1 format [01] as it is implied. You only need to send the 13 digit item number. The check digit will be generated by the printer.

GS1-Databar-Expanded

RSS Expanded (GS-1 DataBar Expanded) encodes the EAN.UCC item identification plus supplementary element strings such as weight and date in a symbol that can be omni-directionally scanned.

Function	**GS1-Databar Expanded**
ASCII	[ESC] b <23> {information} [NUL]
Hexadecimal	1BH 62H <23> {information} [NUL]
Decimal	<27> <98> <23> {information} [NUL]



[ESC] b <23>
[01]12345678901234[11]100909[30]123456[NUL]

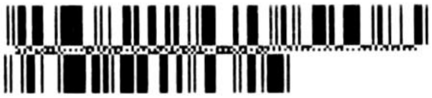
Figure 29 GS1-Databar Expanded Example

Note: The data for expanded RSS barcodes must be formatted using AI fields as defined by the GS-1 specification and comply with the EAN.UCC format. The GS-1 ISO/IEC 24724 standard specifies these formats.
Note: It is possible to define an RSS Expanded barcode that will not fit in the print zone of the EPIC 3000 printer. If this occurs, the printer will attempt to scale the barcode to fit. If the scaling fails, the barcode may be unreadable.

GS1-Databar-Expanded Stacked

RSS Expanded Stacked (GS-1 DataBar Expanded Stacked) encodes the EAN.UCC item identification plus supplementary element strings such as weight and date in a stacked symbol that can be omni-directionally scanned. The ratios between the top and bottom of this symbol are fixed and are not adjustable. There is also a separator between the barcodes that has a specific relationship to the top and bottom bars. This barcode must be printed within a controlled aspect ratio. GS1-Databar Stacked symbols are typically printed without HRI. To allow HRI to be printed if required the Stacked HRI may be enabled with a special configuration command.

Function	**GS1-Databar Expanded Stacked**
ASCII	[ESC] b <24> {information} [NUL]
Hexadecimal	1BH 62H <24> {information} [NUL]
Decimal	<27> <98> <24> {information} [NUL]



[ESC] b <24>
[01]12345678901234[11]100909[30]123456[NUL]

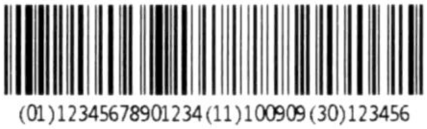
Figure 30 GS1-Databar Expanded Stacked Example

EAN-128 (GS1-128)

The GS1-128 Bar Code has been designed through joint co-operation between GS1 and Automatic Identification Manufacturers, Inc. (AIM). The GS1-128 barcode is a subset of the more general Code 128 barcode. By agreement between AIM, Inc. and GS1, use of the Function 1 Symbol Character (FNC1) in Code 128 Symbols in the first symbol character position following the Start Character has been reserved exclusively for the GS1 System.

The EPIC 3000 printer provides a GS-1 coded input to the barcode and encodes all the EAN-128 requirements into the data.

Function	**GS1-128**
ASCII	[ESC] b <11>{information} [NUL]
Hexadecimal	1BH 62H <11>{information} [NUL]
Decimal	<27> <98> <11>{information} [NUL]



[ESC] b <11>
[01]12345678901234[11]100909[30]123456[NUL]

Figure 31 GS1-128 Example

Note: The data for GS1-128 barcodes must be formatted using AI fields as defined by the GS-1 specification and comply with the EAN.UCC format. The GS-1 ISO/IEC 24724 standard specifies these formats.

2D Barcodes

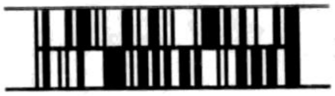
The EPIC 3000 supports a number of 2D and Stacked barcodes. The terms stacked barcode or multi-row barcode code are more accurately applied to those barcodes made up of a series of one-dimensional barcodes. The term Matrix code generally applies to 2-D codes that code the data based on the position of black spots within a matrix. Each black element is the same dimension and it is the position of the element that encodes the data.

Note: When printing 2D barcodes HRI (Human Readable Information) is generally not used. The EPIC 3000 Supports HRI when printing 2D barcodes but makes no attempt to align the text with the barcode.
Note: Space between the barcode and any HRI if active is controlled by the printer. If HRI is not active, the same white spacing is used. It is up to the user to enforce a quite zone between the barcode and any user data.

Code 49

Code 49 barcode is a stacked barcode containing between 2 and 8 rows, each separated by a separator bar. Each row contains 16 "words" or character pairs with a start and stop character. The last row also contains the number of rows in the barcode. There are 2400 possible words which can be generated from each pair of characters. Symbols with less than 7 rows contain 2 check digits in the final row. Symbols with 7 or 8 rows contain 3 check digits in the final row.

Function	Code 49
ASCII	[ESC] b <14> {information} [NUL]
Hexadecimal	1BH 62H <14> {information} [NUL]
Decimal	<27> <98> <14> {information} [NUL]



[ESC] b <14>12345678901234[NUL]

Figure 32 Code 49 Example

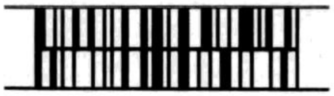
Note: There is no user control of the number of rows or columns.
Note: The scale may be set from 2 to 6
Note: GS-1 encoding may be activated for both Code16. There is an FNC1 indicator in the barcode that indicated that GS1 is active. If GS-1 formatting is used it's up to the user to recognize that GS1 is active and decode the barcode as GS1 data.

Code 16K

The Code 16K bar code is a multiple-row bar code that can encode the full ASCII character set below ASCII 128. It uses existing UPC and Code 128 character set patterns. Up to 77 full ASCII characters or 154 numeric characters can be encoded. These characters are encoded into 2 to 16 rows. Each row is divided by a separator bar. The top and bottom of the symbol also have separator bars that extend to the ends of the minimum quiet zones.

Like Code128 there are various ways to encode the data into the bar pattern. The encoder for Code16K automatically selects the proper encoding method to produce the most compact barcode. If the bar code has four or more consecutive numbers, the numbers are encoded in number pairing mode. This means that two numbers are encoded into one character width, making the size of the bar code smaller. The Code 16K bar code has three forms of error detection. Parity is checked for each character, start and stop characters are used to identify each row, and two checksum characters are always appended to the end of the bar code

Function	**Code 16K**
ASCII	[ESC] b <17> {information} [NUL]
Hexadecimal	1BH 62H <17> {information} [NUL]
Decimal	<27> <98> <17> {information} [NUL]



[ESC] b <17>12345678901234[NUL]

Figure 33 Code 16K Example

Note: There is no user control of the number of rows or columns.
Note: The scale may be set from 2 to 6
Note: GS-1 encoding may be activated for both Code16. There is an FNC1 indicator in the barcode that indicated that GS1 is active. If GS-1 formatting is used it's up to the user to recognize that GS1 is active and decode the barcode as GS1 data.

PDF 417, Truncated PDF417 and Micro PDF 417

PDF 417, Truncated PDF417 and Micro PDF 417 are a two dimensional barcode that will encode the full ASCII character set. As it encodes the full set including control characters, the length of the following data must be provided to the printer.

PDF 417

Large amounts of text and data can be encoded when using the PDF417 barcodes. The printed symbol consists of several linear rows of stacked code words. Each codeword represents 1 of 929 possible values from one of three different clusters. A different cluster is chosen for each row, repeating after every three rows. Because the code words in each cluster are unique, the scanner is able to determine what line each cluster is from.

PDF417 uses Reed Solomon error correction instead of check digits. This error correction allows the symbol to endure some damage without causing loss of data. AIM standards recommend a minimum error correction level of 2.

The X dimension is the width of the narrowest bar in a printed codeword. The Y dimension is the height of each row within the PDF417 symbol. The PDF417 barcode is usually printed at an X to Y ratio of 1:2 to 1:5.   The EPIC 3000 printer defaults to a 1:ratio. By lowering the ratio, a significant amount of space can be saved; however, some scanners cannot read X to Y ratios of less than 1:3.

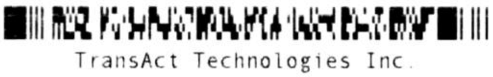
The form of the command is as follows:

Function	**PDF 417**
ASCII	[ESC] b <9><nL><nH><d_1> … <d_n>
Hexadecimal	1BH 62H <9><nL><nH><d_1> … <d_n>
Decimal	<27> <98> <9><nL><nH><d_1> … <d_n>

Where the data length is (nH * 256) + nL. The length is limited to be from 1 to 2048 characters.

Alternate Command

Function	**PDF 417**
ASCII	[ESC] b <10> {information}[NUL]
Hexadecimal	1BH 62H <10> {information}[NUL]
Decimal	<27> <98> <10>{information}[NUL]



[ESC]b<10>TransAct Technologies Inc.[NUL]

Figure 34 PDF 417 Example

Truncated PDF 417

Truncated PDF417 is two-dimensional (2D), multi-row barcode, derived from PDF417. A truncated PDF417 symbol uses less area than the normal PDF417 barcode as the right hand side of the symbol is removed or truncated. This option should be used where damage to the barcode is unlikely as it is not as easily read ad normal PDF417.

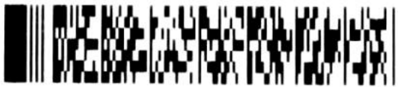
The form of the command is as follows:

Function	**Truncated PDF 417**
ASCII	[ESC] b <38><nL><nH><d_1> … <d_n>
Hexadecimal	1BH 62H <38><nL><nH><d_1> … <d_n>
Decimal	<27> <98> <38><nL><nH><d_1> … <d_n>

Where the data length is (nH * 256) + nL. The length is limited to be from 1 to 150 characters.

Alternate Command

Function	**Truncated PDF 417**
ASCII	[ESC] b <39> {information}[NUL]
Hexadecimal	1BH 62H <39> {information}[NUL]
Decimal	<27> <98> <39>{information}[NUL]



[ESC]b<39>TransAct Technologies Inc.[NUL]

Figure 35 Truncated PDF 417 Example

Note: Micro PDF 417 shares a control table with PDF417 however the right side is removed.
Note: The PDF417 specification does not reference GS-1 formatting; however GS-1 encoding may be activated for both Truncated PDF417. There is no indicator in the barcode that indicated that GS1 is active. If GS-1 formatting is used it's up to the user to recognize that GS1 is active and decode the barcode as GS1 data.

Micro PDF 417

MicroPDF417 is two-dimensional (2D), multi-row barcode, derived from PDF417. Micro PFD417 can encode up to 150 bytes. Micro-PDF417 is designed for applications requiring improved area efficiency

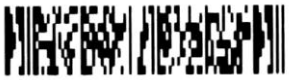
The form of the command is as follows:

Function	**Micro PDF 417**
ASCII	[ESC] b <33><nL><nH><d$_1$> … <d$_n$>
Hexadecimal	1BH 62H <33><nL><nH><d$_1$> … <d$_n$>
Decimal	<27> <98> <33><nL><nH><d$_1$> … <d$_n$>

Where the data length is (nH * 256) + nL. The length is limited to be from 1 to 150 characters.

Alternate Command

Function	**Micro PDF 417**
ASCII	[ESC] b <34> {information}[NUL]
Hexadecimal	1BH 62H <34> {information}[NUL]
Decimal	<27> <98> <34>{information}[NUL]



[ESC]b<32>TransAct Technologies Inc.[NUL]

Figure 36 Micro PDF 417 Example

Note: Micro PDF 417 shares a control table with PDF417. PDF417 allows the columns to be set from 1 to 30, however Micro PDF417 only allows 1-4. (0 will auto set the columns) If the columns are set out of range, it will be limited to 4.
Note: The encryption level cannot be changed for Micro PDF 417.
Note: MicroPDF417 is used for composite barcodes which use GS1 formatting. When used for composit barcodes GS1 encoding is active.
Note: The PDF417 and Micro PDF 417 specifications do not reference GS-1 formatting; however GS-1 encoding may be activated for MicroPDF417. There is no indicator in the barcode that indicates that GS1 is active. If GS-1 formatting is used it's up to the user to recognize that GS1 is active and decode the barcode as GS1 data.

Data Matrix Bar Code

This is a 2-D matrix symboligy barcode that can encode numeric or alphanumeric data and includes error correction. There are 4 original error correction ECC-000 – ECC-140 as well as the standard Reed-Solomon ECC-200 error correction. Most current scanners only support ECC-200. In theory 3116 numeric and 2335 characters can be encoded in Data Matrix. From a practical point of view the limited print width of the EPIC 3000 limits the maximum size of the barcode.

The Data Matrix barcode matrix may be square or rectangular and only specific sizes are allowed. You may select a size and you may limit the selection to square only, however, if you limit the size, you must leave enough room for the data or the printer will automatically select a larger matrix. The commands to control 2 dimensional barcode options start on page 150.

Data Matrix Encoding

As referred to above, there are 2 basic encoding schemes defined for Data Matrix barcodes; ECC 000 – 140 and ECC 200. ECC 200 which uses Reed-Solomon error correction and is recommended for all new applications. ECC 000 - 140 is the older system and only supported by a few older scanners. As ECC200 is the recommended encoding, it is the only version supported by the EPIC 3000.

Data Matrix ECC200

ECC200 is actually 6 encoding schemes. The data to be encoded is scanned and the optimal encoding scheme or combination of schemes is used to encode the symbol. The resulting encoded data is then processed with the Reed-Solomon error correction algorithm and the resulting information is formatted into a Data Matrix barcode.

Note: The ISO/IEC 16022:2006 specification defines how the ECC200 encoding schemes are to be evoked, however in some cases the resulting size of the compressed data is identical with 2 or more different encoding sequences. The ISO standard is followed by the EPIC 3000 however it is possible that other generators will generate different barcode patterns that are equivalent.

Encodation scheme	**Characters**	**Bits per data character**
ASCII	double digit numerics	4
	ASCII values 0 - 127	8
	Extended ASCII values 128 - 255	16
C40	Upper-case alphanumeric	5,33
	Lower case and special characters	10,66*
Text	Lower-case alphanumeric	5,33
	Upper case and special characters	10,66**
X12	ANSI X12 EDI data set	5,33
EDIFACT	ASCII values 32 - 94	6
Base 256	All byte values 0 - 255	8
* encoded as two C40 values as result of use of a shift character		
** encoded as two Text values as result of use of a shift character		

Table 6 Encoding schemes for ECC 200

Data Matrix barcodes are fixed sizes and depending on how the data compresses, two different strings with the same character count may print different size barcodes. It is possible to select a minimum barcode size and unused data locations will be filled with pad data.

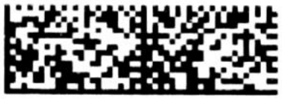
Data Matrix Commands

There are two version of the command, one is null terminated and the other allows a length to be specified (Note that the length is a two byte field as the symbol may contain more than 256 characters.

Function	**Data Matrix**
ASCII	[ESC] b <28>{information} [NUL]
Hexadecimal	1BH 62H <28>{information} [NUL]
Decimal	<27> <98> <28>{information} [NUL]

Function	**Data Matrix**
ASCII	[ESC] b <27><nL><nH><d_1> … <d_n>
Hexadecimal	1BH 62H <27><nL><nH><d_1> … <d_n>
Decimal	<27> <98> <27><nL><nH><d_1> … <d_n>

Note: The normal Select barcode Width and Select Barcode height commands do not affect Data matrix barcodes. The rules governing how the barcode is printed do not allow variations in the aspect ratios. There are Data Matrix control commands that will allow some control over how the barcode is printed.



[ESC] b <28>TransAct Technologies Inc. 20 Bomax Drive, Ithaca New York[NUL]

Figure 37 Data Matrix Example

Testing Data Matrix barcodes

There is a special reference symbol that is useful for control testing. It is a 16 x 16 ECC 200 symbol and can be printed which encodes the data “30Q324343430794<OQQ”. As shown in the figure below. This reference symbol has a region of parallel bars and spaces which can be linearly scanned and then evaluated for print growth using the edge-measurement methodologies of ISO/IEC 15416.

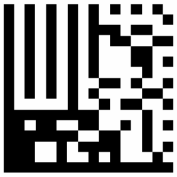


Figure 38 ECC 200 reference symbol encoding "30Q324343430794<OQQ"

Refer to the ISO/IEC 16022:2006 standard for more information.

Maxicode

Maxicode is 2D barcode originally created and used by United Parcel Service. Suitable for tracking and managing the shipment of packages, it resembles a barcode, but uses dots arranged in a hexagonal grid instead of bars. Maxicode has been standardized under "ISO/IEC 16023" or "AIM BC10 ISS – Maxicode"

A Maxicode is sometimes referred to as a "Bird's Eye", "Target", or "ups code". It generally appears as a 1 inch square[11], with a bull's-eye in the middle, surrounded by a pattern of hexagonal dots. It can store about 93 characters of information, and up to 8 Maxicode symbols can be chained together to convey more data. The centered symmetrical bull's-eye is useful in automatic symbol location regardless of orientation, and it allows Maxicode symbols to be scanned even on a package traveling rapidly.

Maxicode symbols optionally include a structured carrier message containing key information about a package. This information is protected with a strong Reed-Solomon error correction code, allowing it to be read even if a portion of the symbol is damaged. These fields include:

- A 4-bit indication of the mode in use, currently either mode 2 or mode 3.
- A national or international postal code. MaxiCode supports both numeric postal codes (e.g. a ZIP Code), and alphanumeric postal codes. (148501200 Transact)
- A 3-digit country code encoded per ISO 3166 (231 for the US)
- A 3-digit class of service code assigned by the carrier (3 ground)

- Mode 0 - Obsolete mode superseded by modes 2 and 3. However, the EPIC 3000 will use a mode setting of zero to indicate the mode should automatically be determined from the data supplied. In Auto Mode Modes 2,3 or 4 are used.
- Mode 1 – is obsolete and not supported by the EPIC 3000
- Mode 2 - Formatted data containing a structured carrier message with a 10 digit numeric postal code and 3 digit country and service code. (US domestic)
- Mode 3 - Formatted data containing a structured Carrier Message with a 6 character alphanumeric postal code and 3 digit country and service code. (international destinations)
- Mode 4 - Unformatted data with Standard Error Correction.
- Mode 5 - Unformatted data with Enhanced Error Correction.
- Mode 6 - Used for programming hardware devices.

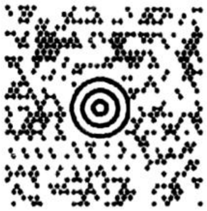


Figure 39 Maxicode Symbol

The above Mode 2 symbol is encoding the TransAct Address as follows: 148501200231003TransAct Technologies Inc. 20 Bomax Drive, Ithaca New York. Where 14850-1200 is the zip code, 231 is the country code and 003 is ground service.

[11] The T480 prints the Maxicode slightly larger 1 inch square to improve readability.

Maxicode Commands

There are two version of the command, one is NUL terminated and the other allows a length to be specified (Note that the length is a two byte field as the symbol may contain more than 256 characters.)

Function	**Maxicode**
ASCII	[ESC] b <16>{information} [NUL]
Hexadecimal	1BH 62H <16>{information} [NUL]
Decimal	<27> <98> <16>{information} [NUL]

Function	**Maxicode**
ASCII	[ESC] b <15><nL><nH><d_1> … <d_n>
Hexadecimal	1BH 62H <15><nL><nH><d_1> … <d_n>
Decimal	<27> <98> <15><nL><nH><d_1> … <d_n>

[ESC] b <16>[GS]148501200[GS]231[GS]3[GS]TransAct Technologies Inc. 20 Bomax Drive, Ithaca New York[NUL]

Figure 40 Data Maxicode Example

Note: The normal Select barcode Width and Select Barcode height commands do not affect Maxicode barcodes. The rules governing how the barcode is printed do not allow variations in the aspect ratios. There are Maxicode control commands that will allow some control over how the barcode is printed.
Note: GS1 encoding is not supported by Maxicode.

MaxiCode Mode 2 and 3 Structured Carrier Message

MaxiCode Mode 2 and Mode 3 are generally use as a destination sorting symbol. In these modes, the primary message is always data specific and encodes postal code, country code and service class. The three primary data elements should be supplied in the above order separated by GS (Group Separator - ASCII 29) immediately followed by the secondary message contents.

There are two commonly used message formats when using Mode 2 and Mode 3:

Messages Beginning with "[)>**RS**01**GS**yy"

Messages which begin with the seven encoded data characters"[)>RS01GSyy" conform to particular open system standards and has the following structure:

- [)>RS (Message Header)
- 01GS (Format Header)
  - 2-digit representing the year (yy) of a date
- Postal/Zip Code:

  - If Mode 2: 5-digit zip code + 4-digit zip code extension
  - If Mode 3: 6-alphanumeric characters zip code (A through Z or 0 to 9)
- GS
  - 3-digit country code
- GS
  - 3-digit class of service
- GS

  NOTE: The following format is not verified by the EPIC 3000
- <tracking number> (Mandatory Data for UPS)
- GS<SCAC> (Mandatory Data for UPS)
- GS<UPS shipper number>
- GS<Julian day of pickup>
- GS<shipment ID number>
- GS<n/x> (Package n/x)
- GS<package weight>
- GS<address validation>
- GS<ship to street address>
- GS<ship to city>
- GS<ship to state>
- RS
- EOT (End of Message)

Where GS (ASCII 29) is used to separate fields in a message; RS (ASCII 30) is used to separate format types and EOT (ASCII 4) is the end of transmission characters.

Note: The EPIC 3000 only enforces the format through the GS after the class of service field. The remainder of the message is not validated and is simply encoded by the printer.

.

Messages NOT Beginning with "[)>RS01GS"

- Postal/Zip Code:
  - If Mode 2 (NUMERIC ONLY): 5-digit zip code + 4-digit zip code extension (if none exists, four zeros 0000 must be specified)
  - If Mode 3 (ALPHANUMERIC): 6-alphanumeric characters zip code (A through Z or 0 to 9)
- GS
  - 3-digit country code (from ISO 3166) - NOTE: Mode 2 supports the US Country Code (840). For other country codes please use Mode 3 instead.
- GS
  - 3-digit class of service
- GS
  - <The secondary message data in the required format>
- EOT (End of Message)

Where GS (ASCII 29) is used to separate fields in a message and EOT (ASCII 4) is the end of transmission character.

QRCode

QR Code is a 2D barcode readable by QR scanners, mobile phones and smart phones with a camera. The code consists of black modules arranged in a square pattern on white background. The information encoded can be most any text data.

Although initially used for tracking parts in vehicle manufacturing, QR Codes are now used in a much broader context, including both commercial tracking applications and convenience-oriented applications aimed at mobile phone users (known as mobile tagging). Denso Wave, Inc. invented the QR Code Barcode. The word “ QR Code” is a registered trademark of DENSO WAVE INCORPORATED. This registered trademark applies only for the word “QR Code”, and not for the QR Code pattern or image.

QR Codes storing addresses and URLs may appear in magazines, on signs, buses, business cards, retail receipts, or on just about any object about which users might need information. Users with a camera phone equipped with the correct reader application can scan the image of the QR Code to display text, contact information, connect to a wireless network, or open a web page in the phone's browser. This act of linking from physical world objects is known as a hard link or physical world hyperlinks.

QR Code data Capacity

Numeric only	Max. 7,089 characters
Alphanumeric	Max. 4,296 characters
Binary (8 bits)	Max. 2,953 bytes
Kanji/Kana	Max. 1,817 characters

NOTE: At this time Kanji and Katakana are not supported by the EPIC 3000
QR codes use the Reed–Solomon error correction and the error correction capacity may be adjusted.

Error correction

Level L	7% of code words can be restored.
Level M	15% of code words can be restored.
Level Q	25% of code words can be restored.
Level H	30% of code words can be restored.

If numeric data is used, the barcode generator will optimize the barcode generation for numeric data. In Alpha numeric mode the printer will encode 0-9, A-Z, space, %, *, +, -, .,/ and :.

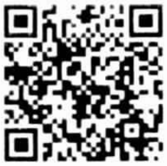


Figure 41 QRCode Symbol

Encoding: TransAct Technologies Inc. 20 Bomax Drive, Ithaca New York

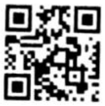


Figure 42 QRCode Symbol for a URL

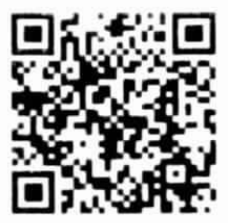
Encoding: www.transact-tech.com

QRCode Commands

There are two version of the command, one is null terminated and the other allows a length to be specified (Note that the length is a two byte field as the symbol may contain more than 256 characters.

Function	**QRCode**
ASCII	[ESC] b <26>{information} [NUL]
Hexadecimal	1BH 62H <26>{information} [NUL]
Decimal	<27> <98> <26>{information} [NUL]

Function	**QRCode**
ASCII	[ESC] b <25><nL><nH><d$_1$> … <d$_n$>
Hexadecimal	1BH 62H <25><nL><nH><d$_1$> … <d$_n$>
Decimal	<27> <98> <25><nL><nH><d$_1$> … <d$_n$>



[ESC] b <26>TransAct Technologies Inc. 20 Bomax Drive, Ithaca New York[NUL]

Figure 43 Data QRCode Example

Note: The normal Select barcode Width and Select Barcode height commands do not affect QRCode barcodes. The rules governing how the barcode is printed do not allow variations in the aspect ratios. There are QRCode control commands that will allow some control over how the barcode is printed.

Note: QR Code Model 1 is obsolete, only Model 2 is supported.

Micro QRCode

Micro QR Code is a 2D barcode readable by most QR scanners, mobile phones and smart phones with a camera. The code is similar to QR code although is restricted to 35 characters

Numeric only	Max. 35 characters
Alphanumeric	Max. 21 characters
Binary (8 bits)	Max. 15 bytes
Kanji/Kana	Max. 9 characters

Note: The number of characters is also affected by the error correction. The above assumes the default level L is used.
Note: Lower case alpha is processed as binary data. URL information works best in upper case.
Note: At this time Kanji and Katakana are not supported by the EPIC 3000

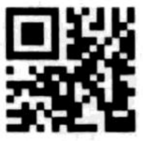
Micro QRCode is controlled by the same commands as QR code. Error correction level H is not available as the number of correction characters would exceed the input length.

Micro QRCode Commands

There are two version of the command, one is null terminated and the other allows a length to be specified (Note that the length is a two byte field as the symbol may contain more than 256 characters.

Function	**Micro QRCode**
ASCII	[ESC] b <37>{information} [NUL]
Hexadecimal	1BH 62H <37>{information} [NUL]
Decimal	<27> <98> <37>{information} [NUL]

Function	**Micro QRCode**
ASCII	[ESC] b <36><nL><nH><d$_1$> … <d$_n$>
Hexadecimal	1BH 62H <36><nL><nH><d$_1$> … <d$_n$>
Decimal	<27> <98> <36><nL><nH><d$_1$> … <d$_n$>



[ESC] b <36>WWW.TRANSACT-TECH.COM[NUL]

Figure 44 Data Micro QRCode Example

Note: The normal Select barcode Width and Select Barcode height commands do not affect Micro QRCode barcodes. The rules governing how the barcode is printed do not allow variations in the aspect ratios. The QRCode control commands that will allow some control over how the barcode is printed.

Aztec

The symbol is built on a square grid with a bulls-eye pattern at its centre for locating the code. Data is encoded in concentric square rings around the bulls-eye pattern. The central bulls-eye is 9×9 or 13×13 pixels, and one row of pixels around that encodes basic coding parameters, producing a "core" of 11×11 or 15×15 squares. Data is added in "layers", each one containing 2 rings of pixels, giving total sizes of 15×15, 19×19, 23×23, and so on.

The corners of the core include orientation marks, allowing the code to be read if rotated or reflected. Decoding begins at the corner with three black pixels, and proceeds clockwise to the corners with two, one and zero black pixels. The variable pixels in the central core encode the size, so it is not necessary to mark the boundary of the code with a blank "quiet zone", although some bar code readers require one.

The compact Aztec code core supports symbols from 15×15 (room for 13 digits or 12 letters) through 27×27. There is additionally a special 11×11 "rune" that encodes one byte of information. The full core supports sizes up to 151x151, which can encode 3832 digits, 3067 letters, or 1914 bytes of data.

The level of Reed–Solomon error correction is used for Aztec and the EPIC 3000 is configurable, to 10%, 23%, 36% or 50% of the data region. The recommended level is 23%.

Aztec Code Standard is ISO/IEC 24778 (published February 2008)
All 8-bit values can be encoded. The default interpretation for values 1[12]–127 is ASCII and for values 128–255, ISO 8859-1

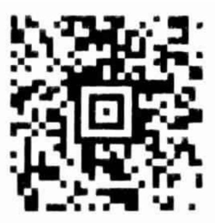


Figure 45 Aztec Symbol

Encoding: TransAct Technologies Inc. 20 Bomax Drive, Ithaca New York

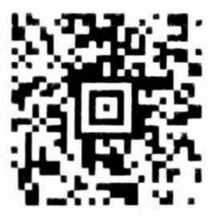
[12] The Aztec Standard allows values from 0 through 255. However, at this time the T480 will not handle a NUL,

Aztec Commands

There are two version of the command, one is null terminated and the other allows a length to be specified (Note that the length is a two byte field as the symbol may contain more than 256 characters.

Function	**Aztec**
ASCII	[ESC] b <30>{information} [NUL]
Hexadecimal	1BH 62H <30>{information} [NUL]
Decimal	<27> <98> <30>{information} [NUL]

Function	**Aztec**
ASCII	[ESC] b <29><nL><nH><d$_1$> … <d$_n$>
Hexadecimal	1BH 62H <29><nL><nH><d$_1$> … <d$_n$>
Decimal	<27> <98> <29><nL><nH><d$_1$> … <d$_n$>



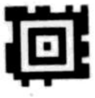
[ESC] b <30>TransAct Technologies Inc. 20 Bomax Drive, Ithaca New York[NUL]

Figure 46 Data Aztec Example

Note: The normal Select barcode Width and Select Barcode height commands do not affect Aztec barcodes. The rules governing how the barcode is printed do not allow variations in the aspect ratios. There are Aztec control commands that will allow some control over how the barcode is printed.

Function	**Aztec Rune**
ASCII	[ESC] b <31>{information} [NUL]
Hexadecimal	1BH 62H <31>{information} [NUL]
Decimal	<27> <98> <31>{information} [NUL]

Aztec Rune will encode one 8 bit byte represented by 3 numeric characters representing 0-255.



[ESC] b <31>25[NUL]

Figure 47 Aztec Rune Example

Note: Many barcode scanners do not support Aztec Rune barcodes.
Note: The intent is that this barcode encodes one 8 bit value

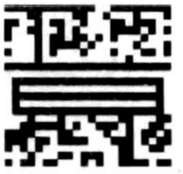
Code One

Code One was invented in 1992 and is the earliest public domain matrix barcode. It uses a finder pattern of horizontal and vertical bars crossing the middle of the symbol. The symbol can encode ASCII data, error correction data, function characters, and binary encoded data.

Code One is currently used in the health care industry for medicine labels and the recycling industry to encode container content for sorting.

.

Function	Code One
ASCII	[ESC] b <32> {information} [NUL]
Hexadecimal	1BH 62H <32> {information} [NUL]
Decimal	<27> <98> <32> {information} [NUL]



[ESC] b <32>12345678901234[NUL]

Figure 48 Code One Example

Note: Setting the Size of the Code One barcode will set the minimum matrix size. If the encoded data requires a larger matrix, the matrix size will be increased as required. (The default is 0 which will auto size the symbol)

Composite Barcodes

GS1 Composite barcode consists of a linear component, that encodes the item's primary data and an adjacent 2D composite component, that encodes supplementary data.

The linear component will be EAN GS1-128, EAN-8, EAN-13, UPC-A, UPC-E or any barcode in the Databar group.

The supplementary, 2D composite component will be one of the following:
Composite Code A barcode, which is derived from the MicroPDF417 specification, for EAN-8, EAN-13, UPC-A, UPC-E, GS1 DataBar barcodes, or Composite Code C barcode, which is derived from the PDF417 standard GS1-128 (EAN-128) barcodes.

MicroPDF417 barcodes are assigned a specific matrix of row/column combinations. This matrix limits the amount of data that may be encoded in 2D composites. The 2D composites can encode up to 56 numeric characters. Alpha characters and certain punctuation characters may also be encoded in the composite, but the maximum number of characters that can be encoded will be reduced significantly.

The composite component is referred to as the CC component and may be encoded as CC-A, CC-B or CC-C

-
  CC-A: a variant of MicroPDF417
- CC-B: a MicroPDF417 symbol with new encoding rules
- CC-C: Composite Code C barcodes, is only used as the composite for GS1-128 and is based on PDF417 standards, can encode approximately 800 characters.

The composite portion is saved in the composite buffer. This buffer is loaded using the composite data command and is processed as GA- data as the barcode is processed.

Composite data

Composite data generally follows the GS-1 standards, however, the GS-1 field processing and data compaction may be controlled by the setting the GS-1 options for Composite barcodes.

Function	**Composite data**
ASCII	[ESC] b @ {information} [NUL]
Hexadecimal	1BH 62H 40H {information} [NUL]
Decimal	<27> <98> <64>{information} [NUL]

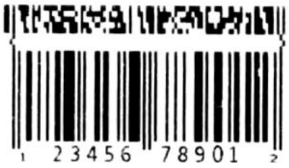
Note:: GS-1 field processing and data compaction occurs when the barcode is generated not when the secondary field is defined.
Note:: The secondary field will remain static until it is redefined.

An example would be:
[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
The data field is a valid GS-1 identification string.
"[01]12345678901234[11]100909[30]123456" will be processed as:
"(01)12345678901234(11)100909(30)123456"

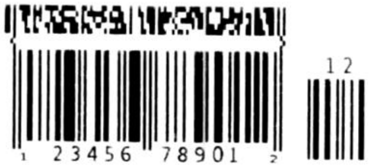
UPC A Composite

UPC A Composite uses a standard 11 digit UPC A code with composite data and optional Addenda data.

Function	**UPC A Composite**
ASCII	[ESC] b A {information} [NUL]
Hexadecimal	1BH 62H 41H {information} [NUL]
Decimal	<27> <98> <65>{information} [NUL]



[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <65> 12345678901[NUL]



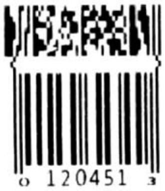
The same as above with a EAN 2 Addenda
[ESC] b <65> 12345678901+12[NUL]

Figure 49 UPC A Composite Example

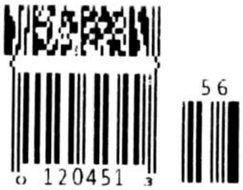
UPC E Composite

UPC E Composite uses a standard zero suppressed 11 digit UPC E code with composite data and optional Addenda data.

Function	UPC E Composite
ASCII	[ESC] b B {information} [NUL]
Hexadecimal	1BH 62H 42H {information} [NUL]
Decimal	<27> <98> <66>{information} [NUL]



[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <66>01210000345[NUL]



The same as above with a EAN 5 Addenda

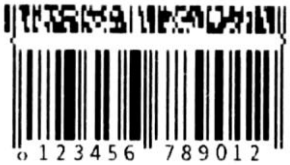
[ESC] b <66>01210000345+56[NUL]

Figure 50 UPC E Composite Example

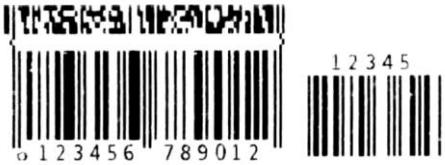
EANX Composite

EANX will process EAN 8, EAN 13 or EAN 14 based on the length of the linear data entered.

Function	EANx Composite
ASCII	[ESC] b C {information} [NUL]
Hexadecimal	1BH 62H 43H {information} [NUL]
Decimal	<27> <98> <67>{information} [NUL]



[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <67> 12345678901[NUL]



The same as above with a EAN 5 Addenda

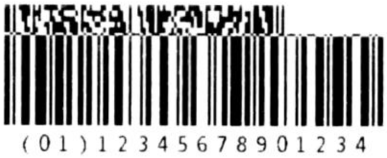
[ESC] b <67> 12345678901+12345[NUL]

Figure 51 EAN-13 Composite Example

EAN GS1-128 Composite

GS1-128 will process a GS1-128 barcode With an CC-C composite component.

Function	GS1-128 Composite
ASCII	[ESC] b D {information} [NUL]
Hexadecimal	1BH 62H 44H {information} [NUL]
Decimal	<27> <98> <68>{information} [NUL]



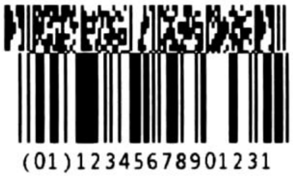
[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <68> [01]12345678901234[NUL]

Figure 52 EAN GS1-128 Composite Example

GS1-Databar-14 Composite

GS1- Databar 14 composite will process a GS1- Databar 14 barcode With an CC-A, or CC-B composite component.

Function	GS1-Databar 14 Composite
ASCII	[ESC] b E {information} [NUL]
Hexadecimal	1BH 62H 45H {information} [NUL]
Decimal	<27> <98> <69>{information} [NUL]



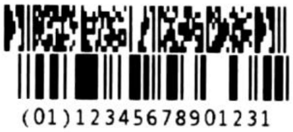
[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <69> 1234567890123[NUL]

Figure 53 GS1-Databar 14 Composite Example

GS1-Databar-Truncated Composite

GS1- Databar truncated composite will process a GS1- Databar truncated barcode With an CC-A, or CC-B composite component.

Function	**GS1-Databar Truncated 14 Composite**
ASCII	[ESC] b F {information} [NUL]
Hexadecimal	1BH 62H 46H {information} [NUL]
Decimal	<27> <98> <70>{information} [NUL]



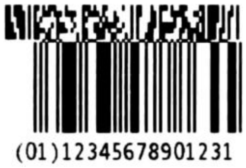
[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <70> 1234567890123[NUL]

Figure 54 GS1-Databar Truncated 14 Composite Example

GS1-Databar-Limited Composite

GS1- Databar limited composite will process a GS1- Databar limited barcode With an CC-A, or CC-B composite component.

Function	**GS1-Databar Limited Composite**
ASCII	[ESC] b G {information} [NUL]
Hexadecimal	1BH 62H 47H {information} [NUL]
Decimal	<27> <98> <71>{information} [NUL]



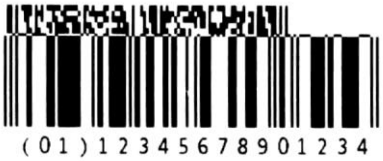
[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <71> 1234567890123[NUL]

Figure 55 GS1-Databar Limited Composite Example

GS1-Databar-Expanded Composite

GS1- Databar Expanded composite will process a GS1- Databar Expanded barcode With an CC-A, or CC-B composite component.

Function	GS1-Databar Expanded Composite
ASCII	[ESC] b H {information} [NUL]
Hexadecimal	1BH 62H 48H {information} [NUL]
Decimal	<27> <98> <72>{information} [NUL]



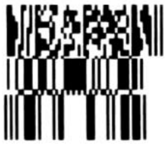
[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <72> [01]12345678901234[NUL]

Figure 56 GS1-Databar Expanded Composite Example

GS1-Databar-14 Stacked Composite

GS1- Databar Stacked composite will process a GS1- Databar Stacked barcode With an CC-A, or CC-B composite component.

Function	GS1-Databar Stacked Composite
ASCII	[ESC] b I {information} [NUL]
Hexadecimal	1BH 62H 49H {information} [NUL]
Decimal	<27> <98> <73>{information} [NUL]



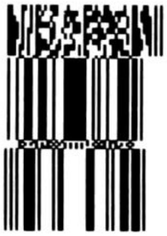
[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <73> 1234567890123[NUL]

Figure 57 GS1-Databar Stacked Composite Example

GS1-Databar-Stacked Omni Composite

GS1- Databar Stacked Omni composite will process a GS1- Databar Stacked barcode With an CC-A, or CC-B composite component.

Function	**GS1-Databar Stacked Omni Composite**
ASCII	[ESC] b J {information} [NUL]
Hexadecimal	1BH 62H 4AH {information} [NUL]
Decimal	<27> <98> <74>{information} [NUL]



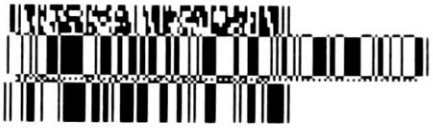
[ESC] b @ [01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <74> 1234567890123[NUL]

Figure 58 GS1-Databar Stacked Omni Composite Example

GS1-Databar-Expanded Stacked Composite

GS1- Databar Expanded Stacked composite will process a GS1- Databar Expanded Stacked barcode With an CC-A, or CC-B composite component.

Function	**GS1-Databar Expanded Stacked Composite**
ASCII	[ESC] b K {information} [NUL]
Hexadecimal	1BH 62H 4BH {information} [NUL]
Decimal	<27> <98> <75>{information} [NUL]



[ESC] b @
[01]12345678901234[11]100909[30]123456[NUL]
[ESC] b <75>
[01]12345678901234[11]100909[30]123456[NUL]

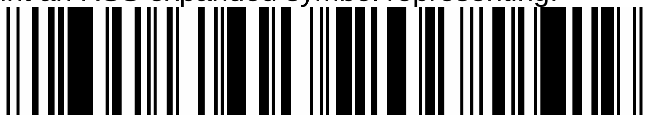
Figure 59 GS1-Databar Expanded Stacked Composite Example

GS-1 Barcodes

The GS1 General Specification defines a global standard for encoding data about products. The full specification is available from www.gs1uk.org web site. Data is encoded as a series of number pairs where the first number, usually shown in (brackets) is an application identifier (AI) (See Table 8 RSS GS1 AI Codes), and the second is a formatted representation of the data. For example (401)6773 can be read as "Consignment Number 6773" where the AI (401) signifies that the data is a consignment number. Note that for the EPIC 3000 AI data is entered using [square] brackets[13]. This allows rounded brackets to be included in the data which as allowed by the specification. When the barcode symbol are generated these square brackets are replaced by rounded brackets in HRI if HRI is active.

Note: RSS-14 barcodes are sometimes printed in a composite form. The EPIC 3000 does not support the composite RSS/EAN13 barcode.

The command to print an RSS expanded symbol representing:



(01)98898765432106(3202)012345(15)991231

Would be:
[ESC]b<23>[01]98898765432106[3202]012345[15]991231<0>
or for stacked version:
 [ESC]b<24>[01]98898765432106[3202]012345[15]991231<0>
Where:
(01) indicates that 98898765432106 is the Global Trade Item Number
(3202) Indicates that 012345 Net weight, of 123.45 pounds
(15) indicates 991231 is the Best Before Date (YYMMDD)

Symbology Name	**RSS-14**	**RSS-14 Truncated**	**RSS-14 Stacked**	**RSS-14 Stacked Omni-directional**	**RSS Limited**	**RSS Expanded**	**RSS Expanded Stacked**
Omni-Directionally Scan-able	Yes	No	No	Yes	No	Yes	Yes
Transmitted Data	AI plus 14-digits	AI plus 14-digits	AI plus 14-digits	AI plus 14-digits	AI plus 14-digits	Primary Identification & other AI element strings	Primary Identification & other AI element strings
Maximum Data Capacity	16 numeric (fixed)	16 numeric (fixed)	16 numeric (fixed)	16 numeric (fixed)	16 numeric (fixed)	74 numeric 41 alpha	74 numeric 41 alpha
Character Set	0 - 9	0 - 9	0 - 9	0 - 9	0 - 9	ASCII See Note	ASCII See Note
Number of rows	1	1	2	2	1	1	2-11
HRI Available	Yes	Yes	No	No	Yes	Yes	No

[13] It is possible for the T480 to accept () rather than [] to delaminate AI fields, however if that is done, the () characters may not be used in the data.

Note: The printer will encode the Value and insert latch sequences based on the ASCII Input. Refer to the ISO/IEC 24724 specification for more information.

Table 7 RSS Characteristics Summary

GS-1 AI definitions

Note: This is not a complete list and is not intended to replace the GS1 General Specification.

AI **See Notes[2,3]**	**Data Content**	**Format** **See Note[1]**	**FNC1** **Note[4]**	**Data Title**
00	SSCC (Serial Shipping Container Code)	n2+n18		SSCC
01	Global Trade Item Number (GTIN)	n2+n14		GTIN
02	GTIN of Contained Trade Items	n2+n14		CONTENT
10	Batch or Lot Number	n2+an..20	(FNC1)	BATCH/LOT
11[2]	Production Date (YYMMDD)	n2+n6		PROD DATE
12[2]	Due Date (YYMMDD)	n2+n6		DUE DATE
13[2]	Packaging Date (YYMMDD)	n2+n6		PACK DATE
15[2]	Best Before Date (YYMMDD)	n2+n6		BEST BEFORE or SELL BY
17[2]	Expiration Date (YYMMDD)	n2+n6		USE BY OR EXPIRY
20	Variant Number	n2+n2		VARIANT
21	Serial Number	n2+an..20	(FNC1)	SERIAL
22	Secondary Data Fields	n2+an..29	(FNC1)	QTY /DATE /BATCH
30	Count of Items (Variable Measure Trade Item)	n2+n..8	(FNC1)	VAR. COUNT
37	Count of Trade Items	n2+n..8	(FNC1)	COUNT
90	Information Mutually Agreed Between Trading Partners	n2+an..30	(FNC1)	INTERNAL
91 to 99	Company Internal Information	n2+an..30	(FNC1)	INTERNAL
240	Additional Item Identification	n3+an..30	(FNC1)	ADDITIONAL ID
241	Customer Part Number	n3+an..30	(FNC1)	CUST. PART NO.
242	Made-to-Order Variation Number	n3+n..6	(FNC1)	MTO VARIANT
250	Secondary Serial Number	n3+an..30	(FNC1)	SECONDARY SERIAL
251	Reference to Source Entity	n3+an..30	(FNC1)	REF. TO SOURCE
253	Global Document Type Identifier (GDTI)	n3+n13+n..17	(FNC1)	DOC. ID
254	GLN Extension Component	n3+an..20	(FNC1)	GLN EXTENSION
400	Customer's Purchase Order Number	n3+an..30	(FNC1)	ORDER NUMBER
401	Consignment Number	n3+an..30	(FNC1)	CONSIGNMENT
402	Shipment Identification Number	n3+n17	(FNC1)	SHIPMENT NO.
403	Routing Code	n3+an..30	(FNC1)	ROUTE
410	Ship to - Deliver to Global Location Number	n3+n13		SHIP TO LOC
411	Bill to - Invoice to Global Location Number	n3+n13		BILL TO
412	Purchased from Global Location Number	n3+n13		PURCHASE FROM
413	Ship for - Deliver for - Forward to Global Location Number	n3+n13		SHIP FOR LOC
414	Identification of a Physical Location - Global Location Number	n3+n13		LOC No
415	Global Location Number of the Invoicing Party	n3+n13		PAY TO
420	Ship to - Deliver to Postal Code Within a Single Postal Authority	n3+an..20	(FNC1)	SHIP TO POST
421	Ship to - Deliver to Postal Code with ISO Country Code	n3+n3+an..12	(FNC1)	SHIP TO POST
422	Country of Origin of a Trade Item	n3+n3	(FNC1)	ORIGIN
423	Country of Initial Processing	n3+n3+n..12	(FNC1)	COUNTRY – INITIAL PROCESS.
424	Country of Processing	n3+n3	(FNC1)	COUNTRY - PROCESS.
425	Country of Disassembly	n3+n3	(FNC1)	COUNTRY -

				DISASSEMBLY
426	Country Covering full Process Chain	n3+n3	(FNC1)	COUNTRY – FULL PROCESS
310n$_3$	Net weight, kilograms (Variable Measure Trade Item)	n4+n6		NET WEIGHT (kg)
311n$_3$	Length of first dimension, meters (Variable Measure Trade Item)	n4+n6		LENGTH (m)
312n$_3$	Width, diameter, or second dimension, meters (Variable Measure Trade Item)	n4+n6		WIDTH (m)
313n$_3$	Depth, thickness, height, or third dimension, meters (Variable Measure Trade Item)	n4+n6		HEIGHT (m)
314n$_3$	Area, square meters (Variable Measure Trade Item)	n4+n6		AREA (m2)
315n$_3$	Net volume, liters (Variable Measure Trade Item)	n4+n6		NET VOLUME (l)
316n$_3$	Net volume, cubic meters (Variable Measure Trade Item)	n4+n6		NET VOLUME (m3)
320n$_3$	Net weight, pounds (Variable Measure Trade Item)	n4+n6		WEIGHT (lb)
321n$_3$	Length or first dimension, inches (Variable Measure Trade Item)	n4+n6		LENGTH (i)
322n$_3$	Length or first dimension, feet (Variable Measure Trade Item)	n4+n6		LENGTH (f)
323n$_3$	Length or first dimension, yards (Variable Measure Trade Item)	n4+n6		LENGTH (y)
324n$_3$	Width, diameter, or second dimension, inches (Variable Measure Trade Item)	n4+n6		WIDTH (i)
325n$_3$	Width, diameter, or second dimension, feet (Variable Measure Trade Item)	n4+n6		WIDTH (f)
326n$_3$	Width, diameter, or second dimension, yards (Variable Measure Trade Item	n4+n6		WIDTH (y)
327n$_3$	Depth, thickness, height, or third dimension, inches (Variable Measure Trade Item)	n4+n6		HEIGHT (i)
328n$_3$	Depth, thickness, height, or third dimension, feet (Variable Measure Trade Item)	n4+n6		HEIGHT (f)
329n$_3$	Depth, thickness, height, or third dimension, yards (Variable Measure Trade Item)	n4+n6		HEIGHT (y)
330n$_3$	Logistic weight, kilograms	n4+n6		GROSS WEIGHT (kg)
331n$_3$	Length or first dimension, meters	n4+n6		LENGTH (m), log
332n$_3$	Width, diameter, or second dimension, meters	n4+n6		WIDTH (m), log
333n$_3$	Depth, thickness, height, or third dimension, meters	n4+n6		HEIGHT (m), log
334n$_3$	Area, square meters	n4+n6		AREA (m2), log
335n$_3$	Logistic volume, liters	n4+n6		VOLUME (l), log
336n$_3$	Logistic volume, cubic liters	n4+n6		VOLUME (m3), log
337n$_3$	Kilograms per square meter	n4+n6		KG PER m^2
340n$_3$	Logistic weight, pounds	n4+n6		GROSS WEIGHT (lb)
341n$_3$	Length or first dimension, inches	n4+n6		LENGTH (i), log
342n$_3$	Length or first dimension, feet	n4+n6		LENGTH (f), log
343n$_3$	Length or first dimension, yards	n4+n6		LENGTH (y), log
344n$_3$	Width, diameter, or second dimension	n4+n6		WIDTH (i), log
345n$_3$	Width, diameter, or second dimension	n4+n6		WIDTH (f), log
346n$_3$	Width, diameter, or second dimension	n4+n6		WIDTH (y), log
347n$_3$	Depth, thickness, height, or third dimension	n4+n6		HEIGHT (i), log

$348n_3$	Depth, thickness, height, or third dimension	n4+n6		HEIGHT (f), log
$349n_3$	Depth, thickness, height, or third dimension	n4+n6		HEIGHT (y), log
$350n_3$	Area, square inches (Variable Measure Trade Item)	n4+n6		AREA (i^2)
$351n_3$	Area, square feet (Variable Measure Trade Item)	n4+n6		AREA (f^2)
$352n_3$	Area, square yards (Variable Measure Trade Item)	n4+n6		AREA (y^2)
$353n_3$	Area, square inches	n4+n6		AREA (i^2), log
$354n_3$	Area, square feet	n4+n6		AREA (f^2), log
$355n_3$	Area, square yards	n4+n6		AREA (y^2), log
$356n_3$	Net weight, troy ounces (Variable Measure Trade Item)	n4+n6		NET WEIGHT (t)
$357n_3$	Net weight (or volume), ounces (Variable Measure Trade Item)	n4+n6		NET VOLUME (oz)
$360n_3$	Net volume, quarts (Variable Measure Trade Item)	n4+n6		NET VOLUME (q)
$361n_3$	Net volume, gallons U.S. (Variable Measure Trade Item)	n4+n6		NET VOLUME (g)
$362n_3$	Logistic volume, quarts	n4+n6		VOLUME (q), log
$363n_3$	Logistic volume, gallons U.S.	n4+n6		VOLUME (g), log
$364n_3$	Net volume, cubic inches (Variable Measure Trade Item)	n4+n6		VOLUME (i3), log
$365n_3$	Net volume, cubic feet (Variable Measure Trade Item)	n4+n6		VOLUME (f3), log
$366n_3$	Net volume, cubic yards (Variable Measure Trade Item)	n4+n6		VOLUME (y3), log
$367n_3$	Logistic volume, cubic inches	n4+n6		VOLUME (q), log
$368n_3$	Logistic volume, cubic feet	n4+n6		VOLUME (g), log
$369n_3$	Logistic volume, cubic yards	n4+n6		VOLUME (i3), log
$390n_3$	Applicable Amount Payable, local currency	n4+n..15	(FNC1)	AMOUNT
$391n_3$	Applicable Amount Payable with ISO Currency Code	n4+n3+n..15	(FNC1)	AMOUNT
$392n_3$	Applicable Amount Payable, single monetary area (Variable Measure Trade Item)	n4+n..15	(FNC1)	PRICE
$393n_3$	Applicable Amount Payable with ISO Currency Code (Variable Measure Trade Item)	n4+n3+n..15	(FNC1)	PRICE
7001	NATO Stock Number (NSN)	n4+n13	(FNC1)	NSN
7002	UN/ECE Meat Carcasses and Cuts Classification	n4+an..30	(FNC1)	MEAT CUT
7003	Expiration Date and Time	n4+n10	(FNC1)	EXPIRY TIME
703s	Approval Number of Processor with ISO Country Code	n4+n3+an..27	(FNC1)	PROCESSOR # s
8001	Roll Products (Width, Length, Core Diameter, Direction, Splices)	n4+n14	(FNC1)	DIMENSIONS
8002	Cellular Mobile Telephone Identifier	n4+an..20	(FNC1)	CMT No
8003	Global Returnable Asset Identifier (GRAI)	n4+n13+an..16	(FNC1)	GRAI
8004	Global Individual Asset Identifier (GIAI)	n4+an..30	(FNC1)	GIAI
8005	Price Per Unit of Measure	n4+n6	(FNC1)	PRICE PER UNIT
8006	Identification of the Components of a Trade Item	n4+n14+n2+n2	(FNC1)	GCTIN
8007	International Bank Account Number (IBAN)	n4+an..30	(FNC1)	IBAN
8008	Date and Time of Production	n4+n8+n..4	(FNC1)	PROD TIME

8018	Global Service Relation Number (GSRN)	n4+n18	(FNC1)	GSRN
8020	Payment Slip Reference Number	n4+an..25	(FNC1)	REF No
8100	GS1-128 Coupon Extended Code	n4+n6	(FNC1)	-
8101	GS1-128 Coupon Extended Code	n4+n1+n5+n4	(FNC1)	-
8102	GS1-128 Coupon Extended Code	n4+n1+n1	(FNC1)	-
8110	Coupon Code Identification for Use in North America	n4+an..30	(FNC1)	-

Notes: As of GS1 General Specifications Version 10.0 *Issue 1, Jan-2010*

Note$_1$ The first position indicates the length (number of digits) of the GS1 Application Identifier.
The following value refers to the format of the data content. The following conventions applied:
n numeric digit
an any valid character
n3 3 numeric digits, fixed length
n..3 up to 3 numeric digits
an..3 up to 3 valid characters

Note$_2$ If only year and month are available, DD must be filled with two zeroes.

Note$_3$ The fourth digit of this GS1 Application Identifier indicates the implied decimal point position.
Example:
3100 Net weight in kg without a decimal point
3102 Net weight in kg with two decimal points

Note$_4$ **(FNC1):** All GS1 Application Identifiers indicated with (FNC1) are defined as variable length and must be limited by a Function 1 Symbol Character unless this Element String is the last one to be encoded in the symbol. The printer will automatically insert FNC1's when and as required.

Table 8 RSS GS1 AI Codes

Note: It is beyond the scope of this document to define all the AI fields, there meaning, requirements and restrictions. The EPIC 3000 will do minimal validation of the AI fields to assure that the barcode can be generated. It will not assure that the barcode meets the GS-1 standard. Refer to the GS-1 General specification for additional information.

Mandatory AI Associations

Some AI fields must be associated with other fields. For example a date field must be associated with some item so cannot be used alone.

AI	Definition	AI	
01 or 02 with N1 = 9	Identification of a Variable Measure Trade Item	30, 3nnn1 or 3nnn2 or 8001	Mandatory association with variable measure information Only GS1-128, ITF-14, and GS1 DataBar Expanded Bar Code Symbols can encode a GTIN with N1 = 9.
02	Identification of logistic unit contents	00	Mandatory association with an SSCC (Serial Shipping Container Code)
02	Identification of logistic unit contents	37	Mandatory count of the contained trade items
10	Batch/lot number	01or 02	Mandatory association with a Global Trade Item Number (GTIN) or with the identification of logistic unit contents
11, 13, 15, 17	Dates	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
12	Due date	8020 and 415	Mandatory association with the payment slip reference number (AI (8020)) and the Global Location Number (GLN) of the invoicing party
20	Product variant	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
21	Serial number	01	Mandatory association with a GTIN of a single trade item (a serial number cannot apply to a grouping of trade items)
22	Secondary data health industry	01	Mandatory association with a GTIN
240	Additional product identification	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
241	Customer part number	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
242	Made-to-Order Variation Number	01 or 02 with N1 = 9	Mandatory association with a GTIN-14 with Indicator Digit 9 represents a Custom Industrial Supply Item
250	Secondary serial number	01	Mandatory association with a GTIN (a secondary serial number cannot apply to a grouping of trade items)
251	Reference to source entity	01	Mandatory association with GTIN of the trade item
254	Extension component of a GLN	414	Mandatory association with AI (414). Only GS1-128, GS1 DataBar Expanded symbologies, and EPC RFID tags are valid. This is used with GLN and not GTIN.
30	Variable count	01 or 02	Mandatory association with a variable measure GTIN (e.g., a GTIN-14 starting with the digit 9) or the identification of variable measure content of a logistic unit
$3nnn_1$	Trade measures that cannot be summed	01	Mandatory association with a variable measure GTIN (e.g., a GTIN-14 starting with the digit 9)
$3nnn_2$	Trade measures that can be summed	01 or 02	Mandatory association with a variable measure GTIN (e.g., a GTIN-14 starting with the digit 9) or the identification of variable measure content of a logistic unit
$3nnn_3$	Logistic measures	00 or 01	Mandatory association with an SSCC or a variable measure GTIN (e.g., a GTIN-14 starting with the digit 9)
337n	Kilograms per square metre	01	Mandatory association with a GTIN
37	Count of units	02	Mandatory association with the identification of

	contained		logistic unit contents
390n	Amount payable – single monetary area	8020 and 415	Mandatory association with the payment slip reference number, AI (8020), and the GLN of the invoicing party
391n	Amount payable – with ISO currency code	8020 and 415	Mandatory association with the payment slip reference number, AI (8020), and the GLN of the invoicing party
392n	Amount payable – single monetary unit	01	Mandatory association with variable measure GTIN (e.g., a GTIN-14 starting with the digit 9)
393n	Amount payable – with ISO currency code	01	Mandatory association with variable measure GTIN (e.g., a GTIN-14 starting with the digit 9)
403	Routing code	00	Mandatory association with an SSCC
415	GLN of the invoicing party	8020	Mandatory association with payment slip reference number, AI (8020)
422	Country of origin	01 or 02	Mandatory association with a GTIN
423	Country of initial	01 or 02	Mandatory association with a GTIN or with the processing identification of logistic unit contents
424	Country of processing	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
425	Country of disassembly	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
426	Country of full processing	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
7001	NATO stock number	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
7002	UN/ECE meat carcasses and cuts classification	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
703(s)	Approval number of processor	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
8001	Variables of roll products	01	Mandatory association with a variable measure GTIN (e.g., an GTIN-14 starting with the digit 9)
8005	Price per unit of measure	01 or 02 with N1 = 9	Mandatory association with a variable measure GTIN or the identification of variable measure content of a logistic unit
8007	International Bank Account Number, AI (8007)	8020 and 415	Mandatory association with the payment slip reference number, AI (8020), and the GLN of the invoicing party
8008	Date and time of production	01 or 02	Mandatory association with a GTIN or with the identification of logistic unit contents
8020	Payment slip reference number, AI (8020)	415	Mandatory association with the GLN of the invoicing party

Note $_1$ Is (3nnn) where the first three digits are 312, 313, 324, 325, 326, 327, 328, and 329

Note $_2$ Is (3nnn) where the first three digits are 310, 311, 314, 315, 316, 320, 321, 322, 323, 350, 351, 352, 356, 357, 360, 361, 364, 365, and 366

Note $_3$ Is (3nnn) where the first three digits are 330, 331, 332, 333, 334, 335, 336, 340, 341, 342, 343, 344, 345, 346, 347, 348, 349, 353, 354, 355, 362, 363, 367, 368, and 369

Table 9 Mandatory AI Code Associations

Note: It is beyond the scope of this document to define all the AI fields, there meaning, requirements and restrictions. The EPIC 3000 will do minimal validation of the AI fields to assure that the barcode can be generated. It will not assure that the barcode meets the GS-1 standard. Refer to the GS-1 General specification for additional information.

Invalid AI associations

Some AI fields cannot be used with other AI fields.

AI	Definition	AI	Definition	
01	Identification of a trade item	01	Identification of a trade item	Duplicate Global Trade Item Numbers(GTINs) with different values
01	Identification of a trade item	02	Identification of logistic unit contents	AI (02) must not be used for the identification of trade items contained in a trade item.
01	Identification of a trade item	37	Count of units contained	The count of units contained would duplicate the master data of the GTIN. AI (37) may only be used with AI (02).
22	Secondary data for the health industry	30	Count	Duplicate counts with different values
22	Secondary data for the health industry	10	Batch/lot number	Duplicate lot numbers with different values
22	Secondary data for the health industry	17	Expiration date	Duplicate expiration date with different values
22	Secondary data for the health industry	21	Serial number	Duplicate serial numbers with different values
242	Made-to-Order Variation	01 or 02 with N1 not equal to 9	Identification of a Variable Measure Trade Item	Made-to-Order Variation can only be used with a GTIN-14, Indicator digit 9. This represents a Custom Industrial Supply Item
420	Ship to postal code, single postal authority	421	Ship to postal code with ISO country code	Only one ship to postal code may be applied on an item
422	Country of origin of a trade item	426	Country of full processing	Duplication of country of origin of a trade item (covered by country of full processing)
423	Country of initial processing	426	Country of full processing	Duplication of country of initial processing(covered by country of full processing)
424	Country of processing	426	Country of full processing	Duplication of country of processing(covered by country of full processing)
425	Country of disassembly	426	Country of full processing	Duplication of country of disassembly(covered by country of full processing)
390n	Amount payable–single monetary area	391n	Amount payable – with ISO currency code	Only one amount payable Element String may be applied on a payment slip
392n	Amount Payable for a Variable Measure Trade Item – Single Monetary Area	393n	Amount Payable for a Variable Measure Trade Item and ISO Currency Code	Only one amount payable Element String may be applied on a Variable Measure Trade Item.
8006	Component identification	01	Identification of a trade item	Other GTINs cannot be used with AI (8006). The trade item is identified by a GTIN contained in the AI (8006).

Table 10 Invalid AI Code Associations

Note: It is beyond the scope of this document to define all the AI fields, there meaning, requirements and restrictions. The EPIC 3000 will do minimal validation of the AI fields to assure that the barcode can be generated. It will not assure that the barcode meets the GS-1 standard. Refer to the GS-1General specification for additional information.

GS-1 AI fields with a Check Digit

Refer to the GS-1 General specification for the method used to calculate the check digit. In some cases the check digit is not encoded into the barcode. However, the check digit is always shown in the human readable interpretation and transmitted by the decoder even though it is not explicitly encoded in RSS-14 and RSS Limited symbols. The printer does not check the check digit and whatever is passed in will be printed in the HRI. The printer however will format the barcode without the check digit and the barcode reader will reinsert it into the resulting data. The result is that if the check digit passed in is not correct, it will be printed in the HRI but may be reported as a different character by the reader.

To make check digit generation easier, the printer will (optionally) automatically generate and insert a check digit in several AI fields. To trigger automatic insertion, the GS-1 AI check digit option should be activated and a “*” character used in the location of the check digit.

AI	Data Content	Format	Check Digit Location	Data Title
00	SSCC (Serial Shipping Container Code)	n2+n18	n18	SSCC
01	Global Trade Item Number (GTIN)	n2+n14	n12, n13, or n14	GTIN
02	GTIN of Contained Trade Items	n2+n14	n14	CONTENT
253	Global Document Type Identifier (GDTI)	n3+n13+n..17	n13	DOC. ID
402	Shipment Identification Number	n3+n17	n17	SHIPMENT NO.
410	Ship to - Deliver to Global Location Number	n3+n13	n13	SHIP TO LOC
411	Bill to - Invoice to Global Location Number	n3+n13	n13	BILL TO
412	Purchased from Global Location Number	n3+n13	n13	PURCHASE FROM
413	Ship for - Deliver for - Forward to Global Location Number	n3+n13	n13	SHIP FOR LOC
414	Identification of a Physical Location - Global Location Number	n3+n13	n13	LOC No
415	Global Location Number of the Invoicing Party	n3+n13	n13	PAY TO
8003	Global Returnable Asset Identifier (GRAI)	n4+n13+an..16	n13	GRAI
8018	Global Service Relation Number (GSRN)	n4+n18	n18	GSRN

Table 11 AI’s supported by automatic check digit generation

Note: Automatic Check Character generation for Price/Weight, Four-Digit Price, and Five-Digit Price fields are not supported by automatic check character generation.
Note: If additional AI fields with check characters are added to the GS1 standards they will not be processed by the printer unless the printer’s firmware is updated.

Controlling Barcodes

Unified Commands

To making control of linear and two dimensional barcode more consistent, all barcode control commands will follow a similar format. The

Function	Control bar code
ASCII	[ESC] [EM] <b> <f> <v>
Hexadecimal	1BH 19H <b> <f> <v>
Decimal	<27> <25> <b> <f> <v>

Where:

<b>	Barcode type	
	b, 62H, 98D	General Barcodes
	d, 64H, 100D	Datamatrix Barcodes
	r, 72H, 114D	GS-1 Databar (RSS) Barcodes
	4, 34H, 52D	Code 49 barcodes
	6, 36H, 54D	Code 49 barcodes
	a, 61H, 97D	Aztec Barcodes
	m, 6DH, 109D	Maxicode Barcodes.
	q, 71Hm 113D	QRCode
	E, 45H, 69D	PDF417
	c,63H,99D	Composite Barcodes.
<f>	Common Format Parameter to all barcodes	
	W, 57H, 87D	Minimum bar width or Scale
	J, 4AH, 74D	Justification 0 = Left, 1 = Center, 3 = Right
	V, 56H, 86D	Vertical Height (value*4) dots at 203 Dots per inch
	G, 47H, 71D	GS1 Mode.
	P, 50H, 80D	HRI Position 0 – Off, 1 = Top, 2 = Bottom, 3 = Both
	F, 46H, 70D	HRI Font
	H, 48H, 72D	Space above and below the barcode in dots
	C, 43H, 67D	Composite Secondary Mode
	Additional barcode specific controls are available	
<v>	Value of the parameter.	

Description The [ESC] [EM] <b> <p> <n> command sets general barcode format parameters. In some cases a specific barcode may set a barcode specific parameter. For example the general element width value is generally not optimal for 2D barcodes so each 2D barcode will use its own parameter.

General Settings

f = W, 57H, 87D Minimum bar width or scale
Minimum bar width or Scale

f = J, 4AH, 74D Justification
0 = Left, 1 = Center, 3 = Right

f = V, 56H, 86D
Vertical Height
Height (value*4) dots at 203 Dots per inch

f = G, 47H, 71D
GS1-Mode
The GS1-Mode is configurable as to how closely the GS1 specification is followed. This allows future changes in GS1 compaction to be processed without error. In some cases this parameter will activate Enhanced features.

The values for GS1-Mode are bit values and are as follows:

0x01 GS1 AI and Compaction Active (If this bit is not set no GS-1 AI processing is performed. The input data is not scanned nor are the AI fields preprocessed or compressed. Not setting this option may produce invalid GS1 barcodes.

0x02 AI field lengths are not strictly enforced.

0x04 No GS1 faults are generated. All AI fields and compaction that can be processed will be but unknown fields or fields in error will not stop the barcode generation. Setting this may produce invalid barcodes.

0x08 AI (01) data field may be passed 12, 13, or 14 bytes. 12 and 13 byte fields will be expanded to 14 with leading zeros.

0x10 The check digit in AI fields (00), (01), (02), (253), (402), (410) trough (415), (8003) and (8018) may be replaced with a '*' and the EPIC 3000 will replace the '*' with a check digit for that field calculated as defined in the GS1 general specification.

0x20 Use () to delineate AI fields rather than []

Default The default value for most barcodes is 0x19 which provides variation of the AI (01) length and optional automatic check digit generation. If you require strict compliance with the GS1 general specification, set the GS1-Mode to 0x01.

f = P, 50H, 80D
HRI Position
f =0 – Off, f = 1 - Top, f =2 - Bottom, f =3 - Both

f = F, 46H, 70D
HRI Font
f = 0 Medium, f=1 Larger, f- 2 Smaller (Note: Fonts may be redefined by using the change legacy font command. See page 215)

f = H, 48H, 72D
Space above and below the barcode between the barcode and any HRI if active. This is in dots.
Typically 2 dots although some 2D barcodes require more.
If HRI is inactive, it is up to the user to enforce any extra quite zone between the barcode and any user data.

f = C, 43H, 67D
Composite Secondary Mode
f = 0=Auto, 1=CC-A, 2=CC-B, or 3=CC-3

Barcode Control Summery Chart

Barcode Type <b>	Format Parameter <f>															
Barcode	W	J	V	G	P	F	H	K	X	Y	C	R	E	Q	S	M
Interleaved 2 of 5	b	b	b		b	b	b									
Code 39	b	b	b		b	b	b									
Code 128	b	b	b		b	b	b									
UPC A	b	b	b		b	b	b				b					
UPC E	b	b	b		b	b	b				b					
EAN-13	b	b	b		b	b	b				b					
EAN-8	b	b	b		b	b	b				b					
Code 93	b	b	b		b	b	b									
Code 93	b	b	b		b	b	b									
Codabar	b	b	b		b	b	b									
Data Bar	r b	r b	r b	r	r b	r b	r b									
Data Bar Stacked	r b	r b	r b	r	r	r	r b	r			b				r	
EAN-128	b	b	b	b	b	b	b				b					
EAN-14	b	b	b	b	b	b	b									
ITF-14	b	b	b		b	b	b									
Code 49	4	4 b	4 b	4	4	4	4 b									
Code 16K	6	6 b	6 b	6	6	6	6 b									
PDF417				E					E	E	E	E	E			
Maxicode	m	m b		m	m	m	m									m
Datamatrix	d			d	d	d	d						d	d		d
QRCode	q	q b		q	q	q	q				q		q			q
Aztec	a	a b		a	a	a	a						a			A
Code One	1	1b			1	1	b									1
Composite	c	b		c	*	*	b	c								c

Table 12 Barcode Control Commands

Note: If there are two letters listed, either command may be used. Both will set a common value.

PDF417 Print Options

Function **PDF 417 bar code control**

ASCII [ESC] [EM] E <f> <v>
Hexadecimal 1BH 19H 45H <f> <v>
Decimal <27> <25> <69><f> <v>
IPCL None
Description This command alters the way PDF 417 barcodes are generated and printed.
Where f = Feature to control and v = the value of the feature.

f = X, 58H, 88D *Set encoding X aspect.*
2 ≤v ≤6
[Default] v = 3

Note: the X aspect is equivalent to the module width and affects the maximum number of columns that will fit in the print zone.

f = Y, 59H, 89D *Set encoding Y aspect.*
2 ≤v ≤8
[Default] v = 3
f = H, 48H, 72D Blank Space before and after the barcode
0 ≤v ≤255. Default is 8 dots
f = C, 43H, 67D Set encoding columns
0 ≤v ≤30 (Values > 8 generally will not fit in the print zone)
[Default] v = 0 (automatic mode)
If v= 0 then automatic processing is used.
When automatic processing is specified, the number of columns is calculated with the number of code words based on the range of the printable area**.**

f = R, 52H, 82D Set encoding rows
v = 0 or 3 ≤v ≤90
[Default] v = 0
f v = 0 then automatic processing is used.
When automatic processing is specified, the number of rows is calculated with the number of code words or the range of the printable area.

Note: If the number of rows is specified, the number of columns must also be specified.
Note: The number of rows times the number of columns must not exceed 928.

Typically the row and columns should be set to 0 so that auto encoding will be used. The X and Y aspect represent the number of dots horizontally and vertically to form the smallest image element. Values of 2 for each produce very small elements, and are probably too small unless good paper is used. The defaults are 3 by 9, which produce easily readable barcodes.

f = E, 45H, 69D *Set error correction level.*
If v between 1 and 40 it is interpreted as a percentage of the data.
If v is between 48 and 56 it is set to a specific level of 0 to 8.

Error correcting levels are selected using one of two methods. The first is a fixed level.

v	Level	Code Word
48	Level 0	2
49	Level 1	4
50	Level 2	8
51	Level 3	16
52	Level 4	32
53	Level 5	64
54	Level 6	128
55	Level 7	256
56	Level 8	512

The second way is to determine correction level based on a percent of the encoded data in the barcode where the value v is a percent between 1% and 40%. The correction level is determined by calculating a correction factor based on the formula: Cf = ((v * Encoded Length) + 50)/100 then using Cf in the following table to set the correction level.

Cf	Level	Code Words
0-3	Level 1	4
4-10	Level 2	8
11-20	Level 3	16
21-45	Level 4	32
46-100	Level 5	64
101-200	Level 6	128
201-400	Level 7	256
401 Up	Level 8	512

For example: if the encoded data length is 80 bytes and v = 10% then Cf = ((10 * 80) +50)/100 or Cf= 8. Therefore Cf is between 4 and 10 so Level 2 correction would be used.

The encoded length is not the same as the input length. The input data is first compressed into Text, Octet and Numeric data and the resulting encoded data length is then used to calculate the correction level.

Note: Setting the Error correction higher will increase the size of the barcode and reduce the available space for data.
Note: The encoding may be set to 0 which will do a 10% error correction however this is not the same as 1. 0 does not do round off and levels 0 or 8 are never used. (This mode provides compatibility with legacy products.)

f = P, 50H, 80D	HRI Position f =0 – Off, f = 1 - Top, f =2 - Bottom, f =3 - Both
f = F, 46H, 70D	HRI Font

f = 0 Medium, f=1 Larger, f- 2 Smaller (Note: Fonts may be redefined by using the change legacy font command.)

Data Matrix Print Options

Function **Data Matrix bar code control**
ASCII [ESC] [EM] d <f> <v>
Hexadecimal 1BH 19H 64H <f> <v>
Decimal <27> <25> <100><f> <v>
IPCL None
Description This command alters the way Data Matrix barcodes are generated and printed.
Where f = Feature to control and v = the value of the feature.

f = H, 48H, 72D Blank Space before and after the barcode
f = M, 4DH, 77 Set minimum matrix size. v = 1-30, 0 sets to auto.

Size(v)	Symbol Size	Size(v)	Symbol Size	Size(v)	Symbol Size
0	Auto	11	36 x 36	21	104 x 104
1	10 x 10	12	40 x 40	22	120 x 120
2	12 x 12	13	44 x 44	23	132 x 132
3	14 x 14	14	48 x 48	24	144 x 144
4	16 x 16	15	52 x 52	25	8 x 18
5	18 x 18	16	64 x 64	26	8 x 32
6	20 x 20	17	72 x 72	27	12 x 26
7	22 x 22	18	80 x 80	28	12 x 36
8	24 x 24	19	88 x 88	29	16 x 36
9	26 x 26	20	96 x 96	30	16 x 48
10	32 x 32				

Figure 60 ECC – 200 Size options

Note: If the data will not fit in the selected size, the printer will revert to auto mode.

f = Q, 52H, 82 if v = 1 Only square matrix sizes will be selected in auto mode y. v = 0-1
f = W 57H, 87D Set the minimum element width and height.
v = 1-10, 6 is the default.
A width of 1 or 2 may be unreadable.
It the barcode will not fit in the print zone, the printer will automatically reduce the width until the barcode will fit. If the barcode will not fit at a width of 1, the printer will not print the barcode.
f = G, 47H, 71D) GS1 Mode. (See General description above)

Note: If in GS1 mode, a FNC1 is added to the beginning of the symbol and that the compression is slightly altered. GS1 data formatting and compaction are active and the input must follow the GS1 rules.

f = P, 50H, 80D HRI Position
f =0 – Off, f = 1 - Top, f =2 - Bottom, f =3 - Both
f = F, 46H, 70D HRI Font
f = 0 Medium, f=1 Larger, f- 2 Smaller (Note: Fonts may be redefined by using the change legacy font command.)

Code 49 Print Options

Function Code 49 Options

ASCII **[ESC] [EM] 4 <f> <v>**

Hexadecimal 1BH 19H 34H <f> <v>

Decimal <27> <25> <52><f> <v>

IPCL None

Description This command alters the way Code49 barcodes are generated and printed.

Where f = Feature to control and v = the value of the feature.

f = W 57H, 87D	Set the minimum element width. v = 1-10, 3 is the default. A width of 1 may be unreadable. It the barcode will not fit in the print zone, the printer will automatically reduce the width until the barcode will fit. If the barcode will not fit at a width of 1, the printer will not print the barcode.
f = H, 48H, 72D	Blank Space between the barcode and the HRI Default 8
f = G, 47H, 71D	GS1 Mode. (See General description above) By default Databar barcodes use GS1 formatting.
f = P, 50H, 80D	HRI Position f =0 – Off, f = 1 - Top, f =2 - Bottom, f =3 - Both
f = F, 46H, 70D	HRI Font f = 0 Medium, f=1 Larger, f- 2 Smaller (Note: Fonts may be redefined by using the change legacy font command.)

Code 16K Print Options

Function Code 16K Options

ASCII **[ESC] [EM] 6 <f> <v>**

Hexadecimal 1BH 19H 36H <f> <v>
Decimal <27> <25> <54><f> <v>
IPCL None
Description This command alters the way code16K barcodes are generated and printed.
Where f = Feature to control and v = the value of the feature.

f = W 57H, 87D	Set the minimum element width. v = 1-10, 3 is the default. A width of 1 may be unreadable. It the barcode will not fit in the print zone, the printer will automatically reduce the width until the barcode will fit. If the barcode will not fit at a width of 1, the printer will not print the barcode.
f = H, 48H, 72D	Blank Space between the barcode and the HRI Default 8 I
f = G, 47H, 71D	GS1 Mode. (See General description above) By default Databar barcodes use GS1 formatting.
f = P, 50H, 80D	HRI Position f =0 – Off, f = 1 - Top, f =2 - Bottom, f =3 - Both
f = F, 46H, 70D	HRI Font f = 0 Medium, f=1 Larger, f- 2 Smaller (Note: Fonts may be redefined by using the change legacy font command.)

QRCode Print Options

Function **QRCode code control**

ASCII [ESC] [EM] q <f> <v>

Hexadecimal 1BH 19H 71H <f> <v>

Decimal <27> <25> <113><f> <v>

IPCL None

Description This command alters the way QRCode barcodes are generated and printed.

Where f = Feature to control and v = the value of the feature.

f = W 57H, 87D Set the minimum element width and height.
v = 1-10, 4 is the default.
A width of 1 may be unreadable.
It the barcode will not fit in the print zone, the printer will automatically reduce the width until the barcode will fit. If the barcode will not fit at a width of 1, the printer will not print the barcode.

f = E, 45H, 69D QRCode Error Correction.
0-4 are accepted.
0 = Auto
1 = L or 7%
2 = M or 15%
3 = Q or 25%
4 = H or 30%

f = H, 48H, 72D Blank Space between the barcode and the following data
Default 8

f = M, 4DH, 77D QRCode Matrix Size. 0-40 are accepted. 0 = Auto

f = G, 47H, 71D) GS1 Mode. Default = 0 (See General description above)

Note: If in GS1 mode, a FNC1 is added to the beginning of the symbol. GS1 data formatting and compaction are active and the input must follow the GS1 rules.
Note: QR Code Model 1 is obsolete, only Model 2 is supported

f = P, 50H, 80D HRI Position
f =0 – Off, f = 1 - Top, f =2 - Bottom, f =3 - Both

f = F, 46H, 70D HRI Font
f = 0 Medium, f=1 Larger, f- 2 Smaller (Note: Fonts may be redefined by using the change legacy font command.)

Maxicode Print Options

Function **Maxicode control**

ASCII [ESC] [EM] m <f> <v>
Hexadecimal 1BH 19H 6DH <f> <v>
Decimal <27> <25> <109><f> <v>
IPCL None
Description This command alters the way Maxicode barcodes are generated and printed.
Where f = Feature to control and v = the value of the feature.

f = W 57H, 87D Set Scale. The scale may be set from 0 to 10, however, only values of 8, 6, 4, and 2 will be used. (default is 4)
(0 will default to 4 and odd numbers will round down.)
A scale of 2 may be unreadable.
It the barcode will not fit in the print zone, the printer will automatically reduce the scale until the barcode will fit. If the barcode will not fit at a width of 1, the printer will not print the barcode.

f = H, 48H, 72D Blank Space between the barcode and the following data
Default 8

f = M, 4DH, 77D Mode 2, 3, 4 and 5 are accepted, 0 = Auto

f = P, 50H, 80D HRI Position
f =0 – Off, f = 1 - Top, f =2 - Bottom, f =3 - Both

f = F, 46H, 70D HRI Font
f = 0 Medium, f=1 Larger, f- 2 Smaller (Note: Fonts may be redefined by using the change legacy font command.)

Aztec Print Options

Function	**Aztec code control**
ASCII	[ESC] [EM] a <f> <v>
Hexadecimal	1BH 19H 61H <f> <v>
Decimal	<27> <25> <97><f> <v>
IPCL	None
Description	This command alters the way Aztec barcodes are generated and printed.
Where	f = Feature to control and v = the value of the feature.

f = W 57H, 87D	Set the minimum element width. v = 1-10, 6 is the default and height. A width of 1 may be unreadable. It the barcode will not fit in the print zone, the printer will automatically reduce the width until the barcode will fit. If the barcode will not fit at a width of 1, the printer will not print the barcode.
f = M, 4DH, 77D	QRCode Matrix Size. 1-36 are accepted. 0 – Auto Size 1-4 are Compact and 5-36 are normal.
f = E, 45H, 69D	QRCode Error Correction. 1-4 are accepted. 0 = default to level 2
f = H, 48H, 72D	Blank Space between the barcode and the following data Default 8
f = G, 47H, 71D)	GS1 Mode. (See General description above) If in GS1 mode the symbol start with an FNC1 and that the compression is slightly altered. GS1 data formatting and compaction are active and the input must follow the GS1 rules.
f = P, 50H, 80D	HRI Position f =0 – Off, f = 1 - Top, f =2 - Bottom, f =3 - Both
f = F, 46H, 70D	HRI Font f = 0 Medium, f=1 Larger, f- 2 Smaller (Note: Fonts may be redefined by using the change legacy font command.)

Code One Print Options

Function	**Code One code control**
ASCII	[ESC] [EM] 1 <f> <v>
Hexadecimal	1BH 19H 31H <f> <v>
Decimal	<27> <25> <49><f> <v>
IPCL	None
Description	This command alters the way Aztec barcodes are generated and printed.
Where	f = Feature to control and v = the value of the feature.

f = W 57H, 87D	Set the minimum element width. v = 1-10, 6 is the default and height. A width of 1 may be unreadable. It the barcode will not fit in the print zone, the printer will automatically reduce the width until the barcode will fit. If the barcode will not fit at a width of 1, the printer will not print the barcode.
f = M, 4DH, 77D	Code One Matrix Size. 1-10 are accepted. 0 – Auto Size 1-8 are 1A – 1H and 9 - 10 allow other shapes.
f = H, 48H, 72D	Blank Space between the barcode and the following data Default 8
f = G, 47H, 71D)	GS1 Mode. (See General description above) If in GS1 mode the symbol start with an FNC1 and that the compression is slightly altered. GS1 data formatting and compaction are active and the input must follow the GS1 rules.
f = P, 50H, 80D	HRI Position f =0 – Off, f = 1 - Top, f =2 - Bottom, f =3 - Both
f = F, 46H, 70D	HRI Font f = 0 Medium, f=1 Larger, f- 2 Smaller (Note: Fonts may be redefined by using the change legacy font command.)

Databar (RSS) Print Options

Function **Databar Options**

ASCII **[ESC] [EM] r <f> <v>**

Hexadecimal 1BH 19H 72H <f> <v>
Decimal <27> <25> <114><f> <v>
IPCL None
Description This command alters the way Databar barcodes are generated and printed.
Where f = Feature to control and v = the value of the feature.

f = W 57H, 87D	Set the minimum element width. v = 1-10, 3 is the default. A width of 1 may be unreadable. It the barcode will not fit in the print zone, the printer will automatically reduce the width until the barcode will fit. If the barcode will not fit at a width of 1, the printer will not print the barcode.
f = S, 53H, 83D	Expanded Stack character segment width. v = 2-10 Default 3
f = K, 4BH, 75D	Activate Stacked symbol HRI. 0 = off, 1 = on. By convention stacked barcode are not printed with HRI, however if required they may be activated
f = G, 47H, 71D	GS1 Mode. Default = 0x19 (See General description above) By default Databar barcodes use GS1 formatting.

Composite Barcode Print Options

Function **Composite Barcode control**

ASCII [ESC] [EM] c <f> <v>
Hexadecimal 1BH 19H 63H <f> <v>
Decimal <27> <25> <99><f> <v>
IPCL None
Description This command alters the way Composite barcodes are generated and printed.
Where f = Feature to control and v = the value of the feature.

f = W 57H, 87D	Set the minimum element width. v = 1-3, 2 is the default and height. A width of 1 may be unreadable. It the barcode will not fit in the print zone, the printer will automatically reduce the width until the barcode will fit. If the barcode will not fit at a width of 1, the printer will not print the barcode.
f = M, 4DH, 77D	Composite CC mode. 1-3 are accepted. 0 – Auto 0=Auto, 1=CC-A, 2=CC-B, or 3=CC-C.
f = G, 47H, 71D)	GS1 Mode. (See General description above) If in GS1 mode the symbol start with an FNC1 and that the compression is slightly altered. GS1 data formatting and compaction are active and the input must follow the GS1 rules.
f = K, 4BH, 75D	HRI On and Font. f =0 – Off, F=1 - Medium, f=2 - Larger, f=3 - Smaller (Note: Fonts may be redefined by using the change legacy font command. Note that this command referances font offset by 1)

Note : The normal HRI options do not affect Composite barcodes. Composite barcode HRI may be on or off and you may select the font. HRI is only available below barcode.

Legacy Commands

Function **Set bar code height**

ASCII [ESC] [EM] B <n>
Hexadecimal 1BH 19H 42H <n>
Decimal <27> <25> <66> <n>
IPCL &%BH <m>
Description The [ESC] [EM] B <n> command sets the bar code height where <n>*24 are the number dots. The default is n = 4, and results in a barcode that is about 0.47 inches high.

Function **Set bar code width**

ASCII [ESC] [EM] W <n>
Hexadecimal 1BH 19H 57H <n>
Decimal <27> <25> <87> <n>
IPCL &%BW <m>
Description The [ESC] [EM] w <n> command sets the minimum bar width. The value may be between 1 and 8. The default is 3.

Note: A value of 1 may result in barcodes that are unreadable with some readers.

An alternate version of this command is available for Interleaved 2 of 5 barcodes. Interleave 2 of 5sometimes require that the bar ratios be altered for unconventional barcode readers.

ASCII [ESC] [EM] W <0><narrow><wide>
Hexadecimal 1BH 19H 57H <0><narrow><wide>
Decimal <27> <25> <87> <0><narrow><wide>
Description The [ESC] [EM] w <0><narrow><wide> only affects the 2 of 5 barcode and sets the narrow bar width to <narrow> dots and the wide bar width to <wide> dots. If this command is used, the scale is set to 1 and the narrow and wide setting provide the scale. No error checking is performed when setting these values and invalid barcodes can be generated.

Function **Set bar code justification, human readable interpretation (HRI) character print mode, and print direction**

ASCII [ESC] [EM] J <n>
Hexadecimal 1BH 19H 4AH <n>
Decimal <27> <25> <74> <n>
IPCL &%BJ <m_1> <m_2>
EPOS none
The power on default is center justified with HRI characters not printed.
Description The [ESC] [EM] J <n> command selects the operation of the bar code justification, HRI characters, and print direction.

Where n bits	n	IPCL	76543210	
			------xx	Justified
	0	00	------00	Left

1	01	`------01`	Center
2	02	`------10`	Right
		`--xx----`	HRI characters
0	00	`--00----`	Not printed
16	16	`--01----`	Printed above the bar code
32	32	`--10----`	Printed below the bar code
48	48	`--11----`	Printed above and below the bar code
		`-x------`	Vertical print mode. (Page mode may work better)
0	00	`-0------`	Bar code printed in horizontal direction (default)
64	64	`-100----`	Bar code printed in vertical direction

Note 1: The [ESC] [EM] J <n> command only affects bar code printing.
Note 2: HRI is not available in vertical print mode. You may print normal barcodes in page mode, which will provide for HRI as well as significantly longer barcodes.

Identifying Barcode Processing Errors

Barcodes that are not entered correctly in most cases will not print. It can be difficult at times to identify why. To help identify the problem there is a command that will return the last barcode error in the form of a 16 bit error ID.

ASCII	ENQ <37>
Hexadecimal	5 25H
Decimal	5 37

Will return: ACK <37> <42> <Error$_{7-0}$ ><Error$_{15-8}$>
Or 05H 25H 2AH <Error$_{7-0}$ ><Error$_{15-8}$>

Where the error codes are as follows:

Error Type	Error ID	Description
Warnings	1	Invalid option, default used
	2	Invalid barcode Command
	3	Unable to process the barcode command
General Errors	10	Too much barcode data.
	11	Not enough barcode data
	12	Too much Addenda data
	13	Invalid bar code data
	14	Invalid check digit
	15	Invalid option
	17	Barcode encoding problem
	18	Barcode won't fit in the print zone.
System Errors[14]	90	This barcode is not active in this code.
	91	HRI Text field pointer is not valid
Not recoverable	98	Unknown Internal ID
	99	Memory allocation error
Aztec	100	Input too long or too many extended ASCII characters
	101	Invalid error correction level - using default instead
	102	Invalid Aztec Code size
	103	Input too large
	104	Not used
	105	Aztec Invalid character (NULL) in input data
Data Matrix	200	Invalid characters in data
	201	Input wrong length
	202	Invalid PZN Data
	203	Value out of range
	300	Data too long to fit in symbol
	301	Data does not fit in selected symbol size
	210	Not Used
	211	Invalid symbol size
	212	Unable to fit data in specified symbol size
GS-1 Data Errors	400	Extended ASCII characters are not supported by GS1
	401	Control characters are not supported by GS1

[14] System Errors are internal printer errors that should not occur. If error codes between 90 and 99 occur, contact customer support.

	402	Data does not start with an AI
	403	Malformed AI in input data (brackets don't match)
	404	Found nested brackets in input data
	405	Invalid AI in input data (AI too long)
	406	Invalid AI in input data (AI too short)
	407	Invalid AI in input data (non-numeric characters in AI)
	408	Invalid data length for AI
	409	Invalid AI value
	410	Invalid Check Digit Calculation Field Length
	411	Invalid Check Digit Calculation Field Data
	412	Too many AI's to handle (25 Max)
Maxicode	500	Invalid Mode
	501	Invalid Secondary String
	502	Invalid option
	503	Invalid Country code field
	504	Invalid Service code field
	505	Requires Postal, Service and Country
	506	Requires Postal, Service and Country
PDF 417	600	Security value out of range
	601	Number of columns out of range
	603	Number of code words per row too small
	604	Data too long for specified number of columns
	605	Internal index out of range
	606	Specified width out of range
	607	Specified symbol size too small for data
UPC	700	Invalid UPC-E data
	701	Invalid ISBN
	702	Incorrect ISBN check
	703	UPC A input wrong length
	704	UPC E input wrong length
Composite	800	Invalid AI 90 data
	801	2D component input data too long
	802	No message in 2D composite
	803	CC-C only valid with GS1-128 linear component
	804	EANX portion is invalid

Miscellaneous Printer Control

[ESC] 8 Disable paper out sensor

ASCII	[ESC] 8
Hexadecimal	1BH 38H
Decimal	<27> <56>
IPCL	&%PF
EPOS	none

Description The [ESC] 8 command temporally disables the paper out sensor. The printer does not stop printing or go off-line when it senses it is out of paper. The inquire commands still return paper out status.

[ESC] 9 Enable paper out sensor

ASCII	[ESC] 9
Hexadecimal	1BH 39H
Decimal	<27> <57>
IPCL	&%PO
EPOS	none

Description The [ESC] 9 command enables paper sensing and is intended to reverse the effect of the disable paper out sensor command. If the printer is out of paper when the command is issued, it goes off-line.

[ESC] X Set left/right print margins

ASCII	[ESC] X <n_1> <n_2>
Hexadecimal	1BH 58H <n_1> <n_2>
Decimal	<27> <88> <n_1> <n_2>
IPCL	none
EPOS	[ESC] l, [ESC] Q

Description The [ESC] X <n_1> <n_2> command sets left and right print margins in characters from the home position. Where n_1 = left margin and n_2 = right margin, the absolute position depends on the current print pitch. If the left and right margins are set to the right of the current horizontal position, the new margins become valid in the same line. If the left margin is set to the left of the current horizontal position and the right margin set to the right of the current horizontal position, the right margin setting becomes valid in the same line, but the left margin setting becomes valid in the next line. When the left and right margins are set to the left of the current horizontal position, both left and right margin settings appear to become valid in the next line because an auto-CR is performed by the subsequent data.

[CAN] Clear print buffer

ASCII	[CAN]
Hexadecimal	18H
Decimal	<24>
IPCL	&%RP
EPOS	[CAN]

Description The [CAN] command clears the print buffer and any unprinted information in the printer received before it. If the input buffer is not being processed because the printer is out of paper or a form is not inserted, the [CAN] command will not be processed until after the error is cleared. The [CAN] command does not restore default conditions; it only clears the print buffer.

[ESC] q Query marker

ASCII	[ESC] q <n>
Hexadecimal	1BH 71H <n>
Decimal	<27> <113> <n>
IPCL	none
EPOS	none

Description The [ESC] q <n> command returns a status to the host when it is processed.

Response [SOH] <n>

The [ESC] q <n> command may be placed in the print data and, when processed by the printer, will return a progress status marker. The value of <n> can be any 8-bit value. It is returned to the host unaltered. The intent is for it to be a sequence number. The command can be used to track the print progress of the printer or verify that data has been printed.

Note: The [ESC] q <n> command is a line terminator that causes the printer to print all previous data. If a normal line terminator like a [CR] is not supplied, right justify and auto-center do not function correctly. All data is left justified. [ESC] q does not perform a [CR] or [LF] function.

[ESC] v Perform Auto Cut

ASCII	[ESC] v
Hexadecimal	1BH 76H <n>
Decimal	<27><118>
IPCL	&%FC &%PC
EPOS	[ESC] i or [ESC] m

Description The [ESC] v command operated the auto cutter.

Note: The auto cutter is optional. If the auto cutter is not installed this command will be ignored.

[ESC] @ Initialize the printer

ASCII	[ESC] @
Hexadecimal	1BH 40H
Decimal	<27> <64>

Description The [ESC] @ command initializes the printer. All settings, including character font and line spacing, are canceled.

[BEL] Audio alert

ASCII	[BEL]
Hexadecimal	07H
Decimal	<7>
IPCL	&%BL
EPOS	[BEL]

Description When enabled, the [BEL] command starts the audio alert sequence. The default is a single sound, lasting the period of time defined by the audio alert setting. If the audio alert is off, it does not function.

[ESC] [BEL] Configure audio alert

ASCII	[ESC] [BEL] <n_1> <n_2> <n_3>
Hexadecimal	1BH 07H <n_1> <n_2> <n_3>
Decimal	<27> <7> <n_1> <n_2> <n_3>
IPCL	None
EPOS	None

Description The [ESC] [BEL] <n_1> <n_2> <n_3> command alters the way the audio alert sounds.

The default is a single sound lasting the period of time defined by the audio alert setting. The [ESC] [BEL] <n_1> <n_2> <n_3> command allows the sound to be altered.

Where <n1> is the number of alert cycles

<n2> is the on time of the alert cycle in ten Ms intervals

<n3> is the off time of the alert cycle in ten Ms intervals

[ESC] p 5 Enable/disable paper feed

ASCII	[ESC] p 5 <n>
Hexadecimal	1BH 70H 35H <n>
Decimal	<27> <112> <53> <n>
Range	$0 \leq n \leq 255$

Description The [ESC] p 5 <n> command enables or disables the FEED button. When the least significant bit (LSB) of <n> = one, the FEED button is disabled; when it is zero, the button is enabled. To prevent problems caused by accidentally pressing the FEED button, use the command to disable it. The FEED button is temporarily enabled, regardless of how the command is set during the wait time set by the [GS] z 0 command for paper insertion and during the recovery confirmation time.

Where <n> Bit 0 = 1 the FEED button is disabled

Bit 0 = 0 the button is enabled.

[ESC] p 4 Select paper sensor(s) to stop printing

ASCII	[ESC] p 4 <n>
Hexadecimal	1BH 70H 34H <n>
Decimal	<27> <112> <52> <n>
Range	0 ≤ n ≤ 255

Description The [ESC] p 4 <n> command selects the sensors that tell the printer to stop printing. The default setting occurs when only the Paper Out sensor stops printing. When the paper roll near-end sensor is enabled and the sensor detects a near-end condition during printing, the printer completes the current line and then automatically goes off-line. Replacing a new paper roll restarts the printing. When the paper roll near-end sensor is disabled and a paper near-end condition is detected during printing, the paper out LED comes on, but the printer continues to print.

Note: The FST does not allow the Paper Out sensor to be disabled. It is always on.

<n> is defined as follows:

Bit	On/Off	Hexadecimal	Decimal	Function
0,1	Off	00H	<0>	Paper roll near-end sensor disabled
	On	01H,02H,03H	<1>,<2>,<3>	Paper roll near-end sensor enabled
7	-	00H	<0>	Undefined

Table 13 Paper Sensor Commands

[ESC] p 3 Select paper sensor(s) to output paper-end signals

ASCII	[ESC] p 3 <n>
Hexadecimal	1BH 70H 33H <n>
Decimal	<27> <112> <51> <n>
Range	0 ≤ n ≤ 255

Description The [ESC] p 3 <n> command selects the paper sensor that outputs a paper-end signal to the parallel interface when a paper-end is detected. The default setting is when all sensors are enabled. (<n> = 15). It is possible to select multiple sensors to output signals. If any of the sensors detect a paper end, the paper end signal is output. The command is only available with a parallel interface. The paper-end sensor is an option. If the sensor is not equipped, the settings of bits 0 and 1 of the command are not effective.

Bit	On/Off	Hexadecimal	Decimal	Function
0	Off	00H	<0>	Paper roll near-end sensor disabled
	On	01H	<1>	Paper roll near-end sensor enabled
1	Off	00H	<0>	Paper roll near-end sensor disabled
	On	02H	<2>	Paper roll near-end sensor enabled
2	Off	00H	<0>	Paper roll end sensor disabled
	On	04H	<4>	Paper roll end sensor enabled
3	Off	00H	<0>	Paper roll end sensor disabled
	On	08H	<8>	Paper roll end sensor enabled
4-7	-	-	-	Undefined

Table 14 Paper Sensor Commands

NOTE: Paper low is not supported by the EPIC 3000

[ESC] y Set control feature commands

ASCII [ESC] y <n>
Hexadecimal 1BH 79H <n>
Decimal <27> <121> <n>
IPCL &%Y0-9 or &%YX <m$_1$> <m$_2$> <m$_3$> (for numbers greater than nine)
EPOS [ESC] y <n>
Where n
0 Reinitializes the printer and forces Citizen mode
1 Reinitializes the printer and forces Star mode
2 Reinitializes the printer and forces EPIC 3000 mode
3 Reinitializes the printer and forces extended EPOS mode
4 Disables IPCL commands
5 Enables IPCL commands
6 Disables inquire processing (All preprocessing is disabled.)
7 Enables inquire processing
8 Enables extended diagnostics
9 Print Current Configuration
10 Not used
11 Not used
12 Not used
13 Not used
14 Not used
15 Print Current totals
16 Not used
17 Requests the printer to enter remote Standby. (See page 238)
18 Requests the printer to exit remote Standby.
20 Print alignment settings
21 Force Off Line Mode to Normal operation
22 Force Off Line Mode to Buffer full only.
34 Reinitializes the printer and forces Model 50 mode
35 Reinitializes the printer and forces Microline mode
48 Force Microline status to False
49 Force Microline status to True
78 Reset all character attributes to disabled
87 Force the printer into Windows mode. (203x203 and No Wrap)

Description The [ESC] y <n> command enables and disables command set features. It is possible that the IPCL commands will interfere with print data. If this occurs, the IPCL can be disabled with an [ESC] y <4> command.

Note 1: Once IPCL commands are disabled, the Enable IPLC command will not be a valid IPCL code.
Note 2: [ESC] y <0>, <1>, <2>, <3>, <34> and <35> allow the printer to switch between emulation modes. When the switch takes place, the current print buffer is printed, and the printer reinitializes. These commands do not permanently change the configuration. A power on reset restores the mode that was configured in menu mode. A reset by command or from the INIT pin does not restore the mode.
Note 3: [ESC] y <6> and <7> enable and disable the inquire process. These commands are not processed as they are received, but are buffered then processed. The buffering process allows inquire commands sent after a disable to be answered. In addition, inquires sent after an enable may not be answered. (See additional notes 3 and 4 on the next page).
Note 5: In EPOS mode, the [ESC] y command is active.

[ESC] ~ Extended Configuration and Control

ASCII	[ESC] ~ <n>
Hexadecimal	1BH 7EH <n>
Decimal	<27> <126> <n>
IPCL	none
EPOS	[ESC] ~ <n>

Description The [ESC] ~ <n> commands are extended diagnostics commands. They must all be preceded with an enabling command. These commands (in general) are not intended to be used by the end user.

Printer Status

Status Inquire
The EPIC 3000 Printer is designed for use as part of an automated system where the host computer makes every attempt to correct problems with the printer. In addition, the host application requires that it be able to obtain more information from the printer than is typical of normal computer printers. For example, a normal computer printer does not have cash drawers, such additional features require that the standard printer protocol be extended to deal with the added features of a point-of-sale (POS) printer.

All inquire commands are processed as they are received (preprocessed or real time) and require a response from the printer. Consequently, parallel, IEEE 1284 bidirectional communications, USB or bidirectional serial operation is required.

The EPIC 3000 Printer looks at and evaluates all commands as they are received, and does not respond to inquire commands that happen to be embedded in graphics or other commands. (Refer to the buffer and preprocessor descriptions in later sections.)

In all cases, inquire commands are responded to by an acknowledged (ACK) or a not acknowledged (NAK) and then the command ID, which allows the host application to make multiple requests and receive identifiable responses. If the printer is configured for serial or USB operation the status is automatically returned to the host. If the printer is configured for parallel, IEEE 1284 operation, the HOST must initiate a reverse channel request to return the status.

Serial and USB Mode Inquire
All inquire (ENQ) commands require a response from the printer. During serial operation, all inquire commands are responded to by an acknowledged (ACK) or not acknowledged (NAK), the command ID, and in some cases status. Most status responses sequences contain a length field to help decode and separate responses.

In general the printer should be configured for “Buffer Full Only” off-line operation if inquire commands are used. This prevents the printer from using flow control for anything but buffer full. The programmer must take on the responsibility for assuring that inquire commands are used to maintain status of the printer.

The printer always accepts serial data even when it is off-line. The printer has reserve buffer space that allows additional information to be received even if the printer is signaling buffer full or off-line. Because inquire commands are processed before they go into the buffer, the printer responds even when it is busy printing.

In serial mode, the response to an inquiry should be received by the host before another inquire command is issued to the printer. When the printer receives an inquiry, it generates a response. If inquiries are sent to the printer too quickly, the printer spends all of its time responding and does not have time to print.

IEEE 1284 Mode Inquire
In parallel, IEEE 1284 mode, status information can be returned to the host through the IEEE 1284 reverse channel. After the host makes an inquire request, it activates IEEE 1284 mode 0 reverse channel and waits for a response from the printer. The response to the inquire is identical to serial mode in format.

The printer always accepts IEEE 1284 reverse-channel requests but does not accept inquire commands when off-line. It is possible to obtain status when off-line by placing the printer in dynamic response mode before the printer goes off-line. The IEEE 1284 reverse channel responds to status changes even when the printer is off-line.

In general, the printer should be configured for "Buffer Full Only" off-line operation if inquire commands are used. This prevents the printer from using flow control for anything but buffer full. The programmer must take on the responsibility for assuring that inquire commands are used to maintain status of the printer.

Inquire Commands

[ENQ] Inquire printer status

ASCII	[ENQ] <n>
Hexadecimal	05H <n>
Decimal	<5> <n>
IPCL	none
EPOS	[GS] r or [DLE] [ENQ] or [DLE] [EOT]

Description The [ENQ] <n> command inquires about the printer's status and returns a result.

Note: When the printer is off-line, inquires may not be accepted.

[ENQ] <4> Inquire receipt paper out status

ASCII	[ENQ] <4>	
Hexadecimal	05H 04H	
Decimal	<5> <4>	
Function	Receipt paper exhausted	
Response	ACK <4> (06H 04H)	Receipt paper is present
	NAK <4> (15H 04H)	Receipt paper is exhausted

[ENQ] <8> Inquire cover open status

ASCII	[ENQ] <8>	
Hexadecimal	05H 08H	
Decimal	<5> <8>	
Function	Cover open/closed status	
Response	ACK <8> (06H 08H)	The cover is closed
	NAK <8> (15H 08H)	The cover is open

[ENQ] <9> Is the buffer empty?

ASCII	[ENQ] <9>
Hexadecimal	05H 09H
Decimal	<5> <9>

Function The [ENQ] <9> command allows the host to know when the print buffer is empty. If IEEE 1284 is active, the command also clears the response buffer.

Response ACK <9> (06H 09H) The buffer is empty.
NAK <9> (15H 09H) The buffer is not empty.

[ENQ] <10> Request printer reset

ASCII	[ENQ] <10>	
Hexadecimal	05H 0AH	
Decimal	<5> <10>	
Function	Reset printer	
Response	Serial	Parallel
	ACK <10> (06H 0AH)	No response
	The command was accepted.	
	NAK <10> (15H 0AH)	
	The command was rejected.	

Description The ENQ <10>, EPOS DLE ENQ <n> commands and the INIT pin all have the same effect and are referred to as reset commands. To prevent data loss, the printer tries to finish printing the buffered data. When operator intervention with the printer is required for any reason, data loss results. The reset operation is saved until the printer goes idle.

When the printer processes a reset command, the printer goes off-line and/or busy until the reset completes.

In serial mode, the printer may have information in its high-speed buffer that was received after the reset command but before the reset was processed. If the host application continues to send information to the printer after a reset command, some of that information may be processed before the reset is processed.

In parallel mode, the printer goes busy after the reset is received but before the next byte is accepted. The printer accepts an [ENQ] <10> in parallel mode. It, however, is not acknowledged. If both the serial and parallel ports are active, the serial reset is not acknowledged either because the reset operation removes the parallel response. In IEEE 1284 mode, the response buffer is cleared by a reset command, which prevents responses in IEEE 1284 mode as well.

In USB mode the USB link is not affected by the reset. Any data sent to the printer after the reset and before the printer completes the reset may be lost however the USB link will acknowledge all packets.

Note: If reset inhibit is set in the configuration menu, this command is ignored.

[ENQ] <11> Inquire power cycle status

ASCII	[ENQ] <11>
Hexadecimal	05H 0BH
Decimal	<5> <11>
Function	Has the printer been power cycled since the last request?
Response	ACK <11> (06H 0BH)
	Printer has been power cycled since the last [ENQ] <11>

NAK <5> (15H 0BH)
Printer has not power cycled since the last [ENQ] <11>

Description The first time after a reset, the command returns [ACK] <11>, after that the command returns [NAK] <11>. The command allows the application to determine if the printer has been power cycled and needs to be reinitialized. The [ENQ] <10> command and the INIT signal on the parallel port both cause the printer to return power up status.

[ENQ] <14> Inquire Mechanical error status

ASCII	[ENQ] <14>	
Hexadecimal	05H 0EH	
Decimal	<5> <14>	
Function	Error status	
Response	ACK <14> (06H 0EH)	No mechanical errors
	NAK <14> (15H 0EH)	Mechanical error has occurred (Use [ENQ]<22> to identify the error)

Note: For this status request to function, the "Buffer Full Only" off-line option should be selected.

[ENQ] <15> Inquire printer state

ASCII [ENQ] <15>
Hexadecimal 05H 11H
Decimal <5> <15>
Function The [ENQ] <15> command returns the current printer state
Note [ENQ] <17> also returns the current printer state, but it should not be used as it conflicts with XON/XOFF flow control.
Response [ACK] <15> <n> <r$_1$> <r$_2$>…

Where **<15>** is the echo of the command ID.
<n> is the number of return bytes + 40 (28H)
(to prevent confusion with XON/XOFF).
<r$_1$>: bit 0 = 1 always
bit 1 = Cover is closed.
bit 2 = Receipt paper is out.
bit 3 = 0
bit 4 = Printer is waiting in an error mode. Use [ENQ]<22> to identify the specific error and [ENQ]<10> to recover
bit 5 = 0
bit 6 = 1 always
bit 7 = 0 always
<r$_2$>: bit 0 - 5 = 0 always
bit 6 = 1 always
bit 7 = 0 always

Note: For this status request to function, the "Buffer Full Only" off-line option should be selected.

[ENQ] <20> Inquire all printer status

ASCII [ENQ] <20>
Hexadecimal 05H 14H
Decimal <5> <20>
Function The [ENQ] <20> command returns all status flags
Response [ACK] <20> <n> <r$_1$> <r$_2$>…

Where **<20>** is the echo of command ID.
<n> is the number of return bytes + 40
(28H) (to prevent confusion with XON/XOFF).
<r$_1$>: bit 0 = Cash Drawer 1 is open.
bit 1 = Cash Drawer 2 is open
bit 2 = Receipt paper is out.
bit 3 = 0
bit 4 = Receipt paper error occurred. (low or out)
bit 5 = 0
bit 6 = 1 always
bit 7 = 0 always
<r$_2$>: bit 0 = 1 always
bit 1 = Cover is closed.
bit 2 = Buffer is empty.
bit 3 = Printer has been power cycled,
Reading this does not affect the state of the power-cycled flag.
(Use [ENQ] <11> to reset the power cycled bit.)
bit 4 = Printer is waiting in an error mode. Use [ENQ]<22> to identify the specific error and [ENQ]<10> to recover
bit 5 = 0
bit 6 = 1 always
bit 7 = 0 always
<r$_3$>: bit 0 = 1 always (Receipt Station)
bit 1 = 0
bit 2 = Printer Jam Detected.
bit 3 = 0
bit 4 = Undefined
bit 5 = Printer is blocking print (Cover is open or out of paper.)
bit 6 = 1 always
bit 7 = 0 always
<r$_4$>: bit 0 = Printer supports receipts.
bit 1 = Printer supports inserted forms.
bit 2 = Printer supports multiple colors
bit 3 = Printer supports cutter.
bit 4 = Printer supports partial cuts.
bit 5 = 0
bit 6 = 1 always
bit 7 = 0 always
<r$_5$>: Percentage of ink remaining on Head 1 (0-100) + 40 (28H)
<r$_6$>: Percentage of ink remaining on Head 2 (0-100) + 40 (28H)
<r$_7$>: Current multi-head alignment (0-16, 8 = 0 offset)

[ENQ] <21> Inquire printer ID

ASCII	[ENQ] <21>
Hexadecimal	05H 15H
Decimal	<5> <21>
Function	The [ENQ] <21> command returns the printer IEEE 1284 ID string.
Response	[ACK] <21> <n> {ID string}

Where <21> is the echo of the command ID and <n> is the number of return bytes in the ID string {ID string} is the IEEE ID return string, which follows:

```
MFG:TransAct.;
CMD:M9100CL,IPCL;
CLS:PRINTER;
MDL:M9000 PcOS;
DES:Ithaca-M9100;
REV:PE9100-0M.NN
OPTS:$63xy
```

Where x is a bit field defined as follows:
bit 0 = 1 Color support
bit 1 = 0
bit 2 = 1 Periodic Status Supported
bit 3 = Always 0
bit 4 = Always 1
bit 5 = Always 1
bit 6 = Always 0
bit 7 = Always 0

The y is a bit field defined as follows:
bit 0 = 0
bit 1 = Knife is installed.
bit 2 = 0
bit 3 = Always 0
bit 4 = Always 1
bit 5 = Always 1
bit 6 = Always 0
bit 7 = Always 0

[ENQ] <22> Inquire Error status

ASCII [ENQ] <22>
Hexadecimal 05H 17H
Decimal <5> <22>
Function The [ENQ] <22> command reports on the error status.
Response [ACK] <22> <n> <r>

Where **<22>** is the echo of the command ID.
<n> is the number of return bytes + 40 (28H)
(to prevent confusion with XON/XOFF).
<r$_1$>: Bit status as follows:
bit 0 = Cover is open.
bit 1 = Receipt Paper is Low
bit 2 = Receipt paper is out.
bit 3 = Not used
bit 4 = The Printer as detected a paper jam.
bit 5 = The Auto-Cutter has faulted.
bit 6 = 1 always
bit 7 = An serious error has occurred.

Note 1: If bit 7 is set, a serious error has occurred. The printer is not able to recover from type of error without operator intervention. If bit 7 is set without bit 5 (Auto-cutter fault) then the print carriage has faulted, which is probably caused by a paper jam or a component failure. The host system may issue an [ENQ]<10> (Reset Request command) to attempt to recover. The Reset Request will reset the printer to an initial power up state. All data will be lost.
Note 2: When a serious error occurs (bit 7 set) the printer enters a static state. Status responses will reflect the state of the printer when the error occurred.
Note 3: For this status request to function during a serious error, the "Buffer Full Only" off line option should be selected.

[ENQ] <23> Inquire user-store status

ASCII	[ENQ] <23>
Hexadecimal	05H 17H
Decimal	<5> <23>
Function	The [ENQ] <23> command reports on the user-store status.
Response	[ACK] <23> <Report> <0>

Where <23> is the echo of command ID. The report is a null terminated string with the following format:

12345[CR][LF](Free user store)
12345 Type Name…[CR][LF] (First entry) etc.
12345 Type Name…[CR][LF] (Last entry) <0>

Type The type field describes the type of information.
M = macro
C = character definition

[ENQ] <24> Inquire Color status

ASCII	[ENQ] <24>
Hexadecimal	05H 18H
Decimal	<5> <24>
Function	The [ENQ] <24> command reports Color Cartridge status.
Response	[ACK] <24> <Length+40><n_1><n_2><n_3>

Where <24>	Is the echo of command
<n_1>	Secondary Paper Color 0 = Not installed, 1 = Red, 2 = Green, 4 = Blue
<n_2>	Primary Paper Color 1 = Red, 2 = Green, 4 = Blue 16 = Black (always Black)
<n_3>	Pen Status bit 0 = Not defined 0 always bit 1 = Not defined 0 always bit 2 = Not defined 0 always bit 3 = Not defined 0 always bit 4 = Not defined 0 always bit 5 = Not defined 0 always bit 6 = 1 always bit 7 = 0 always

[ENQ] <25> Inquire Electronic Journal

ASCII	[ENQ] <25>
Hexadecimal	05H 19H
Decimal	<5> <25>
Function	The [ENQ] <25> command returns current status and free space in the electronic journal
Response	[NAK] <25> <n> <r_1> <r_2>… (Electronic Journal NOT active) [ACK] <25> <n> <r_1> <r_2>… (Electronic Journal active)

Where **<25>** is the echo of command ID.

<n> is the number of return bytes + 40
(28H) (to prevent confusion with XON/XOFF).

<r$_1$>: MSB of size in K bytes.

<r$_2$>: LSB of size in K bytes.

[ENQ] <26> Inquire current print zone

ASCII [ENQ] <26>

Hexadecimal 05H 1AH

Decimal <5> <26>

Function The [ENQ] <26> command returns current print zone in dots

Response [ACK] <26> <n> <r$_1$> <r$_2$>

Where **<26>** is the echo of command ID.

<n> is the number of return bytes + 40
(28H) (to prevent confusion with XON/XOFF).

<r$_1$>: MSB of the print zone.

<r$_2$>: LSB of the print zone.

[ENQ] <28> Inquire Buffer status

ASCII [ENQ] <28>

Hexadecimal 05H 1CH

Decimal <5> <28>

Function The [ENQ] <28> command returns current print zone in dots

Response [ACK] <28> <n> <r$_1$>

Where **<28>** is the echo of command ID.

<n> is the number of return bytes + 40
(28H) (to prevent confusion with XON/XOFF).

<r$_1$>: 0-100 Percent of input buffer being used.

[ENQ] <29> Inquire Jam Status

ASCII [ENQ] <29>
Hexadecimal 05H 1DH
Decimal <5> <29>
Function The [ENQ] <29> command reports Jam and Transport status.
Response [ACK] <29> <41><n>

Where
<29> Is the echo of command
<n> Status
bit 0 = Paper Jammed before the cut
bit 1 = Paper Jammed after the cut.
bit 2 = 0 always
bit 3 = 0 always
bit 4 = Jam Sensor
bit 5 = 1 always
bit 6 = 0 always
bit 7 = 0 always

[ENQ] <31> Inquire Power on status

ASCII [ENQ] <31>
Hexadecimal 05H 1FH
Decimal <5> <31>
Function The [ENQ] <31> command returns File and POR system status
Response [ACK] <31> <n> <r_1> <r_2> <r_3>

Where **<31>** is the echo of command ID.
<n> is the number of return bytes + 40
(28H) (to prevent confusion with XON/XOFF).
<r_1>: Power up system status where the bits are defined as follows:
0x01 No Font present
0x02 No Files present
0x04 File system faulted and has been cleared
0x08 POR.INI file was missing and a default has been generated.
0x10 Code page file requested Is missing.
0x20 The primary FAT was corrupt and the alternate was used.
0x40 The primary and alternate FAT's are corrupt and the file system has been initialized.
<r_2>: Font system status
0x00 Font system closed
0x01 Font system Ready
0x02 Font system Fault.
All other values are errors that will also set the R3 status byte.
<r_3>: Font imaging system extended status.
Extended status information being present is a serious error in the printer. Typically generated by a faulty font file. Any value here should be interpreted as a fault.

[ENQ] <33> Inquire Paper Size.

ASCII	[ENQ] <33>
Hexadecimal	05H 21H
Decimal	<5> <33>
Function	The [ENQ] <33> command Paper Size Status.
Response	[ACK] <33> <41> <n>

Where

<33>	Is the echo of command
<41>	Length + 40
<n>	Paper Size Statu.
	bit 0,1 0 = 40 mm paper
	1 = 58 mm paper (Note[15]: 40mm can be reported as 58mm)
	2 = 80 mm paper
	3 = 82 mm paper
	bit 2 = Not defined 0 always
	bit 3 = Not defined 0 always
	bit 4 = 1 40mm paper divider installed.
	bit 5 = Not defined 0 always
	bit 6 = 1 always
	bit 7 = 0 always

[ENQ] <34> Inquire Firmware ID and Rev.

ASCII	[ENQ] <34>
Hexadecimal	05H 22H
Decimal	<5> <34>
Function	The [ENQ] <34> command returns the Firmware ID and revision.
Response	[ACK] <34> <51> PE9100-X.XX.

Where

<33>	Is the echo of command
<51>	Length + 40
Firmware ID	PE9100
Separator	- (0x2D or 45D)
Rev	X.XX (Current Revision in ASCII)

[15] In some cases a small 58 mm paper print zone is used on 40mm paper although some of the print may miss the paper.

[ENQ] <35> Inquire USB Watch Dog Resets.

ASCII [ENQ] <35>
Hexadecimal 05H 23H
Decimal <5> <35>
Function The [ENQ] <35> command returns the number of USB Watch dog resets and then resets the count to zero.
Response [ACK] <35> <41> <n>

Where
<35> Is the echo of command
<41> Length + 40
<n> The number of USB Watchdogs since the last inquire.

[ENQ] <36> Inquire current UTF encoding mode.

ASCII [ENQ] <36>
Hexadecimal 05H 24H
Decimal <5> <36>
Function The [ENQ] <36> command returns the current UTF encoding mode.
Response [ACK] <36> <41> <n>

Where
<36> Is the echo of command
<41> Length + 40
<n> The UTF Mode where:
0 = ASCII
1 = UTF8 Text only
2 = UTF8
4 = UTF16
8 = UTF16BE
16 = UTF16LE
Note: This command must be sent in the current encoding. Only ASCII, UFT8 Text, and UTF Modes use identical data structures.

[ENQ] <37> Inquire barcode error status

ASCII [ENQ] <37>
Hexadecimal 05H 25H
Decimal <5> <37>
Function The [ENQ] <37> command returns the last barcode error code.
Response ACK <37> <42> <Error$_{7\text{-}0}$ ><Error$_{15\text{-}8}$>

Where <Error$_{7\text{-}0}$ ><Error$_{15\text{-}8}$> indicate a 16 bit barcode error code.

The returned error code reflects the last barcode print attempt. See page 161 for a detailed list of barcode errors. 0 indicates no error was detected.

[ENQ] <38> Inquire currently loaded firmware file name

ASCII [ENQ] <38>
Hexadecimal 05H 26H
Decimal <5> <38>
Function The [ENQ] <38> command returns the file name of the currently loaded firmware.
Response ACK <38> <Len+40> Firmware Id String<0>

[ENQ] <39> Inquire update Process status and CRC.

ASCII [ENQ] <39>
Hexadecimal 05H 27H
Decimal <5> <39>
Function The [ENQ] <39> command returns File and POR system status
Response [ACK] <39> <n> <r_1> <r_2> <r_{3L}><r_{4H}>

Where **<39>** is the echo of command ID.
<n> is the number of return bytes + 40 (28H) (to prevent confusion with XON/XOFF).
<r_1>: CBT Update Process Status:

0x00	Idle
0x01	Get Name
0x11	Fault
0x03	Get Data
0x04	Write
0x05	Done
0x15	Not Processed
0x25	Done No Validation requested
0x12	Bad CRC
0x06	Roll Back performed
0x08	Not yet done

<r_2>: CBL Update Process Status:

0x00	Idle
0x01	Get Name
0x11	Fault
0x03	Get Data
0x04	Write
0x05	Done
0x15	Not Processed
0x25	Done No Validation requested
0x12	Bad CRC
0x06	Roll Back performed
0x08	Not yet done

<r_{3L}><r_{4H}>: Current CBL process CRC.

[ENQ] <40> Inquire Current Firmware CRC

ASCII [ENQ] <40>
Hexadecimal 05H 28H
Decimal <5> <40>
Function The [ENQ] <40> command returns current firmware CRC.
Response [NAK] <31> <CRC$_L$> <CRC$_H$> Not calculated yet
[ACK] <31> <CRC$_L$> <CRC$_H$> Calculation complete

Where **<40>** is the echo of command ID.
<n> is the number of return bytes + 40 (28H) (to prevent confusion with XON/XOFF).
<CRC$_L$> 16 bit CRC low 8 bits
<CRC$_H$> 16 bit CRC upper 8 bits

[ENQ] <41> Inquire Current Firmware CRC

ASCII [ENQ] <41>
Hexadecimal 05H 29H
Decimal <5> <41>
Function The [ENQ] <41> command firmware update process status.
Response [ACK] <41> <n> <r$_1$> <r$_2$>

Where **<39>** is the echo of command ID.
<n> is the number of return bytes + 40 (28H) (to prevent confusion with XON/XOFF).
<r$_1$>: CBT Update Process Status:

0x00	Idle
0x01	Get Name
0x11	Fault
0x03	Get Data
0x04	Write
0x05	Done
0x15	Not Processed
0x25	Done No Validation requested
0x12	Bad CRC
0x06	Roll Back performed
0x08	Not yet done

<r$_2$>: CBL Update Process Status:

0x00	Idle
0x01	Get Name
0x11	Fault
0x03	Get Data
0x04	Write
0x05	Done
0x15	Not Processed
0x25	Done No Validation requested
0x12	Bad CRC
0x06	Roll Back performed

0x08 Not yet done

[ESC] [EM]P<n> Activate Periodic Status Back

ASCII	[ESC] [EM] P<n>
Hexadecimal	1BH 19H 50H <n>
Decimal	<27> <25> <80><n>
IPCL	None
EPOS	None

Description This command activates the periodic status back feature. It will automatically return an [ENQ]<20> status (See page **Error! Bookmark not defined.**) on a periodic bases. The value of n is the period in 100 MS intervals. This command is saved through power cycles. Once set it need not be set again, however you can set it the same value repeatedly as it is only saved if it is changed. In general it should not be changed on a regular bases.

Where n = Interval in 100 MS increments. IE 20 = 2 Seconds. Setting the value to 0 disables the feature.

Note: Periodic status back can also be activated with the [ESC][EM]p command, however it is not save during a power cycle.

[ESC] [EM]p<n> Activate Periodic Status Back

ASCII	[ESC] [EM] p<n>
Hexadecimal	1BH 19H 70H <n>
Decimal	<27> <25> <112><n>
IPCL	None
EPOS	None

Description This command temporarily disables and/or enables the periodic status back feature if previously activated with the [ESC][EM]P command.

Where n = 0 disables PSB and n = (non zero) Enables PSB at the interval defined by the [ESC][EM]P command.

If not previously activated with the [ESC][EM]P command, this command will activate it but the setting is not saved through a power cycle.

Where n = Interval in 100 MS increments. IE 20 = 2 Seconds. Setting the value to 0 disables the feature.

ESC/POS^{TM} Codes

The EPOS codes that are supported by the EPIC 3000 Printer are listed in this section.

Note: The Epson emulation does not support Unicode, Double Byte or downloaded True Type fonts.

Differences between Epson TM Printers and EPIC 3000

In a few minor ways, the operation and programming EPIC 3000 is not identical to a TM-T90. The following is a summary of key differences.

Page Mode

The page mode in the EPIC 3000 operates in the same way as the TM-T88 and the TM-T90, however, the page size can be larger if the ESC/POS emulation is selected.

Undocumented Epson Commands

There are a number of undocumented commands in Epson printers. Where TransAct is aware of these commands, they have been duplicated in the EPIC 3000 Printer.

Barcodes

The barcode generation in the EPIC 3000 generates barcode differently and at a different resolution than the Epson printer. In most cases the barcodes are equivalent, however, in some cases the EPIC 3000 may generate a larger or smaller image and may be encoded differently.

Real-time Status

The EPIC 3000 Printer is available in USB, serial, parallel and Ethernet versions. Epson supports parallel operation through a parallel to serial interface. Consequently, the response times for the EPIC 3000 Printer are generally faster. The EPIC 3000 Printer implements the IEEE 1284 nibble-mode standard. If an application requires real-time status from the printer, the IEEE 1284 bi-directional protocol must be used. The EPIC 3000 Printer does not support byte mode. If the EPIC 3000 Printer is used in an IEEE 1284 compliant system, byte mode should be an extension and the default should be nibble mode. The EPIC 3000 provides real time status using the UDP IP protocol. This is not the same as the Epson printer. In general the EPIC 3000 when operating through an Ethernet interface will not be interchangeable.

The EPIC 3000 Printer supports the EPOS real-time status commands [DLE] [ENQ] and [DLE] [EOT] and are preprocessed by the printer. The printer supports all the response bit fields as defined by the TM-T88 and TM-T90 Printers. The EPIC 3000 Printer looks at and evaluates all commands as they are received and does not respond to [DLE] [ENQ] or [DLE] [EOT] commands that happen to be embedded in graphics or other commands. (Refer to the buffer and preprocessor descriptions in later sections.)

Epson Licensed Firmware

Epson has patented several features used by the TM-88 and TM90. If you use automatic status back or the real time cash drawer command, This printer does not support those patented features.

EPIC 3000 Graphics

Printing Graphics

The EPIC 3000 Printer has bit-image graphic capability and a full PC-compatible graphic character set. The bit image format is similar to that used on other personal computer printers. Three modes of operation are available. Single-density is the fastest mode. It makes a single unidirectional 60 dpi print pass. Full-speed double-density mode makes two passes with a half-dot offset. Full-speed double-density mode is half the speed of single-density mode, but it prints at 120 dpi. Half-speed double-density mode is half the speed of full-speed double-density; however, the print quality is enhanced.

Character Graphics

Character graphics is the term for joining individual characters together to produce a mosaic of characters that form a graphic image. The simplest method uses an * (or any other character) to form an image. For example, ITHACA® might be formed as follows.

```
***  ***  *   *   **    ***    **
 *    *   *   *  *  *  *   *  *  *
 *    *   *****  ****  *      ****
 *    *   *   *  *  *  *   *  *  *
***   *   *   *  *  *   ***   *  *
```

Figure 61 Example of Character Graphics

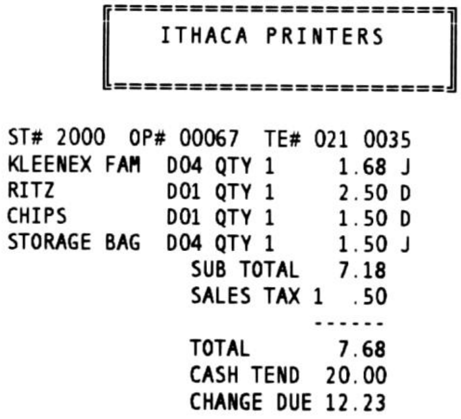
The extended character set of the printer supports line graphic characters that can be combined to form windows and other shapes. For the shapes to join from line to line, the spacing must be set properly.

All characters are in a nine-dot-high character cell. The dots are 1/60 inch apart. Therefore, the line spacing should be 10/60 or 1/6 inch. The set fine line space command can be used to set the line spacing. If possible, the spacing should be reduced slightly to overlap the rows, which prevents any white spacing from appearing between the lines. The following example illustrates the use of extended character graphics.

```
[ESC]:
[ESC]3<27>
[ESC]a<1>
<201>
<205><205><205><205><205><205><205><205><205>
<205><205><205><205><205><205><205><205><205>
<205><205><205><205>
<187>[CR][LF]
<186>   ITHACA PRINTERS    <186>[CR][LF]
<186>                      <186>[CR][LF]
<200>
<205><205><205><205><205><205><205><205><205>
<205><205><205><205><205><205><205><205><205>
<205><205><205><205>
<188>[CR][LF]
[ESC]a<0>
[ESC]0
[ESC][P<15>
[CR][LF]

ST# 2000  OP# 00067  TE# 021 0035[CR][LF]
KLEENEX FAM  D04 QTY 1     1.68 J[CR][LF]
RITZ         D01 QTY 1     2.50 D[CR][LF]
CHIPS        D01 QTY 1     1.50 D[CR][LF]
STORAGE BAG  D04 QTY 1     1.50 J[CR][LF]
               SUB TOTAL   7.18[CR][LF]
               SALES TAX 1  .50[CR][LF]
                         ------[CR][LF]
               TOTAL       7.68[CR][LF]
               CASH TEND  20.00[CR][LF]
               CHANGE DUE 12.23[CR][LF]
[CR][LF]
```

Figure 62 Example Commands for a Sample Receipt



```
╔═══════════════════════╗
║    ITHACA PRINTERS    ║
║                       ║
╚═══════════════════════╝

ST# 2000  OP# 00067  TE# 021 0035
KLEENEX FAM  D04 QTY 1     1.68 J
RITZ         D01 QTY 1     2.50 D
CHIPS        D01 QTY 1     1.50 D
STORAGE BAG  D04 QTY 1     1.50 J
               SUB TOTAL   7.18
               SALES TAX 1  .50
                          ------
               TOTAL       7.68
               CASH TEND  20.00
               CHANGE DUE 12.23
```

Figure 63 Sample Receipt

When printing line graphics, the data path to the printer must be eight bits. Seven-bit serial protocols do not access the extended characters. The extended characters require that the form be reverse fed. Consequently, printing line graphics on the receipt may be less than acceptable.

APA Graphics

The printer is capable of all-points-addressable (APA) or horizontal graphics. Generating a graphic image by hand is time consuming and tedious. It is recommended that a graphic package be used to create a graphic image. The following procedure will help with the setup.

Procedure for APA graphics:

1. Generate the graphic image in the program of your choice. APA graphics only support monochrome images.
2. Make sure the paper size chosen fits the printer (3 inches wide with 0.25-inch margins). If the paper size cannot be set, print a portion of the page.
3. Print the graphic to a file using a generic, IBM, graphic, 9-pin driver. The standard IBM resolutions are 240 x 216 dpi, 120 x 72 dpi, and 60 x 72 dpi. The EPIC 3000 Printer supports all three resolutions. For good resolution with reasonable speed, use 120 x 72 dpi.
4. Print the graphic image to a file.
5. Edit the resulting file to remove any unwanted form control, and insert the EPIC 3000 form control.
6. Make the resulting file available to your application, so it can be sent to the printer when required.

Color Graphics

The EPIC 3000 has a two-color option. The EPIC 3000 has a two-color and is not a full color printer. A full color printer forms the various colors of an image by mixing inks of different colors on the paper. The amount of each color determines the hue. Typically the paper is white and no ink produces a white. Mixing yellow and cyan produces a red and mixing cyan, magenta, and yellow in equal amounts produces a black or gray.

The EPIC 3000 is a two-color printer. It has white (the paper), and two predefined colors. Typically one of the colors is black, and the other is a highlight color. The high light color is primarily used in text printing to highlight a line, and typically is red.

When the EPIC 3000 generates graphics, both colors are used to generate the image. Because the program that generates the graphic image data should not need to be aware of the colors available in the printer, three color planes are sent to the printer. The printer is aware of what colors the cartridges are that will generate the image. The printer will converts the full color image information into a two color image.

If the EPIC 3000 printer is configured with a black color and a highlight color, there is no point in the printers mixing the highlight color with black. If however, the EPIC 3000 printer is configured with two colors other than black, the printer attempts to mix the colors to generate black. (Note that in text mode the colors are not mixed.) This requires that there be two basic modes of color graphic generation.

If the EPIC 3000 printer is configured with two colors, the printer will mix the two colors on the thermal paper, using the primary and the secondary color. Because there are three primary colors, the printer is unable to print full color. The printer follows the rules for mixing colors and it if the result is ambiguous (because of the missing color), the printer will print with the primary color. This results in a image made up of primary color dots, secondary color dots and mixed dots. The mixed dots color may not be black but a mixture of whatever colors are installed.

In text mode, the primary color is assigned as the default color and the secondary color is assigned as the alternate highlight color. In most cases the darkest color should be initialized as the primary color.

The Ithaca® Windows print driver provides the translation from Windows color to the three color planes. When a graphic is created for the EPIC 3000 Printer, the colors used must take into account that colors other than the highlight color will print black. The Ithaca® Windows driver helps adjust the color content and generates the desired print from a full color image[16].

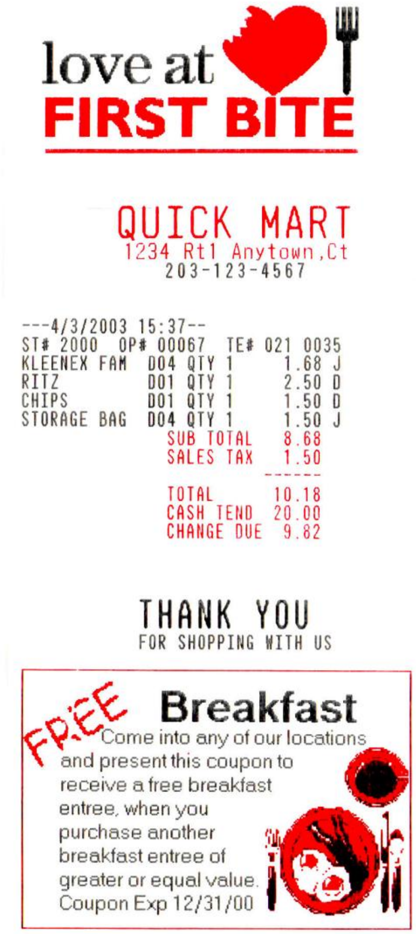
Procedure for color horizontal graphics:

There are many ways to generate graphics for the EPIC 3000 printer. The easiest is to use the PJColor program. (See the next section.) The other way is to use the Ithaca windows driver and capture the output to a file.

To generate an image using the windows print driver:

1. Generate the graphic image in the program of your choice. Use colors consistent with the two colors installed in the printer.
2. Make sure the paper size you pick fits the printer (3 inches wide with 0.25-inch margins).
3. The EPIC 3000 print driver should be installed (even if the printer is not connected).
4. Set up the print driver to print the graphic to a file using whatever resolution is required.
5. Print the graphic image to a file.
6. Later, when you want to print the graphic, simply copy the file to the printer with your application.
7. You may also copy the graphic to the graphic save buffer and then request the printer to retrieve and print it without re-sending the graphic data.

Optionally, process the color image with the Ithaca® color image processor[16] in the PJColor program. The program can print the image or generate a printable file.



love at
FIRST BITE

```
                  QUICK MART
               1234 Rt1 Anytown,Ct
                  203-123-4567

---4/3/2003 15:37--
ST# 2000  OP# 00067  TE# 021 0035
KLEENEX FAM  D04 QTY 1      1.68 J
RITZ         D01 QTY 1      2.50 D
CHIPS        D01 QTY 1      1.50 D
STORAGE BAG  D04 QTY 1      1.50 J
                SUB TOTAL   8.68
                SALES TAX   1.50
                          ------
                TOTAL      10.18
                CASH TEND  20.00
                CHANGE DUE  9.82


               THANK YOU
            FOR SHOPPING WITH US
```

FREE Breakfast

Come into any of our locations
and present this coupon to
receive a free breakfast
entree, when you
purchase another
breakfast entree of
greater or equal value.
Coupon Exp 12/31/00

Figure 64 Receipt with graphics

As with all graphics, the data path to the printer must be eight bits. Seven-bit protocols do not work.

[16] Ithaca® Color Image Processor is patent pending

EPIC 3000 Universal Color Graphics

The EPIC 3000 Printer Color[17] firmware supports the ability to print two color graphics in all emulations. This support is an enhancement to the original TM-U200, ESC/POS, Star, and Citizen printer emulations.

The intent of Universal Color Graphic support is to allow the EPIC 3000 color graphics capability to be used in existing applications that do not support color graphics.

There are several ways to add color graphics to an existing application. The easiest for you will depend on how much control you have over your application. At a minimum, you should be able to change the name printed on the top of a receipt. With the PJColor[18] program you can store a named graphic into the printer an print it by changing the text name to match the stored graphic. For example, if your receipt has a name like "Joe's Market", you can save a graphic in the printer named "Joe's" and then change the "Joe's Market" to "&%URJoe's&" When the EPIC 3000 finds "&%URJoe's&", it is replaced with the stored graphic.

Some applications allow a graphic file to be sent to the printer. In this case PJColor can generate the graphic file and then your application can send it to the printer. This file will support two color print.

PJColor also has a feature that will allow you to generate a file that will define the graphic to be stored into the printer. You can then use this file to setup any number of printers with the same graphic.

If you are using a windows print driver (other than the Transact EPIC 3000 driver) to support your printer, you will not be able to send color graphics to the printer through the print driver. The print driver will not support universal color graphics. You can however, store the graphic in the printer and use IPCL commands to print the stored graphic. (You must select a printer resident font for this to work.)

The following is a short summary and how to description of these features.

Print File Graphics

PJColor can generate a print file that may be sent to the printer in any emulation and produce a two color print graphic.

To generate a print file.

1) Start PJColor
2) Under Settings, Select the EPIC 3000 printer. Then select the emulation that machines the printer.

[17] Color is supported with special color firmware option.

[18] The PJColor program has been enhanced to provide several ways to create color Logos and Coupons and make them available to the T480 printer.

3) Select the resolution you would like to have the printer use to print the graphics. Low resolutions are faster, high resolutions produce better graphics.
4) Load the graphics Image you wish to print.
5) Select the communications port and configuration.
6) Adjust the image to produce the effect you would like. The lower graphic window displays an approximation of the printed image.
7) When you are satisfied with the graphic, press the "Print to a File" button. PJColor will ask what file you would like to receive the print data.
8) This file can be sent to the printer and the graphic will be printed.

Store Graphics in the printer:

PJColor can store a graphic in the EPIC 3000 Printer or generate a file that will store a graphic in the printer.

To Store a graphic in the printer

1) Start PJColor
2) Under Settings, Select the EPIC 3000 printer. Then select the emulation that you will be setting the printer to.
3) Select the resolution you would like to have the printer use to print the graphics. Low resolutions are faster, high resolutions produce better graphics.
4) Select the communications port and protocol that is to be used to communicate to the printer.
5) Load the graphics Image you wish to print.
6) Adjust the image to produce the effect you would like. The lower graphic window displays an approximation of the printed image.
7) When you are satisfied with the graphic, press the "Store in Printer" button. PJColor will attempt to interrogate the printer and will display the graphics currently in the printer if any. (Note: If PJColor cannot communicate with the printer, only the "Save to File" option will be allowed.)
8) Make sure there is enough room in the printer for the graphic.
9) Insert a name in the "Macro Name" box. Keep it simple, this name will be used later to print the graphic.
10) Record the graphic in the printer.

Print a stored graphic.

In the data stream to the printer enter "&%URName&" and the graphic will print in place of the "&%URName&" data. The "Name" must be identical to the name entered earlier.

Generate a file to store graphics into a printer

To generate a file that will store a graphic into a printer, follow the same procedure to store a graphic in a printer up through step 8. Then:

1) Insert a name in the "Macro Name" box. Keep it simple, this name will be used later to print the graphic
2) Press the "Save to File" button. This will allow you to select a file where the stored universal graphic information is saved.

3) This file contains an "erase any previous graphic with the same name" command, "a save new graphic with this name" command and the graphics information.
4) This file can then be sent to the printer and the graphic will be saved in the printer. Note: If the target printer does not have enough room for the graphic information to be stored, the graphic will not be stored.

How universal graphics is done

The printer extends all the emulations to support two additional escape sequences and adds limited IPCL support.

IPCL (Ithaca Printer Control Language) is an ASCII method of sending printer commands to the printer. In Ithaca PcOS emulation, IPCL command support is extensive. In other emulations, IPCL support is limited to the following commands.

&%CR	Insert a [CR]
&%LF	Insert a [LF]
&%UAxxx	Feed xxx paper steps and cycle auto-cutter
&%CLx	Select Color where x = 0 for the Primary or 1,2,3 for secondary
&%UBName&	Begin defining universal graphic "Name"
&%UGName&	End the definition of "Name"
&%URName&	Run (print) universal graphic "Name"
&%UDName&	Remove universal graphic "Name" from nonvolatile memory
&%USName&	Flag universal graphic "Name" to be run when the printer is turned on
&%UFALL&	Erase all stored universal graphics. (Erases all User Store)
&%UQ&	Prints a directory of the universal graphics currently stored in the printer
&%UTx	Changes the Name termination character from "&" to "x". "x" may range from 21H to 255H

Note: The & used to flag the end of the Name string is not valid in PcOS Ithaca emulation mode. You should use <0> or define the terminator with the &%UTx command.

The extended Escape sequences are [ESC][US]… and [ESC][FS]…

The [ESC][US] commands are the same as the PcOS emulation. The [ESC][FS] commands are not intended to be used by the customer. They provide the universal graphics support. Because graphics would be very difficult to generate and are not supported by any graphics drives other than PJColor .

How to use IPCL commands in text strings

If your software allows you to pass text strings to the printer, you should be able to use the universal graphics commands. Most POS software allows user customization of the text message printed at the beginning and the end of the receipt.
To use the Universal IPCL commands simply place them in a text string like the following example; note that your results may vary depending on the operation system, software and the ability to pass ASCII Characters.

Load and store named graphic image

- First you must create the graphic image using the PJColor Color Image Converter and save the image to a file. See the section "**Generate a file to store graphics into a printer**" above.

- Send the following text strings to the printer using whatever means is available to you.

&%UBLogo&	Begin defining macro "Logo"
&%UGLogo&	End the Definition of "Logo"
&%UMLogo&	Save Macro "Logo" to nonvolatile memory

- A graphic image named "Logo" should now be stored in the nonvolatile memory.
- To verify the image is present, use the "&%UQ&" IPCL command or the PJColor Color Image Converter to print the name and size of the stored images.

Recall and print stored named graphic image

- Send the following text string to the printer using whatever means is available to you.

&%URLogo&	Run Macro "Logo" (Print the macro)

Cautions

Universal graphics information is stored in the same place as user defined characters and user defined macros. If you are using an emulation such as ESC/POS that supports macros and/or user defined characters, universal graphics will compete for space with these functions. In addition, the "&%UFALL&" (Erase universal graphics) will also erase any user defined graphics and macros.

If you are using the Ithaca PcOS emulation, these commands are identical with the User Store commands except for the terminator character. You may change the NUL terminator to "&" with "&%UT&" if you find the "&" easier.

Universal Graphic Command Descriptions

&%UB <Name..>& Begin named universal graphic record

IPCL &%UB <Name..>&

Description The &%UB <Name..>& command initializes the universal graphic buffer structure, and redirects the following data to the universal graphic buffer. It uses the <Name..> field as a reference. If the name already exists in the flash user store, the command is ignored. The command must be followed by the "End name universal graphic record" command with the same name. If the data that follows is larger than the universal graphic buffer (about 16K), the universal graphic definition is terminated without saving any data.

&%UG <Name..>& End named universal graphic record

IPCL &%UG <Name..>&

Description The &%UG <Name..>& command ends the universal graphic record operation and saves the universal graphic to flash. It uses the <Name..> field to verify the command end and must match the "Begin named universal graphic record" command. If the name already exists in the flash user store or the universal graphic memory is exceeded, the command is invalid, and the <Name..> field prints.

&%UR <Name..>& Run universal graphic data from user store

IPCL &%UR <Name..>&

Description The &%UR <Name..>& command loads the referenced universal graphic into the universal graphic buffer. The universal graphic buffer is then inserted into the data stream. If the named item does not exist or is not a universal graphic, the EPIC 3000 ignores the command.

&%US <Name.. >& Flag item as a start-up universal graphic

IPCL &%US <Name.. >&

Description The &%US <Name.. >& command flags the referenced item to `be processed at startup. Only one user character definition and one universal graphic may be flagged to run at startup.

&%UD <Name..>& Delete item from user store

IPCL &%UD <Name..>&

Description The &%UD <Name..>&command removes an item from user store and frees up space. If the item does not exist, the EPIC 3000 ignores the command.

&%UFALL& Flush information from user store

IPCL &%UFALL&

Description The &%UFALL& command clears all entries in user store and frees the data space. It must have the name, “ALL” (in uppercase) attached.

&%UQ& **Report on user store**

IPCL &%UQ&

Description The &%UQ& command prints a status report. The intention of the command is to aid in universal graphic development.

&%UT<n> **Redefine User Store Termination Character**

IPCL &%UT<n>

Description This command allows the terminator used to signal the end of the name field in User Store commands to be modified. The value of <n> is used for the terminator. The value of n may be from 0 to 255.

Example If &%UT% were sent to the printer the User Store command to run universal graphic "Demo" would be &%URDemo%.

&%CL **Set Print Color**

IPCL &%CL <n>

Description This command allows various colors to be selected on printer emulations that do not support color text.

Where n: 0 Print with the Left cartridge (Typically Black)

1, 2, 3 Print with the Right cartridge (Typically Red, Blue, or Green)

&%UA **Cycle Auto-Cutter**

IPCL &%UA <m$_1$> <m$_2$> <m$_3$>

Description This command feeds m/96 inches of paper and cycles the auto cutter.

Where m: $m = m_1 * 100 + m_2 *10 + m_3$

Scalable Fonts

Your TransAct® EPIC 3000 printer uses outline and/or stroke based scalable fonts. These fonts provide you with additional font options as well as improved character appearance, while functioning transparently within legacy applications.

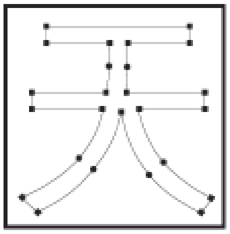
Such fonts represent a substantial improvement over the bitmap fonts that are traditionally used for thermal printers. Bitmap characters are based on a pixel by pixel definition of the characters. With a fixed size and fixed character spacing, these bitmap fonts are limited to specific magnification factors from 2-8X, and required scaling and smoothing at larger font sizes. Moreover, such scaling and smoothing operations were often unsuitable for complex fonts such as Asian characters, where changes to pixel layout actually risk changing character meanings.

To take full advantage of scalable fonts, the EPIC 3000 supports additional commands and features, including:

1) Character size selection by points
2) Character pitch selection by points
3) Variable character spacing if desired
4) Custom fonts
5) Unicode support for international language support
6) Enhanced code page support for ASCII based applications.

Character Generation

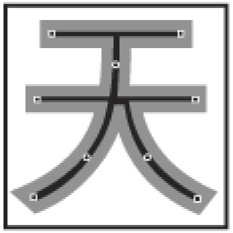
The font technology in the EPIC 3000 printer uses standard outline fonts (sometimes referred to as TrueType fonts) or stroke fonts. Both technologies are scalable, however each has unique advantages.



Outline characters

Outline characters use points along the edge of the character to describe the character. The character generator defines the edge and then fills in the enclosed space to define the character.

This type of character generation produces very well formed characters and produces the best looking characters. However, it requires more storage than stroke fonts, and is best for non-Asian fonts.



Stroke-based characters

With stroke based characters, the points stored are along the center line. Less than half the points are needed to render stroke based characters. This improves character-generation performance and uses less space.

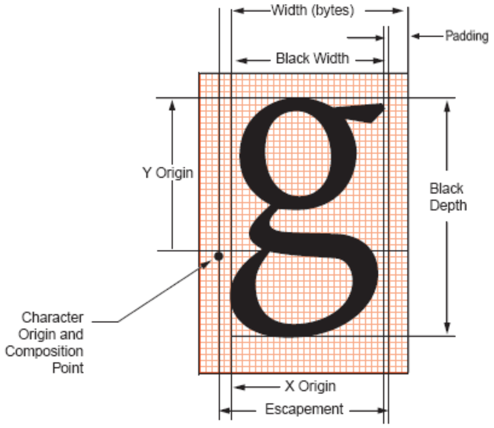
This type of character generation is fast and efficient, and is ideally suited for Asian fonts.

Character Definition

True Type and Stroke fonts are designed as a complete font with character cell size and character position in the cell based on the overall font design. Typical the characters are defined as vectors and stored as coordinates on a character cell grid. The grid is in an arbitrary design unit and may be up to 4096 units on a side.

In most systems character sizes is specified as a point size where the point size refers to the character height. The character width is typically variable and designed to produce the optimal appearance. The font rendering system must take the requested point size and generate a character based on the original design units and produce a character that is the correct size and position for the printer.

In most font designs, the vertical point size includes white space between lines. The font designer defines the height of the character cell in design units for all characters then defines a character origin that will be used for all the characters in the font. The designer then defines individual character sizes based on how the font is supposed to look and all the characters that are to be included. Characters are then positioned in the cell based on this origin. All characters in the font are then based on the same rules. The white space between lines in defined to be above the character.



The information available at print time is listed above. The complete cell is not provided, only the escapement, black width and depth and the x and y offsets to the origin are available. The printer cannot arbitrarily shorten the cell height that was defined by the font designer even though the provided character may fit in a smaller space. Using these rules, may result in characters that at first appear too small with excessive white space between lines, however this is how the font was designed. The printer must allow the minimum line spacing based on the point selection and not on the actual character height of any given characters. Second guessing the font designer can have very bad results when character are encountered that use the full cell.

Asian fonts require slightly different rules for character placement and are not as one would expect. The Asian ideographs are positioned on center of mass, rather than on a baseline. The Latin data in Asian typefaces must be built on the same rules. As a result, when Latin characters are scaled the expected baselines do not line up. Asian fonts are also fixed pitch, however, Latin character in the font are basically ½ the width of the typical Asian character.

Character Size

The character generation engine used in the T480 internally uses a standard point based system to specify the character size. One point is 1/72 of an inch. Therefore a 72 point character would form a character suitable to generate one line per inch printing. The typical application might refer to a 12 point character. This is the character height and not the width. The character width typically varies on character by character bases; for example, the lower case “i” is much narrower than the upper case “W”.

Font selection commands for selecting character sizes in legacy applications are also supported. For printers such as the EPIC 3000 , two or three character sizes are generally predefined with a dot matrix size of 10x24 or 13x24 dots, and this fixed size may then be double or tripled to provide larger characters when needed. To supply legacy support, the EPIC 3000 will automatically select the appropriate character size to support the legacy font and character scaling commands.

In the EPIC 3000 , the horizontal and vertical point size may be set independently. Typically this type of printer would print a tall, narrow, mono-spaced character. Tall narrow mono-spaced character provides a very readable print with easy column alignment while using less paper than standard type fonts. This type of font is sometimes referred to as a condensed font.

Selecting Character Size

The EPIC 3000 provides two ways to specify character size. The legacy or classic font selection method is based on dot matrix size. The second method is based on the standard type points system. The advantage of the type point system is that the print produced by the printer will match what is displayed by the host system, as both use the same system for describing the characters produced.

Legacy or Classic Method

With the classic method, the application selects a character size and then sets the character spacing by adding or removing dot spaces between the characters. Using the scaled font, the EPIC 3000 provides 3 basic predefined character sizes. The smallest is a 10x24 dot-like[19] font and is typically printed in pitches from 16 to 20 characters per inch (CPI). The next larger font is 13 x 24 dot-like, and is typically printed in pitches from 14 to 16 CPI. The largest font is 14 x 24 dot-like font and is typically printed at pitches from 10 to 14 CPI.

The EPIC 3000 always prints at 203 dots per inch (dpi) and always uses the scalable font to form characters. The resulting characters are not necessarily exactly the dot size

[19] Characters are dot-like because they are not guaranteed to be exactly at an exact dot equivalent. They are spaced in a fixed character cell that provides equivalent spacing and alignment as a fixed character size, however the actual character size is defined by the font designer.

indicated, but are always spaced in a fixed dot cell provided that the legacy commands are used. Adding or subtracting space between characters achieves different character pitches with a fixed character cell size. As each dot has a fixed size and position, only specific pitches are possible.

Character spacing may also be selected by requesting a print pitch based on characters pre inch. Once again, the results are not exact.

If the requested spacing is zero, the character spacing will be defined by the character definition, and will result in variable spacing.

Line spacing

The legacy commands select line spacing as lines per inch. With scalable characters, the lines per inch is a minimum spacing. If a character is larger then the spacing between lines, the line spacing will be increased to allow enough room for the characters on that line.

Selecting character size by points.

In addition to the legacy or classic method of character size selection, the EPIC 3000 allows selection by point size. Point sizes from 4 to 72 points may be selected for both the horizontal and vertical axes. If a horizontal point size of zero (0) is selected for the horizontal spacing the characters are printed using variable spacing based on the character definition, using the vertical point size for the horizontal point size.

To provide fine control over character size, two commands are available, one of which specifies the size in points, and the other of which specifies the size in ¼-point increments.

If the font is a stroke font, the boldness of the characters is controlled through variations in individual stroke width.

Diacritical Marks

It is sometimes required that diacritical marks like accents be placed on characters. The Unicode standard defines three basic ways to accomplish this.

1) Define a character with the diacritical mark as part of the character definion.
2) Define combining diacritical marks that may be added to existing characters.
3) Define diacritical marks as characters that may be positioned and combined with other characters.

The first is the easiest as the character is addressed as a single character and renders the character with the the character with the diacritical mark. The second defines a character with a negative position that places the character over the previous character when it is rendered. The third requires that the diacritical mark be positioned over the character.

The Unicode standard does not always make it clear how the diacritical marks are defined. In general characters from 0x300 through 0x36F are defined as combining diacritical marks. In this case, the character to character to be modified should be followed by the combining diacritical mark. The EPIC 3000 printer will position the diacritical mark over the previous character as defined by the diacritical mark. The printer does not control the positioning of the diacritical mark. The diacritical mark defines the fall back amount.

In the case where the diacritical mark is defined as a character, there is no negative motion in the diacritical marks definition so the host system must position the character entry position back to the previous character. This is may be done with a set position command or a simple backspace command. If backspace is used the printer will keep track of the width of the previous character and reposition the entry pointer to the end of the original character after the diacritical mark is rendered.

The EPIC 3000 printer does not handle multiple diacritical marks on the same character. The printer will not shift the second diacritical mark to prevent it from interfering with the previous. The EPIC 3000 processes the diacritical marks as defined in the font.

Character Cache

The EPIC 3000 supports a high speed character cache that can cache character bitmaps, outlines, tiles, or bands of characters. The caching technique is based on a least-recently used (LRU) algorithm. The Cache contains an entry for each size of character used. The amount of space in the cache used by each character is dependant on the size of the character.

At power up, the character cache is empty. The first ticket will require that all characters to be printed be generated from the vector data. This will typically make the first ticket slightly slower than subsequent tickets. It is also possible to define a user defined font with different metrics that will cause the cache to be flushed when any of the user characters are being accessed. If this should occur, the user defined font should be changed to match the other fonts in the printer.

Stacked or Linked fonts

The EPIC 3000 also uses a font stacking technology where fonts are linked together. This means that as each character is looked up, the first font in the stack containing the character is then printed. For example, if a customer would like to replace several standard characters with custom characters, a user defined font can be provided that would, if first in the link, replace the characters n the standard font.

It may also be desirable to stack fonts to provide a precedent for how individual characters are looked up. For example:

- An application may wish to alter the appearance of several characters for security.
- It might be necessary to define a group of special symbols for a specific application.
- It might be necessary to supplement a font with special characters like OCR characters.

To provide this flexibility, the EPIC 3000 has the ability to link up to 8 fonts together. When this is done, the first font in the link is searched first. If the character is not defined, the next font in the link is searched. This process is continued until the character is found or the last font is searched.

There are two ways to define a linked font. This first is to define a default linked font in the POR.INI file. If it link font is defined in the POR.INI file, it will be selected as the default power on font. Selecting font 0 will select the linked font provided that a link font is defined.

It is also possible to dynamically define a linked font. This requires that the fonts to be linked be aliased to a font id in the POR.INI file. Defining a linked font does not necessarily activate it. If the linked font was not already active, it must be selected by selecting font 0.

Bitmap Fonts

It is possible to use bitmap fonts with the EPIC 3000 Printer. Bitmap fonts are fixed pitch and are not scalable. They will function as legacy fonts or may be selected. The printer is supplied with 4 legacy bitmap fonts. They are in a 10x24, 12x24, 16x24 draft fonts format and a 16x24 OCR B font. They are defined as follows:

BMFont0 = chr10x24.bft 10 x 24 draft font with typical spacing of 16 characters per inch
BMFont1 = chr12x24.bft 12 x 24 draft font with typical spacing of 14 characters per inch
BMFont2 = chr16x24.bft 16 x 24 NLQ font with typical spacing of 12 characters per inch
BMFont4 = ocr16x24.bft 16 x 24 OCR font with typical spacing of 12 characters per inch

It is possible to define a custom bitmap font. Transact supports a bitmap font compiler that will convert a bitmap font picture file into a compressed bitmap file that may be loaded into the printer as a bitmap (.bft) font file.

The input to the program is a text file in a predefined format consisting of a font description followed by character definitions consisting of the character ID and then the character definition. The format is as follows: (Note lines preceded with # characters are comments.

```
# 11 by 24 font with a base line at 22
# F=X is not used in Thermal Products
W=11 H=24 B=22 F=0
; N=0000 U=0000  NULL
01      ...........
02      ...........
03      ...........
04      ...........
05      ...........
06      ...........
07      ...........
08      ...........
09      ...........
10      ...........
11      ...........
12      ...........
13      ...........
14      ...........
15      ...........
16      ...........
17      ...........
18      ...........
19      ...........
20      ...........
21      ...........
22      ...........
23      ...........
24      ...........
; N=0001 U=0020  SPACE
01      ...........
02      ...........
03      ...........
04      ...........
05      ...........
06      ...........
07      ...........
08      ...........
09      ...........
10      ...........
11      ...........
12      ...........
13      ...........
14      ...........
15      ...........
16      ...........
17      ...........
18      ...........
19      ...........
20      ...........
21      ...........
```

```
22      ...........
23      ...........
24      ...........
; N=0002 U=0021  EXCLAMINATION MARK
01      ...........
02      ...........
03      ...00......
04      ...00......
05      ...00......
06      ...00......
07      ...00......
08      ...00......
09      ...00......
10      ...00......
11      ...00......
12      ...00......
13      ...00......
14      ...00......
15      ...00......
16      ...00......
17      ...........
18      ...........
19      ...........
20      ..0000.....
21      ..0000.....
22      ..0000.....
23      ...........
24      ...........
; N=0003 U=0022  QUOTATION MARK
01      ...........
02      ...........
03      .00..00....
04      .00..00....
05      .00..00....
06      .00..00....
07      .00..00....
08      ...........
09      ...........
10      ...........
11      ...........
12      ...........
13      ...........
14      ...........
15      ...........
16      ...........
17      ...........
18      ...........
19      ...........
20      ...........
21      ...........
22      ...........
23      ...........
24      ...........
; N=0004 U=0023  NUMBER SIGN
01      ...........
02      ...........
03      ...........
04      .....00....
05      ..00.00....
06      ..00.00....
07      ..00.00....
08      ..00.0000..
09      ..0000000..
10      0000000....
11      0000.00....
12      ..00.00....
13      ..00.00....
14      ..00.0000..
15      ..0000000..
16      0000000....
17      0000.00....
18      ..00.00....
19      ..00.00....
20      ..00.00....
21      ..00.......
22      ...........
23      ...........
24      ...........
```

The characters must be in sequential order and must be assigned Unicode character codes. N=0001 is a sequence number in hex and is not used in controlling character generation or order. U=0021 is the Unicode address in hex and the characters must be in ascending Unicode address order.

Custom Fonts

The EPIC 3000 supports TrueType fonts. There are several companies that will provide custom character sets. The EPIC 3000 uses fonts provided by Monotype. You can contract Monotype through their website www.fonts.com, or by phone in U.S. & Canada (toll-free 1-800-424-8973, directly at 1-781-970-6020), or the United Kingdom (Free Phone 0800 371242, direct +44 (0)1737 765959.)

Internal Fonts

The EPIC 3000 is provided by default with a standard WGL4.0 outline-based font, and several bitmap fonts. Optionally the WGL4.0 font may be replaces with a GB18030 stroke-based font. Additional user defined outline or stroke fonts may be used as required.

The default font provided with the EPIC 3000 is called "Andale Mono Regular" from Monotype. This is a fixed-pitch font that produces good results when compared to legacy bitmap fonts. As another alternative, the printer may be ordered with a variable-pitch font, which will allow variable-pitch printing.

Font Storage

The EPIC 3000 supports a Flash file system used to store fonts, custom graphic and custom macros. A file system interface is provided for this system, where the host application may download files. In addition, TransAct Technologies provides a file loading tool that runs on Windows® based systems.

To allow flexible and easy support for all kinds of fonts, fonts are stored in the EPIC 3000 printer as a standard font file. These files are typically not visible to the user, however; TransAct Technologies provides a support tool that will allow the user to load their own font directly from Windows and change the way fonts are printed. It is also possible for the host application to load fonts into the printer.

The printer can contain up to 99 unique and selectable fonts. Any font may be selected at any time. In addition up to 8 fonts may be linked or stacked together. Some standard character size and character pitch commands are supported as legacy commands. The appearance of the print using those commands has been optimized using the TransAct WGL4 font. If you elect to use your own font, or the GB18030 font, you may wish to use the scalable font control commands to select the character size and spacing rather than the legacy commands.

TransAct Technologies provides a basic WGL4 font with the printer. This may be supplemented or replaced with a GB18030 Chinese font upon request. The printer will accept TrueType and compressed stroke fonts as defined by Monotype. If required, the customer may supplement the TransAct supplied fonts with their own custom fonts.

WARNING: If you elect to load fonts into the printer you must have proper rights to that font. **Do not** download a font to the printer if you do not have the right to use the font as a downloaded printer font**.**

Font Control Commands

Function **Select Font** **All**

ASCII [ESC] + 3 <ID>
Hexadecimal 1BH 2BH 31H
Decimal <27> <43> <51>
Description The [ESC] + 3 command selects the font for printing. This command is used to select a previously loaded font based on its alias.

Note: Selecting font 0 will select the linked font. If the selected font does not exist, the previous font will remain in effect.
Note: Selecting font 100 will select bit map font 0. 101 is bitmap font 1 and so on. Up to 7 bitmap fonts may be present.

Function **Define a Stacked or Linked Font** **All**

ASCII [ESC] + S <ID$_1$> <ID$_2$> <ID$_2$> … <0>
Hexadecimal 1BH 2BH 53H
Decimal <27> <43> <83>
Description The [ESC] + S command defines but does not select a stacked or linked font set. This command will define a linked list of previously loaded and aliased fonts into a linked font stack. The font ID is the same ID as in the select font command. Up to 8 fonts may be linked. The last entry must be 0. If the font does not exist, it will not be made part of the link.

Note: You must select font 0 to activate the linked font.
Note: You can not use bitmap fonts as part of a linked font

Function **Select Font by name** **All**

ASCII [ESC] + N <FileName> <0>
Hexadecimal 1BH 2BH 31H
Decimal <27> <43> <51>
Description The [ESC] + N command selects the font for printing by file name. This command is used to select a previously loaded font by its file name. If the selected font does not exist, the previous font will remain in effect.

Note: This command may be undesirable because it embeds in the application a file name that you may wish to change in the future. By using the Alias ID the font name may change, but the application will remain constant.
Note: This command may be used to load and select a bitmap font. If a bitmap font is selected with this command, it will define bitmap font 7 and select it. Once loaded, it may be reselected by selecting font 107.

Font Size and Spacing

The font typically defines the character size and line spacing. The typical font is proportional spaced. That is the spacing between characters varies. This is not always the most desirable mode of operation. To give the programmer some additional control over character spacing and line height, the Accutherm Supreme provides a width and height override command.

The following table converts dots to CPI (Characters Per Inch) points and ¼ points and is useful in calculating point size settings.

Dots 1/203”	Characters per Inch (CPI)	Points 1/72”	¼ Points 1/288”
8	25.38	2.84	11.35
9	22.56	3.19	12.77
10	20.30	3.55	14.19
11	18.45	3.90	15.61
12	16.92	4.26	17.02
13	15.62	4.61	18.44
14	14.50	4.97	19.86
15	13.53	5.32	21.28
16	12.69	5.67	22.70
17	11.94	6.03	24.12
18	11.28	6.38	25.54
19	10.68	6.74	26.96
20	10.15	7.09	28.37
21	9.67	7.45	29.79
22	9.23	7.80	31.21
23	8.83	8.16	32.63
24	8.46	8.51	34.05
25	8.12	8.87	35.47
26	7.81	9.22	36.89
27	7.52	9.58	38.31
28	7.25	9.93	39.72
29	7.00	10.29	41.14
41	4.95	14.54	42.56
48	4.23	17.02	68.10
51	3.98	18.09	72.35
68	2.99	24.12	96.47
101	2.01	35.82	143.29

Font Size and Spacing command interactions

There are interactions between some of the following commands and some of the legacy font selection commands. These interactions need to be considered when developing a application for this printer.

This printer uses a font rendering engine that relies on the font to provide character size and spacing information. Unfortunately, legacy applications assume all characters are the same and that the character size and spacing is fixed. To force the characters rendered by the font rendering engine to conform to legacy modes of operation, some post generation processing is performed to reposition the characters into a fixed size cell.

The set minimum character height and width ([ESC] + P and [ESC] + p), the set character spacing ([ESC] + I, [ESC] + i, [ESC] + J and [ESC] + j), the set minimum line spacing ([ESC] + V and [ESC] + v), and the legacy font select and spacing commands all interact.

The set minimum character height and width ([ESC] + P and [ESC] + p) commands set character size but in two different ways. In most systems a character point size refers only to the line spacing and indirectly to the character height. That is also true. The vertical character height referenced in these commands refer to the character height including the white space between lines. The horizontal character width is defined by the font. Normally only the character height would be specified and the width would be defined by the font and that's how these commands work if the Width is defined as zero. If the width is defined as zero this is used as a flag to the printer to generate characters as defined by the font and use the character width returned by the font. In effect the vertical point size passed to the font rendering engine is the same as the horizontal value. The added effect of the width being passed as zero is that any enforced horizontal spacing is disables. IE the effect of the [ESC] + I, [ESC] + i, [ESC] + J and [ESC] + j commands are disabled. If the width is not zero, the [ESC] + I, [ESC] + i, [ESC] + J and [ESC] + j remain in effect and only the resulting character size is changed, the horizontal spacing is not changed.

The legacy [ESC] ! <n> select the print mode effectively issues a set minimum character height and width command followed by a set character spacing command without effecting the pseudo fixed spacing flag.

The pseudo fixed spacing flag is a further complication required for dealing with fonts that are not truly fixed pitch. In some cases a fixed pitch font will have more that one character size depending on what the character is used for. This generally only affects Asian fonts where the ideograms are generally twice as wide as Latin characters. In fixed spacing mode, the printer will put the rendered character at whatever spacing is requested even if they don't fit. If the character is too big, it will overlap the previous and next character. To allow a fixed pitch operation that deals with small and large fixed pitch character, the printer has a pseudo-fixed pitch flag that will increase the spacing in multiples of the requested spacing until it fits.

The following table lists the commands and how they interact.

Command	Zero	Character width	Character height	Cell Width	Pseudo Fixed pitch flag
[ESC] + P, [ESC] + p	Width 0	Same as Height	From command	From Font	No effect
[ESC] + P, [ESC] + p	Width Not Zero	From Command	From command	Based on set character spacing command	Will be used if previously set and character spacing is not being defined by the font
[ESC] + I, [ESC] + i	Value 0	No effect	No effect	From Font	Set Off but has no effect
[ESC] + I, [ESC] + i	Value Not zero	No effect	No effect	From Command	Set Off
[ESC] + J, [ESC] + j	Value 0	No effect	No effect	From Font	Set On but has no effect
[ESC] + J, [ESC] + j	Value Not zero	No effect	No effect	A multiple of the value defined by the command	Set On
[ESC]I <n>		From POR.INI definition	From POR.INI definition	As defined by command	Will be used if previously set.

Function	**Set minimum character height and width in points.**	**All**
ASCII	[ESC] + P <w><h>	
Hexadecimal	1BH 2BH 50H	
Decimal	<27> <43> <80>	
Range	w = 0, 4 – 72 h = 4 - 72	

The [ESC] + P command will set the minimum character width or height based on "w" for the width and "h" for height, where "w" and "h" are in points, defined as 1/72nd of an inch increments.

If the character width is set to zero, the height will be used for the width and proportional spacing will be used.

NOTE: The set pitch command will take precedence unless this command selects 0 width.

Function	**Set minimum character height and width in ¼ points.**		**All**
ASCII	[ESC] + p <w><h>		
Hexadecimal	1BH 2BH 70H		
Decimal	<27> <43> <112>		
Range	w = 0, 16 – 255	h = 16 - 255	

The [ESC] + p command will set the minimum character width or height based on "w" for the width and "h" where "w" and "h" are in ¼ points or 1/288th of an inch increments. This approximates setting characters by dot.

If the character width is set to zero, the height will be used for the width and proportional spacing will be used.

NOTE: The set pitch command will take precedence unless this command selects 0 width.

Function	**Set Character spacing in points.**	**All**
ASCII	[ESC] + I <d>	
Hexadecimal	1BH 2BH 49H	
Decimal	<27> <43> <73>	
Range	d = 0, 4 – 72	

The [ESC] + I command will set the character spacing in points, where one point is defined as 1/72nd of an inch. This command will force mono-space printing. It will override any character spacing set by the set character height and width commands defined above. This spacing will be enforced until deactivated by setting the value to 0 or if the set character height and width commands use a 0 for the width indicating proportional spacing should be used. This command differs from the [ESC] + J command in that all characters are centered on the fixed cell size. It the character is too big for the cell, it may overlap the previous and next character. The character size is not adjusted to fit the cell.

If d = 0 variable spacing is selected.

NOTE: If the current character size is too large for the selected spacing, the characters will overlap. Variable spacing is recommended.

Function	**Set Character spacing in ¼ points.**	**All**
ASCII	[ESC] + i <d>	
Hexadecimal	1BH 2BH 69H	
Decimal	<27> <43> <105>	
Range	d = 0, 16 – 255	

The [ESC] + i command will set the character spacing in points, where ¼ point is defined as 1/288th of an inch. This command will force mono-space printing. It will override any character spacing set by the set character height and width commands defined above. This spacing will be enforced until deactivated by setting the value to 0 or if the set character height and width commands use a 0 for the width indicating proportional spacing should be used. This command differs from the [ESC] + j command in that all characters are centered on the fixed cell size. It the character is too big for the cell, it may overlap the previous and next character. The character size is not adjusted to fit the cell.

If d = 0 variable spacing is selected.

NOTE: If the current character size is too large for the selected spacing, the characters will overlap. Variable spacing is recommended.

Function	**Set Character spacing in points with adjustment.**	**All**
ASCII	[ESC] + J <d>	
Hexadecimal	1BH 2BH 4AH	
Decimal	<27> <43> <74>	
Range	d = 0, 4 – 72	

The [ESC] + J command will set the character spacing in points, where one point is defined as 1/72nd of an inch. This command will force mono-space printing. It will override any character spacing set by the set character height and width commands defined above. This spacing will be enforced until deactivated by setting the value to 0 or if the set character height and width commands use a 0 for the width indicating proportional spacing should be used. This command differs from the [ESC] + I command in that if the character is too large for the cell, the cell will be expanded in multiples of <d> until the character fits.

If d = 0 variable spacing is selected. However, note that the cell adjustment flag will remain set and if legacy commands are used they will allow the cell to be expanded.

NOTE: If the current character size is too large for the selected spacing, the cell size will be expanded

Function	**Set Character spacing in points with adjustment.**	**All**
ASCII	[ESC] + j <d>	
Hexadecimal	1BH 2BH 6AH	
Decimal	<27> <43> <106>	
Range	d = 0, 16 – 255	

The [ESC] + j command will set the character spacing in points, where ¼ point is defined as 1/288th of an inch. This command will force mono-space printing. It will override any character spacing set by the set character height and width commands defined above. This spacing will be enforced until deactivated by setting the value to 0 or if the set character height and width commands use a 0 for the width indicating proportional spacing should be used. This command differs from the [ESC] + i command in that if the character is too large for the cell, the cell will be expanded in multiples of <d> until the character fits.

If d = 0 variable spacing is selected. However, note that the cell adjustment flag will remain set and if legacy commands are used they will allow the cell to be expanded.

NOTE: If the current character size is too large for the selected spacing, the characters will overlap. Variable spacing is recommended.

Function	**Set minimum Line Spacing in Points**	**All**
ASCII	[ESC] + V <d>	
Hexadecimal	1BH 2BH 56H	
Decimal	<27> <43> <86>	
Range	d = 0, 4 – 72	

The [ESC] + V command will set the line spacing in points, where one point is defined as 1/72nd of an inch.

If d = 0 variable spacing is selected.

NOTE: This is the minimum spacing. If the character height setting requires a larger spacing, the character height will override this setting.

Function	**Set minimum Line Spacing in ¼ Points**	**All**
ASCII	[ESC] + v <d>	
Hexadecimal	1BH 2BH 76H	
Decimal	<27> <43> <118>	
Range	d = 0, 16 – 255	

The [ESC] + v command will set the line spacing in ¼ points, where ¼ point is defined as 1/288th of an inch.

ASCII	[ESC] + v <d>
Hexadecimal	1BH 2BH 76H
Decimal	<27> <43> <118>
Range	d = 0, 16 – 255

If d = 0 variable spacing is selected.

NOTE: This is the minimum spacing. If the character height setting requires a larger spacing, the character height will override this setting.

Function	**Set stroke font brush size.**	**All**
ASCII	[ESC] + B <w>	
Hexadecimal	1BH 2BH 42H	
Decimal	<27> <43> <66>	
Range	w = 0, 6 – 200	

Description The [ESC] + B command will set brush stroke percentage for stroke fonts. If the brush size is set to zero the font design stroke width will be used.

Values from 6 to 200 represent 0.4 to 12% of the em-width of the font. The default for most fonts is about 3%. The EPIC 3000 using the GB18030 font supplied by TransAct Technologies produces the best characters with a brush size of about 100.

Note: The default value for the Brush stroke may be set in the POR.INI file.

Function **Redefine Legacy Font definitions.** **All**

ASCII [ESC] + r < ID > < FontID ><Horz><Vert><Spacing>
Hexadecimal 1BH 2BH 74H
Decimal <27> <43> <116>

Description The [ESC] + r command will allow the legacy fonts defined in the POR.INI file to be dynamically redefined.

The ID is the legacy font ID

Epson Fonts	ID	Ithaca Mode Fonts	ID	Ithaca Mode Barcodes	ID	Ithaca Mode OCR	ID
Epson1	0	PcOS1	0	Bar Code1	4	OCR	7
Epson2	1	PcOS2	1	Bar Code2	5		
Epson3	2	PcOS3	2	Bar Code3	6		
		PcOS4	3				

The Font ID is the logical font assignment in the POR.INI file where 0-99 refer to scalable fonts and 100-199 refer to bitmap fonts.

The Horz and Vert value are the horizontal and vertical size of the font in 1/8 points.

Spacing is the absolute spacing in dots. If this is 0 the font definition is used for the spacing.

Note: The default value for the Brush stroke may be set in the POR.INI file.

[ESC] [P Set character pitch (Legacy mode command) PcOS

ASCII [ESC] [P <n>
Hexadecimal 1BH 5BH 50H <n>
Decimal <27> <91> <80> <n>
IPCL &%F1, &%F2, &%F3, &%F4, &%F5, &%F6, &%F7
EPOS [ESC] [SP] <n>

Description The [ESC] [P <n> command sets character per inch print pitch to <n>. The printer resolution limits the exact print pitch. The following table lists the exact pitch for various values on

<n>	Resulting Characters per Inch	IPCL	<n>	Resulting Characters per Inch	IPCL
1	1.00		16	16.00	
2	2.00		17	17.33	&%F1
3	3.01		18	17.33	
4	4.00		19	18.91	
5	4.95		20	20.80	&%F5
6	5.94		21	20.80	
7	6.93		22	23.11	
8	8.00	&%F7	23	23.11	
9	9.04		24	23.11	&%F4
10	9.90	&%F3	25	23.11	
11	10.95		26	26.00	
12	12.23	&%F2	27	26.00	
13	13.00		28	26.00	
14	13.87		29	29.71	
15	14.86	&%F6	30	29.71	

Table 15 Character Pitch

This command disables any right-side spacing set by the [ESC] V command. It enforces this spacing on the current font selection even if the character is too large for the spacing. In addition, when font changes are made, the character pitch is maintained.

Unicode

As computer systems started to address more and more international environments, the classic ASCII standard with code pages became unworkable. Several competing systems were developed. however it was clear that a standard needed to be developed. In 1991 Version 1.0 of the Unicode standard was developed, to standardize how and where characters are to be addressed in an expanded addressing scheme. In 2006 Version 5.0 of the Unicode standard was published and generally accepted. The EPIC 3000 follows this standard for character placement and encoding and Unicode addresses from 0 to 1114111 (0x00 to 0x010FFFF) are supported by the EPIC 3000 Printer.

Note: If a custom font is used that is not in Unicode order, the order of the font will be used as if it were in Unicode order. Any subsequent character mappings will assume to font is in Unicode order and may not produce the desired effects.

Unicode Encoding

The EPIC 3000 Printer supports Unicode character addressing using Unicode Transform Format or UTF as defined in the Version 5.0 Unicode Specification. There are several forms of UTF encoding, UTF32 big and little-endian, UTF16 big and little endian and UTF8.

UTF-32

UTF-32 is a straight forward although not very efficient way to access characters above 255. UTF-32 essentially sends four 8-bit bytes that form a 32-bit address to access the desired character. Basic UTF-32 does not define the byte order. If you wish to use UTF-32 and allow the printer to determine the byte order, you must send the byte order mark (BOM) (0x0000FEFF) before you send any characters. It is difficult to prevent loss of byte order synchronization with UTF32 however, sending the BOM periodically can sometimes resynchronize the 8bit byte to the 32bit address. If UTF-32 is selected, all data sent to the printer must be 32 bits. All commands and command parameters are also 32 bit, however only values between 0 and 255 are valid. This makes UTF32 rather inefficient.

UTF-32BE uses the big-endian method of sending the four bytes. This method sends the high byte first and then the low byte. It is not required to send the byte order mark (BOM) (0x0000FEFF) for the correct byte order to be initialized. Sending the BOM in big-endian would be as follows:

0x00 0x00 0xFE 0xFF

UTF-32LE uses the little-endian method of sending the four bytes. This method sends the low byte first and then the high byte. It is not required to send the byte order mark (BOM) (0x0000FEFF) for the correct byte order to be initialized. Sending the BOM in little-endian would be as follows:

0xFF 0xFE 0x00 0x00

UTF-16

UTF-16 is the most straightforward way to access characters above 255. UTF-16 essentially sends two 8-bit bytes that form a 16-bit address to access the desired character. Basic UTF-16 does not define the byte order. If you wish to use UTF-16 and allow the printer to determine the byte order, you must send the byte order mark (0xFEFF) before you send any characters. To prevent loss of byte order synchronization, you should periodically send the byte order mark to resynchronize the printer with your application. If UTF-16 is selected, all data sent to the printer must be 16 bits. All commands and command parameters are also 16 bit, however only values between 0 and 255 are valid. Extended addressing uses surrogate pairs to encode values above 0xFFFF.

Scalar Value	UTF-16	
xxxxxxxxxxxxxxxx	xxxxxxxx xxxxxxxx	
000uuuuuxxxxxxxxxxxxxxxx	110110wwwwxxxxxx	110111xxxxxxxxxx

Note: wwww = uuuuu – 1 and uuuuu may not be larger than 10000

Unicode Extended UTF16 Encoding Example

3 Byte output Example

Hex Character Code: 0x00010302

00 01 03 02 - Original Hex Code(s)

00000000 00000001 00000011 00000010 - Converted to Binary

00000000 000 00001 000000 11 00000010 - Bit Sections

Discarded

0000 000000 11 00000010 - Adjusted by subtracting 1 from uuuuu

0000000000 1100000010 - Merged into two 10 bit values

1101100000000000 1101111100000010 - Surrogates added

1101 1000 0000 0000 1101 1111 0000 0010 - 2 16 bit values

0xD800 0xDF02 - 2 16 bit hex values

UTF-16

With UTF16 values from 0x0000000000 to 0x0000D7FF and 0x0000E000 to 0x0010FFFF may be represented Values from 0x0000D800 to 0x0000DFFF and above 0x0010FFFF are not valid and in fact are not valid for any characters in Unicode rev 5.0 regardless of encoding.

UTF-16BE uses the big-endian method of sending the two bytes. This method sends the high byte first and then the low byte. It is not required to send the byte order mark

(0xFEFF) for the correct byte order to be initialized. Sending the BOM in big-endian would be as follows:

0xFE 0xFF

UTF-16LE uses the little-endian method of sending the two bytes. This method sends the low byte first and then the high byte. It is not required to send the byte order mark (0xFEFF) for the correct byte order to be initialized. Sending the BOM in big-endian would be as follows:

0xFF 0xFE

UTF-8

UTF-8 uses a Multiple Byte Character Sequence (MBCS) to identify the desired Unicode character. This encoding method is less straightforward but preserves some of the 8-bit character of ASCII encoding.. This method uses unique bit sequences at the MSBs of a byte to determine its location and meaning within the MBCS encoding. See the table below for more information. If UTF-8 is selected all data sent to the printer must be encoded. All command parameters over 127 must be encoded in UTF-8.

UTF-8TXT uses a Multiple Byte Character Sequence (MBCS) to identify the desired Unicode character. This encoding method is identical to UTF-8 except commands and command parameters over 127 are not UTF-8 encoded. They must be sent unmodified as 8-bit values.

UTF-8TXT is probably the easiest mode to use. It allows normal 8 bit commands and graphics while still supporting the full Unicode character encoding range. Typically EPIC 3000 Printers are shipped set to this mode**.**

Scalar Value	1st Byte	2nd Byte	3rd Byte	4th Byte
0000000 00000000 0xxxxxxx	0xxxxxxx			
0000000 00000yyy yyxxxxxx	110yyyyy	10xxxxxx		
0000000 zzzzyyyy yyxxxxxx	1110zzzz	10yyyyyy	10xxxxxx	
00uuuuu zzzzyyyy yyxxxxxx	11110uuu	10uuzzzz	10yyyyyy	10xxxxxx

NOTE: Where values from 0x001FFFFF may be encoded, only values up to 0x0010FFFF are valid in Unicode 5.0

Unicode UTF8 Encoding Example

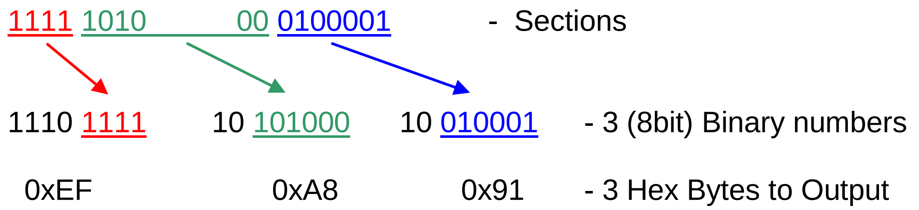
3 Byte output Example

Hex Character Code: FA11

FA   11   - Original Hex Code(s)

↓   →

11111010   000100001   - Converted to Binary



1111 1010 00 0100001 - Sections

1110 1111 10 101000 10 010001 - 3 (8bit) Binary numbers

0xEF 0xA8 0x91 - 3 Hex Bytes to Output

Unicode Encoding Control Commands

Function	**Initiate Unicode UTF-32BE Encoding**	**All**
ASCII	[ESC] + h	
Hexadecimal	1BH 2BH 68H	
Decimal	<27> <43> <104>	

Description The [ESC] + h command will put the printer into UTF-32BE character encoding mode of operation. If you wish to access characters above 255, you must select a Unicode encoding such as UTF-32BE.

UTF-16 is the most straightforward way to access characters above 255, however UFT32 does not require Surrogates to address character values greater than 65535.

UTF-32BE uses the big-endian method of sending the four bytes. This method sends the high byte first and then the lower bytes.

Note: Once selected, all information sent to the printer must then use this encoding, even for non-print commands

Function	**Initiate Unicode UTF-32LE Encoding**	**All**
ASCII	[ESC] + l	
Hexadecimal	1BH 2BH 6CH	
Decimal	<27> <43> <108>	

Description The [ESC] + l command will put the printer into UTF-32LE character encoding mode of operation. If you wish to access characters above 255, You must select a Unicode encoding such as UTF-16LE.

UTF-16 is the most straightforward way to access characters above 255, however UFT32 does not require Surrogates to address character values greater than 65535.

UTF-32LE uses the little-endian method of sending the four bytes. This method sends the low byte first and then the higher bytes.

Note: Once selected, all information sent to the printer must then use this encoding, even for non-print commands

Function	**Initiate Unicode UTF-16BE Encoding**	**All**
ASCII	[ESC] + H	
Hexadecimal	1BH 2BH 48H	
Decimal	<27> <43> <72>	

Description The [ESC] + H command will put the printer into UTF-16BE character encoding mode of operation. If you wish to access characters above 255, you must select a Unicode encoding such as UTF-16BE.

UTF-16 is the most straightforward way to access characters above 255, sending two 8-bit bytes that form a 16-bit address to access the desired character.

UTF-16BE uses the big-endian method of sending the two bytes. This method sends the high byte first and then the low byte.

Note: Once selected, all information sent to the printer must then use this encoding, even for non-print commands

---

Function **Initiate Unicode UTF-16LE Encoding** **All**
ASCII [ESC] + L
Hexadecimal 1BH 2BH 4CH
Decimal <27> <43> <76>
Description The [ESC] + L command will put the printer into UTF-16LE character encoding mode of operation. If you wish to access characters above 255, You must select a Unicode encoding such as UTF-16LE.

UTF-16 is the most straightforward way to access characters above 255, sending two 8-bit bytes that form a 16-bit address to access the desired character.

UTF-16LE uses the little-endian method of sending the two bytes. This method sends the low byte first and then the high byte.

Note: Once selected, all information sent to the printer must then use this encoding, even for non-print commands

---

Function **Initiate Unicode UTF-8 Encoding (MBCS)** **All**
ASCII [ESC] + M
Hexadecimal 1BH 2BH 4DH
Decimal <27> <43> <77>
Description The [ESC] + M command will put the printer into UTF-8 character encoding mode of operation. If you wish to access characters above 255, You must select a Unicode encoding such as UTF-8.

UTF-8 uses a Multiple Byte Character Sequence (MBCS) to identify the desired Unicode character. This encoding method is less straightforward. This method uses unique bit sequences at the MSBs of a byte to determine its location and meaning within the MBCS encoding. See the table below for more information.

Note: Once selected, all information sent to the printer must then use this encoding, even for non-print commands

Scalar Value	1st Byte	2nd Byte	3rd Byte	3rd Byte

00000000 0xxxxxxx	0xxxxxxx			
00000yyy yyxxxxxx	110yyyyy	10xxxxxx		
zzzzyyyy yyxxxxxx	1110zzzz	10yyyyyy	10xxxxxx	
000uuuuu zzzzyyyy yyxxxxxx	11110uuu	10uuzzzz	10yyyyyy	10xxxxxx

Function **Initiate Unicode UTF-8 Text only Encoding (MBCS)** **All**

ASCII [ESC] + T
Hexadecimal 1BH 2BH 54H
Decimal <27> <43> <84>
Description The [ESC] + T command will put the printer into UTF-8 Text only character encoding mode of operation. This mode is identical to the UTF-8 mode described above, except commands and there parameters are not UTF encoded. For example the following command would be used to select underline on:

[ESC] W 128.

If true UTF-8 encoding were in effect, the 128 parameter would be UTF encoded to [ESC] W 194 128. With UTF-8 Text only mode this command is simply [ESC] W 128.

Note: This command also applies to graphic data being sent to the printer. The graphic data is a command and not text. It is not UTF-8 encoded.

Function **Initiate Normal 8-bit ASCII Character Encoding** **All**

ASCII [ESC] + A
Hexadecimal 1BH 2BH 41H
Decimal <27> <43> <65>
Description The [ESC] + A command will put the printer into normal character encoding mode of operation. One byte = one character. In this mode international characters must be selected by selecting the appropriate code page for translation.

Legacy Printer Features that Have Changed

Because this product employs outline and stroke font character generation, support for several legacy features are changed from standard EPIC 3000 version.

Dynamic code page definition

Dynamic code page definition is still supported when in ASCII mode, however Unicode is now used for the source character locations.

File System

File System Interface

The EPIC 3000 provides a file system to support fonts, configuration information, user graphics and macros.

There are a number of commands that are provided to support the file system. In general, files need to be opened for read or write, read or written, and then closed. There is a command that will delete a file, and print or return a file directory.

TransAct Technologies provides a Windows® based tool that will interact with the EPIC 3000 and provide a drag and drop interface to the file system.

The file system in the EPIC 3000 is partitioned into two sections, one for internal system use by the printer, and one for user information.

The system partition is referred to as partition 0. It is reserved for fonts, configurations and code page files. This partition cannot be deleted or completely erased. The second partition is for all other information. There is a command that will erase all the files in this partition. The partition where files are placed is determined by the three character extension. There is a third RAM file partition that may be used to saving temporary information.

File System commands

Function	**Open File command.**	**All**
ASCII	[ESC] [RS] O < Mode > <space> < Filename ><0>	
Hexadecimal	1BH 1EH 4FH	
Decimal	<27> <30> <79>	
Mode	Mode of operation "r" for read or "w" for write.	
FileName	File name from 1 to 30 characters including a three character extension.	

Description The [ESC] [RS]O command will select and open a file for the selected operation. If the file being opened for write exists, the existing file will be overwritten. Note that only one file may be open for external operations at any one time.
The Mode and FileName take the following format and must be null terminated:
r Filename.ext<0>
Valid Modes are:

"r"	Read.
"w"	Write
"w+"	Write Append (Future enhancement)
"ram"	open a RAM file for write.

Note: RAM files capability is an option and not available on all products.

File Extensions are any three characters. The following are predefined and reserved for internal use.

Extension	Partition 0= System 1= User	Definition
.udf	1	Undefined macro type
.mac	1	Command Macro
.img	1	Graphic image. (Internal format)
.bgp	1	Bitmapped internal graphic
.chr	1	User Character definition
.cfg	0	configuration.
.ttf	0	true type font
.ccc	0	compressed stroke font
.btf	0	Bitmap font definition
.cpm	0	code page map.
.bmp	1	bitmap graphic file
.gph	1	raster graphic file.
.ini	0	System information file
.fcg	0	Field Configuration File
.upd	0	System Update file
.sys	0	Load image
.sy_	0	Compressed load image.

Note: All other file extensions will be placed in Partition 1 (user space)

Function **Return Free space for Open FIle.** **All**

ASCII [ESC] [RS] S
Hexadecimal 1BH 1EH 53H
Decimal <27> <30> <83>
Description The [ESC] [RS]S command will return an identifier byte and 4 additional bytes representing a 32 bit value (LSB First) representing the amount of free space in the partition containing the open file.

The format is as follows:

S <B_{7-0}><B_{15-8}><B_{23-16}><B_{31-24}>

Function **Return Free space for this partition.** **All**

ASCII [ESC] [RS] s<n>
Hexadecimal 1BH 1EH 73H
Decimal <27> <30> <115>
Where n = The partition
Description The [ESC] [RS]s command will return an identifier byte and 4 additional bytes representing a 32 bit value (LSB First) representing the amount of free space in the partition.

The format is as follows:

S <B_{7-0}><B_{15-8}><B_{23-16}><B_{31-24}>

Function **Close File command.** **All**
ASCII [ESC] [RS] C
Hexadecimal 1BH 1EH 43H
Decimal <27> <30> <67>
Description The [ESC] [RS]C command will close the currently open file.

Function **Close All Files command.** **All**
ASCII [ESC] [RS] K
Hexadecimal 1BH 1EH 4BH
Decimal <27> <30> <75>
Description The [ESC] [RS]K command will close the font system and close all currently open files. Internal fonts will be reopened automatically if used.

Function **Delete File command.** **All**
ASCII [ESC] [RS] D <Filename><0>
Hexadecimal 1BH 1EH 44H
Decimal <27> <30> <68>
FileName File name from 1 to 30 characters including a three character extension, null terminated.
Description The [ESC] [RS]D command will select and delete a file.

Note: Some of the system files are protected and cannot be deleted.

Function **Set/Clear File Attributes command.** **All**
ASCII [ESC] [RS] A < Attbs > <space> < Filename ><0>
Hexadecimal 1BH 1EH 41H
Decimal <27> <30> <64>
Attbs File attributes to modify.
FileName File name from 1 to 30 characters including a three character extension.

Each file has several attributes associated to it. They include S, R, and H.

Attribute	Syntax	Name	Use
S	+S or -S	System	This is a system file.
R	+R or -R	Read Only	This file cannot be erased or modified.
H	+H or -H	Hidden	This file is hidden and not displayed in the directory listing.

NOTE: Attributes can be combined, however, each needs to have the + or – as a prefix.

To allow these attributes to be set and cleared, the [ESC][RS]A command can be used. The format is as follows:

[ESC][RS]A-R-S FileName<0>
This command will remove the Read only and System attributes form File name.

Function **Return the last file command status.** **All**
ASCII [ESC] [RS] ?
Hexadecimal 1BH 1EH 3FH
Decimal <27> <30> <63>

The [ESC][RS]? Command requests the file system to return the status of the last file operation.

This command, returns an identifier byte, followed by 2 bytes indicating the status results of the last file command. The format will be as follows:

?<Status><Details> or 3F, (47 or 42), <Details>

Where:
Status = 'G' for success and 'B' for Failure
Detail = Detailed status as a binary byte with bit definitions as follows:

Bit	Hex	Decimal	Function
0	01	1	File Open
1	02	2	File in Write Mode
2	04	4	The Read response is shorter than requested and EOF has been encountered.
3	08	8	The file is already open.
4	10	16	The file system has no space for the preceding operation. Could be out of Flash or out of Buffer space.
5	20	32	A write operation has been attempted to a read only file.
6	40	64	File requested was not found
8	80	128	An error has occurred. Other bits may be set that give additional detail. (This bit determines the G or B status in the previous byte)

Function **Return File CRC command.** **All**
ASCII [ESC] [RS] G <Filename><0>
Hexadecimal 1BH 1EH 47H
Decimal <27> <30> <71>
FileName File name from 1 to 30 characters including a three character extension, null terminated.
Description The [ESC] [RS]G command will return the CRC of the specified file.

Note: If the file does not exist a CRC will not be valid and the last file system status will be updated.

Function **Query File Status.** **All**

ASCII [ESC] [RS] q
Hexadecimal 1BH 1EH 71H
Decimal <27> <30> <113>
Description The [ESC] [RS]q command will return the CRC and length of the currently open file.

Return

ACK	File open	NAK	File not open
Length	High	0	
Length	Low	0	
CRC	High	0	
CRC	Low	0	

Function **Write File command.** **All**

ASCII [ESC] [RS] W <L_L><L_H><… data ..>
Hexadecimal 1BH 1EH 57H
Decimal <27> <30> <87>
Description The [ESC] [RS]W command sends data to the printer to be stored in the file. The <LL><LH> parameters specify the length of data that will follow where the length is LH * 256 + LL. The data is treated as binary data with no translations.

Function **Read File command.** **All**

ASCII [ESC] [RS] R <L_L><L_H>
Hexadecimal 1BH 1EH 52H
Decimal <27> <30> <82>
Description The [ESC] [RS]R command requests that data be read from the file and returned to the host.

The <LL><LH> parameters specify the length of data that should be returned where LH * 256 + LL specifies the number of returned bytes. The data is treated as binary data with no translations. If there is not enough data in the file to make up the requested length, only the available data is returned.

Function **Generate and return a file directory report.** **All**

ASCII [ESC] [RS] I
Hexadecimal 1BH 1EH 49H
Decimal <27> <30> <73>
Description The [ESC] [RS]I command requests that a formatted text directory be returned from the printer. Each line is null terminated.

Function **Erase all files in a partition** **All**

ASCII [ESC] [RS] X <p>
Hexadecimal 1BH 1EH 58H
Decimal <27> <30> <88>
<p> selects the partition. 0 = System, 1 = User.
Description The [ESC] [RS]X command requests that the selected partition be reformatted. Reformatting the system partition <0> is not recommended, as it will erase all fonts and render the printer unusable.

Note: The system partition cannot be erased using this command. The system partition can only be erased in Maintenance mode.

Function **De-fragment the file system.** **All**

ASCII [ESC] [RS] F

Hexadecimal 1BH 1EH 46H

Decimal <27> <30> <70>

Description The [ESC] [RS]F command forces the file system to go through the file system and clean up deleted file sectors. All sectors flagged for deletion are actually erased and consolidated when this command is issued.

Note: File space is not necessarily freed up by a file delete. Sectors may be marked for deletion but still be present but inactive in the file system. These sectors take up flash space. Each cluster has a fixed number of sectors, and if the number of deleted sectors in a cluster exceed a predefined threshold, the cluster is de-fragmented automatically. This command forces all clusters to be de-fragmented

---

Function **Verify files.** **All**

ASCII [ESC] [RS] V

Hexadecimal 1BH 1EH 56H

Decimal <27> <30> <86>

Description The [ESC] [RS]V opens and reads the VERIFY.CFG file. This file contains a list of all files that are to be validated in the printer and the expected CRC of the file. If all the files verify, this command will return VG followed by it's 2 byte CRC. If any of the files do not verify, the command will return VB followed by it's 2 byte CRC.

For example the file might look like this:

```
Por.ini  0x06FF
```

Only the Por.ini file will be checked in this example.

An additional and optional feature of this command is that it can verify the CRC of the operating firmware. By adding "Firmware" as a file name, this command will recalculate the Firmware CRC and compare it to the master value. If the recalculation does not match the master value, this command will return a failed response. The file would be as follows to add the Firmware check. By using the master value, this file need not be updated if the firmware is updated.

```
Por.ini  0x06FF
Firmware
```

By knowing the CRC of the Verify.cfg file, the host application can verify that all the other files are correct (and optionally the firmware) without knowing anything about the other files of firmware.

Note: The typical printer is not shipped with a Verify.cfg file.

Note: This command is not performed as a condition of normal operation. It is up to the host application to refuse to use the printer if this command

returns a fail to verify status.
Note: if the Verify.cfg file is not present, the verify command will return VB and a 0 CRC.

POR.INI file

The POR.INI file is used to control how fonts are encoded, named, identified and linked, as well as allowing how the font to be printed is controlled. In addition the POR.INI file defines how each legacy font is defined and printed. It also controls several other features and functions.

The POR.INI file is divided into sections:

- [encoding]
  This section defines how the printer encodes the character set. It may be ASCII with code page, or Unicode. If ASCII is selected the default code page is selected and/or defined in this section.
- [font]This section defines how the scalable fonts are named and generated. You can control the font cache size and partitions, the Hinting, link fonts and font abstractions.
- [bmfont]
  This section defines and abstracts any bitmap font definitions.
- [Legacy]
  This section defines how the previously defined fonts are used to generate legacy fonts for each emulation.

The following is an example of the POR.INI file:

```
; Default System Configuration.PE9105-1.04
[encoding]
;To remap Unicode characters, define a UniRemap.cpm file.
UniMapfile = UniRemap.cpm
mode = ASCII
;NOTE: A code page is only used in ASCII mode.
;To specify a code page, use one of the following forms:
CodePage = 437
;CPFile = CP8959-1.cpm
;To remap Unicode characters, define a UniRemap.cpm file.
;UniMapfile = UniRemap.cpm
[font]
;Optionally specify the Cache Partitions
;Fontcache = 320,256,64
;True Type font hinting may be disabled by setting Nohint to 1
;Nohint = 0
;True Type font line spacing fit; 1-Min 2-Typ, 3-Max
;TTFit = 2
;Extra Character Bolding
;Bolding = 300,10
;Specify Linked fonts starting with LinkFont1.
;LinkFont1 will be searched first.
;You may specify up to 8 linked fonts.
;if Link Fonts are defined, they will be Font0.
LinkFont1 = TactMOD.ttf
LinkFont2 = TactWGL_M.ttf
LinkFont3 = TactWGL_V.ttf
```

```
LinkFont4 = TactGB18030.ccc
;From 1 to 99 fonts may be defined
Font1 = TactMOD.ttf
Font2 = TactWGL_M.ttf
Font3 = TactGB18030.ccc
Font4 = TactOCR.ttf
Font5 = TactWGL_V.ttf
;The brush size effects only stroke fonts.
Brush = 100
[legacy]
;EmulationMode = Font,Horizontal,Vertical,Width.
; Where:
; If Font = 0 Use Linkfont else 1-4 above.
; If Font is 100 or greater use BMFont (Font - 100). BMFonts only used
Width.
; Horizontal and Vertical are in 8th points, Width in Dots.
; See the Programmers guide for more information.
Epson1 = 0,56,72,10
Epson2 = 0,64,72,14
Epson3 = 0,50,60,8
PcOS1 = 0,56,72,10
PcOS2 = 0,64,72,14
PcOS3 = 0,80,80,18
PcOS4 = 0,80,80,20
PcOSOCR = 4,80,80,20
BarC1 = 0,56,72,10
BarC2 = 0,64,72,14
BarC3 = 0,50,60,8
; Up to 8 user defined fonts may be defined.
USRFont1 = 100,9,24,10
USRFont2 = 101,12,24,13
USRFont3 = 4,56,72,0
USRFont4 = 4,64,72,0
[gtech]
;Add custom GTECH Prams here
;Pram0 = Pram0Return
[bmfont]
;There may be up to 8 bitmap fonts.
;Bitmap fonts are fixed sizes and have no options
BMFont0 = chr10x24.bft
BMFont1 = chr13x24.bft
BMFont2 = chr15x24.bft
BMFont3 = ocr16x24.bft
[usb]
;USB Driver
;Usbclass may be 255, 7, GSA, 108 ...
;Default is class 7 (Printer device)
Usbclass = 7
;;UsbSpeed may be 0 for Full or 1 for High
Usbspeed = 1
;USB Plug and Play Print driver. 1 = PnP Active.
;USB_PnP = 1
[options]
; to remove white space set Linespace to 0
```

Font1 through Font99 may be defined, and the font number is the alias used by the set font command, e.g. Font23 is selected by doing a select font 23 command. Font 0 is reserved for selecting the linked font.

The printer may contain one default linked font. A linked font is a method of allowing the user to replace characters in a standard font with custom characters, described in more detail in an earlier section. The POR.INI file is one way of defining a linked font. In the above POR.INI file link the link font consists of User, TactWGL, and the TactGB18030 fonts. When a character is to be printed, the user font will be searched followed by TactWGL and then the TactGB18030 font. The first font containing the character will define the character.

Bitmap fonts are not recommended, they should only be used if an exact bitmap is required. Bitmap fonts are not scalable like true type fonts, only the normal 2X, 3X and so on scaling is available. TransAct Technologies can upon request and signing an NDA provide tools to allow the customer to develop there own bitmap fonts. These fonts must be in Unicode order but only need support the specific characters needed in the font. NOTE: When loaded and made available the legacy select font commands should select the bitmap font by adding 100 to the font ID. For example to use BMFont0, select font 100 in the Legacy font definition. (Note: Only the font ID is used from the legacy font definition if a bitmap font is selected.)

Legacy font definitions define the Size, spacing, font, quality and spot size to be used to generate a legacy font. For example the PcOS [esc] I<0> font select command will select PcOS1 font. (Note that the font select and the Font ID are offset by one.) The font size is defined in 1/8th point units. This gives a lot of control of the size. Because the EPIC 3000 printer prints discreet dots that are relatively large (from a font point of view), changes in character size will appear to jump. That is small changes in size will not appear to alter the character and them an 1/8th point change will make a dramatic change. If you wish to change the font and/or the character size select a size in the middle of the step. Even though small changes don't appear to have an effect, small changes do occur in the font generation and a value in the middle works best. The X and Y Spot size effects how the characters are generated. The spot size is equivalent to selecting a paint brush. A small spot is a small paint brush and therefore requires a lot more brush strokes to form the character. As the dots printed by the printer are always the same size, defining a small spot will cause more dots to be used to form the character. Depending on how the font was defined, it may be advantageous to select a large brush size and in effect reduce the number of dots in the character. There are no real guidelines for selecting Spot size. You have to experiment with it until you get acceptable results with your font. This value can generally be set to nominal with most fonts. (Nominal is 100).

File system Support

The EPIC 3000 Printer supports a file system to support TransAct Technologies fonts and allow the user to load and link custom fonts.

The POSFile tool provides a Windows interface to the printer and will allow fonts and configuration files to be loaded into the printer. This tool can read and write the POR.INI file, however the TransAct supplied fonts can not be read or deleted from the printer.

TrueType[20] and Compressed Stroke Fonts[21] are supported by the EPIC 3000 Printer. User-defined TrueType fonts many be defined and loaded into the printer, however, once in the printer they can not be extracted. (This protects the copyrights on the font.)

[20] Some but not all features of Open Type fonts are supported. Open Type fonts are not recommended.

[21] Compressed Stroke fonts are supplied by MonoType Inc.

EPIC 3000 Extended Printer Control

The EPIC 3000 printer has a number of Extended Control commands that allow an application to better track and maintain the printer. These commands are in all emulations[22].

EPIC 3000 Internal Logs

The printer maintains a log of printer activity. This activity may be returned to the host with the [ESC]~ T command. This command returns a ~T followed by four binary bits that make up a 32 bit unsigned integer. The description of the command below describes the format in full.

The printer also contains a number of commands that will force the printer to perform specific functions to help maintain the printer or print information about the printer. The functions available are:

1) Print Current Configuration
2) Print current log totals

Note: Each of these commands follow the ESC~ or ESCy format. Other functions are performed by these basic commands. Do not attempt to use any undocumented version of these commands. The extended diagnostics commands may affect the print quality and performance of the printer. In some cases, the commands may degrade the performance of the print cartridge or mechanism.

[22] The Star emulation does not allow the use of [ESC] y commands for extended diagnostics. The [ESC] y commands are not available.

Function **Read and Return Totals**

ASCII [ESC] ~ T <n>
Hexadecimal 1BH 7EH 54H <n>
Decimal <27> <126> <84> <n>
Description This command returns the current statistics for parameter n. The value returned will be ~T<n> with the next 4 bytes being an unsigned integer. For example:

[ESC]~T<1> Request cover open count
Returns: ~T<1><0><0><1><100> or 256 + 100 or 356 cover opens

Values of n	Request:		
0	Not Used	16	Configuration Faults
1	Not Used	17	Cash Drawer Opens
2	Not Used	18	Auto Cutter Cycles
3	Cover Opens	19	Not Used
4	Paper Outs	20	Not Used
5	Line Feeds	21	Standby Cycles
6	Characters Printed	22	Power Up Cycles
7	Cash Drawer	23	Fault Resets
8	Not Used	24	Power On Time (Min.)
9	Init Requests	25	System Active Time (Min.)
10	Auto Cutter Faults	26	Error Vectors
11	Over Temperature	27	Flash File Faults
12	Auto Cutter Re-home	28	USB Watch Dog Count
13	Not Used	29	FAT Flash Erase Cycles
14	Not Used	30	Ext Flash Erases
15	Missed TOF (if equipped)	31	System Memory Faults

Function **Print Current Configuration and Totals**

ASCII [ESC] y <9> or [ESC] ~ <9>
Hexadecimal 1BH 79H 09H
Decimal <27> <121> <9>
Description This command forces the printer to print the current configuration. To function correctly it must be issued with the printer in the proper emulation mode. It is intended to be printed in the default Ithaca® configuration but will print in any configuration.

Note: This command must be preceded with an ESC y <8>.

Function **Print Current Totals**

ASCII [ESC] y <15>
Hexadecimal 1BH 79H 0FH
Decimal <27> <121> <15>
Description This command forces the printer to print the current totals log.

Note: This command must be preceded with an ESC y <8>.

Function **Print Current Print Setup Values**

ASCII [ESC] y <20> or [ESC] ~ <20>
Hexadecimal 1BH 79H 14H
Decimal <27> <121> <20>
Description This command forces the printer to print the current setup values.

Note: This command must be preceded with an ESC y <8>.

EPIC 3000 Green and Sleep Power Control

The EPIC 3000 Printer has a remote power control command that instructs it to enter sleep mode. When the command is issued, the printer enters sleep and consumes less power. Unlike pushing the Power button, remote sleep mode leaves the communications active. All commands except the exit power down command are ignored.

Green mode is only available in USB mode. In Green, USB Vbus is monitored and the printer enters a very low power state until Vbus is restored.

[ESC] y Remote Power Control

ASCII	[ESC] y <n>
Hexadecimal	1BH 79H <n>
Decimal	<27> <121> <n>
IPCL	&%YX17 or &%YX18
EPOS	[ESC] y <n>
Where n	17 Requests the printer to enter remote sleep mode. 18 Requests the printer to exit remote sleep mode

Description Inquiry (ENQ) commands are accepted and answered in remote power down mode. The printer reactivates if the power button is pressed or a wake up command is received.

Note 1: If power is lost after the power down command is issued, the printer remembers it is in power down mode but does not reactivate the communications link. The power button must be pushed to return the printer to full operation.

Note 2: This command is not available in STAR mode.

[ESC] ~S Set Sleep Wait

ASCII	[ESC] ~ S<n>
Hexadecimal	1BH 7EH 53H<n>
Decimal	<27> <126> <83><n>
IPCL	None
EPOS	None
Where n	Specifies the sleep timer in 10 second increments. (1 = 10 seconds, and so on, 0 = Disable sleep.

Description If no data is received after this period of time, the printer will enter a sleep mode where less power is consumed. Once in this mode, the printer will resume normal operation after a slight delay when data is received.

Note 1: At low temperatures, this delay will include a print head preheat delay while the print head is heated to a level that will allow normal operation.

Note 2: Sleep mode also removes the feed motor hold current. It is possible that the first line of print may be slightly altered if the paper is moved while in sleep mode.

[ESC] ~ G USB Green

ASCII [ESC] ~ G <n>
Hexadecimal 1BH 7EH 47H <n>
Decimal <27> <126><71><n>
IPCL None
EPOS None
Where n sets the Mode. 0 = Disable. Non zero, enables greem mode after n seconds.

Description The USB Vbus is monitored by the printer. This command sets up what the printer will do when Vbus is removed.

If this value is non zero, the printer will enter Green mode after n seconds.

Note 1: At low temperatures, this delay will include a print head preheat delay while the print head is heated to a level that will allow normal operation.
Note 2: Sleep mode also removes the feed motor hold current. It is possible that the first line of print may be slightly altered if the paper is moved while in sleep mode.

Communications

Protocol and Print Buffers

The following figure illustrates the communication flow from host computer to printer and from printer to cash drawer.

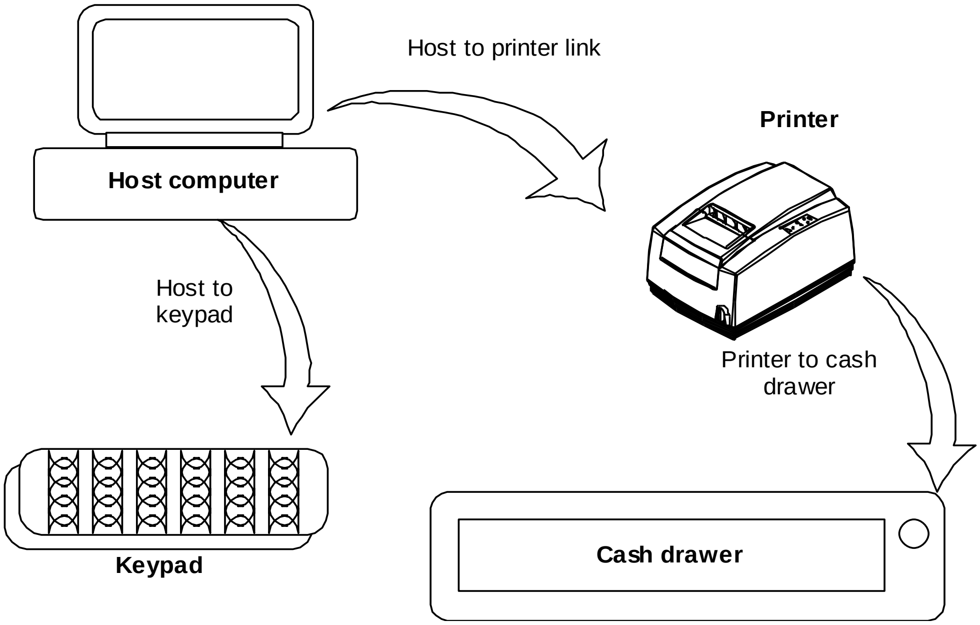


Figure 65 Typical POS System

For the host to printer communication link, the EPIC 3000 printer supports serial or parallel, USB and Ethernet communications. The serial, parallel, USB and Ethernet ports all follow standards developed for the personal computer environment.

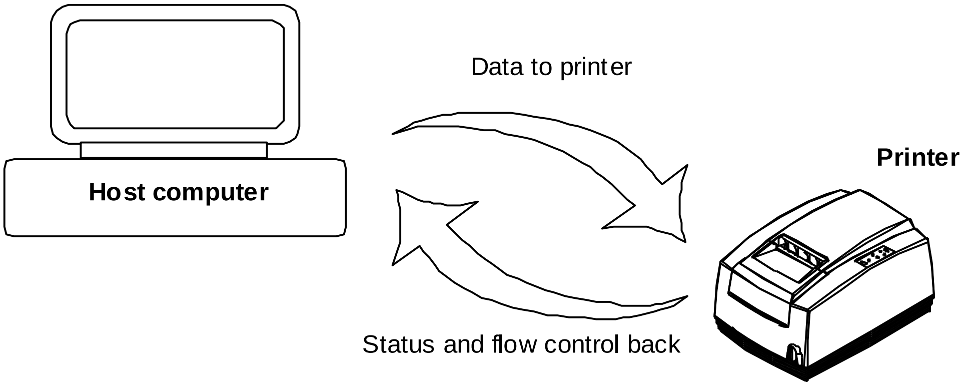


Figure 66 Host to Printer Link

In most cases, the host computer is capable of sending information to the printer much faster than the printer can print it. To prevent information from being lost, a flow control mechanism is provided. The mechanism is called the flow control protocol. The goal of the flow control protocol is to exchange as much information as possible as fast as possible without losing any data. The EPIC 3000 printer supports three flow control protocols, two in serial mode and one in parallel.

From the printer's point of view, four basic functions are required of communications. All four are common to all three flow control protocols. There must be a communication driver, status inquire procedure, storage buffer, and print control mechanism that is using the data.

The communication port is either the serial port hardware or the parallel port hardware and the associated communication software driver.

A means for the host to bypass the buffer for status information, referred to here as an inquirer processor, is also required because the buffer offsets the printer in time from the host. (The printer is generally behind the host).

The storage print buffer is a software implemented, first-in first-out (FIFO) circular buffer. It stores information in an asynchronous fashion where information can be placed in it at any rate and retrieved from it at any rate, but the information order is not altered. All buffers have a finite size, and if information is put in faster than it is removed, the buffer will overflow. To avoid overflow, a flow control mechanism is required.

The print control mechanism is the remainder of the printer hardware and control software. It interprets control codes and operates the control panel, print head, and cash drawer interface.

The following figure illustrates the four basic parts of printer flow control.

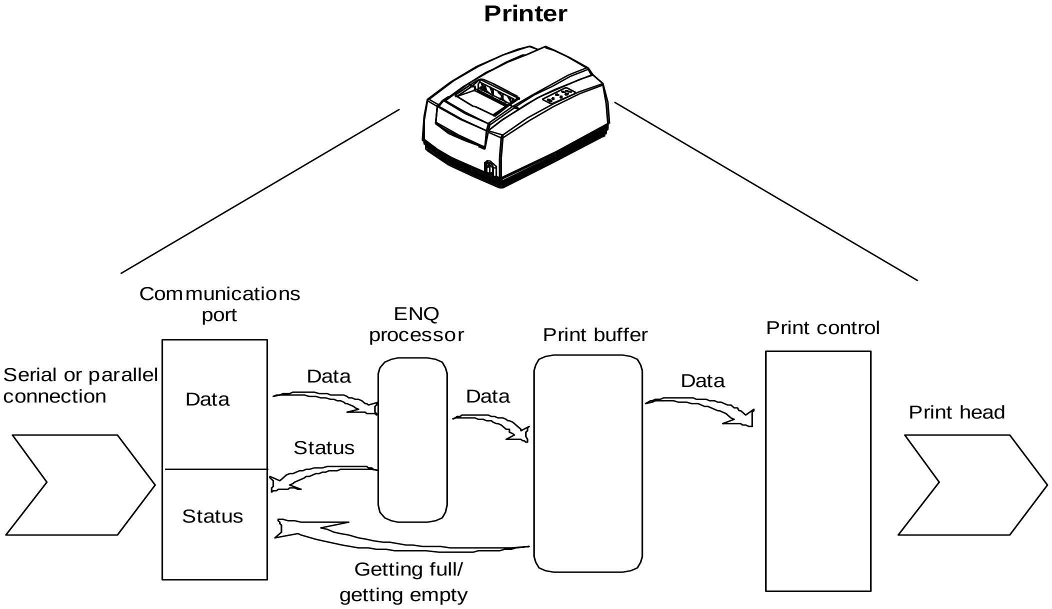


Figure 67 Printer Communications Buffer Flow

The communication port is either serial or parallel and is controlled by a software communication driver. The driver receives data and returns requested status. As information is received from the host, data is taken from the communication port hardware by the driver, preprocessed to look for status inquire commands, and placed in the buffer. When the buffer input function finds that the buffer is getting full, it notifies the communication driver to implement flow control. Flow control attempts to stop further information from being sent from the host.

The print control software takes information from the buffer, as it needs it and can use it. When the buffer output function finds that the buffer is getting low, it notifies the communication driver that the information flow can be resumed (if it was stopped) and allows more information to be placed in the buffer.

The EPIC 3000 printer has a configurable input buffer. The printer can be configured to allow from 40 to 8196 bytes of input buffer. Consequently, up to 8196 characters (or control codes) can be sent to the printer before they are interpreted and printed. In effect, the host computer can get 8196 characters ahead of the printer. In all cases, the buffer resumes communications when it is half empty. For example, if the buffer is configured to be 2048, the printer will signal stop when 2048 bytes are in the buffer; resume will be signaled when 1024 bytes remain. Inquire commands are preprocessed, which means they are found in the input data stream and acted upon as they are received. The status returned is valid as of the time the command is received. This is termed real-time status even though inquire commands are preprocessed and still placed in the buffer. Real-time status assures that data is not lost when the inquire sequence is part of another command. However, the buffer may also be filled by inquire commands if the printer is waiting for some activity.

USB

USB stands for Universal Serial Bus. A technical discussion of USB is beyond the scope of this document. If you would like more information about USB, visit the USB web site at www.usb.org. and http://www.usb.org/developers/devclass_docs

USB Support

The EPIC 3000 printer supports a standard USB Printer class interface. (See Appendix B). This interface standard is supported by most operating system including Windows and Linux. In Windows this interface is supported by the USB Printing Support Driver (usbprint.sys) and will present the printer as USB001-USB999. The ID is assigned by Windows and not by the printer. In Linux, the printer is also supported by a USB Printing Support driver, however the exact details can vary based on the Linux implementation. The printer will generally be presented as lp0 – lp99 again depending on the implementation.

The EPIC 3000 fully supports the USB Printer clas device including Read/Write and Status.

Transact has written a POSPrinter.OCX ActiveX that will allow you to easily interface to our printers. It is used by all of our demonstration programs. This OCX is available for use with customer applications. It works with printers installed on Serial COM ports, LPT ports, TCP/IP and USB. A USB compatible version is available from customer support or from our web site.

See: www.usb.org/developers/devclass_docs/usbprint11.pdf

Remote Printer Reset

It is possible to generate a software printer reset in serial mode. The [ENQ] <10> command requests that the printer reset. (This is not a hardware reset). The reset completely initializes the hardware and software, but the printer does not recover from a loss of software control.

Note: If the printer mode was changed by the [ESC] y <2> or <3> command, a soft reset will not return to the power on default.

The command flow is as follows:

- The [ENQ] <10> is acknowledged. During cleanup and initialization, the printer is placed off-line. Before the printer initializes, it tries to clean up its input buffer and other internal processes.
- The printer's software is reinitialized.
- The power-cycled flag is set.
- The print head is homed and re-calibrated.
- The printer goes back on-line.

Miscellaneous Communication Features

Power-cycle Recovery

Sometimes the host needs to know if the printer was power cycled. An example would be after the receipt tape was changed. It is not necessary to turn off the printer to change the receipt. However, if the operator does, any information sent to the printer before the power cycle will be lost.

The EPIC 3000 Printer has a flag that is set after a reset. The flag stays set until the host requests a reset. The [ENQ] <11> command reads the flag. If the command returns power-cycled status, the power has been reset or power cycled since the last request. All unprinted information has been lost.

If the print operation is critical, it is a good idea to check the power-cycle flag before and after all transactions. An alternate approach is to check the flag after every off-line to on-line transition.

Note: If the printer mode was changed by the [ESC] y <2> or <3> command, a power-cycle reset will return to the initial configuration.

Off-line Active

A configuration flag that prevents the printer from going off-line (in most cases) is available. Off-line mode allows the application to query the printer for status rather than assume a status from the control signals. The feature allows the host application to query the printer at all times except when there is no power; a full input buffer; or a hard failure. For example, when the printer's cover is open, the printer stops printing but still accepts data and inquiries. The inquire cover status command returns, "Cover open."

Hard failures result when there is no power or a printer fault occurs. If the printer is off-line, either the input buffer is full or a hard fault has occurred. The host application should not allow the input buffer to fill.

Recovery from Mechanical Errors

The Ithaca Inquire commands and the Epson [DLE][ENQ] and [DLE][EOT] commands allow most printer error status to be read and in some cases recovery attempted. Paper jams and auto-cutter faults can be recovered, however, any data not previously printed will be lost.

If the application is to support error recovery, the application should use the appropriate status request commands to query the printer periodically. If an error response is such that the fault is recoverable, the host application should interact with the operator and request that the fault be corrected. (IE. Clear Paper Jam) When the operator indicates that the problem has been corrected, the host can issue a reset request.

If a serious error occurs, the printer will halt and enter fault error mode. If the fault is such that the printer can maintain communications with the host, the print status request and response system will remain active. The status of the system will however remain static, i.e. the status responses will reflect the state of the system when the fault occurred.

There are errors that cannot be reported to the host system. These errors are such that the integrity of the printer do not allow continued operation. That is there is no way to report the error. These errors occur (for the most part) during power up diagnostics. They are as follows:

EEPROM READ ERROR **Power up only**

The Internal EEPROM is not readable or the check sum is bad.
Pressing the Power Button will attempt to rewrite the configuration information.
This may leave the printer configured incorrectly.

EEPROM WRITE ERROR **Power up/down and Configuration only**

The Internal EEPROM is defective. There is no recovery.

SOFTWARE ERROR VECTOR **Can occur at any time.**

These errors can occur during operation. They signal a serious problem with the system. In most cases this error will also generate a Watch Dog reset. A power cycle will generally recover normal printer operation. An ESD event or a firmware bug generally causes these errors. The printer maintains an error log, this log contains additional information about the fault and is printed during manual configuration. The information in this log should be reported to Transact to identify the exact cause of the fault.

USERSTORE FORMAT ERROR **Power up or User Store Write Operations**

The User Store data in FLASH has an invalid format. Pressing the Power Button will erase the user store and reformat it. This can be caused by a firmware update.

FLASH WRITE ERROR **Power up or User Store Write Operation.**

The Program Flash has failed. There is no recovery from this error.

COM ADAPTER ERROR **Power up only**

The communications interface card is not supported by the firmware or is missing. There is no recovery.

Programmer's Notes

The best time to configure the printer by the host with remote configuration is during system setup or software update.

Appendix A: CRC16

The CRC16 function used in the EPIC 3000 is as follows:

```
const static short crctab[] =/* CRC lookup table */
{
 0x0000, 0xC0C1, 0xC181, 0x0140, 0xC301, 0x03C0, 0x0280, 0xC241,
 0xC601, 0x06C0, 0x0780, 0xC741, 0x0500, 0xC5C1, 0xC481, 0x0440,
 0xCC01, 0x0CC0, 0x0D80, 0xCD41, 0x0F00, 0xCFC1, 0xCE81, 0x0E40,
 0x0A00, 0xCAC1, 0xCB81, 0x0B40, 0xC901, 0x09C0, 0x0880, 0xC841,
 0xD801, 0x18C0, 0x1980, 0xD941, 0x1B00, 0xDBC1, 0xDA81, 0x1A40,
 0x1E00, 0xDEC1, 0xDF81, 0x1F40, 0xDD01, 0x1DC0, 0x1C80, 0xDC41,
 0x1400, 0xD4C1, 0xD581, 0x1540, 0xD701, 0x17C0, 0x1680, 0xD641,
 0xD201, 0x12C0, 0x1380, 0xD341, 0x1100, 0xD1C1, 0xD081, 0x1040,
 0xF001, 0x30C0, 0x3180, 0xF141, 0x3300, 0xF3C1, 0xF281, 0x3240,
 0x3600, 0xF6C1, 0xF781, 0x3740, 0xF501, 0x35C0, 0x3480, 0xF441,
 0x3C00, 0xFCC1, 0xFD81, 0x3D40, 0xFF01, 0x3FC0, 0x3E80, 0xFE41,
 0xFA01, 0x3AC0, 0x3B80, 0xFB41, 0x3900, 0xF9C1, 0xF881, 0x3840,
 0x2800, 0xE8C1, 0xE981, 0x2940, 0xEB01, 0x2BC0, 0x2A80, 0xEA41,
 0xEE01, 0x2EC0, 0x2F80, 0xEF41, 0x2D00, 0xEDC1, 0xEC81, 0x2C40,
 0xE401, 0x24C0, 0x2580, 0xE541, 0x2700, 0xE7C1, 0xE681, 0x2640,
 0x2200, 0xE2C1, 0xE381, 0x2340, 0xE101, 0x21C0, 0x2080, 0xE041,
 0xA001, 0x60C0, 0x6180, 0xA141, 0x6300, 0xA3C1, 0xA281, 0x6240,
 0x6600, 0xA6C1, 0xA781, 0x6740, 0xA501, 0x65C0, 0x6480, 0xA441,
 0x6C00, 0xACC1, 0xAD81, 0x6D40, 0xAF01, 0x6FC0, 0x6E80, 0xAE41,
 0xAA01, 0x6AC0, 0x6B80, 0xAB41, 0x6900, 0xA9C1, 0xA881, 0x6840,
 0x7800, 0xB8C1, 0xB981, 0x7940, 0xBB01, 0x7BC0, 0x7A80, 0xBA41,
 0xBE01, 0x7EC0, 0x7F80, 0xBF41, 0x7D00, 0xBDC1, 0xBC81, 0x7C40,
 0xB401, 0x74C0, 0x7580, 0xB541, 0x7700, 0xB7C1, 0xB681, 0x7640,
 0x7200, 0xB2C1, 0xB381, 0x7340, 0xB101, 0x71C0, 0x7080, 0xB041,
 0x5000, 0x90C1, 0x9181, 0x5140, 0x9301, 0x53C0, 0x5280, 0x9241,
 0x9601, 0x56C0, 0x5780, 0x9741, 0x5500, 0x95C1, 0x9481, 0x5440,
 0x9C01, 0x5CC0, 0x5D80, 0x9D41, 0x5F00, 0x9FC1, 0x9E81, 0x5E40,
 0x5A00, 0x9AC1, 0x9B81, 0x5B40, 0x9901, 0x59C0, 0x5880, 0x9841,
 0x8801, 0x48C0, 0x4980, 0x8941, 0x4B00, 0x8BC1, 0x8A81, 0x4A40,
 0x4E00, 0x8EC1, 0x8F81, 0x4F40, 0x8D01, 0x4DC0, 0x4C80, 0x8C41,
 0x4400, 0x84C1, 0x8581, 0x4540, 0x8701, 0x47C0, 0x4680, 0x8641,
 0x8201, 0x42C0, 0x4380, 0x8341, 0x4100, 0x81C1, 0x8081, 0x4040
};
//
// Update a CRC check on the given buffer.
```

// $x^{16} + x^{15} + x^2 + 1$

```
unsigned short crcbuf(unsigned short crc, int len, unsigned char* buf)
{
int i;
      for (i=0; i<len; i++)
            crc = ((crc >> 8) & 0xff) ^ crctab[(crc ^ *buf++) & 0xff];

      return (crc);
}
```

Appendix B: USB Printer Class Specification

The USB interface conforms to USB Printer class standard published by the USB standards organization Version 1.1 dated January 2000.

The EPIC 3000 Printer will appear to the host system as a USB class 7 device.

There are 3 class specific requests that may be made to the printer after enumeration.

Label	bmRequest Type	bRequest	wValue	wIndex	wLength	Data
GET_DEVICE_ID	10100001B	0	Config Index	Interface & Alternate Setting	Length Maximum	1284 Device ID String
GET_PORT_STATUS	10100001B	1	Zero	Interface	1	BYTE
SOFT_RESET	00100001B	2	Zero	Interface	Zero	[None]

GET_DEVICE_ID (bRequest = 0)

This class-specific request returns a device ID string that is compatible with IEEE 1284. See IEEE 1284 for syntax and formatting information. The wValue field is used to specify a zero-based configuration index. The high-byte of the wIndex field is used to specify the zero-based interface index. The low-byte of the wIndex field is used to specify the zero-based alternate setting. The device ID string is returned in the following format:

Offset	Type	Description
0...n-1	Data	IEEE 1284 device ID string (including length in the first two bytes in big endian format).

GET_PORT_STATUS (bRequest = 1)

This class-specific request returns the printer's current status, in a format which is compatible with the status register of a standard PC parallel port. The following table defines the data returned.

Bit(s)	Field	Description
7 .. 6	Reserved	Reserved for future use; device shall return these bits reset to zero.
5	Paper Empty	1 = Paper Empty, 0 = Paper Not Empty
4	Select	1 = Selected, 0 = Not Selected
3	Not Error	1 = No Error, 0 = Error
2 .. 0	Reserved	Reserved for future use; device shall return these bits reset to zero.

SOFT_RESET (bRequest = 2)

This class-specific request flushes all the USB buffers and resets the Bulk OUT and Bulk IN pipes to their default states. This request clears all USB stall conditions. This reset does NOT change the USB addressing or USB configuration or actually reset the printer.

Note: Version 1.0 of the USB printer class specification incorrectly stated that the bmReqestType for SOFT_RESET was 00100011B.

EPIC 3000 USB Interface

The EPIC 3000 USB interface supports all three interfaces. The Get Device ID interface is configurable and may be disabled to prevent the host operating system from attempting to automatically install a Print Driver[23].

Note: The Print Driver is not the USB printer driver. It is a higher level driver that may or may not be used in the host system.

The Soft Reset will not reset the operating firmware nor will it do a complete hardware reset. It only resets the USB bulk pipes.

[23] The Print Driver is not the same as the USB Printing Support Driver. The USB Printing Support driver supports the USB interface but does not provide print services. The Print Driver provides a printer specific API on top of the USB Driver.

Appendix C: Internal Code Pages

Code Page	Country Code/Language Set	Decimal <n_h> <n_l>	Hex <n_h> <n_l>
64	USA (Slashed 0)	0,64	0H,040H
65	USA (Unslashed 0)	0,65	0H,041H
66	British	0,66	0H,042H
67	German	0,67	0H,043H
68	French	0,68	0H,044H
69	Swedish I	0,69	0H,045H
70	Danish	0,70	0H,046H
71	Norwegian	0,71	0H,047H
72	Dutch	0,72	0H,048H
73	Italian	0,73	0H,049H
74	French Canadian	0,74	0H,04AH
75	Spanish	0,75	0H,04BH
76	Swedish II	0,76	0H,04CH
77	Swedish III	0,77	0H,04DH
78	Swedish IV	0,78	0H,04EH
79	Turkish	0,79	0H,04FH
80	Swiss I	0,80	0H,050H
81	Swiss II	0,81	0H,051H
437	USA	1,181	1H,0B5H
737	Greek	2,225	2H,0E1H
850	Multilingual	3,82	3H,052H
852	East Europe Latin II-852	3,84	3H,054H
855	Cyrillic I-855	3,87	3H,057H
857	Turkey 857	3,89	3H,059H
858	Multilingual Euro	3,90	3H,05AH
866	Cyrillic II-866	3,98	3H,062H
1004	IS08859	3,236	3H,0ECH
1250	Windows 1250 Central Europe	4,226	4H,0E2H
1251	Windows 1251 Cyrillic	4,227	4H,0E3H
1252	Windows 1252 Latin 1	4,228	4H,0E4H
1253	Windows 1253 Greek	4,229	4H,0E5H
1254	Windows 1254 Turkish	4,230	4H,0E6H
1257	Windows 1257 Baltic	4,233	4H,0E9H
28591	ISO8859-1 Latin 1	111,175	6FH,AFH
28592	ISO8859-2 Latin 2	111,176	6FH,B0H
28593	ISO8859-3 Latin 3	111,177	6FH,B1H
28594	ISO8859-4 Baltic 4	111,178	6FH,B2H
28595	ISO8859-5 Cyrillic	111,179	6FH,B3H
28597	ISO8859-7 Greek	111,181	6FH,B5H
28599	ISO8859-9 Turkish	111,183	6FH,B7H
28605	ISO8859-15 Latin 9	111,189	6FH,BDH

Note: The [ESC]! Select international character set command uses Code Pages 64-81 and represent code page maps. They are provided to support of legacy applications. They are not recommended for new applications.

Appendix D: ASCII Code Table

Hex	Decimal	ASCII	Hex	Decimal	ASCII	Hex	Decimal	ASCII	Hex	Decimal	ASCII
00	0	NULL	20	32	(SP)	40	64	@	60	96	`
01	1	SOH	21	33	!	41	65	A	61	97	a
02	2	STX	22	34	"	42	66	B	62	98	b
03	3	ETX	23	35	#	43	67	C	63	99	c
04	4	EOT	24	36	$	44	68	D	64	100	d
05	5	ENQ	25	37	%	45	69	E	65	101	e
06	6	ACK	26	38	&	46	70	F	66	102	f
07	7	BEL	27	39	'	47	71	G	67	103	g
08	8	BS	28	40	(	48	72	H	68	104	h
09	9	HT	29	41	)	49	73	I	69	105	i
0A	10	LF	2A	42	*	4A	74	J	6A	106	j
0B	11	VT	2B	43	+	4B	75	K	6B	107	k
0C	12	FF	2C	44	,	4C	76	L	6C	108	l
0D	13	CR	2D	45	-	4D	77	M	6D	109	m
0E	14	SO	2E	46	.	4E	78	N	6E	110	n
0F	15	SI	2F	47	/	4F	79	O	6F	111	o
10	16	DLE	30	48	0	50	80	P	70	112	p
11	17	DC1	31	49	1	51	81	Q	71	113	q
12	18	DC2	32	50	2	52	82	R	72	114	r
13	19	DC3	33	51	3	53	83	S	73	115	s
14	20	DC4	34	52	4	54	84	T	74	116	t
15	21	NAK	35	53	5	55	85	U	75	117	u
16	22	SYN	36	54	6	56	86	V	76	118	v
17	23	ETB	37	55	7	57	87	W	77	119	w
18	24	CAN	38	56	8	58	88	X	78	120	x
19	25	EM	39	57	9	59	89	Y	79	121	y
1A	26	SUB	3A	58	:	5A	90	Z	7A	122	z
1B	27	ESC	3B	59	;	5B	91	[	7B	123	{
1C	28	FS	3C	60	<	5C	92	\	7C	124	|
1D	29	GS	3D	61	=	5D	93	]	7D	125	}
1E	30	RS	3E	62	>	5E	94	^	7E	126	~
1F	31	US	3F	63	?	5F	95	_	7F	127	(sp)

Appendix E: Unicode Character Addresses

Note: This information is based on the Unicode 3.0 Standard. For specific character locations see the Unicode standard.
Note: The EPIC 3000 does not contain all possible Unicode characters. The default character sets are defined by the WGL4 and GB18030 standards

Unicode Range		Use
0x0000	0x007F	C0 Controls and Basic Latin
0x0000	0x001F	C0 controls
0x0020	0x007F	ASCII
0x0080	0x00FF	C1 Controls and Latin-1 Supplement
0x00A0	0x00FF	Latin1
0x0100	0x017F	Latin Extended-A
0x0180	0x024F	Latin Extended-B
0x0250	0x02AF	IPA Extensions
0x02B0	0x02FF	Spacing Modifier Letters
0x0300	0x036F	Combining Diacritical Marks
0x0370	0x03FF	Greek
0x0400	0x04FF	Cyrillic
0x0500	0x052F	Unassigned zone 0500-052F
0x0530	0x058F	Armenian
0x0590	0x05FF	Hebrew
0x0600	0x06FF	Arabic
0x0700	0x08FF	Unassigned 0700-08FF
0x0900	0x097F	Devanagari. Based on ISCII 1988
0x0980	0x09FF	Bengali. Based on ISCII 1988
0x0A00	0x0A7F	Gurmukhi. Based on ISCII 1988
0x0A80	0x0AFF	Gujarati. Based on ISCII 1988
0x0B00	0x0B7F	Oriya. Based on ISCII 1988
0x0B80	0x0BFF	Tamil. Based on ISCII 1988
0x0C00	0x0C7F	Telugu. Based on ISCII 1988
0x0C80	0x0CFF	Kannada. Based on ISCII 1988
0x0D00	0x0D7F	Malayalam. Based on ISCII 1988
0x0D80	0x0DFF	Unassigned zone 0D80-0DFF
0x0D80	0x0DFF	Sinhala (Pre-Unicode 2.0)
0x0E00	0x0E7F	Thai. Based on TIS 620-2529
0x0E80	0x0EFF	Lao. Based on TIS 620-2529
0x0F00	0x0F7F	Burmese (Pre-Unicode 2.0)
0x0F00	0x0FBF	Tibetan
0x0F80	0x0FFF	Khmer (Pre-Unicode 2.0)
0x1000	0x105F	Tibetan (Pre-Unicode 2.0)
0x1060	0x109F	Mongolian (Pre-Unicode 2.0)
0x10A0	0x10FF	Georgian
0x1100	0x11FF	Hangul Jamo
0x1100	0x11F9	Korean combining alphabet
0x1200	0x137F	Ethiopian (Post-Unicode 2.0)
0x13A0	0x13FF	Cherokee (Post-Unicode 2.0)
0x1400	0x167F	Canadian Syllabics (Post-Unicode 2.0)
0x16A0	0x1DFF	Unassigned zone 16A0-1DFF
0x1E00	0x1EFF	Latin Extended Additional
0x1F00	0x1FFF	Greek Extended
0x2000	0x206F	General Punctuation

0x2070	0x209F	Superscripts and Subscripts
0x20A0	0x20CF	Currency Symbols
0x20D0	0x20FF	Combining Diacritical Marks for Symbols
0x2100	0x214F	Letter like Symbols
0x2150	0x218F	Number Forms
0x2190	0x21FF	Arrows
0x2200	0x22FF	Mathematical Operators
0x2300	0x23FF	Miscellaneous Technical
0x2400	0x243F	Control Pictures
0x2440	0x245F	Optical Character Recognition
0x2460	0x24FF	Enclosed Alphanumerics
0x2500	0x257F	Box Drawing
0x2580	0x259F	Block Elements
0x25A0	0x25FF	Geometric Shapes
0x2600	0x26FF	Miscellaneous Symbols
0x2700	0x27BF	Dingbats
0x27C0	0x27FF	Unassigned zone 27C0-27FF
0x2800	0x28FF	Braille Pattern Symbols (Post-Unicode 2.0)
0x2900	0x2FFF	Unassigned zone 2900-2FFF
0x3000	0x303F	CJK Symbols and Punctuation
0x3040	0x309F	Hiragana
0x30A0	0x30FF	Katakana
0x3100	0x312F	Bopomofo
0x3130	0x318F	Hangul Compatibility Jamo. Based on KSC 5601
0x3190	0x319F	Kanbun
0x31A0	0x31FF	Unassigned zone 31A0-31FF
0x3200	0x32FF	Enclosed CJK Letters and Months
0x3300	0x33FF	CJK Compatibility
0x3400	0x4DFF	CJK Unified Ideograph Extension A (Post-Unicode 2.0)
0x4E00	0x9FA5	CJK Unified Ideographs
0x9FA6	0xABFF	Unassigned zone 9FA6-ABFF
0xAC00	0xD7A3	Hangul Syllables
0xD7A4	0xD7FF	Unassigned zone D7A4-D7FF
0xD800	0xDB7F	High Surrogates
0xDB80	0xDBFF	Private Use High Surrogates
0xDC00	0xDFFF	Low Surrogates
0xE000	0xF8FF	Private Use Area
0xF900	0xFAFF	CJK Compatibility Ideographs
0xFB00	0xFB4F	Alphabetic Presentation Forms
0xFB50	0xFDFF	Arabic Presentation Forms-A
0xFE20	0xFE2F	Combining Half Marks
0xFE30	0xFE4F	CJK Compatibility Forms
0xFE50	0xFE6F	Small Form Variants
0xFE70	0xFEFF	Arabic Presentation Forms-B
0xFEFF	0xFEFF	Special
0xFF00	0xFFEF	Half width and Full width Forms
0xFFF0	0xFFFF	Specials
0xFFF0	0xFFFD	Specials
0xFFFE	0xFFFF	Not character codes

Appendix F: WGL4.0 Character Addresses

There are 654 Characters in this set and does not use address greater than 0xFFFF

Unicode	Character
0020	space
0021	exclamation mark
0022	quotation mark
0023	number sign
0024	dollar sign
0025	percent sign
0026	ampersand
0027	apostrophe
0028	left parenthesis
0029	right parenthesis
002a	asterisk
002b	plus sign
002c	comma
002d	hyphen-minus
002e	period
002f	slash
0030	digit zero
0031	digit one
0032	digit two
0033	digit three
0034	digit four
0035	digit five
0036	digit six
0037	digit seven
0038	digit eight
0039	digit nine
003a	colon
003b	semicolon
003c	less-than sign
003d	equals sign
003e	greater-than sign
003f	question mark
0040	commercial at
0041	Latin capital letter a
0042	Latin capital letter b
0043	Latin capital letter c
0044	Latin capital letter d
0045	Latin capital letter e
0046	Latin capital letter f
0047	Latin capital letter g
0048	Latin capital letter h
0049	Latin capital letter i
004a	Latin capital letter j
004b	Latin capital letter k
004c	Latin capital letter l
004d	Latin capital letter m
004e	Latin capital letter n
004f	Latin capital letter o
0050	Latin capital letter p
0051	Latin capital letter q
0052	Latin capital letter r
0053	Latin capital letter s
0054	Latin capital letter t
0055	Latin capital letter u
0056	Latin capital letter v
0057	Latin capital letter w
0058	Latin capital letter x
0059	Latin capital letter y
005a	Latin capital letter z
005b	left square bracket
005c	backslash
005d	right square bracket
005e	circumflex accent
005f	underline
0060	grave accent
0061	Latin small letter a
0062	Latin small letter b
0063	Latin small letter c
0064	Latin small letter d
0065	Latin small letter e
0066	Latin small letter f
0067	Latin small letter g
0068	Latin small letter h
0069	Latin small letter i
006a	Latin small letter j
006b	Latin small letter k
006c	Latin small letter l
006d	Latin small letter m
006e	Latin small letter n
006f	Latin small letter o
0070	Latin small letter p
0071	Latin small letter q
0072	Latin small letter r
0073	Latin small letter s
0074	Latin small letter t
0075	Latin small letter u
0076	Latin small letter v
0077	Latin small letter w
0078	Latin small letter x
0079	Latin small letter y
007a	Latin small letter z
007b	left curly bracket
007c	vertical line
007d	right curly bracket
007e	tilde
00a0	no-break space
00a1	inverted exclamation mark
00a2	cent sign
00a3	pound sign
00a4	currency sign
00a5	yen sign
00a6	broken bar
00a7	section sign
00a8	diaeresis
00a9	copyright sign
00aa	feminine ordinal indicator
00ab	left guillemet
00ac	not sign
00ad	soft hyphen
00ae	registered trade mark sign
00af	macron, overline
00b0	degree sign
00b1	plus-minus sign
00b2	superscript two
00b3	superscript three
00b4	acute accent
00b5	micro sign
00b6	paragraph sign
00b7	middle dot, kana conjoctive
00b8	cedilla
00b9	superscript one
00ba	masculine ordinal indicator
00bb	right guillemet
00bc	vulgar fraction one quarter
00bd	vulgar fraction one half
00be	vulgar fraction three quarters
00bf	inverted question mark
00c0	Latin capital letter a with grave accent
00c1	Latin capital letter a with acute accent
00c2	Latin capital letter a with circumflex accent
00c3	Latin capital letter a with tilde
00c4	Latin capital letter a with diaeresis
00c5	Latin capital letter a with ring above
00c6	Latin capital letter a with e
00c7	Latin capital letter c with cedilla
00c8	Latin capital letter e with grave accent
00c9	Latin capital letter e with acute accent
00ca	Latin capital letter e with circumflex accent
00cb	Latin capital letter e with diaeresis
00cc	Latin capital letter i with grave accent
00cd	Latin capital letter i with acute accent
00ce	Latin capital letter i with circumflex accent
00cf	Latin capital letter i with diaeresis
00d0	Latin capital letter eth
00d1	Latin capital letter n with tilde
00d2	Latin capital letter o with grave accent
00d3	Latin capital letter o with acute accent
00d4	Latin capital letter o with circumflex accent
00d5	Latin capital letter o with tilde
00d6	Latin capital letter o with diaeresis
00d7	multiplication sign
00d8	Latin capital letter o with oblique stroke
00d9	Latin capital letter u with grave accent
00da	Latin capital letter u with acute accent
00db	Latin capital letter u with circumflex accent
00dc	Latin capital letter u with diaeresis
00dd	Latin capital letter y with acute accent
00de	Latin capital letter thorn
00df	Latin small letter sharp s
00e0	Latin small letter a with grave accent
00e1	Latin small letter a with acute accent

00e2	Latin small letter a with circumflex accent
00e3	Latin small letter a with tilde
00e4	Latin small letter a with diaeresis
00e5	Latin small letter a with ring above
00e6	Latin small letter a with e
00e7	Latin small letter c with cedilla
00e8	Latin small letter e with grave accent
00e9	Latin small letter e with acute accent
00ea	Latin small letter e with circumflex accent
00eb	Latin small letter e with diaeresis
00ec	Latin small letter i with grave accent
00ed	Latin small letter i with acute accent
00ee	Latin small letter i with circumflex accent
00ef	Latin small letter i with diaeresis
00f0	Latin small letter eth
00f1	Latin small letter n with tilde
00f2	Latin small letter o with grave accent
00f3	Latin small letter o with acute accent
00f4	Latin small letter o with circumflex accent
00f5	Latin small letter o with tilde
00f6	Latin small letter o with diaeresis
00f7	division sign
00f8	Latin small letter o with oblique stroke
00f9	Latin small letter u with grave accent
00fa	Latin small letter u with acute accent
00fb	Latin small letter u with circumflex accent
00fc	Latin small letter u with diaeresis
00fd	Latin small letter y with acute accent
00fe	Latin small letter thorn
00ff	Latin small letter y with diaeresis
0100	Latin capital letter a with macron
0101	Latin small letter a with macron
0102	Latin capital letter a with breve
0103	Latin small letter a with breve
0104	Latin capital letter a with ogonek
0105	Latin small letter a with ogonek
0106	Latin capital letter c with acute accent
0107	Latin small letter c with acute accent
0108	Latin capital letter c with circumflex
0109	Latin small letter c with circumflex
010a	Latin capital letter c with dot above
010b	Latin small letter c with dot above
010c	Latin capital letter c with caron
010d	Latin small letter c with caron
010e	Latin capital letter d with hacek
010f	Latin small letter d with hacek
0110	Latin capital letter d with stroke
0111	Latin small letter d with stroke
0112	Latin capital letter e with macron
0113	Latin small letter e with macron
0114	Latin capital letter e with breve
0115	Latin small letter e with breve
0116	Latin capital letter e with dot above
0117	Latin small letter e with dot above
0118	Latin capital letter e with ogenek
0119	Latin small letter e with ogenek
011a	Latin capital letter e with hacek
011b	Latin small letter e with hacek
011c	Latin capital letter g with circumflex
011d	Latin small letter g with circumflex
011e	Latin capital letter g with breve
011f	Latin small letter g with breve
0120	Latin capital letter g with dot above
0121	Latin small letter g with dot above
0122	Latin capital letter g with cedilla
0123	Latin small letter g with cedilla
0124	Latin capital letter h with circumflex
0125	Latin small letter h with circumflex
0126	Latin capital letter h with stroke
0127	Latin small letter h with stroke
0128	Latin capital letter i with tilde
0129	Latin small letter i with tilde
012a	Latin capital letter i with macron
012b	Latin small letter i with macron
012c	Latin capital letter i with breve
012d	Latin small letter i with breve
012e	Latin capital letter i with ogonek
012f	Latin small letter i with ogonek
0130	Latin capital letter i with dot above
0131	Latin small letter i without dot above
0132	Latin capital ligature ij
0133	Latin small ligature ij
0134	Latin capital letter j with circumflex
0135	Latin small letter j with circumflex
0136	Latin capital letter k with cedilla
0137	Latin small letter k with cedilla
0138	Latin small letter kra
0139	Latin capital letter l with acute accent
013a	Latin small letter l with acute accent
013b	Latin capital letter l with cedilla
013c	Latin small letter l with cedilla
013d	Latin capital letter l with hacek
013e	Latin small letter l with hacek
013f	Latin capital letter l with middle dot
0140	Latin small letter l with middle dot
0141	Latin capital letter l with stroke
0142	Latin small letter l with stroke
0143	Latin capital letter n with acute accent
0144	Latin small letter n with acute accent
0145	Latin capital letter n with cedilla
0146	Latin small letter n with cedilla
0147	Latin capital letter n with hacek
0148	Latin small letter n with hacek
0149	Latin small letter n preceded by apostrophe
014a	Latin capital letter eng
014b	Latin small letter eng
014c	Latin capital letter o with macron
014d	Latin small letter o with macron
014e	Latin capital letter o with breve
014f	Latin small letter o with breve
0150	Latin capital letter o with double acute accent
0151	Latin small letter o with double acute accent
0152	Latin capital ligature o with e
0153	Latin small ligature o with e
0154	Latin capital letter r with acute accent
0155	Latin small letter r with acute accent
0156	Latin capital letter r with cedilla
0157	Latin small letter r with cedilla
0158	Latin capital letter r with hacek
0159	Latin small letter r with hacek
015a	Latin capital letter s with acute accent
015b	Latin small letter s with acute accent
015c	Latin capital letter s with circumflex
015d	Latin small letter s with circumflex
015e	Latin capital letter s with cedilla
015f	Latin small letter s with cedilla
0160	Latin capital letter s with hacek
0161	Latin small letter s with hacek
0162	Latin capital letter t with cedilla
0163	Latin small letter t with cedilla
0164	Latin capital letter t with hacek
0165	Latin small letter t with hacek
0166	Latin capital letter t with stroke
0167	Latin small letter t with stroke
0168	Latin capital letter u with tilde
0169	Latin small letter u with tilde
016a	Latin capital letter u with macron
016b	Latin small letter u with macron
016c	Latin capital letter u with breve
016d	Latin small letter u with breve
016e	Latin capital letter u with ring above
016f	Latin small letter u with ring above
0170	Latin capital letter u with double acute accent
0171	Latin small letter u with double acute accent
0172	Latin capital letter u with ogonek
0173	Latin small letter u with ogonek
0174	Latin capital letter w with circumflex
0175	Latin cmall letter w with circumflex
0176	Latin capital letter y with circumflex
0177	Latin small letter y with circumflex
0178	Latin capital letter y with diaeresis
0179	Latin capital letter z with acute accent
017a	Latin small letter z with acute accent
017b	Latin capital letter z with dot above
017c	Latin small letter z with dot above
017d	Latin capital letter z with hacek
017e	Latin small letter z with hacek
017f	Latin small letter long s
0192	Latin small letter script f,florin sign
01fa	Latin capital letter a with ring above and acute
01fb	Latin small letter a with ring above and acute
01fc	Latin capital ligature ae with acute
01fd	Latin small ligature ae with acute
01fe	Latin capital letter o with stroke and acute
01ff	Latin small letter o with stroke and acute
02c6	nonspacing circumflex accent
02c7	modifier letter hacek
02c9	modifier letter macron
02d8	breve
02d9	dot above
02da	ring above
02db	ogonek
02dc	nonspacing tilde
02dd	modifier letter double prime

0384	Greek tonos
0385	Greek dialytika tonos
0386	Greek capital letter alpha with tonos
0387	Greek ano teleia
0388	Greek capital letter epsilon with tonos
0389	Greek capital letter eta with tonos
038a	Greek capital letter iota with tonos
038c	Greek capital letter omicron with tonos
038e	Greek capital letter upsilon with tonos
038f	Greek capital letter omega with tonos
0390	Greek small letter iota with dialytika and tonos
0391	Greek capital letter alpha
0392	Greek capital letter beta
0393	Greek capital letter gamma
0394	Greek capital letter delta
0395	Greek capital letter epsilon
0396	Greek capital letter zeta
0397	Greek capital letter eta
0398	Greek capital letter theta
0399	Greek capital letter iota
039a	Greek capital letter kappa
039b	Greek capital letter lamda
039c	Greek capital letter mu
039d	Greek capital letter nu
039e	Greek capital letter xi
039f	Greek capital letter omicron
03a0	Greek capital letter pi
03a1	Greek capital letter rho
03a3	Greek capital letter sigma
03a4	Greek capital letter tau
03a5	Greek capital letter upsilon
03a6	Greek capital letter phi
03a7	Greek capital letter chi
03a8	Greek capital letter psi
03a9	Greek capital letter omega
03aa	Greek capital letter iota with dialytika
03ab	Greek capital letter upsilon with dialytika
03ac	Greek small letter alpha with tonos
03ad	Greek small letter epsilon with tonos
03ae	Greek small letter eta with tonos
03af	Greek small letter iota with tonos
03b0	Greek small letter upsilon with dialytika and tonos
03b1	Greek small letter alpha
03b2	Greek small letter beta
03b3	Greek small letter gamma
03b4	Greek small letter delta
03b5	Greek small letter epsilon
03b6	Greek small letter zeta
03b7	Greek small letter eta
03b8	Greek small letter theta
03b9	Greek small letter iota
03ba	Greek small letter kappa
03bb	Greek small letter lamda
03bc	Greek small letter mu
03bd	Greek small letter nu
03be	Greek small letter xi
03bf	Greek small letter omicron
03c0	Greek small letter pi
03c1	Greek small letter rho
03c2	Greek small letter final sigma
03c3	Greek small letter sigma
03c4	Greek small letter tau
03c5	Greek small letter upsilon
03c6	Greek small letter phi
03c7	Greek small letter chi
03c8	Greek small letter psi
03c9	Greek small letter omega
03ca	Greek small letter iota with dialytika
03cb	Greek small letter upsilon with dialytika
03cc	Greek small letter omicron with tonos
03cd	Greek small letter upsilon with tonos
03ce	Greek small letter omega with tonos
0401	Cyrillic capital letter io
0402	Cyrillic capital letter dje
0403	Cyrillic capital letter gje
0404	Cyrillic capital letter ukrainian ie
0405	Cyrillic capital letter dze
0406	Cyrillic capital letter byelorussian-ukrainian i
0407	Cyrillic capital letter yi
0408	Cyrillic capital letter je
0409	Cyrillic capital letter lje
040a	Cyrillic capital letter nje
040b	Cyrillic capital letter tshe
040c	Cyrillic capital letter kje
040e	Cyrillic capital letter short u
040f	Cyrillic capital letter dzhe
0410	Cyrillic capital letter a
0411	Cyrillic capital letter be
0412	Cyrillic capital letter ve
0413	Cyrillic capital letter ghe
0414	Cyrillic capital letter de
0415	Cyrillic capital letter ie
0416	Cyrillic capital letter zhe
0417	Cyrillic capital letter ze
0418	Cyrillic capital letter i
0419	Cyrillic capital letter short i
041a	Cyrillic capital letter ka
041b	Cyrillic capital letter el
041c	Cyrillic capital letter em
041d	Cyrillic capital letter en
041e	Cyrillic capital letter o
041f	Cyrillic capital letter pe
0420	Cyrillic capital letter er
0421	Cyrillic capital letter es
0422	Cyrillic capital letter te
0423	Cyrillic capital letter u
0424	Cyrillic capital letter ef
0425	Cyrillic capital letter ha
0426	Cyrillic capital letter tse
0427	Cyrillic capital letter che
0428	Cyrillic capital letter sha
0429	Cyrillic capital letter shcha
042a	Cyrillic capital letter hard sign
042b	Cyrillic capital letter yeru
042c	Cyrillic capital letter soft sign
042d	Cyrillic capital letter e
042e	Cyrillic capital letter yu
042f	Cyrillic capital letter ya
0430	Cyrillic small letter a
0431	Cyrillic small letter be
0432	Cyrillic small letter ve
0433	Cyrillic small letter ghe
0434	Cyrillic small letter de
0435	Cyrillic small letter ie
0436	Cyrillic small letter zhe
0437	Cyrillic small letter ze
0438	Cyrillic small letter i
0439	Cyrillic small letter short i
043a	Cyrillic small letter ka
043b	Cyrillic small letter el
043c	Cyrillic small letter em
043d	Cyrillic small letter en
043e	Cyrillic small letter o
043f	Cyrillic small letter pe
0440	Cyrillic small letter er
0441	Cyrillic small letter es
0442	Cyrillic small letter te
0443	Cyrillic small letter u
0444	Cyrillic small letter ef
0445	Cyrillic small letter ha
0446	Cyrillic small letter tse
0447	Cyrillic small letter che
0448	Cyrillic small letter sha
0449	Cyrillic small letter shcha
044a	Cyrillic small letter hard sign
044b	Cyrillic small letter yeru
044c	Cyrillic small letter soft sign
044d	Cyrillic small letter e
044e	Cyrillic small letter yu
044f	Cyrillic small letter ya
0451	Cyrillic small letter io
0452	Cyrillic small letter dje
0453	Cyrillic small letter gje
0454	Cyrillic small letter ukrainian ie
0455	Cyrillic small letter dze
0456	Cyrillic small letter byelorussian-ukrainian i
0457	Cyrillic small letter yi
0458	Cyrillic small letter je
0459	Cyrillic small letter lje
045a	Cyrillic small letter nje
045b	Cyrillic small letter tshe
045c	Cyrillic small letter kje
045e	Cyrillic small letter short u
045f	Cyrillic small letter dzhe
0490	Cyrillic capital letter ghe with upturn
0491	Cyrillic small letter ghe with upturn
1e80	Latin capital letter w with grave
1e81	Latin small letter w with grave
1e82	Latin capital letter w with acute
1e83	Latin small letter w with acute
1e84	Latin capital letter w with diaeresis
1e85	Latin small letter w with diaeresis
1ef2	Latin capital letter y with grave
1ef3	Latin small letter y with grave

2013	en dash
2014	em dash
2015	horizontal bar
2017	double low line
2018	left single quotation mark
2019	right single quotation mark
201a	single low-9 quotation mark
201b	single high-reversed-9 quotation mark
201c	left double quotation mark
201d	right double quotation mark
201e	double low-9 quotation mark
2020	dagger
2021	double dagger
2022	bullet
2026	horizontal ellipsis
2030	per mille sign
2032	prime
2033	double prime
2039	single left-pointing angle quotation mark
203a	single right-pointing angle quotation mark
203c	double exclamation mark
203e	overline
2044	fraction slash
207f	superscript Latin small letter n
20a3	french franc sign
20a4	lira sign
20a7	peseta sign
20ac	euro currency symbol
2105	care of
2113	script small l
2116	numero sign
2122	trademark sign
2126	ohm sign
212e	estimated symbol
215b	vulgar fraction one eighth
215c	vulgar fraction three eighths
215d	vulgar fraction five eighths
215e	vulgar fraction seven eighths
2190	leftwards arrow
2191	upwards arrow
2192	rightwards arrow
2193	downwards arrow
2194	left right arrow
2195	up down arrow
21a8	up down arrow with base
2202	partial differential
2206	increment
220f	n-ary product
2211	n-ary summation
2212	minus sign
2215	division slash
2219	bullet operator
221a	square root
221e	infinity
221f	right angle
2229	intersection
222b	integral
2248	almost equal to
2260	not equal to
2261	identical to
2264	less-than or equal to
2265	greater-than or equal to
2302	house
2310	reversed not sign
2320	top half integral
2321	bottom half integral
2500	box drawings light horizontal
2502	box drawings light vertical
250c	box drawings light down and right
2510	box drawings light down and left
2514	box drawings light up and right
2518	box drawings light up and left
251c	box drawings light vertical and right
2524	box drawings light vertical and left
252c	box drawings light down and horizontal
2534	box drawings light up and horizontal
253c	box drawings light vertical and horizontal
2550	box drawings double horizontal
2551	box drawings double vertical
2552	box drawings down single and right double
2553	box drawings down double and right single
2554	box drawings double down and right
2555	box drawings down single and left double
2556	box drawings down double and left single
2557	box drawings double down and left
2558	box drawings up single and right double
2559	box drawings up double and right single
255a	box drawings double up and right
255b	box drawings up single and left double
255c	box drawings up double and left single
255d	box drawings double up and left
255e	box drawings vertical single and right double
255f	box drawings vertical double and right single
2560	box drawings double vertical and right
2561	box drawings vertical single and left double
2562	box drawings vertical double and left single
2563	box drawings double vertical and left
2564	box drawings down single and horizontal double
2565	box drawings down double and horizontal single
2566	box drawings double down and horizontal
2567	box drawings up single and horizontal double
2568	box drawings up double and horizontal single
2569	box drawings double up and horizontal
256a	box drawings vertical single and horizontal double
256b	box drawings vertical double and horizontal single
256c	box drawings double vertical and horizontal
2580	upper half block
2584	lower half block
2588	full block
258c	left half block
2590	right half block
2591	light shade
2592	medium shade
2593	dark shade
25a0	black square
25a1	white square
25aa	black small square
25ab	white small square
25ac	black rectangle
25b2	black up-pointing triangle
25ba	black right-pointing pointer
25bc	black down-pointing triangle
25c4	black left-pointing pointer
25ca	lozenge
25cb	white circle
25cf	black circle
25d8	inverse bullet
25d9	inverse white circle
25e6	white bullet
263a	white smiling face
263b	black smiling face
263c	white sun with rays
2640	female sign
2642	male sign
2660	black spade suit
2663	black club suit
2665	black heart suit
2666	black diamond suit
266a	eighth note
266b	Beamed eighth notes
f001	fi ligature
f002	fl ligature
fb01	Fi ligature
fb02	Fl ligature

Note: This information is based on the Microsoft's Typography web page

Appendix G: GB18030 Character Addresses

There are 28575 Characters in this set and does not use address greater than 0xFFFF

UNICODE	CHARACTER
0020	SPACE
0021	EXCLAMATION MARK
0022	QUOTATION MARK
0023	NUMBER SIGN
0024	DOLLAR SIGN
0025	PERCENT SIGN
0026	AMPERSAND
0027	APOSTROPHE
0028	LEFT PARENTHESIS
0029	RIGHT PARENTHESIS
002A	ASTERISK
002B	PLUS SIGN
002C	COMMA
002D	HYPHEN-MINUS
002E	FULL STOP
002F	SOLIDUS
0030	DIGIT ZERO
0031	DIGIT ONE
0032	DIGIT TWO
0033	DIGIT THREE
0034	DIGIT FOUR
0035	DIGIT FIVE
0036	DIGIT SIX
0037	DIGIT SEVEN
0038	DIGIT EIGHT
0039	DIGIT NINE
003A	COLON
003B	SEMICOLON
003C	LESS-THAN SIGN
003D	EQUALS SIGN
003E	GREATER-THAN SIGN
003F	QUESTION MARK
0040	COMMERCIAL AT
0041	LATIN CAPITAL LETTER A
0042	LATIN CAPITAL LETTER B
0043	LATIN CAPITAL LETTER C
0044	LATIN CAPITAL LETTER D
0045	LATIN CAPITAL LETTER E
0046	LATIN CAPITAL LETTER F
0047	LATIN CAPITAL LETTER G
0048	LATIN CAPITAL LETTER H
0049	LATIN CAPITAL LETTER I
004A	LATIN CAPITAL LETTER J
004B	LATIN CAPITAL LETTER K
004C	LATIN CAPITAL LETTER L
004D	LATIN CAPITAL LETTER M
004E	LATIN CAPITAL LETTER N
004F	LATIN CAPITAL LETTER O
0050	LATIN CAPITAL LETTER P
0051	LATIN CAPITAL LETTER Q
0052	LATIN CAPITAL LETTER R
0053	LATIN CAPITAL LETTER S
0054	LATIN CAPITAL LETTER T
0055	LATIN CAPITAL LETTER U
0056	LATIN CAPITAL LETTER V
0057	LATIN CAPITAL LETTER W
0058	LATIN CAPITAL LETTER X
0059	LATIN CAPITAL LETTER Y
005A	LATIN CAPITAL LETTER Z
005B	LEFT SQUARE BRACKET
005C	REVERSE SOLIDUS
005D	RIGHT SQUARE BRACKET
005E	CIRCUMFLEX ACCENT
005F	LOW LINE
0060	GRAVE ACCENT
0061	LATIN SMALL LETTER A
0062	LATIN SMALL LETTER B
0063	LATIN SMALL LETTER C
0064	LATIN SMALL LETTER D
0065	LATIN SMALL LETTER E
0066	LATIN SMALL LETTER F
0067	LATIN SMALL LETTER G
0068	LATIN SMALL LETTER H
0069	LATIN SMALL LETTER I
006A	LATIN SMALL LETTER J
006B	LATIN SMALL LETTER K
006C	LATIN SMALL LETTER L
006D	LATIN SMALL LETTER M
006E	LATIN SMALL LETTER N
006F	LATIN SMALL LETTER O
0070	LATIN SMALL LETTER P
0071	LATIN SMALL LETTER Q
0072	LATIN SMALL LETTER R
0073	LATIN SMALL LETTER S
0074	LATIN SMALL LETTER T
0075	LATIN SMALL LETTER U
0076	LATIN SMALL LETTER V
0077	LATIN SMALL LETTER W
0078	LATIN SMALL LETTER X
0079	LATIN SMALL LETTER Y
007A	LATIN SMALL LETTER Z
007B	LEFT CURLY BRACKET
007C	VERTICAL LINE
007D	RIGHT CURLY BRACKET
007E	TILDE
007F	<CONTROL>
00A4	CURRENCY SIGN
00A7	SECTION SIGN
00A8	DIAERESIS
00B0	DEGREE SIGN
00B1	PLUS-MINUS SIGN
00B7	MIDDLE DOT
00D7	MULTIPLICATION SIGN
00E0	LATIN SMALL LETTER A WITH GRAVE
00E1	LATIN SMALL LETTER A WITH ACUTE
00E8	LATIN SMALL LETTER E WITH GRAVE
00E9	LATIN SMALL LETTER E WITH ACUTE
00EA	LATIN SMALL LETTER E WITH CIRCUMFLEX
00EC	LATIN SMALL LETTER I WITH GRAVE
00ED	LATIN SMALL LETTER I WITH ACUTE
00F2	LATIN SMALL LETTER O WITH GRAVE
00F3	LATIN SMALL LETTER O WITH ACUTE
00F7	DIVISION SIGN
00F9	LATIN SMALL LETTER U WITH GRAVE
00FA	LATIN SMALL LETTER U WITH ACUTE
00FC	LATIN SMALL LETTER U WITH DIAERESIS
0101	LATIN SMALL LETTER A WITH MACRON
0113	LATIN SMALL LETTER E WITH MACRON
011B	LATIN SMALL LETTER E WITH CARON
012B	LATIN SMALL LETTER I WITH MACRON
0144	LATIN SMALL LETTER N WITH ACUTE
0148	LATIN SMALL LETTER N WITH CARON
014D	LATIN SMALL LETTER O WITH MACRON
016B	LATIN SMALL LETTER U WITH MACRON
01CE	LATIN SMALL LETTER A WITH CARON
01D0	LATIN SMALL LETTER I WITH CARON
01D2	LATIN SMALL LETTER O WITH CARON
01D4	LATIN SMALL LETTER U WITH CARON
01D6	LATIN SMALL LETTER U WITH DIAERESIS AND MACRON
01D8	LATIN SMALL LETTER U WITH DIAERESIS AND ACUTE
01DA	LATIN SMALL LETTER U WITH DIAERESIS AND CARON
01DC	LATIN SMALL LETTER U WITH DIAERESIS AND GRAVE
0251	LATIN SMALL LETTER ALPHA
0261	LATIN SMALL LETTER SCRIPT G
02C7	CARON (MANDARIN CHINESE THIRD TONE)
02C9	MODIFIER LETTER MACRON (MANDARIN CHINESE FIRST TONE)
02CA	MODIFIER LETTER ACUTE ACCENT (MANDARIN CHINESE SECOND TONE)
02CB	MODIFIER LETTER GRAVE ACCENT (MANDARIN CHINESE FOURTH TONE)
02D9	DOT ABOVE (MANDARIN CHINESE LIGHT TONE)
0391	GREEK CAPITAL LETTER ALPHA
0392	GREEK CAPITAL LETTER BETA
0393	GREEK CAPITAL LETTER GAMMA
0394	GREEK CAPITAL LETTER DELTA
0395	GREEK CAPITAL LETTER EPSILON
0396	GREEK CAPITAL LETTER ZETA
0397	GREEK CAPITAL LETTER ETA
0398	GREEK CAPITAL LETTER THETA
0399	GREEK CAPITAL LETTER IOTA
039A	GREEK CAPITAL LETTER KAPPA
039B	GREEK CAPITAL LETTER LAMDA
039C	GREEK CAPITAL LETTER MU
039D	GREEK CAPITAL LETTER NU
039E	GREEK CAPITAL LETTER XI
039F	GREEK CAPITAL LETTER OMICRON

03A0	GREEK CAPITAL LETTER PI
03A1	GREEK CAPITAL LETTER RHO
03A3	GREEK CAPITAL LETTER SIGMA
03A4	GREEK CAPITAL LETTER TAU
03A5	GREEK CAPITAL LETTER UPSILON
03A6	GREEK CAPITAL LETTER PHI
03A7	GREEK CAPITAL LETTER CHI
03A8	GREEK CAPITAL LETTER PSI
03A9	GREEK CAPITAL LETTER OMEGA
03B1	GREEK SMALL LETTER ALPHA
03B2	GREEK SMALL LETTER BETA
03B3	GREEK SMALL LETTER GAMMA
03B4	GREEK SMALL LETTER DELTA
03B5	GREEK SMALL LETTER EPSILON
03B6	GREEK SMALL LETTER ZETA
03B7	GREEK SMALL LETTER ETA
03B8	GREEK SMALL LETTER THETA
03B9	GREEK SMALL LETTER IOTA
03BA	GREEK SMALL LETTER KAPPA
03BB	GREEK SMALL LETTER LAMDA
03BC	GREEK SMALL LETTER MU
03BD	GREEK SMALL LETTER NU
03BE	GREEK SMALL LETTER XI
03BF	GREEK SMALL LETTER OMICRON
03C0	GREEK SMALL LETTER PI
03C1	GREEK SMALL LETTER RHO
03C3	GREEK SMALL LETTER SIGMA
03C4	GREEK SMALL LETTER TAU
03C5	GREEK SMALL LETTER UPSILON
03C6	GREEK SMALL LETTER PHI
03C7	GREEK SMALL LETTER CHI
03C8	GREEK SMALL LETTER PSI
03C9	GREEK SMALL LETTER OMEGA
0401	CYRILLIC CAPITAL LETTER IO
0410	CYRILLIC CAPITAL LETTER A
0411	CYRILLIC CAPITAL LETTER BE
0412	CYRILLIC CAPITAL LETTER VE
0413	CYRILLIC CAPITAL LETTER GHE
0414	CYRILLIC CAPITAL LETTER DE
0415	CYRILLIC CAPITAL LETTER IE
0416	CYRILLIC CAPITAL LETTER ZHE
0417	CYRILLIC CAPITAL LETTER ZE
0418	CYRILLIC CAPITAL LETTER I
0419	CYRILLIC CAPITAL LETTER SHORT I
041A	CYRILLIC CAPITAL LETTER KA
041B	CYRILLIC CAPITAL LETTER EL
041C	CYRILLIC CAPITAL LETTER EM
041D	CYRILLIC CAPITAL LETTER EN
041E	CYRILLIC CAPITAL LETTER O
041F	CYRILLIC CAPITAL LETTER PE
0420	CYRILLIC CAPITAL LETTER ER
0421	CYRILLIC CAPITAL LETTER ES
0422	CYRILLIC CAPITAL LETTER TE
0423	CYRILLIC CAPITAL LETTER U
0424	CYRILLIC CAPITAL LETTER EF
0425	CYRILLIC CAPITAL LETTER HA
0426	CYRILLIC CAPITAL LETTER TSE
0427	CYRILLIC CAPITAL LETTER CHE
0428	CYRILLIC CAPITAL LETTER SHA
0429	CYRILLIC CAPITAL LETTER SHCHA
042A	CYRILLIC CAPITAL LETTER HARD SIGN
042B	CYRILLIC CAPITAL LETTER YERU
042C	CYRILLIC CAPITAL LETTER SOFT SIGN
042D	CYRILLIC CAPITAL LETTER E
042E	CYRILLIC CAPITAL LETTER YU
042F	CYRILLIC CAPITAL LETTER YA
0430	CYRILLIC SMALL LETTER A
0431	CYRILLIC SMALL LETTER BE
0432	CYRILLIC SMALL LETTER VE
0433	CYRILLIC SMALL LETTER GHE
0434	CYRILLIC SMALL LETTER DE
0435	CYRILLIC SMALL LETTER IE
0436	CYRILLIC SMALL LETTER ZHE
0437	CYRILLIC SMALL LETTER ZE
0438	CYRILLIC SMALL LETTER I
0439	CYRILLIC SMALL LETTER SHORT I
043A	CYRILLIC SMALL LETTER KA
043B	CYRILLIC SMALL LETTER EL
043C	CYRILLIC SMALL LETTER EM
043D	CYRILLIC SMALL LETTER EN
043E	CYRILLIC SMALL LETTER O
043F	CYRILLIC SMALL LETTER PE
0440	CYRILLIC SMALL LETTER ER
0441	CYRILLIC SMALL LETTER ES
0442	CYRILLIC SMALL LETTER TE
0443	CYRILLIC SMALL LETTER U
0444	CYRILLIC SMALL LETTER EF
0445	CYRILLIC SMALL LETTER HA
0446	CYRILLIC SMALL LETTER TSE
0447	CYRILLIC SMALL LETTER CHE
0448	CYRILLIC SMALL LETTER SHA
0449	CYRILLIC SMALL LETTER SHCHA
044A	CYRILLIC SMALL LETTER HARD SIGN
044B	CYRILLIC SMALL LETTER YERU
044C	CYRILLIC SMALL LETTER SOFT SIGN
044D	CYRILLIC SMALL LETTER E
044E	CYRILLIC SMALL LETTER YU
044F	CYRILLIC SMALL LETTER YA
0451	CYRILLIC SMALL LETTER IO
2010	HYPHEN
2013	EN DASH
2014	EM DASH
2015	HORIZONTAL BAR
2016	DOUBLE VERTICAL LINE
2018	LEFT SINGLE QUOTATION MARK
2019	RIGHT SINGLE QUOTATION MARK
201C	LEFT DOUBLE QUOTATION MARK
201D	RIGHT DOUBLE QUOTATION MARK
2025	TWO DOT LEADER
2026	HORIZONTAL ELLIPSIS
2030	PER MILLE SIGN
2032	PRIME
2033	DOUBLE PRIME
2035	REVERSED PRIME
203B	REFERENCE MARK
20AC	EURO SIGN
2103	DEGREE CELSIUS
2105	CARE OF
2109	DEGREE FAHRENHEIT
2116	NUMERO SIGN
2121	TELEPHONE SIGN
2160	ROMAN NUMERAL ONE
2161	ROMAN NUMERAL TWO
2162	ROMAN NUMERAL THREE
2163	ROMAN NUMERAL FOUR
2164	ROMAN NUMERAL FIVE
2165	ROMAN NUMERAL SIX
2166	ROMAN NUMERAL SEVEN
2167	ROMAN NUMERAL EIGHT
2168	ROMAN NUMERAL NINE
2169	ROMAN NUMERAL TEN
216A	ROMAN NUMERAL ELEVEN
216B	ROMAN NUMERAL TWELVE
2170	SMALL ROMAN NUMERAL ONE
2171	SMALL ROMAN NUMERAL TWO
2172	SMALL ROMAN NUMERAL THREE
2173	SMALL ROMAN NUMERAL FOUR
2174	SMALL ROMAN NUMERAL FIVE
2175	SMALL ROMAN NUMERAL SIX
2176	SMALL ROMAN NUMERAL SEVEN
2177	SMALL ROMAN NUMERAL EIGHT
2178	SMALL ROMAN NUMERAL NINE
2179	SMALL ROMAN NUMERAL TEN
2190	LEFTWARDS ARROW
2191	UPWARDS ARROW
2192	RIGHTWARDS ARROW
2193	DOWNWARDS ARROW
2196	NORTH WEST ARROW
2197	NORTH EAST ARROW
2198	SOUTH EAST ARROW
2199	SOUTH WEST ARROW
2208	ELEMENT OF
220F	N-ARY PRODUCT
2211	N-ARY SUMMATION
2215	DIVISION SLASH
221A	SQUARE ROOT
221D	PROPORTIONAL TO
221E	INFINITY
221F	RIGHT ANGLE
2220	ANGLE
2223	DIVIDES
2225	PARALLEL TO
2227	LOGICAL AND
2228	LOGICAL OR
2229	INTERSECTION
222A	UNION
222B	INTEGRAL
222E	CONTOUR INTEGRAL
2234	THEREFORE
2235	BECAUSE
2236	RATIO
2237	PROPORTION
223D	REVERSED TILDE (LAZY S)
2248	ALMOST EQUAL TO

224C	ALL EQUAL TO
2252	APPROXIMATELY EQUAL TO OR THE IMAGE OF
2260	NOT EQUAL TO
2261	IDENTICAL TO
2264	LESS-THAN OR EQUAL TO
2265	GREATER-THAN OR EQUAL TO
2266	LESS-THAN OVER EQUAL TO
2267	GREATER-THAN OVER EQUAL TO
226E	NOT LESS-THAN
226F	NOT GREATER-THAN
2295	CIRCLED PLUS
2299	CIRCLED DOT OPERATOR
22A5	UP TACK
22BF	RIGHT TRIANGLE
2312	ARC
2460	CIRCLED DIGIT ONE
2461	CIRCLED DIGIT TWO
2462	CIRCLED DIGIT THREE
2463	CIRCLED DIGIT FOUR
2464	CIRCLED DIGIT FIVE
2465	CIRCLED DIGIT SIX
2466	CIRCLED DIGIT SEVEN
2467	CIRCLED DIGIT EIGHT
2468	CIRCLED DIGIT NINE
2469	CIRCLED NUMBER TEN
2474	PARENTHESIZED DIGIT ONE
2475	PARENTHESIZED DIGIT TWO
2476	PARENTHESIZED DIGIT THREE
2477	PARENTHESIZED DIGIT FOUR
2478	PARENTHESIZED DIGIT FIVE
2479	PARENTHESIZED DIGIT SIX
247A	PARENTHESIZED DIGIT SEVEN
247B	PARENTHESIZED DIGIT EIGHT
247C	PARENTHESIZED DIGIT NINE
247D	PARENTHESIZED NUMBER TEN
247E	PARENTHESIZED NUMBER ELEVEN
247F	PARENTHESIZED NUMBER TWELVE
2480	PARENTHESIZED NUMBER THIRTEEN
2481	PARENTHESIZED NUMBER FOURTEEN
2482	PARENTHESIZED NUMBER FIFTEEN
2483	PARENTHESIZED NUMBER SIXTEEN
2484	PARENTHESIZED NUMBER SEVENTEEN
2485	PARENTHESIZED NUMBER EIGHTEEN
2486	PARENTHESIZED NUMBER NINETEEN
2487	PARENTHESIZED NUMBER TWENTY
2488	DIGIT ONE FULL STOP
2489	DIGIT TWO FULL STOP
248A	DIGIT THREE FULL STOP
248B	DIGIT FOUR FULL STOP
248C	DIGIT FIVE FULL STOP
248D	DIGIT SIX FULL STOP
248E	DIGIT SEVEN FULL STOP
248F	DIGIT EIGHT FULL STOP
2490	DIGIT NINE FULL STOP
2491	NUMBER TEN FULL STOP
2492	NUMBER ELEVEN FULL STOP
2493	NUMBER TWELVE FULL STOP
2494	NUMBER THIRTEEN FULL STOP
2495	NUMBER FOURTEEN FULL STOP
2496	NUMBER FIFTEEN FULL STOP
2497	NUMBER SIXTEEN FULL STOP
2498	NUMBER SEVENTEEN FULL STOP
2499	NUMBER EIGHTEEN FULL STOP
249A	NUMBER NINETEEN FULL STOP
249B	NUMBER TWENTY FULL STOP
2500	BOX DRAWINGS LIGHT HORIZONTAL
2501	BOX DRAWINGS HEAVY HORIZONTAL
2502	BOX DRAWINGS LIGHT VERTICAL
2503	BOX DRAWINGS HEAVY VERTICAL
2504	BOX DRAWINGS LIGHT TRIPLE DASH HORIZONTAL
2505	BOX DRAWINGS HEAVY TRIPLE DASH HORIZONTAL
2506	BOX DRAWINGS LIGHT TRIPLE DASH VERTICAL
2507	BOX DRAWINGS HEAVY TRIPLE DASH VERTICAL
2508	BOX DRAWINGS LIGHT QUADRUPLE DASH HORIZONTAL
2509	BOX DRAWINGS HEAVY QUADRUPLE DASH HORIZONTAL
250A	BOX DRAWINGS LIGHT QUADRUPLE DASH VERTICAL
250B	BOX DRAWINGS HEAVY QUADRUPLE DASH VERTICAL
250C	BOX DRAWINGS LIGHT DOWN AND RIGHT
250D	BOX DRAWINGS DOWN LIGHT AND RIGHT HEAVY
250E	BOX DRAWINGS DOWN HEAVY AND RIGHT LIGHT
250F	BOX DRAWINGS HEAVY DOWN AND RIGHT

2510	BOX DRAWINGS LIGHT DOWN AND LEFT
2511	BOX DRAWINGS DOWN LIGHT AND LEFT HEAVY
2512	BOX DRAWINGS DOWN HEAVY AND LEFT LIGHT
2513	BOX DRAWINGS HEAVY DOWN AND LEFT
2514	BOX DRAWINGS LIGHT UP AND RIGHT
2515	BOX DRAWINGS UP LIGHT AND RIGHT HEAVY
2516	BOX DRAWINGS UP HEAVY AND RIGHT LIGHT
2517	BOX DRAWINGS HEAVY UP AND RIGHT
2518	BOX DRAWINGS LIGHT UP AND LEFT
2519	BOX DRAWINGS UP LIGHT AND LEFT HEAVY
251A	BOX DRAWINGS UP HEAVY AND LEFT LIGHT
251B	BOX DRAWINGS HEAVY UP AND LEFT
251C	BOX DRAWINGS LIGHT VERTICAL AND RIGHT
251D	BOX DRAWINGS VERTICAL LIGHT AND RIGHT HEAVY
251E	BOX DRAWINGS UP HEAVY AND RIGHT DOWN LIGHT
251F	BOX DRAWINGS DOWN HEAVY AND RIGHT UP LIGHT
2520	BOX DRAWINGS VERTICAL HEAVY AND RIGHT LIGHT
2521	BOX DRAWINGS DOWN LIGHT AND RIGHT UP HEAVY
2522	BOX DRAWINGS UP LIGHT AND RIGHT DOWN HEAVY
2523	BOX DRAWINGS HEAVY VERTICAL AND RIGHT
2524	BOX DRAWINGS LIGHT VERTICAL AND LEFT
2525	BOX DRAWINGS VERTICAL LIGHT AND LEFT HEAVY
2526	BOX DRAWINGS UP HEAVY AND LEFT DOWN LIGHT
2527	BOX DRAWINGS DOWN HEAVY AND LEFT UP LIGHT
2528	BOX DRAWINGS VERTICAL HEAVY AND LEFT LIGHT
2529	BOX DRAWINGS DOWN LIGHT AND LEFT UP HEAVY
252A	BOX DRAWINGS UP LIGHT AND LEFT DOWN HEAVY
252B	BOX DRAWINGS HEAVY VERTICAL AND LEFT
252C	BOX DRAWINGS LIGHT DOWN AND HORIZONTAL
252D	BOX DRAWINGS LEFT HEAVY AND RIGHT DOWN LIGHT
252E	BOX DRAWINGS RIGHT HEAVY AND LEFT DOWN LIGHT
252F	BOX DRAWINGS DOWN LIGHT AND HORIZONTAL HEAVY
2530	BOX DRAWINGS DOWN HEAVY AND HORIZONTAL LIGHT
2531	BOX DRAWINGS RIGHT LIGHT AND LEFT DOWN HEAVY
2532	BOX DRAWINGS LEFT LIGHT AND RIGHT DOWN HEAVY
2533	BOX DRAWINGS HEAVY DOWN AND HORIZONTAL
2534	BOX DRAWINGS LIGHT UP AND HORIZONTAL
2535	BOX DRAWINGS LEFT HEAVY AND RIGHT UP LIGHT
2536	BOX DRAWINGS RIGHT HEAVY AND LEFT UP LIGHT
2537	BOX DRAWINGS UP LIGHT AND HORIZONTAL HEAVY
2538	BOX DRAWINGS UP HEAVY AND HORIZONTAL LIGHT
2539	BOX DRAWINGS RIGHT LIGHT AND LEFT UP HEAVY
253A	BOX DRAWINGS LEFT LIGHT AND RIGHT UP HEAVY
253B	BOX DRAWINGS HEAVY UP AND HORIZONTAL
253C	BOX DRAWINGS LIGHT VERTICAL AND HORIZONTAL
253D	BOX DRAWINGS LEFT HEAVY AND RIGHT VERTICAL LIGHT
253E	BOX DRAWINGS RIGHT HEAVY AND LEFT VERTICAL LIGHT
253F	BOX DRAWINGS VERTICAL LIGHT AND HORIZONTAL HEAVY
2540	BOX DRAWINGS UP HEAVY AND DOWN HORIZONTAL LIGHT
2541	BOX DRAWINGS DOWN HEAVY AND UP HORIZONTAL LIGHT
2542	BOX DRAWINGS VERTICAL HEAVY AND HORIZONTAL LIGHT
2543	BOX DRAWINGS LEFT UP HEAVY AND RIGHT DOWN LIGHT
2544	BOX DRAWINGS RIGHT UP HEAVY AND LEFT DOWN LIGHT
2545	BOX DRAWINGS LEFT DOWN HEAVY AND RIGHT UP LIGHT

2546	BOX DRAWINGS RIGHT DOWN HEAVY AND LEFT UP LIGHT
2547	BOX DRAWINGS DOWN LIGHT AND UP HORIZONTAL HEAVY
2548	BOX DRAWINGS UP LIGHT AND DOWN HORIZONTAL HEAVY
2549	BOX DRAWINGS RIGHT LIGHT AND LEFT VERTICAL HEAVY
254A	BOX DRAWINGS LEFT LIGHT AND RIGHT VERTICAL HEAVY
254B	BOX DRAWINGS HEAVY VERTICAL AND HORIZONTAL
2550	BOX DRAWINGS DOUBLE HORIZONTAL
2551	BOX DRAWINGS DOUBLE VERTICAL
2552	BOX DRAWINGS DOWN SINGLE AND RIGHT DOUBLE
2553	BOX DRAWINGS DOWN DOUBLE AND RIGHT SINGLE
2554	BOX DRAWINGS DOUBLE DOWN AND RIGHT
2555	BOX DRAWINGS DOWN SINGLE AND LEFT DOUBLE
2556	BOX DRAWINGS DOWN DOUBLE AND LEFT SINGLE
2557	BOX DRAWINGS DOUBLE DOWN AND LEFT
2558	BOX DRAWINGS UP SINGLE AND RIGHT DOUBLE
2559	BOX DRAWINGS UP DOUBLE AND RIGHT SINGLE
255A	BOX DRAWINGS DOUBLE UP AND RIGHT
255B	BOX DRAWINGS UP SINGLE AND LEFT DOUBLE
255C	BOX DRAWINGS UP DOUBLE AND LEFT SINGLE
255D	BOX DRAWINGS DOUBLE UP AND LEFT
255E	BOX DRAWINGS VERTICAL SINGLE AND RIGHT DOUBLE
255F	BOX DRAWINGS VERTICAL DOUBLE AND RIGHT SINGLE
2560	BOX DRAWINGS DOUBLE VERTICAL AND RIGHT
2561	BOX DRAWINGS VERTICAL SINGLE AND LEFT DOUBLE
2562	BOX DRAWINGS VERTICAL DOUBLE AND LEFT SINGLE
2563	BOX DRAWINGS DOUBLE VERTICAL AND LEFT
2564	BOX DRAWINGS DOWN SINGLE AND HORIZONTAL DOUBLE
2565	BOX DRAWINGS DOWN DOUBLE AND HORIZONTAL SINGLE
2566	BOX DRAWINGS DOUBLE DOWN AND HORIZONTAL
2567	BOX DRAWINGS UP SINGLE AND HORIZONTAL DOUBLE
2568	BOX DRAWINGS UP DOUBLE AND HORIZONTAL SINGLE
2569	BOX DRAWINGS DOUBLE UP AND HORIZONTAL
256A	BOX DRAWINGS VERTICAL SINGLE AND HORIZONTAL DOUBLE
256B	BOX DRAWINGS VERTICAL DOUBLE AND HORIZONTAL SINGLE
256C	BOX DRAWINGS DOUBLE VERTICAL AND HORIZONTAL
256D	BOX DRAWINGS LIGHT ARC DOWN AND RIGHT
256E	BOX DRAWINGS LIGHT ARC DOWN AND LEFT
256F	BOX DRAWINGS LIGHT ARC UP AND LEFT
2570	BOX DRAWINGS LIGHT ARC UP AND RIGHT
2571	BOX DRAWINGS LIGHT DIAGONAL UPPER RIGHT TO LOWER LEFT
2572	BOX DRAWINGS LIGHT DIAGONAL UPPER LEFT TO LOWER RIGHT
2573	BOX DRAWINGS LIGHT DIAGONAL CROSS
2581	LOWER ONE EIGHTH BLOCK
2582	LOWER ONE QUARTER BLOCK
2583	LOWER THREE EIGHTHS BLOCK
2584	LOWER HALF BLOCK
2585	LOWER FIVE EIGHTHS BLOCK
2586	LOWER THREE QUARTERS BLOCK
2587	LOWER SEVEN EIGHTHS BLOCK
2588	FULL BLOCK
2589	LEFT SEVEN EIGHTHS BLOCK
258A	LEFT THREE QUARTERS BLOCK
258B	LEFT FIVE EIGHTHS BLOCK
258C	LEFT HALF BLOCK
258D	LEFT THREE EIGHTHS BLOCK
258E	LEFT ONE QUARTER BLOCK
258F	LEFT ONE EIGHTH BLOCK
2593	DARK SHADE
2594	UPPER ONE EIGHTH BLOCK
2595	RIGHT ONE EIGHTH BLOCK
25A0	BLACK SQUARE
25A1	WHITE SQUARE
25B2	BLACK UP-POINTING TRIANGLE
25B3	WHITE UP-POINTING TRIANGLE
25BC	BLACK DOWN-POINTING TRIANGLE
25BD	WHITE DOWN-POINTING TRIANGLE
25C6	BLACK DIAMOND
25C7	WHITE DIAMOND
25CB	WHITE CIRCLE
25CE	BULLSEYE
25CF	BLACK CIRCLE
25E2	BLACK LOWER RIGHT TRIANGLE
25E3	BLACK LOWER LEFT TRIANGLE
25E4	BLACK UPPER LEFT TRIANGLE
25E5	BLACK UPPER RIGHT TRIANGLE
2605	BLACK STAR
2606	WHITE STAR
2609	SUN
2640	FEMALE SIGN
2642	MALE SIGN
2FF0	IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO RIGHT
2FF1	IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO BELOW
2FF2	IDEOGRAPHIC DESCRIPTION CHARACTER LEFT TO MIDDLE AND RIGHT
2FF3	IDEOGRAPHIC DESCRIPTION CHARACTER ABOVE TO MIDDLE AND BELOW
2FF4	IDEOGRAPHIC DESCRIPTION CHARACTER FULL SURROUND
2FF5	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM ABOVE
2FF6	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM BELOW
2FF7	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM LEFT
2FF8	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM UPPER LEFT
2FF9	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM UPPER RIGHT
2FFA	IDEOGRAPHIC DESCRIPTION CHARACTER SURROUND FROM LOWER LEFT
2FFB	IDEOGRAPHIC DESCRIPTION CHARACTER OVERLAID
3000	IDEOGRAPHIC SPACE
3001	IDEOGRAPHIC COMMA
3002	IDEOGRAPHIC FULL STOP
3003	DITTO MARK
3005	IDEOGRAPHIC ITERATION MARK
3006	IDEOGRAPHIC CLOSING MARK
3007	IDEOGRAPHIC NUMBER ZERO
3008	LEFT ANGLE BRACKET
3009	RIGHT ANGLE BRACKET
300A	LEFT DOUBLE ANGLE BRACKET
300B	RIGHT DOUBLE ANGLE BRACKET
300C	LEFT CORNER BRACKET
300D	RIGHT CORNER BRACKET
300E	LEFT WHITE CORNER BRACKET
300F	RIGHT WHITE CORNER BRACKET
3010	LEFT BLACK LENTICULAR BRACKET
3011	RIGHT BLACK LENTICULAR BRACKET
3012	POSTAL MARK
3013	GETA MARK
3014	LEFT TORTOISE SHELL BRACKET
3015	RIGHT TORTOISE SHELL BRACKET
3016	LEFT WHITE LENTICULAR BRACKET
3017	RIGHT WHITE LENTICULAR BRACKET
301D	REVERSED DOUBLE PRIME QUOTATION MARK
301E	DOUBLE PRIME QUOTATION MARK
3021	HANGZHOU NUMERAL ONE
3022	HANGZHOU NUMERAL TWO
3023	HANGZHOU NUMERAL THREE
3024	HANGZHOU NUMERAL FOUR
3025	HANGZHOU NUMERAL FIVE
3026	HANGZHOU NUMERAL SIX
3027	HANGZHOU NUMERAL SEVEN
3028	HANGZHOU NUMERAL EIGHT
3029	HANGZHOU NUMERAL NINE
303E	IDEOGRAPHIC VARIATION INDICATOR
3041	HIRAGANA LETTER SMALL A
3042	HIRAGANA LETTER A
3043	HIRAGANA LETTER SMALL I
3044	HIRAGANA LETTER I
3045	HIRAGANA LETTER SMALL U
3046	HIRAGANA LETTER U
3047	HIRAGANA LETTER SMALL E
3048	HIRAGANA LETTER E
3049	HIRAGANA LETTER SMALL O
304A	HIRAGANA LETTER O
304B	HIRAGANA LETTER KA
304C	HIRAGANA LETTER GA
304D	HIRAGANA LETTER KI
304E	HIRAGANA LETTER GI

304F	HIRAGANA LETTER KU
3050	HIRAGANA LETTER GU
3051	HIRAGANA LETTER KE
3052	HIRAGANA LETTER GE
3053	HIRAGANA LETTER KO
3054	HIRAGANA LETTER GO
3055	HIRAGANA LETTER SA
3056	HIRAGANA LETTER ZA
3057	HIRAGANA LETTER SI
3058	HIRAGANA LETTER ZI
3059	HIRAGANA LETTER SU
305A	HIRAGANA LETTER ZU
305B	HIRAGANA LETTER SE
305C	HIRAGANA LETTER ZE
305D	HIRAGANA LETTER SO
305E	HIRAGANA LETTER ZO
305F	HIRAGANA LETTER TA
3060	HIRAGANA LETTER DA
3061	HIRAGANA LETTER TI
3062	HIRAGANA LETTER DI
3063	HIRAGANA LETTER SMALL TU
3064	HIRAGANA LETTER TU
3065	HIRAGANA LETTER DU
3066	HIRAGANA LETTER TE
3067	HIRAGANA LETTER DE
3068	HIRAGANA LETTER TO
3069	HIRAGANA LETTER DO
306A	HIRAGANA LETTER NA
306B	HIRAGANA LETTER NI
306C	HIRAGANA LETTER NU
306D	HIRAGANA LETTER NE
306E	HIRAGANA LETTER NO
306F	HIRAGANA LETTER HA
3070	HIRAGANA LETTER BA
3071	HIRAGANA LETTER PA
3072	HIRAGANA LETTER HI
3073	HIRAGANA LETTER BI
3074	HIRAGANA LETTER PI
3075	HIRAGANA LETTER HU
3076	HIRAGANA LETTER BU
3077	HIRAGANA LETTER PU
3078	HIRAGANA LETTER HE
3079	HIRAGANA LETTER BE
307A	HIRAGANA LETTER PE
307B	HIRAGANA LETTER HO
307C	HIRAGANA LETTER BO
307D	HIRAGANA LETTER PO
307E	HIRAGANA LETTER MA
307F	HIRAGANA LETTER MI
3080	HIRAGANA LETTER MU
3081	HIRAGANA LETTER ME
3082	HIRAGANA LETTER MO
3083	HIRAGANA LETTER SMALL YA
3084	HIRAGANA LETTER YA
3085	HIRAGANA LETTER SMALL YU
3086	HIRAGANA LETTER YU
3087	HIRAGANA LETTER SMALL YO
3088	HIRAGANA LETTER YO
3089	HIRAGANA LETTER RA
308A	HIRAGANA LETTER RI
308B	HIRAGANA LETTER RU
308C	HIRAGANA LETTER RE
308D	HIRAGANA LETTER RO
308E	HIRAGANA LETTER SMALL WA
308F	HIRAGANA LETTER WA
3090	HIRAGANA LETTER WI
3091	HIRAGANA LETTER WE
3092	HIRAGANA LETTER WO
3093	HIRAGANA LETTER N
309B	KATAKANA-HIRAGANA VOICED SOUND MARK
309C	KATAKANA-HIRAGANA SEMI-VOICED SOUND MARK
309D	HIRAGANA ITERATION MARK
309E	HIRAGANA VOICED ITERATION MARK
30A1	KATAKANA LETTER SMALL A
30A2	KATAKANA LETTER A
30A3	KATAKANA LETTER SMALL I
30A4	KATAKANA LETTER I
30A5	KATAKANA LETTER SMALL U
30A6	KATAKANA LETTER U
30A7	KATAKANA LETTER SMALL E
30A8	KATAKANA LETTER E
30A9	KATAKANA LETTER SMALL O
30AA	KATAKANA LETTER O
30AB	KATAKANA LETTER KA
30AC	KATAKANA LETTER GA
30AD	KATAKANA LETTER KI
30AE	KATAKANA LETTER GI
30AF	KATAKANA LETTER KU
30B0	KATAKANA LETTER GU
30B1	KATAKANA LETTER KE
30B2	KATAKANA LETTER GE
30B3	KATAKANA LETTER KO
30B4	KATAKANA LETTER GO
30B5	KATAKANA LETTER SA
30B6	KATAKANA LETTER ZA
30B7	KATAKANA LETTER SI
30B8	KATAKANA LETTER ZI
30B9	KATAKANA LETTER SU
30BA	KATAKANA LETTER ZU
30BB	KATAKANA LETTER SE
30BC	KATAKANA LETTER ZE
30BD	KATAKANA LETTER SO
30BE	KATAKANA LETTER ZO
30BF	KATAKANA LETTER TA
30C0	KATAKANA LETTER DA
30C1	KATAKANA LETTER TI
30C2	KATAKANA LETTER DI
30C3	KATAKANA LETTER SMALL TU
30C4	KATAKANA LETTER TU
30C5	KATAKANA LETTER DU
30C6	KATAKANA LETTER TE
30C7	KATAKANA LETTER DE
30C8	KATAKANA LETTER TO
30C9	KATAKANA LETTER DO
30CA	KATAKANA LETTER NA
30CB	KATAKANA LETTER NI
30CC	KATAKANA LETTER NU
30CD	KATAKANA LETTER NE
30CE	KATAKANA LETTER NO
30CF	KATAKANA LETTER HA
30D0	KATAKANA LETTER BA
30D1	KATAKANA LETTER PA
30D2	KATAKANA LETTER HI
30D3	KATAKANA LETTER BI
30D4	KATAKANA LETTER PI
30D5	KATAKANA LETTER HU
30D6	KATAKANA LETTER BU
30D7	KATAKANA LETTER PU
30D8	KATAKANA LETTER HE
30D9	KATAKANA LETTER BE
30DA	KATAKANA LETTER PE
30DB	KATAKANA LETTER HO
30DC	KATAKANA LETTER BO
30DD	KATAKANA LETTER PO
30DE	KATAKANA LETTER MA
30DF	KATAKANA LETTER MI
30E0	KATAKANA LETTER MU
30E1	KATAKANA LETTER ME
30E2	KATAKANA LETTER MO
30E3	KATAKANA LETTER SMALL YA
30E4	KATAKANA LETTER YA
30E5	KATAKANA LETTER SMALL YU
30E6	KATAKANA LETTER YU
30E7	KATAKANA LETTER SMALL YO
30E8	KATAKANA LETTER YO
30E9	KATAKANA LETTER RA
30EA	KATAKANA LETTER RI
30EB	KATAKANA LETTER RU
30EC	KATAKANA LETTER RE
30ED	KATAKANA LETTER RO
30EE	KATAKANA LETTER SMALL WA
30EF	KATAKANA LETTER WA
30F0	KATAKANA LETTER WI
30F1	KATAKANA LETTER WE
30F2	KATAKANA LETTER WO
30F3	KATAKANA LETTER N
30F4	KATAKANA LETTER VU
30F5	KATAKANA LETTER SMALL KA
30F6	KATAKANA LETTER SMALL KE
30FC	KATAKANA-HIRAGANA PROLONGED SOUND MARK
30FD	KATAKANA ITERATION MARK
30FE	KATAKANA VOICED ITERATION MARK
3105	BOPOMOFO LETTER B
3106	BOPOMOFO LETTER P
3107	BOPOMOFO LETTER M
3108	BOPOMOFO LETTER F
3109	BOPOMOFO LETTER D
310A	BOPOMOFO LETTER T
310B	BOPOMOFO LETTER N
310C	BOPOMOFO LETTER L
310D	BOPOMOFO LETTER G
310E	BOPOMOFO LETTER K

310F	BOPOMOFO LETTER H
3110	BOPOMOFO LETTER J
3111	BOPOMOFO LETTER Q
3112	BOPOMOFO LETTER X
3113	BOPOMOFO LETTER ZH
3114	BOPOMOFO LETTER CH
3115	BOPOMOFO LETTER SH
3116	BOPOMOFO LETTER R
3117	BOPOMOFO LETTER Z
3118	BOPOMOFO LETTER C
3119	BOPOMOFO LETTER S
311A	BOPOMOFO LETTER A
311B	BOPOMOFO LETTER O
311C	BOPOMOFO LETTER E
311D	BOPOMOFO LETTER EH
311E	BOPOMOFO LETTER AI
311F	BOPOMOFO LETTER EI
3120	BOPOMOFO LETTER AU
3121	BOPOMOFO LETTER OU
3122	BOPOMOFO LETTER AN
3123	BOPOMOFO LETTER EN
3124	BOPOMOFO LETTER ANG
3125	BOPOMOFO LETTER ENG
3126	BOPOMOFO LETTER ER
3127	BOPOMOFO LETTER I
3128	BOPOMOFO LETTER U
3129	BOPOMOFO LETTER IU
3220	PARENTHESIZED IDEOGRAPH ONE
3221	PARENTHESIZED IDEOGRAPH TWO
3222	PARENTHESIZED IDEOGRAPH THREE
3223	PARENTHESIZED IDEOGRAPH FOUR
3224	PARENTHESIZED IDEOGRAPH FIVE
3225	PARENTHESIZED IDEOGRAPH SIX
3226	PARENTHESIZED IDEOGRAPH SEVEN
3227	PARENTHESIZED IDEOGRAPH EIGHT
3228	PARENTHESIZED IDEOGRAPH NINE
3229	PARENTHESIZED IDEOGRAPH TEN
3231	PARENTHESIZED IDEOGRAPH STOCK
32A3	CIRCLED IDEOGRAPH CORRECT
338E	SQUARE MG
338F	SQUARE KG
339C	SQUARE MM
339D	SQUARE CM
339E	SQUARE KM
33A1	SQUARE M SQUARED
33C4	SQUARE CC
33CE	SQUARE KM CAPITAL
33D1	SQUARE LN
33D2	SQUARE LOG
33D5	SQUARE MIL
3400-4DB5	CJK UNIFIED IDEOGRAPH EXTENSION A
4E00-9FA5	CJK UNIFIED IDEOGRAPH
E78D-E796	PRIVATE USE AREA
E7C7-E7C8	PRIVATE USE AREA
E815-E864	PRIVATE USE AREA
F92C	CJK COMPATIBILITY IDEOGRAPH-F92C
F979	CJK COMPATIBILITY IDEOGRAPH-F979
F995	CJK COMPATIBILITY IDEOGRAPH-F995
F9E7	CJK COMPATIBILITY IDEOGRAPH-F9E7
F9F1	CJK COMPATIBILITY IDEOGRAPH-F9F1
FA0C	CJK COMPATIBILITY IDEOGRAPH-FA0C
FA0D	CJK COMPATIBILITY IDEOGRAPH-FA0D
FA0E	CJK COMPATIBILITY IDEOGRAPH-FA0E
FA0F	CJK COMPATIBILITY IDEOGRAPH-FA0F
FA11	CJK COMPATIBILITY IDEOGRAPH-FA11
FA13	CJK COMPATIBILITY IDEOGRAPH-FA13
FA14	CJK COMPATIBILITY IDEOGRAPH-FA14
FA18	CJK COMPATIBILITY IDEOGRAPH-FA18
FA1F	CJK COMPATIBILITY IDEOGRAPH-FA1F *
FA20	CJK COMPATIBILITY IDEOGRAPH-FA20
FA21	CJK COMPATIBILITY IDEOGRAPH-FA21
FA23	CJK COMPATIBILITY IDEOGRAPH-FA23 *
FA24	CJK COMPATIBILITY IDEOGRAPH-FA24
FA27	CJK COMPATIBILITY IDEOGRAPH-FA27
FA28	CJK COMPATIBILITY IDEOGRAPH-FA28
FA29	CJK COMPATIBILITY IDEOGRAPH-FA29
FE30	PRESENTATION FORM FOR VERTICAL TWO DOT LEADER
FE31	PRESENTATION FORM FOR VERTICAL EM DASH
FE33	PRESENTATION FORM FOR VERTICAL LOW LINE
FE34	PRESENTATION FORM FOR VERTICAL WAVY LOW LINE
FE35	PRESENTATION FORM FOR VERTICAL LEFT PARENTHESIS
FE36	PRESENTATION FORM FOR VERTICAL RIGHT PARENTHESIS
FE37	PRESENTATION FORM FOR VERTICAL LEFT CURLY BRACKET
FE38	PRESENTATION FORM FOR VERTICAL RIGHT CURLY BRACKET
FE39	PRESENTATION FORM FOR VERTICAL LEFT TORTOISE SHELL BRACKET
FE3A	PRESENTATION FORM FOR VERTICAL RIGHT TORTOISE SHELL BRACKET
FE3B	PRESENTATION FORM FOR VERTICAL LEFT BLACK LENTICULAR BRACKET
FE3C	PRESENTATION FORM FOR VERTICAL RIGHT BLACK LENTICULAR BRACKET
FE3D	PRESENTATION FORM FOR VERTICAL LEFT DOUBLE ANGLE BRACKET
FE3E	PRESENTATION FORM FOR VERTICAL RIGHT DOUBLE ANGLE BRACKET
FE3F	PRESENTATION FORM FOR VERTICAL LEFT ANGLE BRACKET
FE40	PRESENTATION FORM FOR VERTICAL RIGHT ANGLE BRACKET
FE41	PRESENTATION FORM FOR VERTICAL LEFT CORNER BRACKET
FE42	PRESENTATION FORM FOR VERTICAL RIGHT CORNER BRACKET
FE43	PRESENTATION FORM FOR VERTICAL LEFT WHITE CORNER BRACKET
FE44	PRESENTATION FORM FOR VERTICAL RIGHT WHITE CORNER BRACKET
FE49	DASHED OVERLINE
FE4A	CENTRELINE OVERLINE
FE4B	WAVY OVERLINE
FE4C	DOUBLE WAVY OVERLINE
FE4D	DASHED LOW LINE
FE4E	CENTRELINE LOW LINE
FE4F	WAVY LOW LINE
FE50	SMALL COMMA
FE51	SMALL IDEOGRAPHIC COMMA
FE52	SMALL FULL STOP
FE54	SMALL SEMICOLON
FE55	SMALL COLON
FE56	SMALL QUESTION MARK
FE57	SMALL EXCLAMATION MARK
FE59	SMALL LEFT PARENTHESIS
FE5A	SMALL RIGHT PARENTHESIS
FE5B	SMALL LEFT CURLY BRACKET
FE5C	SMALL RIGHT CURLY BRACKET
FE5D	SMALL LEFT TORTOISE SHELL BRACKET
FE5E	SMALL RIGHT TORTOISE SHELL BRACKET
FE5F	SMALL NUMBER SIGN
FE60	SMALL AMPERSAND
FE61	SMALL ASTERISK
FE62	SMALL PLUS SIGN
FE63	SMALL HYPHEN-MINUS
FE64	SMALL LESS-THAN SIGN
FE65	SMALL GREATER-THAN SIGN
FE66	SMALL EQUALS SIGN
FE68	SMALL REVERSE SOLIDUS
FE69	SMALL DOLLAR SIGN
FE6A	SMALL PERCENT SIGN
FE6B	SMALL COMMERCIAL AT
FF01	FULLWIDTH EXCLAMATION MARK
FF02	FULLWIDTH QUOTATION MARK
FF03	FULLWIDTH NUMBER SIGN
FF04	FULLWIDTH DOLLAR SIGN
FF05	FULLWIDTH PERCENT SIGN
FF06	FULLWIDTH AMPERSAND
FF07	FULLWIDTH APOSTROPHE
FF08	FULLWIDTH LEFT PARENTHESIS
FF09	FULLWIDTH RIGHT PARENTHESIS
FF0A	FULLWIDTH ASTERISK
FF0B	FULLWIDTH PLUS SIGN
FF0C	FULLWIDTH COMMA
FF0D	FULLWIDTH HYPHEN-MINUS
FF0E	FULLWIDTH FULL STOP
FF0F	FULLWIDTH SOLIDUS
FF10	FULLWIDTH DIGIT ZERO
FF11	FULLWIDTH DIGIT ONE
FF12	FULLWIDTH DIGIT TWO
FF13	FULLWIDTH DIGIT THREE
FF14	FULLWIDTH DIGIT FOUR
FF15	FULLWIDTH DIGIT FIVE
FF16	FULLWIDTH DIGIT SIX
FF17	FULLWIDTH DIGIT SEVEN
FF18	FULLWIDTH DIGIT EIGHT
FF19	FULLWIDTH DIGIT NINE
FF1A	FULLWIDTH COLON
FF1B	FULLWIDTH SEMICOLON
FF1C	FULLWIDTH LESS-THAN SIGN

FF1D	FULLWIDTH EQUALS SIGN
FF1E	FULLWIDTH GREATER-THAN SIGN
FF1F	FULLWIDTH QUESTION MARK
FF20	FULLWIDTH COMMERCIAL AT
FF21	FULLWIDTH LATIN CAPITAL LETTER A
FF22	FULLWIDTH LATIN CAPITAL LETTER B
FF23	FULLWIDTH LATIN CAPITAL LETTER C
FF24	FULLWIDTH LATIN CAPITAL LETTER D
FF25	FULLWIDTH LATIN CAPITAL LETTER E
FF26	FULLWIDTH LATIN CAPITAL LETTER F
FF27	FULLWIDTH LATIN CAPITAL LETTER G
FF28	FULLWIDTH LATIN CAPITAL LETTER H
FF29	FULLWIDTH LATIN CAPITAL LETTER I
FF2A	FULLWIDTH LATIN CAPITAL LETTER J
FF2B	FULLWIDTH LATIN CAPITAL LETTER K
FF2C	FULLWIDTH LATIN CAPITAL LETTER L
FF2D	FULLWIDTH LATIN CAPITAL LETTER M
FF2E	FULLWIDTH LATIN CAPITAL LETTER N
FF2F	FULLWIDTH LATIN CAPITAL LETTER O
FF30	FULLWIDTH LATIN CAPITAL LETTER P
FF31	FULLWIDTH LATIN CAPITAL LETTER Q
FF32	FULLWIDTH LATIN CAPITAL LETTER R
FF33	FULLWIDTH LATIN CAPITAL LETTER S
FF34	FULLWIDTH LATIN CAPITAL LETTER T
FF35	FULLWIDTH LATIN CAPITAL LETTER U
FF36	FULLWIDTH LATIN CAPITAL LETTER V
FF37	FULLWIDTH LATIN CAPITAL LETTER W
FF38	FULLWIDTH LATIN CAPITAL LETTER X
FF39	FULLWIDTH LATIN CAPITAL LETTER Y
FF3A	FULLWIDTH LATIN CAPITAL LETTER Z
FF3B	FULLWIDTH LEFT SQUARE BRACKET
FF3C	FULLWIDTH REVERSE SOLIDUS
FF3D	FULLWIDTH RIGHT SQUARE BRACKET
FF3E	FULLWIDTH CIRCUMFLEX ACCENT
FF3F	FULLWIDTH LOW LINE
FF40	FULLWIDTH GRAVE ACCENT
FF41	FULLWIDTH LATIN SMALL LETTER A
FF42	FULLWIDTH LATIN SMALL LETTER B
FF43	FULLWIDTH LATIN SMALL LETTER C
FF44	FULLWIDTH LATIN SMALL LETTER D
FF45	FULLWIDTH LATIN SMALL LETTER E
FF46	FULLWIDTH LATIN SMALL LETTER F
FF47	FULLWIDTH LATIN SMALL LETTER G
FF48	FULLWIDTH LATIN SMALL LETTER H
FF49	FULLWIDTH LATIN SMALL LETTER I
FF4A	FULLWIDTH LATIN SMALL LETTER J
FF4B	FULLWIDTH LATIN SMALL LETTER K
FF4C	FULLWIDTH LATIN SMALL LETTER L
FF4D	FULLWIDTH LATIN SMALL LETTER M
FF4E	FULLWIDTH LATIN SMALL LETTER N
FF4F	FULLWIDTH LATIN SMALL LETTER O
FF50	FULLWIDTH LATIN SMALL LETTER P
FF51	FULLWIDTH LATIN SMALL LETTER Q
FF52	FULLWIDTH LATIN SMALL LETTER R
FF53	FULLWIDTH LATIN SMALL LETTER S
FF54	FULLWIDTH LATIN SMALL LETTER T
FF55	FULLWIDTH LATIN SMALL LETTER U
FF56	FULLWIDTH LATIN SMALL LETTER V
FF57	FULLWIDTH LATIN SMALL LETTER W
FF58	FULLWIDTH LATIN SMALL LETTER X
FF59	FULLWIDTH LATIN SMALL LETTER Y
FF5A	FULLWIDTH LATIN SMALL LETTER Z
FF5B	FULLWIDTH LEFT CURLY BRACKET
FF5C	FULLWIDTH VERTICAL LINE
FF5D	FULLWIDTH RIGHT CURLY BRACKET
FF5E	FULLWIDTH TILDE
FFE0	FULLWIDTH CENT SIGN
FFE1	FULLWIDTH POUND SIGN
FFE2	FULLWIDTH NOT SIGN
FFE3	FULLWIDTH MACRON *
FFE4	FULLWIDTH BROKEN BAR
FFE5	FULLWIDTH YEN SIGN

Appendix H Windows 1252 Latin 1

Windows 1252 Latin 1 to Unicode translation

ASCII	Unicode	Character
0x00	0x0000	NULL
0x01	0x0001	START OF HEADING
0x02	0x0002	START OF TEXT
0x03	0x0003	END OF TEXT
0x04	0x0004	END OF TRANSMISSION
0x05	0x0005	ENQUIRY
0x06	0x0006	ACKNOWLEDGE
0x07	0x0007	BELL
0x08	0x0008	BACKSPACE
0x09	0x0009	HORIZONTAL TABULATION
0x0A	0x000A	LINE FEED
0x0B	0x000B	VERTICAL TABULATION
0x0C	0x000C	FORM FEED
0x0D	0x000D	CARRIAGE RETURN
0x0E	0x000E	SHIFT OUT
0x0F	0x000F	SHIFT IN
0x10	0x0010	DATA LINK ESCAPE
0x11	0x0011	DEVICE CONTROL ONE
0x12	0x0012	DEVICE CONTROL TWO
0x13	0x0013	DEVICE CONTROL THREE
0x14	0x0014	DEVICE CONTROL FOUR
0x15	0x0015	NEGATIVE ACKNOWLEDGE
0x16	0x0016	SYNCHRONOUS IDLE
0x17	0x0017	END OF TRANSMISSION BLOCK
0x18	0x0018	CANCEL
0x19	0x0019	END OF MEDIUM
0x1A	0x001A	SUBSTITUTE
0x1B	0x001B	ESCAPE
0x1C	0x001C	FILE SEPARATOR
0x1D	0x001D	GROUP SEPARATOR
0x1E	0x001E	RECORD SEPARATOR
0x1F	0x001F	UNIT SEPARATOR
0x20	0x0020	SPACE
0x21	0x0021	EXCLAMATION MARK
0x22	0x0022	QUOTATION MARK
0x23	0x0023	NUMBER SIGN
0x24	0x0024	DOLLAR SIGN
0x25	0x0025	PERCENT SIGN
0x26	0x0026	AMPERSAND
0x27	0x0027	APOSTROPHE
0x28	0x0028	LEFT PARENTHESIS
0x29	0x0029	RIGHT PARENTHESIS
0x2A	0x002A	ASTERISK
0x2B	0x002B	PLUS SIGN
0x2C	0x002C	COMMA
0x2D	0x002D	HYPHEN-MINUS
0x2E	0x002E	FULL STOP
0x2F	0x002F	SOLIDUS
0x30	0x0030	DIGIT ZERO
0x31	0x0031	DIGIT ONE
0x32	0x0032	DIGIT TWO
0x33	0x0033	DIGIT THREE
0x34	0x0034	DIGIT FOUR
0x35	0x0035	DIGIT FIVE
0x36	0x0036	DIGIT SIX
0x37	0x0037	DIGIT SEVEN
0x38	0x0038	DIGIT EIGHT
0x39	0x0039	DIGIT NINE
0x3A	0x003A	COLON
0x3B	0x003B	SEMICOLON
0x3C	0x003C	LESS-THAN SIGN
0x3D	0x003D	EQUALS SIGN
0x3E	0x003E	GREATER-THAN SIGN
0x3F	0x003F	QUESTION MARK
0x40	0x0040	COMMERCIAL AT
0x41	0x0041	LATIN CAPITAL LETTER A
0x42	0x0042	LATIN CAPITAL LETTER B
0x43	0x0043	LATIN CAPITAL LETTER C
0x44	0x0044	LATIN CAPITAL LETTER D
0x45	0x0045	LATIN CAPITAL LETTER E
0x46	0x0046	LATIN CAPITAL LETTER F
0x47	0x0047	LATIN CAPITAL LETTER G
0x48	0x0048	LATIN CAPITAL LETTER H
0x49	0x0049	LATIN CAPITAL LETTER I
0x4A	0x004A	LATIN CAPITAL LETTER J
0x4B	0x004B	LATIN CAPITAL LETTER K
0x4C	0x004C	LATIN CAPITAL LETTER L
0x4D	0x004D	LATIN CAPITAL LETTER M
0x4E	0x004E	LATIN CAPITAL LETTER N
0x4F	0x004F	LATIN CAPITAL LETTER O
0x50	0x0050	LATIN CAPITAL LETTER P
0x51	0x0051	LATIN CAPITAL LETTER Q
0x52	0x0052	LATIN CAPITAL LETTER R
0x53	0x0053	LATIN CAPITAL LETTER S
0x54	0x0054	LATIN CAPITAL LETTER T
0x55	0x0055	LATIN CAPITAL LETTER U
0x56	0x0056	LATIN CAPITAL LETTER V
0x57	0x0057	LATIN CAPITAL LETTER W
0x58	0x0058	LATIN CAPITAL LETTER X
0x59	0x0059	LATIN CAPITAL LETTER Y
0x5A	0x005A	LATIN CAPITAL LETTER Z
0x5B	0x005B	LEFT SQUARE BRACKET
0x5C	0x005C	REVERSE SOLIDUS
0x5D	0x005D	RIGHT SQUARE BRACKET
0x5E	0x005E	CIRCUMFLEX ACCENT
0x5F	0x005F	LOW LINE
0x60	0x0060	GRAVE ACCENT
0x61	0x0061	LATIN SMALL LETTER A
0x62	0x0062	LATIN SMALL LETTER B
0x63	0x0063	LATIN SMALL LETTER C
0x64	0x0064	LATIN SMALL LETTER D
0x65	0x0065	LATIN SMALL LETTER E
0x66	0x0066	LATIN SMALL LETTER F
0x67	0x0067	LATIN SMALL LETTER G
0x68	0x0068	LATIN SMALL LETTER H
0x69	0x0069	LATIN SMALL LETTER I
0x6A	0x006A	LATIN SMALL LETTER J
0x6B	0x006B	LATIN SMALL LETTER K
0x6C	0x006C	LATIN SMALL LETTER L
0x6D	0x006D	LATIN SMALL LETTER M
0x6E	0x006E	LATIN SMALL LETTER N
0x6F	0x006F	LATIN SMALL LETTER O
0x70	0x0070	LATIN SMALL LETTER P
0x71	0x0071	LATIN SMALL LETTER Q
0x72	0x0072	LATIN SMALL LETTER R
0x73	0x0073	LATIN SMALL LETTER S
0x74	0x0074	LATIN SMALL LETTER T
0x75	0x0075	LATIN SMALL LETTER U
0x76	0x0076	LATIN SMALL LETTER V
0x77	0x0077	LATIN SMALL LETTER W
0x78	0x0078	LATIN SMALL LETTER X
0x79	0x0079	LATIN SMALL LETTER Y
0x7A	0x007A	LATIN SMALL LETTER Z
0x7B	0x007B	LEFT CURLY BRACKET
0x7C	0x007C	VERTICAL LINE
0x7D	0x007D	RIGHT CURLY BRACKET
0x7E	0x007E	TILDE
0x7F	0x007F	DELETE
0x80	0x20AC	EURO SIGN
0x81	0x0000	
0x82	0x201A	SINGLE LOW-9 QUOTATION MARK
0x83	0x0192	LATIN SMALL LETTER F WITH HOOK
0x84	0x201E	DOUBLE LOW-9 QUOTATION MARK
0x85	0x2026	HORIZONTAL ELLIPSIS
0x86	0x2020	DAGGER
0x87	0x2021	DOUBLE DAGGER
0x88	0x02C6	MODIFIER LETTER CIRCUMFLEX ACCENT
0x89	0x2030	PER MILLE SIGN
0x8A	0x0160	LATIN CAPITAL LETTER S WITH CARON

0x8B	0x2039	SINGLE LEFT-POINTING ANGLE QUOTATION MARK
0x8C	0x0152	LATIN CAPITAL LIGATURE OE
0x8D	0x0000	
0x8E	0x017D	LATIN CAPITAL LETTER Z WITH CARON
0x8F	0x0000	
0x90	0x0000	
0x91	0x2018	LEFT SINGLE QUOTATION MARK
0x92	0x2019	RIGHT SINGLE QUOTATION MARK
0x93	0x201C	LEFT DOUBLE QUOTATION MARK
0x94	0x201D	RIGHT DOUBLE QUOTATION MARK
0x95	0x2022	BULLET
0x96	0x2013	EN DASH
0x97	0x2014	EM DASH
0x98	0x02DC	SMALL TILDE
0x99	0x2122	TRADE MARK SIGN
0x9A	0x0161	LATIN SMALL LETTER S WITH CARON
0x9B	0x203A	SINGLE RIGHT-POINTING ANGLE QUOTATION MARK
0x9C	0x0153	LATIN SMALL LIGATURE OE
0x9D	0x0000	
0x9E	0x017E	LATIN SMALL LETTER Z WITH CARON
0x9F	0x0178	LATIN CAPITAL LETTER Y WITH DIAERESIS
0xA0	0x00A0	NO-BREAK SPACE
0xA1	0x00A1	INVERTED EXCLAMATION MARK
0xA2	0x00A2	CENT SIGN
0xA3	0x00A3	POUND SIGN
0xA4	0x00A4	CURRENCY SIGN
0xA5	0x00A5	YEN SIGN
0xA6	0x00A6	BROKEN BAR
0xA7	0x00A7	SECTION SIGN
0xA8	0x00A8	DIAERESIS
0xA9	0x00A9	COPYRIGHT SIGN
0xAA	0x00AA	FEMININE ORDINAL INDICATOR
0xAB	0x00AB	LEFT-POINTING DOUBLE ANGLE QUOTATION MARK
0xAC	0x00AC	NOT SIGN
0xAD	0x00AD	SOFT HYPHEN
0xAE	0x00AE	REGISTERED SIGN
0xAF	0x00AF	MACRON
0xB0	0x00B0	DEGREE SIGN
0xB1	0x00B1	PLUS-MINUS SIGN
0xB2	0x00B2	SUPERSCRIPT TWO
0xB3	0x00B3	SUPERSCRIPT THREE
0xB4	0x00B4	ACUTE ACCENT
0xB5	0x00B5	MICRO SIGN
0xB6	0x00B6	PILCROW SIGN
0xB7	0x00B7	MIDDLE DOT
0xB8	0x00B8	CEDILLA
0xB9	0x00B9	SUPERSCRIPT ONE
0xBA	0x00BA	MASCULINE ORDINAL INDICATOR
0xBB	0x00BB	RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK
0xBC	0x00BC	VULGAR FRACTION ONE QUARTER
0xBD	0x00BD	VULGAR FRACTION ONE HALF
0xBE	0x00BE	VULGAR FRACTION THREE QUARTERS
0xBF	0x00BF	INVERTED QUESTION MARK
0xC0	0x00C0	LATIN CAPITAL LETTER A WITH GRAVE
0xC1	0x00C1	LATIN CAPITAL LETTER A WITH ACUTE
0xC2	0x00C2	LATIN CAPITAL LETTER A WITH CIRCUMFLEX
0xC3	0x00C3	LATIN CAPITAL LETTER A WITH TILDE
0xC4	0x00C4	LATIN CAPITAL LETTER A WITH DIAERESIS
0xC5	0x00C5	LATIN CAPITAL LETTER A WITH RING ABOVE
0xC6	0x00C6	LATIN CAPITAL LETTER AE
0xC7	0x00C7	LATIN CAPITAL LETTER C WITH CEDILLA
0xC8	0x00C8	LATIN CAPITAL LETTER E WITH GRAVE
0xC9	0x00C9	LATIN CAPITAL LETTER E WITH ACUTE
0xCA	0x00CA	LATIN CAPITAL LETTER E WITH CIRCUMFLEX
0xCB	0x00CB	LATIN CAPITAL LETTER E WITH DIAERESIS
0xCC	0x00CC	LATIN CAPITAL LETTER I WITH GRAVE
0xCD	0x00CD	LATIN CAPITAL LETTER I WITH ACUTE
0xCE	0x00CE	LATIN CAPITAL LETTER I WITH CIRCUMFLEX
0xCF	0x00CF	LATIN CAPITAL LETTER I WITH DIAERESIS
0xD0	0x00D0	LATIN CAPITAL LETTER ETH
0xD1	0x00D1	LATIN CAPITAL LETTER N WITH TILDE
0xD2	0x00D2	LATIN CAPITAL LETTER O WITH GRAVE
0xD3	0x00D3	LATIN CAPITAL LETTER O WITH ACUTE
0xD4	0x00D4	LATIN CAPITAL LETTER O WITH CIRCUMFLEX
0xD5	0x00D5	LATIN CAPITAL LETTER O WITH TILDE
0xD6	0x00D6	LATIN CAPITAL LETTER O WITH DIAERESIS
0xD7	0x00D7	MULTIPLICATION SIGN
0xD8	0x00D8	LATIN CAPITAL LETTER O WITH STROKE
0xD9	0x00D9	LATIN CAPITAL LETTER U WITH GRAVE
0xDA	0x00DA	LATIN CAPITAL LETTER U WITH ACUTE
0xDB	0x00DB	LATIN CAPITAL LETTER U WITH CIRCUMFLEX
0xDC	0x00DC	LATIN CAPITAL LETTER U WITH DIAERESIS
0xDD	0x00DD	LATIN CAPITAL LETTER Y WITH ACUTE
0xDE	0x00DE	LATIN CAPITAL LETTER THORN
0xDF	0x00DF	LATIN SMALL LETTER SHARP S
0xE0	0x00E0	LATIN SMALL LETTER A WITH GRAVE
0xE1	0x00E1	LATIN SMALL LETTER A WITH ACUTE
0xE2	0x00E2	LATIN SMALL LETTER A WITH CIRCUMFLEX
0xE3	0x00E3	LATIN SMALL LETTER A WITH TILDE
0xE4	0x00E4	LATIN SMALL LETTER A WITH DIAERESIS
0xE5	0x00E5	LATIN SMALL LETTER A WITH RING ABOVE
0xE6	0x00E6	LATIN SMALL LETTER AE
0xE7	0x00E7	LATIN SMALL LETTER C WITH CEDILLA
0xE8	0x00E8	LATIN SMALL LETTER E WITH GRAVE
0xE9	0x00E9	LATIN SMALL LETTER E WITH ACUTE
0xEA	0x00EA	LATIN SMALL LETTER E WITH CIRCUMFLEX
0xEB	0x00EB	LATIN SMALL LETTER E WITH DIAERESIS
0xEC	0x00EC	LATIN SMALL LETTER I WITH GRAVE
0xED	0x00ED	LATIN SMALL LETTER I WITH ACUTE
0xEE	0x00EE	LATIN SMALL LETTER I WITH CIRCUMFLEX
0xEF	0x00EF	LATIN SMALL LETTER I WITH DIAERESIS
0xF0	0x00F0	LATIN SMALL LETTER ETH
0xF1	0x00F1	LATIN SMALL LETTER N WITH TILDE
0xF2	0x00F2	LATIN SMALL LETTER O WITH GRAVE
0xF3	0x00F3	LATIN SMALL LETTER O WITH ACUTE
0xF4	0x00F4	LATIN SMALL LETTER O WITH CIRCUMFLEX
0xF5	0x00F5	LATIN SMALL LETTER O WITH TILDE
0xF6	0x00F6	LATIN SMALL LETTER O WITH DIAERESIS
0xF7	0x00F7	DIVISION SIGN
0xF8	0x00F8	LATIN SMALL LETTER O WITH STROKE
0xF9	0x00F9	LATIN SMALL LETTER U WITH GRAVE
0xFA	0x00FA	LATIN SMALL LETTER U WITH ACUTE
0xFB	0x00FB	LATIN SMALL LETTER U WITH CIRCUMFLEX
0xFC	0x00FC	LATIN SMALL LETTER U WITH DIAERESIS
0xFD	0x00FD	LATIN SMALL LETTER Y WITH ACUTE
0xFE	0x00FE	LATIN SMALL LETTER THORN
0xFF	0x00FF	LATIN SMALL LETTER Y WITH DIAERESIS

Index