

438M
PROGRAMMING
GUIDE

PART NUMBER 880043-0101

Revised: June 22, 2021 CPC

© Copyright 2021 by Microcom Corporation, Lewis Center, Ohio – All rights reserved.

Printed in the United States of America

This Page Intentionally Left Blank

This Page Intentionally Left Blank

Proprietary Statement

This manual contains information proprietary to Microcom Corporation. This information is intended solely for the use of parties operating and maintaining such equipment described herein.

Product Enhancements

Microcom Corporation is committed to the continual improvement of performance and quality in our products. For this reason, specifications are subject to change without notice.

Liability Disclaimer

Microcom Corporation makes every effort to assure that all information and specifications contained in this manual are accurate; however, mistakes are sometimes made. Microcom Corporation shall not be liable for any damages resulting in the use or misuse of this product. The exclusion or limitation involving consequential or incidental damage does not apply to all states; therefore limitation mentioned above may or may not apply.

FCC Compliance Statement

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy, and if not installed and used in accordance with the instructions contained in this manual, may cause harmful interference to radio communications.

SolarWinds® is a registered trademark of SolarWinds.Net, Inc.

Tera Term Pro® is a registered trademark of T. Teranishi.

HyperTerminal® and HyperAccess® are trademarks of Hilgraeve Inc.

Centronics® is a registered trademark of Data Computer Corporation.

Procomm Plus® is a registered trademark of Intuitive Communications.

Arial® is a registered trademark of The Monotype Corporation.

TrueType® is a registered trademark of Apple Computer, Inc.

Microsoft®, Windows 95®, Windows 98®, Windows 2000®, Windows NT®, Windows Notepad® are registered trademarks of Microsoft Corporation.

IBM® products and name are registered trademarks of International Business Machines Corporation.

Other products and company names mentioned herein may be trademarks of their respective owners.

Table of Contents

INTRODUCTION.....	1
CHAPTER 1: DESIGNING USING 438M	3
1.1 SPECIAL CONTROL CODES	3
1.2 SCRIPT CREATION: AN OVERVIEW.....	4
1.3 BEGIN SCRIPT - ^A)FILENAME	6
1.4 SCRIPT HEADER	6
1.5 FORMAT ENTRY COMMAND - ^FN).....	9
1.6 TEXT ENTRY COMMAND - ^TN).....	16
CHAPTER 2: PRINTER COMMANDS.....	19
2.1 SCRIPT MANAGEMENT COMMANDS.....	19
2.1.1 Begin Script - ^A)filename	19
2.1.2 Format Entry Command - ^Fn)XB,YB,CI,SW,SH,AI,DN,FO,FJ,FW,CS,FC,CC	20
2.1.3 Text Entry Command - ^Tn)text data.....	20
2.2 2XX SERIES COMMANDS.....	20
2.2.1 ^D200)LSX,LSY,GAP,DRM,SPD,DET,OFX,OFY (Script Header).....	20
2.3 3XX SERIES COMMANDS.....	21
2.3.1 ^D300)n (Print Command).....	21
2.3.2 ^D313)n1,n2 (Advance Media)	21
2.3.3 ^D323)n (List Enablements)	21
2.3.4 ^D324)n (List Statistics).....	21
2.3.5 ^D326)n (List Settings)	21
2.3.6 ^D327)1 (List Status).....	22
2.3.7 ^D340)filename (Load/Save File)	22
2.3.8 ^D341) (List Catalog).....	22
2.3.9 ^D342)filename (Delete File).....	22
2.3.10 ^D343)filename (List Script)	23
2.3.11 ^D390)1 (Reset Printer).....	23
2.3.12 ^D391)1 (Restore Configuration Defaults).....	23
2.3.13 ^D392)1 (Restore Property Defaults).....	23
2.4 5XX SERIES COMMANDS.....	23
2.4.1 ^D506)n (Select Cutter Type).....	23

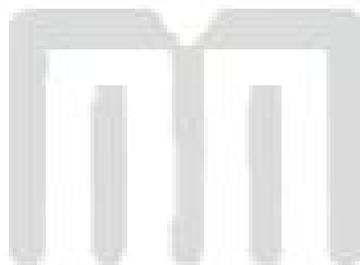
2.4.2	^D508)n (Select Label Present Detector Type).....	23
2.4.3	^D509)n (Select INPUT1 Response Polarity and Action)	23
2.4.4	^D550)n (Select Printer Response Set)	24
2.4.5	^D564)n (Select Measurement Units)	25
2.4.6	^D567)filename (Select Power-up Script).....	25
2.5	6XX SERIES COMMANDS	25
2.5.1	^D631)n (Set Media Width).....	25
2.5.2	^D632)n (Set Media Height).....	25
2.5.3	^D633)n (Set GAP Size)	25
2.5.4	^D634)n (Distance To Registration Mark).....	25
2.5.5	^D635)n (Set Print Speed).....	26
2.5.6	^D636)n (Set Detection Mode).....	26
2.5.7	^D642)n (Set Advance/Retract Distance)	28
2.5.8	^D643)n (Set Print Head to LPD Distance)	28
2.5.9	^D645)n1 (Set Advance/Retract Speed)	28
2.5.10	^D647)dc,dp (Set Print Head Parameters).....	28
2.5.11	^D648)n (Set Blackline Sensor to Print Head Distance)	29
2.5.12	^D649)n (Set Gap Sensor to Print Head Distance).....	29
2.5.13	^D655)n (Set Labels Printed).....	30
2.5.14	^D680)n (Set Gap Threshold)	30
2.5.15	^D681)n (Set Blackline Threshold)	30
2.5.16	^D682)n (Set Stock Sensor Threshold)	30
2.5.17	^D690)n (Set Darkness Range)	30
2.6	7XX SERIES COMMANDS	30
2.6.1	^D720)n (Enable Auto Power-Up Script)	31
2.6.2	^D721)n (Enable Auto Enquiry After Print).....	31
2.6.3	^D726)n (Enable Auto ACK/NAK).....	31
2.6.4	^D727)n (Enable Auto Error Reporting)	31
2.6.5	^D735)n (Enable Immediate Graphic Printing).....	31
2.6.6	^D736)n (Enable Immediate Graphic Centering)	32
2.6.7	^D740)n (Enable Cutter)	32
2.7	9XX SERIES COMMANDS	32
2.7.1	^D945)n (List Diagnostic Dump)	32

CHAPTER 3: UPDATING FLASH MEMORY	33
3.1 COMPRESSED FILE UPDATE.....	33
CHAPTER 4: SPECIAL FEATURES	35
4.1 REVERSE VIDEO	35
4.2 LINE DRAW.....	36
CHAPTER 5: BAR CODES	39
5.1 GENERATING BAR CODES.....	39
5.1.1 CI Parameter	39
5.1.2 SW Parameter	39
5.1.3 SH Parameter.....	39
5.1.4 AI Parameter	39
5.2 LINEAR BAR CODES	39
5.2.1 Codabar (Rationalized).....	39
5.2.2 Code 128 (Manual Compression).....	40
5.2.3 Code 128 (Automatic Compression).....	41
5.2.4 Code 39.....	41
5.2.5 Code 93.....	42
5.2.6 Code I2of5 (Interleaved 2 of 5).....	42
5.2.7 MSI0 (Modified Plessey).....	42
5.2.8 MSI1 (Modified Plessey).....	42
5.2.9 MSI2 (Modified Plessey).....	42
5.2.10 Postnet	42
5.3 2D BAR CODES	42
5.3.1 Data Matrix	43
5.3.2 PDF-417	47
CHAPTER 6: STATUS RESPONSES	51
6.1 STATUS RESPONSES.....	51
6.2 FATAL ERROR RESPONSES.....	53
6.2.1 >STOCK OUT< (01)	53
6.2.2 >STOCK JAMMED< (03).....	53
6.2.3 >CUTTER JAMMED< (10)	53
6.2.4 >CUTTER INOPERATIVE< (11)	53

6.2.5	>SCRIPT ERROR< (30)	53
6.2.6	>INVALID PARAMETER< (31)	54
6.2.7	>PARSER ERROR< (32)	54
6.2.8	>COMM ERROR< (40)	54
6.2.9	>FLASH ERROR< (51)	54
6.3	BASIC STATUS RESPONSES	54
6.3.1	>OK< (06)	54
6.3.2	>INPUT 1< (21)	54
6.3.3	>RESTARTED< (80)	55
6.3.4	>READY< (81)	55
6.3.5	>STOCK ADVANCED< (87)	55
6.3.6	>STOCK NOT LOADED< (8F)	55
6.3.7	>FLASH MEMORY LOW< (B0)	55
6.3.8	>TEMPERATURE OUT OF RANGE< (D0)	55
6.3.9	>NONEXISTENT COMMAND< (D1)	55
6.3.10	>ZIP FILE ERROR< (D3)	55
6.3.11	>UNSUPPORTED FILE TYPE< (D5)	56
6.3.12	>FILE NOT FOUND< (D6)	56
6.3.13	>FILE IS READ ONLY< (D9)	56
6.3.14	>INADEQUATE FLASH SPACE< (DA)	56
6.3.15	>FONT/GRAPHIC NOT FOUND< (DC)	56
CHAPTER 7: COMMUNICATIONS		57
7.1	RS-232 SERIAL	57
7.1.1	RS-232 Printer Cables	57
7.2	USB PORT	58
APPENDIX A – ASCII CHART		59
GLOSSARY		60
INDEX		63
USER NOTES		65

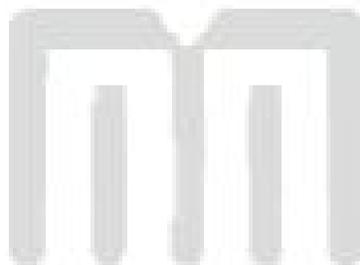
List of Tables

Table 1	Resident Bitmapped Fonts	10
Table 2	Bar Code Index	12
Table 3	Attribute Index	14
Table 4	Field Justifications	15
Table 5	^Dxxx Series Commands	19
Table 6	Print Head Parameters	29
Table 7	Blackline Sensor to Print Head Distance	29
Table 8	Gap Sensor to Print Head Distance.....	29
Table 9	Code 128 Special Function Access.....	40
Table 10	Data Matrix Configuration Parameters	44
Table 11	Data Matrix Encoding Schemes	45
Table 12	ASCII Codeword Values.....	46
Table 13	Data Matrix Special Characters	47
Table 14	PDF-417 Error Correction Level	49
Table 15	Enquiry Responses	52
Table 16	RS-232 Cable Configurations.....	57



List of Figures

Figure 1	Sample Script Format.....	5
Figure 2	Script Parameters.....	7
Figure 3	Distance-to-Registration Mark (DRM).....	8
Figure 4	Format Entry Command.....	9
Figure 5	Text Entry Command.....	17
Figure 6	Detection Mode - None.....	26
Figure 7	Detection Mode - Blackline.....	27
Figure 8	Detection Mode - Die-cut and Blowhole	28
Figure 9	Reverse Video.....	35
Figure 10	Line Draw Reverse Video.....	36
Figure 11	Line Draw Feature.....	37
Figure 12	Two Lines.....	37
Figure 13	Box Creation.....	38
Figure 14	Code 128 Subset Switching	40
Figure 15	Data Matrix Format Structure	43
Figure 16	PDF-417 Format Structure	47

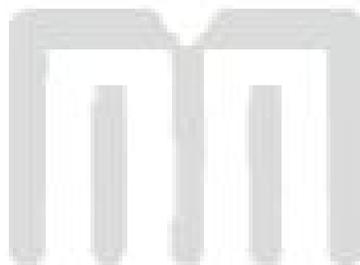


Introduction

This programming guide provides the user with an overview of the 438M command structure and label formatting requirements.

The Label Design Software (438M) is the resident printer control language that is used to create, store and print label scripts, while also controlling the other various printer features.

438M can be used in conjunction with mini-computers, mainframes, personal computers, and most other special purpose computers. The examples provided within this guide use an IBM® compatible personal computer with an available serial port, and a terminal emulation software program (i.e., Procomm Plus, HyperTerminal, HyperAccess or Tera Term Pro). Microcom Corporation printers are shipped with serial parameters set to 115200 baud, no parity, 8 data bits, and 1 stop bit unless setup with a custom configuration. The emulator program will have to be set to these parameters for proper serial communication. 438M printers require the use of hardware flow control (CTS/RTS) for proper data management to occur.



Chapter 1: Designing Using 438M

Label formats or scripts are very easy to design using 438M by following these basic steps below.

1. Determine the media type, media size, and print speed desired for your application; enter this data into the script header.
2. Determine the type of handling operation desired and use the appropriate 438M printer control commands.
3. Define the formatting instructions for the type and placement of data.
4. Enter the text data that corresponds to the formatting instructions.
5. Send the script to the printer.

1.1 Special Control Codes

The printer can accept either the one-character control code (“Ctrl + E”) or the two-character caret (^) plus alpha character (“^E”). Control codes are processed immediately while the ^Dxx commands are processed in the order they are issued. It is recognized that some mainframe and mini-computers cannot use the ASCII carat “^” character. The ASCII pipe symbol “|” can be used as a substitute for these instances. Below is a list of special control codes used by 438M.

- ^A Begin Script:** Marks the beginning of an 438M script file entry.
- ^C Clear Halting Error:** When issued five consecutive times with a one second delay between each ^C the printer will clear blocking or halting error conditions. . Halting error messages are conditions that must be corrected before printing may resume. Refer to section 6.3 for additional information regarding error conditions.
- ^D Standard Command Preface:** Combined with three-digit number issues a command. See Chapter 2 for more information regarding ^D commands.
- ^E Printer Enquiry:** Causes the printer to list its current status to an active communications port.
- ^H Back Space:** Causes the printer to issue a back space.
- ^J Line Feed:** Represents a Line Feed (LF) character or 0x0A.
- ^L Form Feed:** Scrolls one label out of the printer using the previous formats layout.
- ^M Carriage Return:** Represents a Carriage Return (CR) character or 0x0D.

- ^P Print:** Causes the printer to print. *Note: If no label scripts are loaded or stored in the printer's memory, the printer will not print a label until a script is processed.*
- ^R Reset Printer:** Instructs the printer to perform a reset through software that duplicates a hard reset. Issuing the ^R causes the reset to occur immediately, bypassing commands sent but not yet processed. The ^D390)0 command is a queued or serialized version of the ^R and should be used if a reset condition is desired programmatically.
- ^V List Settings:** Causes the printer to list the current configured settings of the printer through an active communications port.
- ^W List Statistics:** Causes the printer to list the current statistics of the printer through an active communications port.
- ^Y List Enablements:** Causes the printer to list the current configured enablements of the printer through an active communications port. .
- ^Z End Script:** Marks the end of a script file. .
- ^] Cancel All Jobs:** Causes all jobs currently queued to be deleted returning the printer to its idle state. Refer to the ^D330 command. .

1.2 Script Creation: An Overview

This section is intended to show a sample script file that will be used to explain the individual components of the script.

A script consists of printer commands, a script header, field formats, and text data. The script is initiated by sending an “^A)” (Begin Script) and continues until a “^Z)” (End Script) has been received. All commands, unless otherwise noted, are terminated with either another command or carriage return (ASCII 0x0D) character. .The printer ignores the LF (0x0A) character so it may be helpful to terminate each line with a CR+LF to assist in the creation and troubleshooting of script files. Script files may contain up to a maximum total of 20,000 characters and up to 1000 different commands. This provides the flexibility in script creation to handle even the most demanding applications.

The sample label script is listed below and will be referred to as the individual components are discussed in the subsequent sections of Chapter 1. The actual programming instructions that have been typed into a text file, using Windows Notepad, are in **bold** typeface. These instructions are followed by a brief explanation shown within the parentheses “()”.

^A)Sample Script

(^A indicates the beginning of the script file followed by an optional script title)

^D200)3.3, 1.9, 0.125, 1.063, 5, 1, 0, 0

(Script header used to define the label format or script file. Units have been entered using inches as the unit of measure in this example.)

^F1)0.15, 1.50, @normal_14

(Field #1 formatting instructions containing text using a normal 14 point font)

^F2)0.30, 0.50, @code128auto, 3, 0.70

(Field #2 formatting instructions containing a Code128 bar code using automatic compression)

^F3)0.60, 0.13, @normal_10

(Field #3 formatting instructions containing text using a normal 10 point font)

^T1)MICROCOM CORPORATION

(Text data for field #1)

^T2)12345678901234567

(Text data for field #2)

^T3)12345 678 90123 45 6 7 8901234

(Text data for field #3)

^D300)1

(Instructs the printer to print one copy once the ^Z instruction is received)

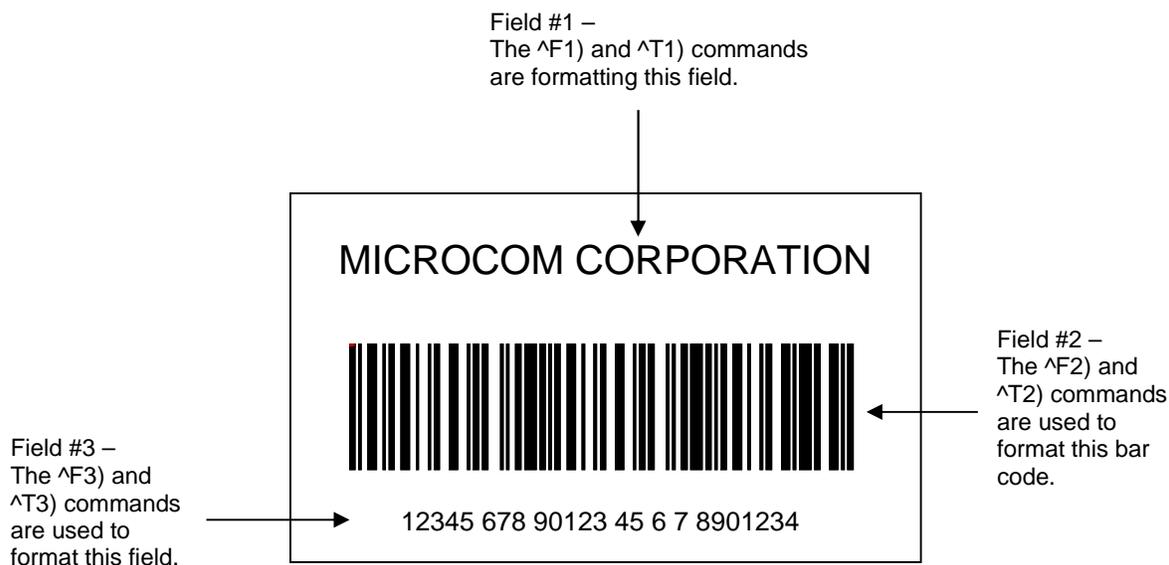
^Z) (Marks the end of the script file)

Figure 1 Sample Script Format

1.3 Begin Script - ^A)filename

This command is used to mark the beginning of a script file. The script may be saved into the printer's non-volatile flash memory if a title or filename is placed after the command. The filename may contain to a maximum of 20 characters in length and include any character from ASCII 0x20 and 0x41 to 0x7A (alphabetical characters plus [\]_'and SP (space character)) with the exception of the caret (^) character. Script filenames can not use a leading zero ("0") or 0x30 characters. A script that contains a "filename" will only save the script and will not process and print. If the script does not contain a "filename", the script is processed normally and will be volatile in nature.

For Example: The previous example script's first line is "**^A)Sample Script**" which would save this script into the printer's flash memory (non-volatile) as "Sample Script".

1.4 Script Header

The script header consists of the "**^D200**") command followed by eight parameters that control the media layout as well as printer operation. The parameters must be separated by a delimiter, comma by default, and are entered using the current unit of measure selected by the "**^D564**") command. The origin or the starting coordinate is always the lower left corner as the media exits the printer. This command may be issued from outside a script but is typically created within a script file.

The script header is used to temporarily override the individual header parameters when issued from within a script file. The parameters may be defaulted to the current persistent value by not entering a value between the delimiters. Each of the header parameters has an "**^D6xx**") command equivalent that when set outside a script file remains persistent.

For Example: "**^D200)LSX,LSY,,DRM**", would cause the "GAP" parameter to be defaulted to the current "**^D633**") value that was set outside a script file.

The persistent commands are helpful in setting the printer up for a basic configuration and then override them within the script. It is recommended that a script header contain values for each and every parameter if access to the persistent command listings is not available.

^D200)LSX, LSY, GAP, DRM, SPD, DET, OFX, OFY

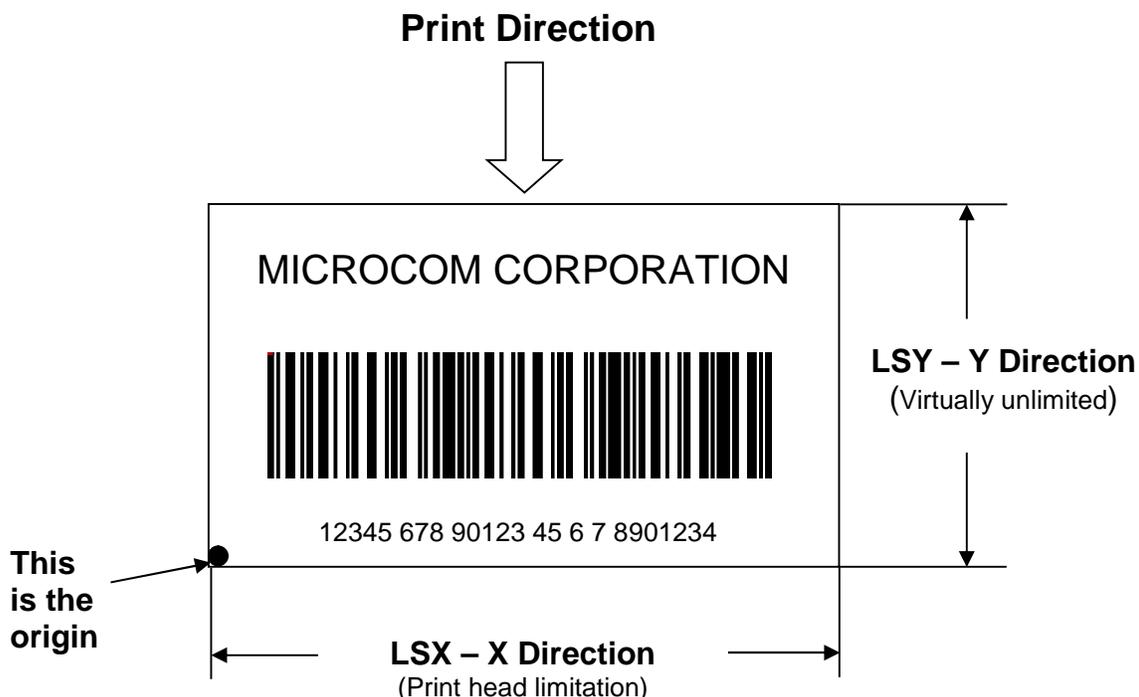


Figure 2 Script Parameters

1.4.1 LSX (Label Width)

LSX, LSY, GAP, DRM, SPD, DET, OFX, OFY

Specifies the width or “X” direction of the label in the unit of measure selected by the “^D564)” command. The maximum LSX is the width of the print head installed in the printer. This parameter is always required for proper script operation. See also ^D631)n command.

1.4.2 LSY (Label Height)

LSX, LSY, GAP, DRM, SPD, DET, OFX, OFY

Specifies the height or “Y” direction of the label in the unit of measure selected by the “^D564)” command. The maximum LSY height is 24”. This parameter is always required for proper script operation. See also ^D632)n command.

1.4.3 GAP (Gap Size)

LSX, LSY, GAP, DRM, SPD, DET, OFX, OFY

The GAP is the height of the registration mark on each label and/or tag. This would be the backing material found between die-cut labels, blowhole sense mark, and/or the blackline height. The value is entered using the current unit of measure selected by the “^D564)” command. See also ^D633)n command.

1.4.4 DRM (Distance to Registration Mark)

LSX, LSY, GAP, DRM, SPD, DET, OFX, OFY

This parameter is typically only used when the registration mark is not located at the edge of the media. The “DRM” is the distance from the leading edge of the media to the beginning of the registration mark. The DRM is entered using the current selected unit of measure selected by the “^D564)” command. See also ^D634)n command.

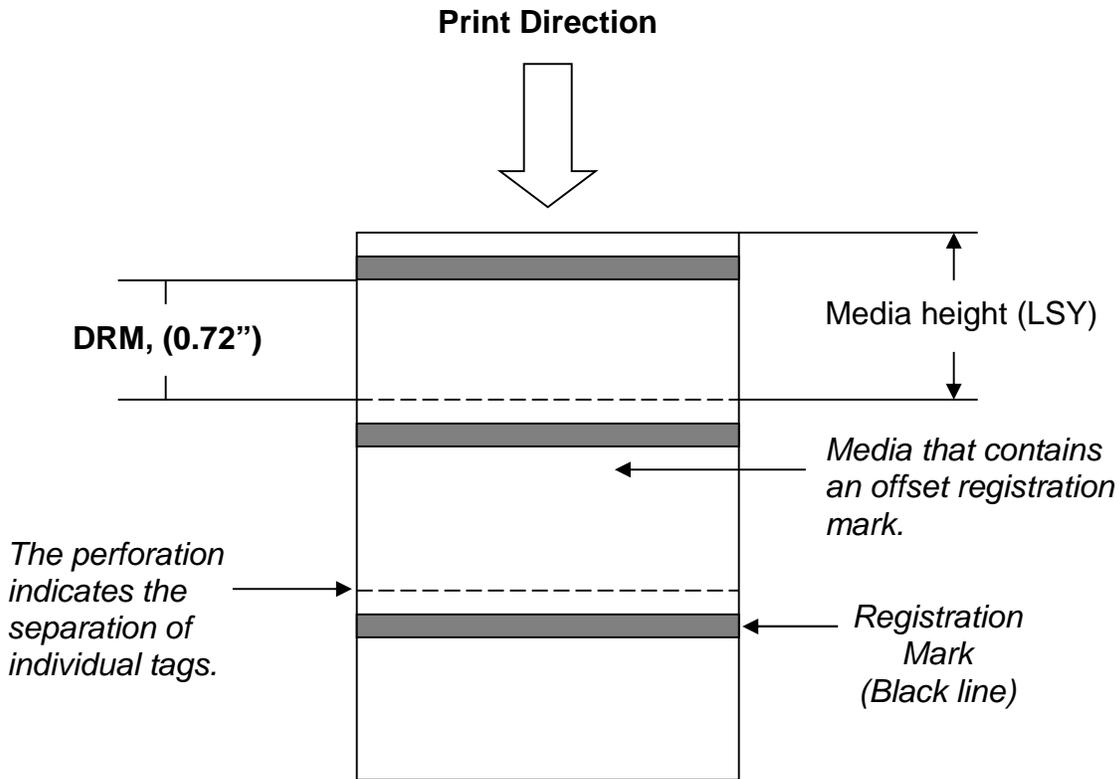


Figure 3 Distance-to-Registration Mark (DRM)

1.4.5 SPD (Print Speed)

LSX, LSY, GAP, DRM, **SPD**, DET, OFX, OFY

This is the speed of the printer while printing, which is defined in "inches/second (ips)" by the "^D564)" command. The print speed can be adjusted anywhere from 1 ips to 6 ips. See also ^D635)n command.

1.4.6 DET (Detection Method)

LSX, LSY, GAP, DRM, SPD, **DET**, OFX, OFY

This parameter selects the type of method the printer uses for detecting registration marks on different media types. A selection of a "0" disables registration detection and is used in conjunction with media that does not contain a registration mark like continuous media. The detectors are only used to detect Stock Out conditions when a selection of "0" is selected. A selection of a "1" uses the reflective detection method and is used when detecting registration marks located on the bottom side of the media. A selection of a "2" uses the transmissive detection method which requires both the upper and the lower gap detectors to be installed on the printer. This method can be used with any type of media but is not recommended when only detecting registration marks located on the bottom side of the media. A selection of a "3" also uses the transmissive detection method and is used in conjunction with media containing a blowhole. The blowhole and die-cut selections differ on how they detect an out of stock condition. See also ^D636)n command.

0	None	(gap sensors are disabled)
1	Blackline	(lower gap sensor is enabled)
2	Die-cut	(upper and lower gap sensors are enabled)
3	Blowhole	(upper and lower gap sensors are enabled)

1.5 Format Entry Command - ^Fn)

The *Format Entry* command is used to define the characteristics, placement and representation of the corresponding text data of the individual fields. This command starts with the “^Fn)” structure which is then followed by thirteen different parameters that define and control the desired field. The “n” is an identifier that represents the field number being formatted and corresponds to the equivalent “^Tn)” line. As with the ^Dxxx commands, multiple parameters must be separated by the comma (“,”) delimiter. The format entry command can be issued from outside a script but is typically created within a script file. Unlike the “^Dxx)” commands and the script header, the Format Entry Command typically does contain default values as there are no equivalent commands for the individual parameters. The default values are discussed under the individual parameter descriptions.

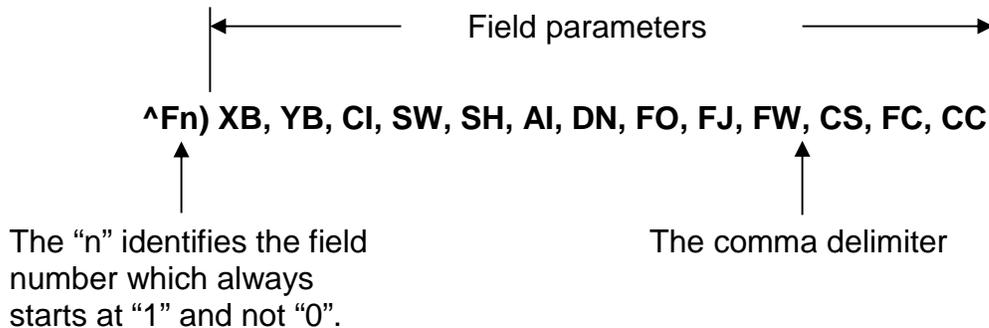


Figure 4 Format Entry Command

1.5.1 XB (X Coordinate)

XB, YB, CI, SW, SH, AI, DN, FO, FJ, FW, CS, FC, CC

This parameter is used to set the X coordinate of the field. The X coordinate is measured from the left edge of the media, as viewed from the front of the printer. The value is entered using the current unit of measure selected by the “^D564)” command. A leading zero before the decimal point is not required but should be used to make trouble shooting scripts easier (i.e., .75 should be entered as 0.75). This parameter does not contain a default value and is required.

1.5.2 YB (Y Coordinate)

XB, YB, CI, SW, SH, AI, DN, FO, FJ, FW, CS, FC, CC

This parameter is used to set the Y coordinate of the field. The Y coordinate is measured from the bottom edge of the media, as viewed from the front of the printer. The value is entered using the current unit of measure selected by the “^D564)” command. A leading zero before the decimal point is not required but should be used to make trouble shooting scripts easier (i.e., .75 should be entered as 0.75). This parameter does not contain a default value and is required.

1.5.3 CI (Conversion Identifier)

XB, YB, CI, SW, SH, AI, DN, FO, FJ, FW, CS, FC, CC

The Conversion Identifier (CI) determines how the associated text field will be rendered. This parameter can be used to select from the available fonts graphics, line draw feature, and/or bar codes that will be used to render the associated Text Entry command line (“^Tn)”). Objects that are stored in the printer’s flash memory are selected using the system character (“@”) and the appropriate text identifier. This parameter can also be used to select other user stored files by entering the name of the saved file into the CI parameter. The entered values for the CI parameter are not case sensitive because the printer will internally translate as lower case.

Resident Bitmapped Fonts

The resident fonts stored in the printer are listed in the table below. To access the resident fonts, enter the appropriate Font Name into the CI parameter. The printer may store up to nineteen bitmapped fonts and custom font sets are available upon request.

CI - Font Name	Font Sample
@normal_06	6 Point – 01234567890ABCDEFGHIJKLMNQPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
@bold_06	6 Point – 01234567890ABCDEFGHIJKLMNQPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
@normal_08	8 Point – 01234567890ABCDEFGHIJKLMNQPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
@bold_08	8 Point – 01234567890ABCDEFGHIJKLMNQPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz
@normal_10	10 Point – 01234567890ABCDEFGHIJKLMNQPQRSTUVWXYZab
@bold_10	10 Point – 01234567890ABCDEFGHIJKLMNQPQRSTUVWXYZa
@normal_12	12 Point – 01234567890ABCDEFGHIJKLMNQPQRSTUVWXYZab
@bold_12	12 Point – 01234567890ABCDEFGHIJKLMNQPQRSTUVWXYZa
@normal_14	14 Point – 01234567890ABCDEFGHIJKLMNQPQRSTU
@bold_14	14 Point – 01234567890ABCDEFGHIJKLMNQPQRST
@normal_16	16 Point – 01234567890ABCDEFGHIJKL
@bold_16	16 Point – 01234567890ABCDEFGHIJKL
@normal_20	20 Point – 01234567890ABCDEFGHIJ
@bold_20	20 Point – 01234567890ABCDEFGHI
@normal_24	24 Point – 0123456ABCDEGabc
@bold_24	24 Point – 012345ABCabcdefgh
@ocra_12	12 Point - 0123456789ABCDEFGHIJKLMNQPQRSTUVWXYZabc
@ocrb_08	8 Point - 1234567890ABCDEFGHIJKLMNQPQRSTUVWXYZabcdefghijklmnopqr
@ocrb_12	12 Point - 1234567890ABCDEFGHIJKLMNQPQRSTUV

Table 1 Resident Bitmapped Fonts

Font File Type Example:

If a normal 8 point font is desired then “@normal_08” should be entered for the CI parameter in order for the proper font to be printed. Fonts are accessed by entering the

system character “@”, font name followed by the underscore character and then the point size into the CI parameter.

Line Draw

The line draw feature is selected by entering “@line” into the CI parameter. The line draw feature easily creates lines by using the XB and YB coordinates as an insertion point for the line to start and then adjusting the SW and SH parameters to define the width and the height of the line.

Line Draw Example:

```
^A)
^D200)3.3, 1.9, 0.125, 1.063, 5, 1
^F1)1.0, 1.0,@line, 1.11, 0.01
^F1)1.0, 1.0,@normal_14
^T1)Line Draw Feature
^D300)1
^Z)
```

This script would cause a line to be inserted an inch from the left side and an inch from the bottom. The line would be 1.11” in width or length (X direction) and would be 0.01” in height (Y direction). The “^Fn)” field must reference a “^Tn)” field for proper operation. The line draw does not have to reference its own specific Text Entry command (“^Tn)” and may reference an existing Text Entry field in the script file as demonstrated in the above example. Refer to *Chapter 4* for additional information regarding the line draw feature.

Bar Codes

The bar codes or symbologies are stored in the printer and can be selected by entering the system character “@” followed by the text equivalent code of the desired symbology in the CI parameter, refer to Table 2 for the available bar codes.

The bar codes that provide optional check digits can be accessed and printed by entering “cs” immediately following the bar code text equivalent code. This is helpful to meet the individual bar code specifications.

For example: If the user wanted to generate a human readable MSI1 that included the check digits, “msi1cs_hr” would then be entered while “msi1_hr” would not include the check digits in human readable form.

See *Chapter 5* for more detailed information pertaining to bar codes and their individual formatting requirements.

CI - Barcode Name	Bar code Type
@codabar	Codabar
@code128	Code 128 using manual compression
@code128auto	Code 128 using automatic compression
@code39	Code 3 of 9

CI - Barcode Name	Bar code Type
@code93	Code 93
@codei2of5	Code I2 of 5
@msi0	Modified Plessey, User provides both check digits
@msi1	Modified Plessey, User provides one check digit
@msi2	Modified Plessey, User provides no check digits
@postnet	POSTNET
@uccean128	UCC/EAN 128
@data	Data Matrix 2D Bar code
@pdf417	PDF417 2D Bar code

Table 2 Bar Code Index

Graphics

The printer is capable of processing monochrome 1-bit bitmaps. If color 1-bit bitmaps are used, it is recommended that the user use Burkes dithering method at the print resolution of the printer. The bitmap graphic does not need a separate converter to convert to the printer's recognized graphic structure as the printer will do this internally. The actual ".bmp" may be saved to the printer's non-volatile flash memory using the "^D340)" command. User definable graphic images are accessed and printed by entering the file name of the appropriate stored graphic, that was used to save the graphic using the "^D340)" command, into the CI parameter.

Graphics may also be sent to the printer in "zipped" (.zip file extensions) form without specifying the file size. The file size would be provided by the zipped files header and should not be included in the command line. This also reduces the transfer time of sending the graphic to the printer improving throughput.

Graphic File Type Example:

If a BMP graphic is saved as "my graphic" then "my graphic" should be entered for the CI parameter in order for the graphic to be printed.

1.5.4 SW (Size Width)

XB, YB, CI, **SW**, SH, AI, DN, FO, FJ, FW, CS, FC, CC

This parameter is used to adjust the width multiple of fonts, bar codes and/or lines.

Bitmapped Fonts

For bitmapped fonts, the SW can be used to adjust the width multiple of the font chosen using the CI parameter. The parameter is entered in as an integer and has a valid range of 1 to 256. The SW default for bitmapped fonts is set a 1.

Bar Codes

The SW has a different effect when used in conjunction with bar codes. When a 0 or 180 degree bar code is selected using the CI parameter, the SW is used to modify the bar code width by a multiple of the narrowest rendition. This means that a value of "1" will produce

the narrowest rendition related to the bar elements while a value of “3” would produce a Bar code that is three times this width. The default value of “1” will be used if a bar code is selected and this field is left blank. For 90 or 270 degree bar codes, the SW field would specify the actual width or height using the current selected unit of measure. The default SW height, 90 and 270 degree rotations, is set a value of 0.5 inches or 12.7mm depending on the current unit of measure (“^D564” command) setting with the default for 0 and 180 degree rotation being set to a 1.

Line Draw

The SW determines the width of a line when the CI field has selected the line draw feature. This is a required field when used in conjunction with the line draw feature.

1.5.5 SH (Size Height)

XB, YB, CI, SW, **SH**, AI, DN, FO, FJ, FW, CS, FC, CC

This parameter is used to adjust the height multiple of fonts, bar codes and/or lines.

Bitmapped Fonts

For bitmapped fonts, the SH can be used to adjust the height of the font chosen using the CI parameter. The parameter is entered in as an integer and has a valid range of 1 to 256. The SH default for bitmapped fonts is set a 1

Bar Codes

The SH has a different effect when used in conjunction with bar codes. When a 90 or 270 degree bar code is selected using the CI parameter, the SH is used to modify the bar code width by a multiple of the narrowest rendition. This means that a value of “1” will produce the narrowest rendition related to the bar elements while a value of “3” would produce a bar code that is three times this width. The default value of “1” will be used if a bar code is selected and this field is left blank. For 0 or 180 degree bar codes, the SH field would specify the actual height of the bar code using the current selected unit of measure. The default SH height, 0 and 180 degree rotations, is set a value of 0.5 inches or 12.7mm depending on the current unit of measure (“^D564” command) setting with the default for 90 and 270 degree rotation being set to a 1.

Line Draw

The SH determines the height of a line when the CI field has selected the line draw feature. This is a required field when used in conjunction with the line draw feature.

1.5.6 AI (Attribute Index)

XB, YB, CI, SW, SH, **AI**, DN, FO, FJ, FW, CS, FC, CC

The attribute index controls the type of effect that is placed on text and/or bar codes. The default is a setting of “0”, which is normal text or a method called OR. This causes new black dots to be placed regardless of existing dots that may be black or white. Also, a white area added to an existing black area does not become white, but the original black dots stay black. The default value may be entered in as “1” or may be omitted by just entering the comma delimiter (“,” would use the default value). The valid values for the AI parameter are listed in Table 3 below.

Attribute Index Value	Conversion / Effect
0	OR (Default)
2	Reverse Video
2:1	Wide to Narrow Bar Ratio
3:1	Wide to Narrow Bar Ratio
4:2	Wide to Narrow Bar Ratio (3:1 bar; 4:2 space)
5:2	Wide to Narrow Bar Ratio
8:3	Wide to Narrow Bar Ratio

Table 3 Attribute Index

Reverse Video

The reverse video function inserts a black rectangle that is one dot larger than the contents of the text string field that will be printed. The result would be that the text would be placed as white on a black background area. This effect is the same as if the image was XOR'ed onto a black box or line. In fact, if the automatic black box is too small, the line draw feature may be used to increase the size of the background black box. Refer to section 4.1 for additional information regarding the reverse video function.

Ratios

The ratios are used in conjunction with the Codabar, Code 39, and I2 of 5 bar codes. This value determines the size of the wide to narrow bar ratio. The narrow bar module is always one dot by default. The actual size of the dot is based on the type of print head installed on the printer. By using the SW and/or SH parameters, it is possible to multiply the narrow bar module and then apply the various ratios to create the desired bar code size.

For example: An 8 dot per mm or 203 dots per inch print head has a dot every 0.0049". This would mean that the narrow bar module by default is 0.0049" and that a 2:1 ratio would result in the wide bars equaling 0.0098". Using the SW and/or SH parameters, depending on the orientation of the bar code, it is possible to multiply and adjust the narrow bar module. This would mean that for a 0 degree bar code, a SW value of 2 (2 x 0.0049) would adjust the narrow bar module to 0.0098" and then applying a 2:1 ratio would produce wide bars of 0.0196"(0.0098 x 2).

1.5.7 DN (Direction)

XB, YB, CI, SW, SH, AI, **DN**, FO, FJ, FW, CS, FC, CC

This parameter is used to select the direction of the text. The only valid parameter for direction is rightwards text. A selection of rightwards text would cause the characters to be printed left to right.

1 Rightwards Direction (default)

1.5.8 FO (Field Orientation)XB, YB, CI, SW, SH, AI, DN, **FO**, FJ, FW, CS, FC, CC

The values entered in this parameter are expressed in degrees. The normal orientation (0°) is right side up as viewed from the front of the printer. The valid arguments are 0, 90, 180 and 270. The rotation is measured counter-clockwise.

0	0 degree	(default)
90	90 degree	
180	180 degree	
270	270 degree	

1.5.9 FJ (Field Justification)XB, YB, CI, SW, SH, AI, DN, FO, **FJ**, FW, CS, FC, CC

The field justification or FJ controls the alignment, kerning, and the proportionality of text and/or bar codes with respect to the type of composition used. Alignment is done side-to-side (left, right, centered, or full justified to span the entire LSY distance) and vertically (on or below baseline, which is an imaginary line extending from the starting coordinates along the direction of the text), relative to the default (0 degree) orientation of text and/or bar codes and the insertion point of the field. This means that whatever the direction and orientation may be; left will always be the left edge of the characters when the media exits the printer.

Kerning is when one character in a proportional font can be slid under another character to achieve a more appealing result. Kerning is enabled and used by default unless a value for the FJ parameter is entered. If a value is entered then kerning would have to be selected by adding "100" to the justification value.

Proportionality refers to whether the characters are all different widths or all the same. A proportional font has characters that are mostly different widths. When selecting the alignment value, text and/or bar codes will be aligned from the XB and YB insertion point. The justified function, to be aligned, can be specified by using a combination of the insertion point and the FW (Field Width) parameter so that the text is spaced to run this entire distance.

The desired alignment value can be determined from the table below:

	Composition		Horizontal Alignment			
	<i>Horizontal</i>	<i>Vertical</i>	<i>Left</i>	<i>Center</i>	<i>Right</i>	<i>Justified</i>
Vertical Alignment	Above Baseline	Top	11	12	13	14
	Below Baseline	Bottom	31	32	33	34

Add 100 for kerning and add 200 for mono-spacing to the FJ value to produce effects.

Table 4 Field Justifications

For Example: A typical European language would be left horizontal and normal vertical alignment, which would be a value of "11" for the "FJ" parameter. If kerning was also desired, the proper value would then be "111" and for mono spacing it would be "211".

1.5.10 FW (Field Width)XB, YB, CI, SW, SH, AI, DN, FO, FJ, **FW**, CS, FC, CC

The field width parameter sets the width of a text field, using the current unit of measure, along the direction of composition. This is used to limit the physical width of the cell where the appropriate data is to be placed. If the desired data is longer than the field width setting, then characters outside the cell width setting will not be printed

1.5.11 CS (Character Spacing)XB, YB, CI, SW, SH, AI, DN, FO, FJ, FW, **CS**, FC, CC

This parameter adjusts the spacing between each character or the inter-character spacing. The change is made relative to a default, where omitting this parameter or entering a "0" causes the default to be used. The specified spacing value is in units of typesetting points, where 1 point = 1/72". Positive point values increase the default spacing while negative values reduce the default spacing. The point values are accurate to a single digit following the decimal place (1.4 for example).

1.5.12 FC (First Character)XB, YB, CI, SW, SH, AI, DN, FO, FJ, FW, CS, **FC**, CC

The FC parameter is used to select the first character to be used as data within the text string data. This parameter is helpful in allowing several fields to use sections of the same text string minimizing the amount of data transmitted. If the value specified for the FC parameter is equal to or greater than the included data, the entire data string will be printed. For Example, for the text string **0123456789**, a FC value of 5 would start the data on the fifth character and the resulting data would be **456789**. This parameter is typically used in conjunction with the CC (Character Count) parameter listed below.

1.5.13 CC (Character Count)XB, YB, CI, SW, SH, AI, DN, FO, FJ, FW, CS, FC, **CC**

This parameter is used to determine the number of characters that will be used as data within a text string. If the number of characters in the selected text string is more than the quantity specified by the CC, then the remainder of the text string is ignored. If this parameter is left blank then the entire text string will be used as the data. Using the same example described in section 1.4.12, a CC value of 2 in conjunction with a FC value of 5, would result in a value of **45** to be used as the data.

1.6 Text Entry Command - ^Tn)

The *Text Entry* command is used to define the actual data to be rendered using the output identified by the CI parameter. This command starts with the "^Tn)" structure and then is followed by the subject text or data. The "n" is an identifier that represents the subject text lines and will use the formatting that has been placed in the corresponding "n" of the format entry command ("Fn").

For features like graphics and/or line draw, this command is simply a placeholder for the equivalent "^Fn)" command, so any text may be entered. When printing graphics, text containing the graphic name or simply the word graphic should be entered and can be helpful for debugging purposes. The same would be true when using the line draw feature by entering the word "line" in the "^Tn)", this also assists in debugging easier.

Note: To print either the caret ("^") or the pipe ("|") characters simply enter them twice (example - "^").

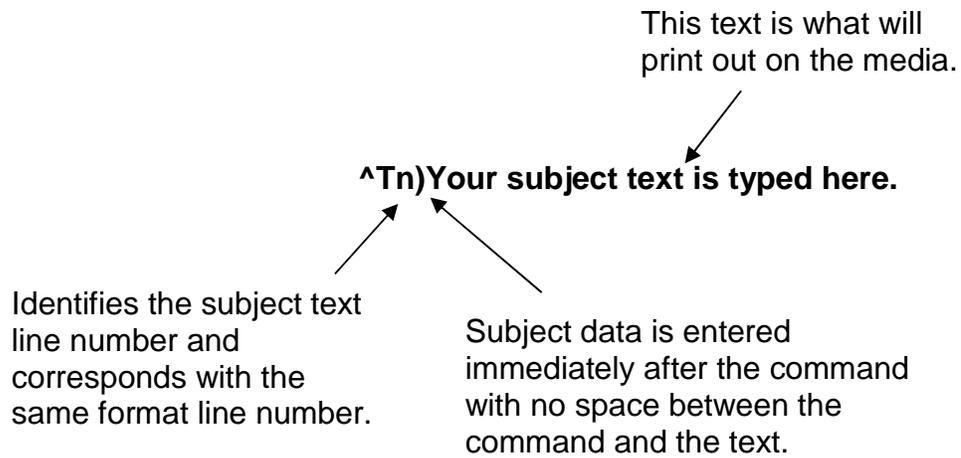
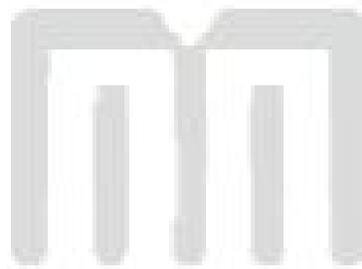


Figure 5 Text Entry Command



Chapter 2: Printer Commands

The script management commands may be issued inside a script file while the ^Dxx printer control commands may be issued from inside or outside a script file. Commands will be persistent and non-volatile if issued from outside a script file and temporary if issued from inside a script. Commands that are issued from inside a script file will override the persistent setting during the processing of the script file and then return to the value of the previous persistent command. The commands require a “)” or ASCII SPACE (0x20) between the command and the first parameter or argument. Commands that contain multiple parameters must be separated with the comma delimiter. The printer control commands follow the standard command preface (“^Dxxx”) structure and are divided into eight sub-categories. These categories are identified by the “hundreds” digit as listed below. The commands are also executed in the order that they are issued. Commands will return the current status and/or setting if the ^Dxxx command is entered without an argument. Entering a question mark (“?”) after the ^Dxxx command causes the printer to report the commands current setting as well as the verbose description and help text associated with that particular command.

^Dxxx Series Number	Series Number Command Definition
2xx	Formatting Commands
3xx	Operation Commands
4xx	Volatile Selection and Setting Commands
5xx	Configurable Selection Commands
6xx	Configurable Setting Commands
7xx	Configurable Enablement Commands
8xx	<i>Reserved for future use.</i>
9xx	Self Test and Factory Commands

Table 5 ^Dxxx Series Commands

2.1 **Script Management Commands**

The script management commands deal specifically with scripts and control how scripts are processed and executed. These commands are only valid when sent inside a script file. Refer to the previous chapter for detailed description on working with script files and their formatting requirements.

2.1.1 **Begin Script - ^A)filename**

This command is used to mark the beginning of a script file. The script may be saved into the printer’s non-volatile flash memory if a title or filename is placed after the command. The filename may contain to a maximum of 20 characters in length and include any

character from ASCII 0x20 and 0x41 to 0x7A (alphabetical characters plus [_]' and SP (space character)) with the exception of the caret (^) character. Script filenames can not use a leading zero ("0") or 0x30 characters. A script that contains a "filename" will only save the script and will not process and print. If the script does not contain a "filename", the script is processed normally and will be volatile in nature.

2.1.2 Format Entry Command - ^Fn)XB,YB,CI,SW,SH,AI,DN,FO,FJ,FW,CS,FC,CC

The *Format Entry* command is used to define the characteristics, placement and representation of the corresponding text data of the individual fields. This command starts with the "^Fn)" structure which is then followed by thirteen different parameters that define and control the desired field. The "n" is an identifier that represents the field number being formatted and corresponds to the equivalent "^Tn)" line. As with the ^Dxxx commands, multiple parameters must be separated by the comma (",") delimiter. The format entry command may be issued from outside a script but is typically created within a script file. Unlike the "^Dxx)" commands and the script header, the Format Entry Command typically does contain default values as there are no equivalent commands for the individual parameters. The default values are discussed under the individual parameter descriptions. Refer to section 1.5 for the individual parameters detailed descriptions.

2.1.3 Text Entry Command - ^Tn)text data

The *Text Entry* command is used to define the actual data to be rendered using the output identified by the CI parameter. This command starts with the "^Tn)" structure and then is followed by the subject text or data. The "n" is an identifier that represents the subject text lines and will use the formatting that has been placed in the corresponding "n" of the format entry command ("Fn)").

For features like graphics and/or line draw, this command is simply a placeholder for the equivalent "^Fn)" command, so any text may be entered. When printing graphics, text containing the graphic name or simply the word graphic should be entered and can be helpful for debugging purposes. The same would be true when using the line draw feature by entering the word "line" in the "^Tn)", this also assists in debugging easier.

Note: To print either the caret (^) or the pipe (|) characters simply enter them twice (example - ^^).

2.2 2xx Series Commands

The 2xx series commands control the formatting commands. Commands will return the current status and/or setting if the ^Dxx command is entered without an argument

2.2.1 ^D200) LSX, LSY, GAP, DRM, SPD, DET, OFX, OFY (Script Header)

This command is followed by eight parameters that control the media layout as well as printer operation. The beginning parameter (LSX) must be separated from the ^D command with a ")" character and each following parameter must be separated by the delimiter, which is the comma (",") by default. It is recommended that since the header is used to identify a certain media configuration that all header parameters are included in a script file. Refer to section 1.4 for additional information.

2.3 3xx Series Commands

The 3xx series commands control basic operation and list commands. Commands will return the current status and/or setting if the ^Dxx command is entered without an argument. The ^D3xx series commands are also required to be sent each and every time because persistent or values set outside a script file are not retained. Entering a question mark (“?”) after the ^Dxx command causes the printer to report the commands current setting as well as the verbose description and help text associated with that particular command.

2.3.1 ^D300)n (Print Command)

This command will print a label or a batch of labels where “n” controls how many labels of the identical script are printed. When this command is used within a script file, it should be placed at the end of the ^D commands. The “^D300)” command is equivalent to the ^P (ASCII 0x10) control code.

2.3.2 ^D313)n1,n2 (Advance Media)

This command is used to advance the media by a programmable distance (n1) at a programmable speed (n2). The distance and speed parameters are specified in the current selectable units (“^D564)” and “^D565”) commands. A negative distance for the n1 parameter causes the printer to retract the specified amount at the speed entered with the n2 parameter.

2.3.3 ^D323)n (List Enablements)

The “^D323)” command will list the printers current configurable enablements along with each of the individual enablement specific ^D7xx command. This command should only be issued from outside a script file and not issued in a script file. The valid parameters for “n” are listed below. If the ^D command is issued without a parameter, the default will be used. The “^D323)” command is equivalent to the ^Y (ASCII 0x19) immediate control code.

1 through active port (default)

2.3.4 ^D324)n (List Statistics)

This command is used to report the printer’s current status such as the firmware version, communication parameters, memory information, and basic usage statistics. The “^D324)” command is equivalent to the ^W (ASCII 0x17) control code.

1 through active port (default)

2.3.5 ^D326)n (List Settings)

The “^D326)” command lists the printers current settings along with each of the individual setting specific ^D6xx) commands. This command should only be issued outside a script file. The valid parameters for “n” are listed below. If the ^D command is issued without a parameter, the default will be used. The “^D326)” command is equivalent to the ^V (ASCII 0x16) control code.

1 through active port (default)

2.3.6 **^D327)1** (List Status)

The “^D327)1” command will report the printers current status through the active port. When the indicator light on the front panel changes from green to amber or red, this command will provide information of the current status (i.e., >STOCK OUT<, >READY<,). The actual response the printer sends depends on the printer response set configuration, refer to the “^D550)” command for additional information. This command may be issued inside or outside a script file. The “^D327)1” command is the queued or serialized equivalent to the ^E (ASCII 0x05) immediate control code. Refer to Chapter 6 for the available status messages.

2.3.7 **^D340)filename** (Load/Save File)

The “^D340)” command causes the printer to take the ensuing data transfer as a binary file upload from the host. The file will then be permanently stored in the printer’s flash memory until it is explicitly erased. The printer will currently accept the following file types: 1-bit BMP graphics, and MCF’s (Microcom Corporation Converted Font). The file transfer can be sent in uncompressed or PKZIP compatible compressed formats. The ^D340) command has three parts, which are name, file size, and comment. Both the name and file size are required while the comment is optional.

The filename and/or comment may contain to a maximum of 20 characters in length and include any character from ASCII 0x20 and 0x41 to 0x7A (alphabetical characters plus [] _ ’ and SP (space character)) with the exception of the caret (^) character. Script filenames can not use a leading zero (“0”) or 0x30 characters.

When saving MCF files, the chosen name should include the “_” (underscore character) followed by the point size; see the examples below.

Command Syntax: ^D340)filename, file size, comment (optional)

Example 1: “^D340)swiss721_10,3092”

Example 2: “^D340) graphic, 2862, my graphic”

Example 2 would save a BMP file titled “graphic” and a comment “my graphic” into the printer’s flash memory. The printer will automatically detect the file type of the ensuing data and save it as such.

2.3.8 **^D341)** (List Catalog)

This command is used to list the files stored in the printer’s flash memory. The list reports the files name (user file CI), any associated comments, file type, and the file size of all file types. This command should only be issued from outside a script file and not entered from a script file. Issuing this command with the “?” as the argument also causes the listing to be generated.

2.3.9 **^D342)filename** (Delete File)

This command is used to delete a stored font, graphic, or script. The filename must match a saved file name. This would include any space characters that were entered when saving the file. The “^D341)” command is useful in determining the exact filename and the valid wild card structure is listed below.

2.3.10 ^D343)filename (List Script)

This command lists the contents of the specified script that has been saved in the printer's flash memory to the active port.

2.3.11 ^D390)1 (Reset Printer)

This command instructs the printer to perform a reset through software, which duplicates a hard reset. It is a good idea to issue an Enquiry command (“^D327)1” or ^E) before sending data after a reset to verify that the printer has returned to a >READY< status. The “^D390)1” command is the queued equivalent to the immediate ^R (ASCII 0x12) control code.

2.3.12 ^D391)1 (Restore Configuration Defaults)

This command is used to restore configuration defaults to a known state. This command may be sent from inside or outside a script. This command should be sent to the printer outside a script file as a part of a Flash memory update.

2.3.13 ^D392)1 (Restore Property Defaults)

This command is used to restore the printer's property command defaults to a known state.

2.4 5xx Series Commands

The 5xx series commands control the printer's configurable selections and may be issued from outside and/or inside a script file. Commands will return the current status and/or setting if the ^Dxxx command is entered without an argument. Entering a question mark (“?”) after the ^Dxxx command causes the printer to report the commands current setting as well as the verbose description and help text associated with that particular command.

2.4.1 ^D506)n (Select Cutter Type)

This command is used to select the type of cutter that is installed on the printer. The valid parameters for “n” are listed below. The printer has to be reset and/or a power cycle must occur after setting this command. To adjust the dispense distance for the available cutters, refer to the “^D642)” command.

0	None
2	Rotary Cutter

2.4.2 ^D508)n (Select Label Present Detector Type)

This command is used to control the optional INPUT 2 sensor. These settings configure how and when the >INPUT 2< or 0x22 message is reported. The valid parameters for “n” are listed below.

0	None
4	Send only a message when INPUT2 signal is active (LOW)
5	Send only a message when INPUT2 signal is inactive (HIGH)

2.4.3 ^D509)n (Select INPUT1 Response Polarity and Action)

The “^D509)” command selects whether the printer issues an >INPUT 1< or 0x21 message, depending on the “^D550)” command setting, when the optional INPUT 1 sensor's signal is active (LOW) , or when it is inactive (HIGH), and whether it sets an error or just sends a message through the printer's active port.

- 0 No response when INPUT 1 signal changes
- 1 Send only a message when INPUT 1 signal is active (LOW)
- 2 Send only a message when INPUT 1 signal is inactive (HIGH)
- 3 Assert an error when INPUT 1 signal is active (LOW)
- 4 Assert an error when INPUT 1 signal is inactive (HIGH)

2.4.4 ^D550)n (Select Printer Response Set)

This command is used to select among the available response sets the printer offers. The response sets determine how the printer will react to commands, functions, and enquiry responses. Issuing a ^Dxx command without a parameter will report the commands current setting regardless of the selected response set. Descriptive text is reported by issuing the question mark character (“?”) after a ^Dxx command and is available in all response sets.

For Example: Issuing the “^D550)? <CR>” command sequence would cause the printer to report text that describes the commands current setting as well as the commands operation and valid settings.

Refer to the following sample dialog:

```

^D550 n -- Selects response mode
      1 - TEXT -- Text responses (default)
      2 - BYTE -- Byte-encoded responses
      9 - VERBOSE -- Text responses with command help text
      and confirmations of settings
Current setting is: 1.
Use a selection index to change the setting.

```

Text: If text responses are selected, the printer responds with simple text responses to enquiries and error conditions. Commands that are issued without an argument will return that commands current setting without descriptive text while in the text mode response set. The printer is capable of multiple line responses so the final response is terminated with a “CR LF” or “0x0D 0x0A” sequence to assist in programming.

Byte: When the byte response set is selected, the printer returns all responses in binary form. Commands that are issued without an argument will return that commands current setting without descriptive text while in the byte mode response set. The printer is capable of multiple line responses so the final response is terminated with the ^Z or 0x1A character to assist in programming.

Verbose: If verbose responses are selected, the printer will include elaborate text help about commands when that command is issued without an argument, a summary of any action that has just resulted from a command, and listings of printer parameters, where the listed values are each identified by descriptive text. This mode is very helpful when learning how to use the printer for the first time and is not intended to be used for actual programming.

- 1 Text (Default)
- 2 Byte
- 9 Verbose

2.4.5 **^D564)n (Select Measurement Units)**

This command is used to select the prevailing unit of measure the printer will use. Some commands and/or calculations use a constant unit of measure and will not change with the issuance of this command. These commands and/or calculations that use a constant unit of measure are noted in their respective descriptions. This is helpful in forcing a unit of measure for troubleshooting and support capabilities.

- | | |
|---|-------------|
| 1 | Inches |
| 2 | Millimeters |

2.4.6 **^D567)filename (Select Power-up Script)**

The “^D567)” command is used to select the script that is processed on a printer power-up. It is used in conjunction with the “^D720)” command to automatically process the selected script on a power-up. Issuing the “^D567)” without a parameter builds and lists the scripts that have been saved into the printers flash memory. The filename may contain to a maximum of 20 characters in length and include any character from ASCII 0x20 and 0x41 to 0x7A (alphabetical characters plus []_ ’and SP (space character)) with the exception of the caret (^) character. Script filenames can not use a leading zero (“0”) or 0x30 characters.

2.5 **6xx Series Commands**

The 6xx series commands control the configurable settings and may be issued outside and/or inside a script file. Commands will return the current status and/or setting if the ^Dxxx command is entered without an argument. Entering a question mark (“?”) after the ^Dxxx command causes the printer to report the commands current setting as well as the verbose description and help text associated with that particular command.

2.5.1 **^D631)n (Set Media Width)**

This command specifies the width of the media using the current selected prevailing unit of measure selected by the “^D564)” command. The maximum width is the width of the print head installed on the printer. This command is also a parameter in the “^D200)” script header command. See also, LSX.

2.5.2 **^D632)n (Set Media Height)**

This command specifies the height of the media using the current selected prevailing unit of measure selected by the “^D564)” command. The maximum LSY height is 24”. This command is also a parameter in the “^D200)” script header command. See also, LSY.

2.5.3 **^D633)n (Set GAP Size)**

This command sets the height distance, as viewed inside the printer, of the registration mark that identifies the label and/or tag size and is entered using the current selected prevailing unit of measure selected by the “^D564)” command. This would be the backing material found between die-cut labels, blowhole sense mark, and/or the blackline height. This command is also a parameter in the “^D200)” script header command. See also, GAP.

2.5.4 **^D634)n (Distance To Registration Mark)**

This command is only used when the registration mark is not located at the edge of the media. The “DRM” is the distance from the leading edge of the media to the beginning of the registration mark and is entered using the current selected unit of measure. This command is also a parameter in the “^D200)” script header command. See also, DRM.

2.5.5 ^D635)n (Set Print Speed)

The “^D635)” command sets the print speed of the printer using the current selected unit of measure. Parameter “n” has a range of 1-6 inches per second. This command is also a parameter in the “^D200)” script header command. See also, SPD.

2.5.6 ^D636)n (Set Detection Mode)

This command selects the type of method the printer uses for detecting registration marks on the different media types. A selection of a “0” disables the detection of registration marks and is used in conjunction with media that does not contain a registration mark like continuous media. A selection of a “1” uses the reflective detection method and is used when detecting registration marks located on the bottom side of the media. A selection of a “2” uses the transmissive detection method which can be used with any type of media but is not recommended when only detecting registration marks located on the bottom side of the media. A selection of a “3” also uses the transmissive detection method and is used in conjunction with media containing a blowhole. Both upper and lower gap detectors are required for stock out detection regardless of the type of registration mark detection. This command is also a parameter in the “^D200)” script header command. See also, DET.

0	None	(gap sensors are disabled)
1	Blackline	(lower gap sensor is enabled)
2	Die-cut	(upper and lower gap sensors are enabled)
3	Blowhole	(upper and lower gap sensors are enabled)

2.5.6.1 Detection Mode - None (0)

This mode is usually used with continuous media. Even though it is not necessary to sense a registration mark, it is still important that the Upper and Lower Sensors are vertically aligned to sense out-of-stock or stock out conditions.

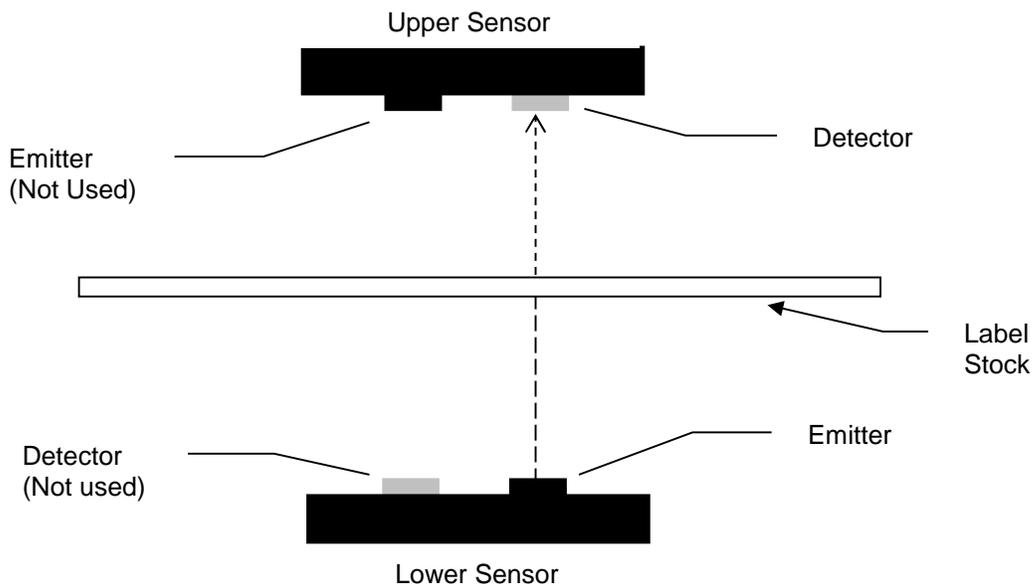


Figure 6 Detection Mode - None

2.5.6.2 Detection Mode - Blackline (1)

This detection mode is used when a black line on the underside of the media is used to register the labels. The Lower Sensor detector “senses” the amount of light reflected off of the label from the Lower Sensor emitter. While registration is determined by the Lower Sensor only in this mode, the upper gap sensor is used for out-of-stock or stock out conditions.

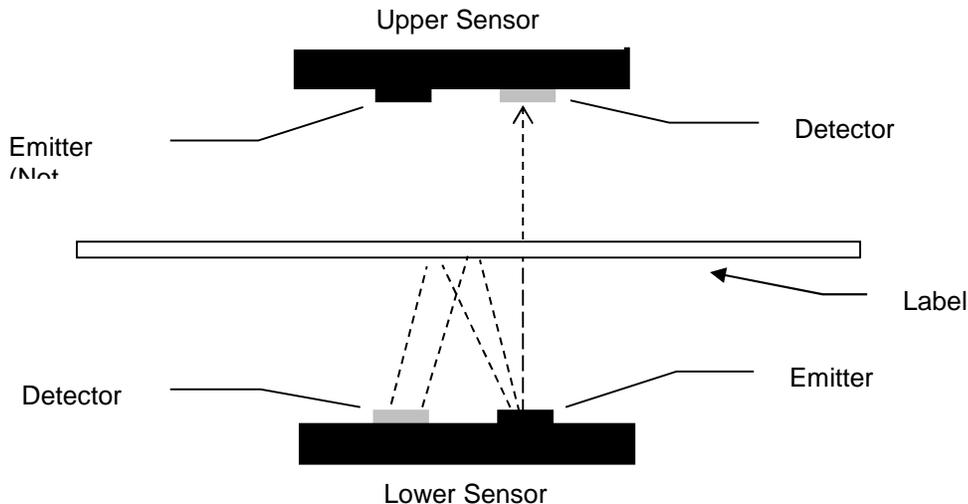


Figure 7 Detection Mode - Blackline

2.5.6.3 Detection Mode - Die-cut (2) and Blow-hole (3)

Die-cut sensing refers to die-cut label registration where the difference in the transparency of [label] and [label + backing] is detected (gap between labels is registration point).

Blow-hole sensing refers to label stock with holes punched out as registration marks. In this case, the Upper and Lower Sensors must be aligned so that the hole in the stock passes over the sensors.

Die-cut and blow-hole use the same basic method of detection. However, because of the difference in light intensity and stock out detection, they each require their own Detection Mode setting (^D636).

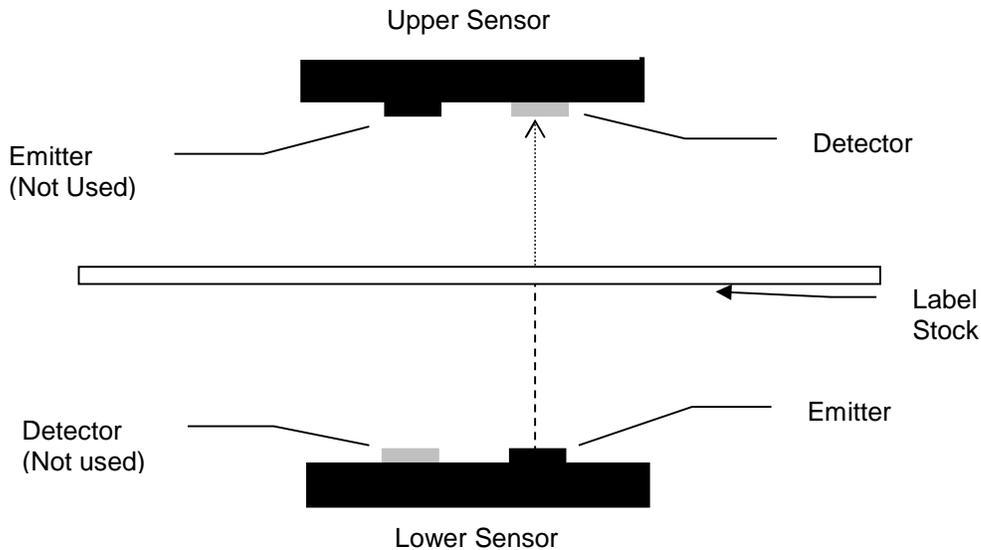


Figure 8 Detection Mode - Die-cut and Blowhole

2.5.7 **^D642)n (Set Advance/Retract Distance)**

This command is used to set the distance the media advances and retracts after printing or when a cutter is installed. The distance “n” is entered in based on the unit of measure selected by the “^D564)” command.

2.5.8 **^D643)n (Set Print Head to LPD Distance)**

This command sets the distance from the print head to the LPD (Label Present Detector) using the unit of measure selected by the “^D564)” command. This distance is the distance media is retracted after reaching the LPD so that the media is properly positioned at the print head. The distance is entered using the current unit of measure selected by the “^D564)” command. This command is a “property” which means that it controls a specific distance that is set at the factory. The property commands may be set back to their default values using the “^D392)1” command.

2.5.9 **^D645)n1 (Set Advance/Retract Speed)**

This command is used to set the speed that the printer advances and retracts while in a “^D597)” dispense mode. The value entered is based on the current unit of measure. If media slippage occurs, the speed should be lowered to reduce this from reoccurring.

2.5.10 **^D647)dc,dp (Set Print Head Parameters)**

The “^D647)” command is used to identify and specify the values for the print head installed on the printer. The “dc” parameter specifies the “dot count” or total number of dots on the print head while the “dp” or “dot pitch” identifies the dots per millimeter value. This command is typically set at the factory and never changed. Refer to the following table for the available print heads and their appropriate values. This command is a “property” which means that it controls a specific distance that is set at the factory. The property commands may be set back to their default values using the “^D392)1” command.

Print Head Part Number	Printer Model Number	dc Parameter	dp Parameter
630020-0000	438M – 203 DPI	832	8
630019-0000	438M – 300 DPI	1280	11.808
630014-0000	438 – 203 DPI	832	8
630017-0000	438 – 300 DPI	1280	11.808

Table 6 Print Head Parameters

2.5.11 ^D648)n (Set Blackline Sensor to Print Head Distance)

The “^D648)” command is used to set the blackline sensor to print head distance at the factory. This command is a “property” which means that it controls a specific distance that is set at the factory. The property commands may be set back to their default values using the “^D392)1” command.

Printer Model Number	Recommended ^D648 Value
438M	1.46”

Table 7 Blackline Sensor to Print Head Distance

2.5.12 ^D649)n (Set Gap Sensor to Print Head Distance)

The “^D649)” command is used to set the gap sensor to print head distance at the factory. This command is a “property” which means that it controls a specific distance that is set at the factory. The property commands may be set back to their default values using the “^D392)1” command.

Printer Model Number	Recommended ^D649 Value
438M	1.46”

Table 8 Gap Sensor to Print Head Distance

2.5.13 ^D655)n (Set Labels Printed)

This command gives the user the ability to set and/or reset the “Labels Printed” counter of the printer that is reported either by issuing the “^D655)” without a parameter or “^D324)” (List Statistics) command.

2.5.14 ^D680)n (Set Gap Threshold)

The “^D680)” command is used to set the Gap threshold. The printer’s default threshold is set to “30” and typically does not require any adjustment. The command is provided in all access modes in the event that this threshold requires adjustment to correct registration issues using different types of die-cut or blow-hole media. The values reported by the “^D900)” Tick-Tick command will increase when sensing the registration mark and decrease when the sensor is looking at label and backing of the media. The threshold should be set to a mid-point between these two points.

2.5.15 ^D681)n (Set Blackline Threshold)

The “^D681)” command is used to set the blackline threshold. The printer’s default threshold is set to “128” and typically does not require any adjustment. The command is provided in all access modes in the event that this threshold requires adjustment to correct registration issues with media using a blackline registration mark. The values reported by the “^D900)” Tick -Tick command will decrease when sensing the black registration mark and increase when the sensor is looking at more reflective portions of the media. The threshold should be set to a mid-point between the black mark and more reflective portion. If the media contains preprint, the threshold should be lowered until proper registration occurs.

2.5.16 ^D682)n (Set Stock Sensor Threshold)

The “^D682)” command is used to set the stock sensor or out of stock detection. The printer’s default threshold is set to “110” and typically does not require any adjustment. The command is provided in all access modes in the event that this threshold requires adjustment to correct the proper detection of stock out conditions. The values reported by the “^D900)” Tick -Tick command will increase when sensing out of stock conditions. The default threshold is set to a point that should detect most stock out conditions. If the media contains preprint, the threshold should be increased until proper registration occurs..

2.5.17 ^D690)n (Set Darkness Range)

The “^D690)” command is used to set the contrast of the printers output and does not correlate to actual print head temperature. The value for “n” is a range from 1 to 100 with a setting of 80 being the printers default. The lowest setting that produces the desired result should be used to extend print head life and performance.

2.6 7xx Series Commands

The 7xx series commands are used to control configurable enablements. These commands may be issued either outside or inside a script file. Commands will return the current status and/or setting if the ^Dxxx command is entered without an argument. Entering a question mark (“?”) after the ^Dxxx command causes the printer to report the commands current setting as well as the verbose description and help text associated with that particular command.

2.6.1 ^D720)n (Enable Auto Power-Up Script)

This command allows scripts stored in the printer's flash memory to be processed on power-up or a reset condition. The printer has a default power-up script, @default, that will be used if a script has not been selected using the “^D567)” command. Once a script has been selected, the “^D720)” command processes the script on power-up as if it was normally received by the printer.

0	Disable
1	Enable

2.6.2 ^D721)n (Enable Auto Enquiry After Print)

The “^D721)” command sends an enquiry or “^D327)1” after each script in the queue has completed printing, or if printing has halted due to an error condition automatically when enabled. Refer to section 6.2 for additional information regarding the possible enquiry responses.

0	Disable
1	Enable

2.6.3 ^D726)n (Enable Auto ACK/NAK)

This command enables the automatic acknowledgement after each command. With this command enabled, the printer automatically issues the command ACK (Human readable descriptive text if in verbose mode, the >OK< response if in text mode, or 0x06 if in byte mode) after a successful command. If a command was not successful, the appropriate error response or NAK would be generated automatically. The responses are sent automatically and do not require sending either the serialized “^D327)1” command or the immediate ^E enquiry command. Refer to section 6.2 for additional information regarding the possible enquiry responses.

0	Disable
1	Enable

2.6.4 ^D727)n (Enable Auto Error Reporting)

The “^D727)” command automatically reports halting errors immediately upon recognition of the error condition when enabled. The “^D550)” commands configuration determines if the error is reported as text or as a byte code. This command automatically reports the error and does not require sending either the serialized “^D327)1” command or the immediate ^E enquiry command.

0	Disable
1	Enable

2.6.5 ^D735)n (Enable Immediate Graphic Printing)

This command enables the immediate printing of any 1-bit (monochrome) BMP file that is received by the printer. The printer places the graphic at dot one or the lower left corner of the media as viewed from the front of the printer. The media size is based either on the last received script or the persistent media layout ^D63x commands if a script has not yet been executed.

0	Disable
1	Enable

2.6.6 ^D736)n (Enable Immediate Graphic Centering)

The “^D736)” command is used to laterally center immediate graphics based on the configured print head size.

0	Disable
1	Enable

2.6.7 ^D740)n (Enable Cutter)

This command is used to control the enablement of a cutter that has been selected using the “^D506)” command. If a cutter is not installed, selected, and this command is enabled, the printer issues the >CUTTER INOPERATIVE< or 0x11 depending on the “^D550)” command configuration.

0	Disable
1	Enable

2.7 9xx Series Commands

The 9xx series commands are reserved for printer self tests, factory configuration, and setup commands. Commands will return the current status and/or setting if the ^Dxxx command is entered without an argument. Entering a question mark (“?”) after the ^Dxxx command causes the printer to report the commands current setting as well as the verbose description and help text associated with that particular command.

2.7.1 ^D945)n (List Diagnostic Dump)

This command causes the printer to list all of the printer’s current setup through the active port and is used for setup and diagnostic purposes. Each line of the listing is terminated with a CR LF (0x0D 0x0A) that may be used to assist in programming delineation while the entire listing is terminated with either an additional CR+LF or 0x1A character depending on the “^D550)” command configuration. The list is generated by issuing the command without the “n” parameter. Issuing the “^D945)” command with the “?” character will always report the “^D550)” commands configuration as “9” for VERBOSE mode because the “?” cause the printer to report the VERBOSE descriptive text. The “n” parameter may be used to select one item from the numbered listing to report only its current value followed by a CR+LF (0x0D+0x0A) used for termination. The “^D945)” command listing is continually updated to provide additional information but all new information is added to the end of the listing for backwards compatibility.

Chapter 3: Updating Flash Memory

The printer's firmware (Microcom Corporation Application; MCA) or Bootloader (Microcom Corporation Bootloader; MCB), which is stored in the flash memory, may be updated using the different methods. The procedures are slightly different depending on the type of update required. This section provides the information necessary to properly update the firmware stored in flash memory.

Both the MCA and MCB code files should always be sent to the printer in the compressed zipped format. The MCB or Bootloader should be updated after the MCA has been updated to avoid any compatibility conflicts.

3.1 Compressed File Update

This method is used to update the firmware stored in flash memory. To update the printer's firmware the following are required: a compressed MCA or MCB (ZIP file), a host computer equipped with a RS-232 serial port, and a Terminal Emulation Program (i.e., HyperAccess, HyperTerminal, Procomm Plus, Tera Term Pro...). The printer is configured at the factory with serial parameters of 115200, n, 8, 1 and the terminal emulation program will need to be set accordingly.

It is a good idea to completely read through the procedure before an update attempt is made. Do not send any unnecessary characters or remove the printer's power during the update or the printer may become inoperable.

1. Make sure the RS-232 serial connection is made between the host and the printer including the proper Baud rate.
2. Contact the Microcom Corporation support department and/or visit the following web address www.microcomcorp.com to acquire the desired MCA file.
3. Send the MCA or MCB file to the printer in compressed form (ZIP) . The MCA must be updated before the MCB if both are being updated. Failure to do this may cause the printer to become unstable.
4. Send the MCA file to the printer in compressed form (ZIP).
5. After the printer has identified the MCA file, the following dialog will be reported to the host: "File received. Unzipping. Please Wait..." This dialog is only reported if the printer is set to use the VERBOSE response set. Refer to the "^D550)" command for additional information regarding the printer response sets.
6. After the file has been unzipped and the checksum verified, the following dialog is reported to the host: "Unzipped. CRC Okay: 0x36410D55" (The checksum shown is for demonstration purposes only. The checksum will reflect the MCA files internal checksum and report any differences automatically.) This dialog is only reported if the printer is set to use the VERBOSE response set. Refer to the "^D550)" command for additional information regarding the printer response sets.

7. Sample MCA update dialog is reported to the host and is listed below. If the MCB code is being updated, the same dialog format would be reported with the word “application” being changed to “bootloader” as the only difference.

```
Application file recognized.  
Erasing old application.....
```

```
Booting application now . . .
```

8. If problems are experienced during the update and the printer has lost the previous application, refer to the following section regarding the recovery file update.
9. After the code has been successfully updated, the “^D391)1”, “^D392)1”, and “^D390)1” commands must be sent to complete the update process. The “^D391)1” and “^D392)1” commands force setting to factory defaults while the “^D390)1” resets the printer and automatically issues an enquiry command on reboot.
10. Once the printer reboots, any custom configuration file and/or commands must be sent to the printer to verify that all commands are set to the desired values before using with the intended application.
11. The “^D945)?” command may now be issued to compare against the previous listing to assist in determining what commands were changed in the update process.



CAUTION: Do not send any unnecessary characters or turn the printer off during the flash memory update. Failure to observe this precaution may result in the printer becoming inoperable.

Chapter 4: Special Features

4.1 Reverse Video

The Reverse Video feature is used to automatically generate a box, using the current foreground color, on top of text. This creates text that is the background color, and is done by setting the "AI" (Attribute Index) field to a "2" in the Format Entry command ("^Fn"). The box can be increased in size by simply overlaying the desired area with a line using the Line Draw feature. The Line Draw format line should be placed before the Reverse Video field to increase the size of the overlaying box.

Auto Reverse Video Example:

```
^A)  
^D200) 3.3, 1.9, 0.125, 1.063, 5, 1  
^F1) 1.0, 1.0, @normal_14, , , 2  
^T1)Reverse Video  
^D300)1  
^Z)
```

← Set to a "2" in the "AI" field to produce Reverse Video.

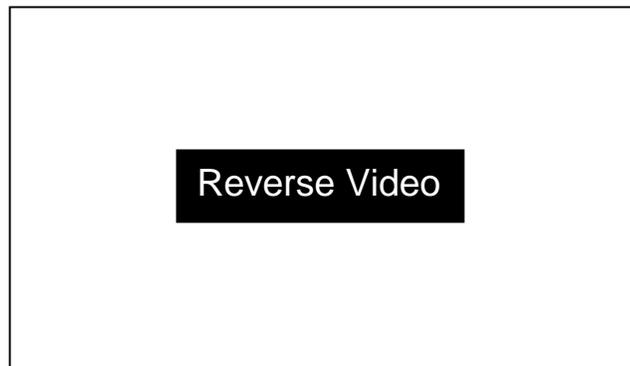


Figure 9 Reverse Video

Line Draw Reverse Video Example:

```

^A)
^D200) 3.3, 1.9, 0.125, 1.063, 5, 1
^F1) 0.25, 0.50, @line, 2.75, 1.00
^F1) 1.0, 1.0, @normal_14, , , 2
^T1)Reverse Video
^D300)1
^Z)

```

← Line Draw feature to produce a larger box for Reverse Video.



Figure 10 Line Draw Reverse Video

4.2 Line Draw

The line draw feature is selected by entering “@line” into the CI parameter of the “^Fn)” Format Entry command. The line draw feature easily creates lines by using the XB and YB coordinates as an insertion point for the line to start. Adjusting the SW and SH parameters determines the width and the height of the line with respect to the rotation or orientation of the FO field.

Line Draw Example:

```

^A)
^D200) 3.3, 1.9, 0.125, 1.063, 5, 1
^F1) 1.0, 1.0, @line, 1.11, 0.01
^F1) 1.0, 1.0, @normal_14
^T1)Line Draw Feature
^D300)1
^Z)

```

This would cause a line to be inserted an inch from the left side and an inch from the bottom. The line would be 0.01” in height and would be 1.11” in width or length. The “^Fn)” field must reference a “^Tn)” field for proper operation. The line draw does not have to reference its own specific Text Entry command (^T) so it could reference an existing Text Entry field in the script file as demonstrated in the above example.

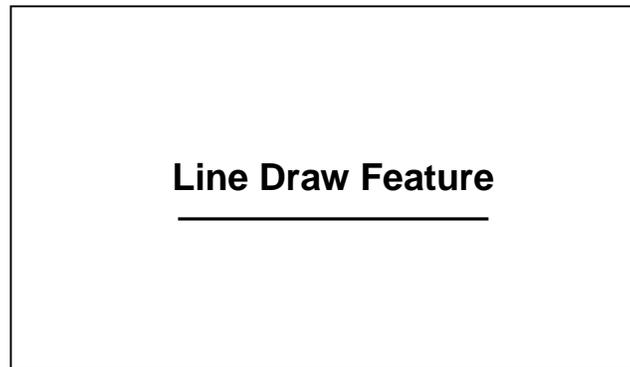


Figure 11 Line Draw Feature

Two Lines Example:

```

^A)
^D200) 3.3, 1.9, 0.125, 1.063, 5, 1
^F1) 1.0, 1.0, @line, 1.11, 0.01
^F1) 1.55,0.25,@line, 0.01, 0.75
^F1) 1.0, 1.0, @normal_14
^T1)Line Draw Feature
^D300)1
^Z)

```

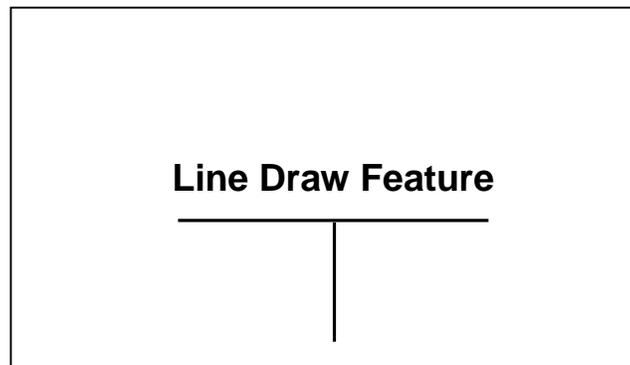


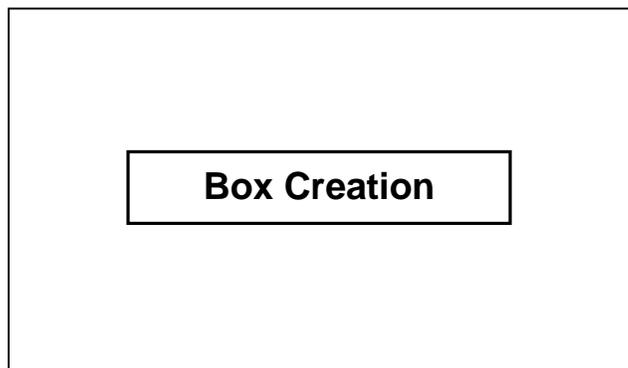
Figure 12 Two Lines

This example shows a line inserted an inch from the left side and an inch from the bottom. This line would be 0.01” in height and would be 1.11” in width or length. A second line is being inserted 1.55” from the left edge and a .25” from the bottom. This line would be 0.75” in height and would be 0.01” in width or length. This demonstrates that the Line Draw feature can be used to draw any amount of lines desired without creating a new “^Tn)” pointer as seen in the example above.

The Line Draw feature can also be used to create boxes. To accomplish this you simply create four lines that connect together to form a box.

Box Creation Example:

```
^A)  
^D200) 3.3, 1.9, 0.125, 1.063, 5, 1  
^F1) 0.90, 1.00, @line, 1.35, 0.01  
^F1) 0.90, 1.25, @line, 1.35, 0.01  
^F1) 0.90, 1.00, @line, 0.01, 0.25  
^F1) 2.24, 1.00, @line, 0.01, 0.25  
^F1) 1.00, 1.00, @normal_14  
^T1)Box Creation  
^D300)1  
^Z)
```

**Figure 13 Box Creation**

Chapter 5: Bar Codes

The purpose of this chapter is to provide information and the formatting requirements of the different bar codes that are stored in the printer's memory.

5.1 Generating Bar Codes

Bar codes that are stored in the printer are generated by using the CI, SW, SH and the AI (when applicable) parameters of the Format Entry Command ("^Fn"). The CI parameter is used to select the text equivalent code of the desired bar code while the SW, SH and AI parameters control the size of the printed bar code.

5.1.1 CI Parameter

The CI or Conversion Identifier selects the bar codes that are stored in the printer by entering the text equivalent code of the desired bar code. Refer to Table 1 or the descriptions below, under the "Bar Code Types" section, for the available bar codes and their equivalent codes. The text equivalent codes may be entered in either uppercase, lowercase, or a combination of the two.

5.1.2 SW Parameter

When used in conjunction with bar codes the SW parameter is used to modify the Bar code width by a multiple of the narrowest rendition. This means that a value of "1" will produce the narrowest rendition related to the bar elements while a value of "3" would produce a Bar code that is three times this width. The default value of "1" will be used if a Bar code is selected and this field is left blank.

5.1.3 SH Parameter

When used in conjunction with bar codes the SH parameter selects the actual height, measured in inches, of the desired bar code. This is a required field when used in conjunction with bar codes.

5.1.4 AI Parameter

Some bar code types combine the use of ratios along with the SW and SH parameters to create the different sizes of printed output. The bar codes that use the different ratios are entered in a ratio format and are listed in the descriptions under the "Bar Code Types" section. The ratio is required for the bar codes that utilize the different ratios and if omitted, the bar code will not be printed.

For example: "3:1" would be entered for the AI parameter for a bar code that allows a three to one ratio with respect to the wide and narrow bars.

5.2 Linear Bar Codes

This section provides information regarding the different requirements of the linear bar codes. Unless stated otherwise in the descriptions below, the printer will automatically generate the necessary check digits when required.

5.2.1 Codabar (Rationalized)

Codabar is a variable length linear symbology that is commonly used in libraries, blood banks, and in the air parcel business. Codabar is capable of encoding the 16 following

characters: 0123456789-\$/ . + and the four start/stop characters ABCD. This symbol uses 2:1, 3:1, 4:2, 5:2, and 8:3 ratios, which may be entered in the AI parameter.

CI text equivalent codes: @codabar

5.2.2 Code 128 (Manual Compression)

The Code 128 bar code is a variable length, high density, alphanumeric symbology that is extensively used worldwide. This bar code uses three subsets (A, B, and C) which allows for the encoding of the full 128 ASCII character set along with special control codes. This bar code uses manual compression, which provides full control by manually shifting between the available subsets using the special function access codes. The codes (#n) are placed within the data string to access the desired functions listed in the table below.

Code	SUBSET A	SUBSET B	SUBSET C
#0	FNC3	FNC3	N/A
#1	FNC2	FNC2	N/A
#2	SHIFT	SHIFT	N/A
#3	CODE C	CODE C	N/A
#4	CODE B	FNC4	CODE B
#5	FNC4	CODE A	CODE A
#6	FNC1	FNC1	FNC1
#7	START	N/A	N/A
#8	N/A	START	N/A
#9	N/A	N/A	START
##	#	#	N/A

Table 9 Code 128 Special Function Access

When using manual compression the printer defaults to subset B if no start code is defined. The printer will also not attempt to compress the data unless subset C is used. Once in subset C the user must send the appropriate code (#n) to return to either subset A or B.

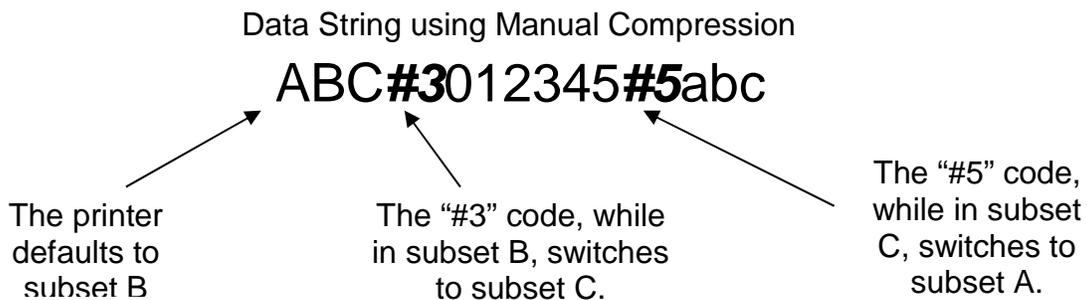


Figure 14 Code 128 Subset Switching

Subset A

This subset contains 106 unique characters, which includes numbers 0 through 9, the English alphabet (uppercase only), punctuation marks, ASCII control codes, and the special function codes listed in table 6.

Subset B

This subset contains 106 unique characters, which includes numbers 0 through 9, the English alphabet (both uppercase and lower case), punctuation marks, and the special function codes listed in table 6. This is the default if a subset is not defined when using manual compression.

Subset C

This subset contains 106 unique characters, which includes matched pair numbers 00 through 99 and the special function codes listed in table 6. Subset C requires that an even amount of digits to be sent because the matched pairs are encoded as a single character. This allows for a greater amount of compression than subsets A and B. If an odd amount of digits and/or non-numeric characters were sent while using subset C, the resulting bar code produced would be invalid.

CI text equivalent codes: @code128, @c128

5.2.3 Code 128 (Automatic Compression)

The Code 128 bar code is a variable length, high density, alphanumeric symbology that is extensively used worldwide. This bar code uses three subsets (A, B, and C) which allows for the encoding of the full 128 ASCII character set along with special control codes. This bar code uses automatic compression, which means that the printer will automatically choose the most efficient subset and insert the appropriate special function code listed in table 6. It is possible to force a subset change by inserting the appropriate code while in automatic compression but is not recommended. Please note that if a subset is forced while using automatic compression the printer will automatically switch to subset C if 6 or more numeric characters are contained in the data string. Once the even number of characters has been processed, the printer will then switch to subset B. This could create problems if a user sent 6 numeric characters while forcing subset A (“#7123456”), the printer would actually automatically switch to subset C and then back to subset B resulting in an undesired bar code. If specific control of the subsets is a requirement, then the manual compression bar code should be used.

CI text equivalent codes: @code128auto

5.2.4 Code 39

Code 39 is a variable length, alphanumeric symbology that is very popular and is extensively used in the non-retail, military, manufacturing and medical industries. Code 39 is capable of encoding numerical characters 0 through 9, the English Alphabet (uppercase only), characters -. *\$/+% along with the “space” character. The Code 39 is framed with a start/stop character represented by the asterisk (“*”) character, which is reserved for this purpose. This symbol uses 2:1, 3:1, 4:2, 5:2, and 8:3 ratios, which may be entered in the AI parameter.

CI text equivalent codes: @code3of9, @code39, @3of9, @c39

5.2.5 Code 93

Code 93 is a variable length; alphanumeric symbology that is capable of encoding 48 different characters and through the use of control characters can encode the full 128 ASCII character set. The 48 characters include the “space”, numerical characters 0 through 9, the English alphabet (uppercase only), characters \$%+-. / and five special characters including the start/stop character).

CI text equivalent codes: @code93, @c93

5.2.6 Code I2of5 (Interleaved 2 of 5)

The I2of5 is a variable length, paired, numerical symbology that is typically used in the industrial and distribution industries. The I2of5 is a paired symbology, which means that the data string being sent must always be an even number amount. If the desired data to be sent is an odd amount, simply pad the beginning with “0” to achieve an even amount of digits. This symbol uses 2:1, 3:1, 4:2, 5:2, and 8:3 ratios, which may be entered in the AI parameter.

CI text equivalent codes: @codei2of5, @i2of5, @i25, @2of5, @c25

5.2.7 MSI0 (Modified Plessey)

The MSI Plessey is a variable length, numeric only symbology used for retail shelf labeling. When using the MSI0 bar code, the user provides both check digits.

CI text equivalent codes: @msi0

5.2.8 MSI1 (Modified Plessey)

The MSI Plessey is a variable length, numeric only symbology used for retail shelf labeling. When using the MSI1 bar code, the user provides one check digit and the printer will calculate the second check digit.

CI text equivalent codes: @msi1

5.2.9 MSI2 (Modified Plessey)

The MSI Plessey is a variable length, numeric only symbology used for retail shelf labeling. When using the MSI2 bar code, the printer will calculate both check digits.

CI text equivalent codes: @msi2

5.2.10 Postnet

The Postnet symbol is used to encode digits for postal applications to sort mail. A Postnet symbol is made up of binary digits, represented as full and half bars. Each character is made up of five bars, which together represent a single numeric digit. The 5-digit ZIP code, 9-digit ZIP+4, and 11-digit Delivery Point Code are all supported.

CI text equivalent codes: @postnet

5.3 2D Bar Codes

This section provides information regarding the different requirements of the two-dimensional bar codes. The 2D symbols generally follow their own unique format structure and not the standard Format Entry Command structure. Unless stated otherwise in the

descriptions below, the printer will automatically generate the necessary check digits when required.

5.3.1 Data Matrix

Data Matrix is a two-dimensional matrix symbology that is capable of encoding up to 2335 alphanumeric characters, 1556 8-bit byte data characters, or 3116 numeric digits. The ECC 200 version is supported and uses the Reed-Solomon error correction. The requirements for generating a data matrix symbol are the subject data, placement information, CI code, configuration, encoding, module size, and date mode. These parameters are set using a Data Matrix unique Format Entry Command used in place of the standard Format Entry Command; listed below.

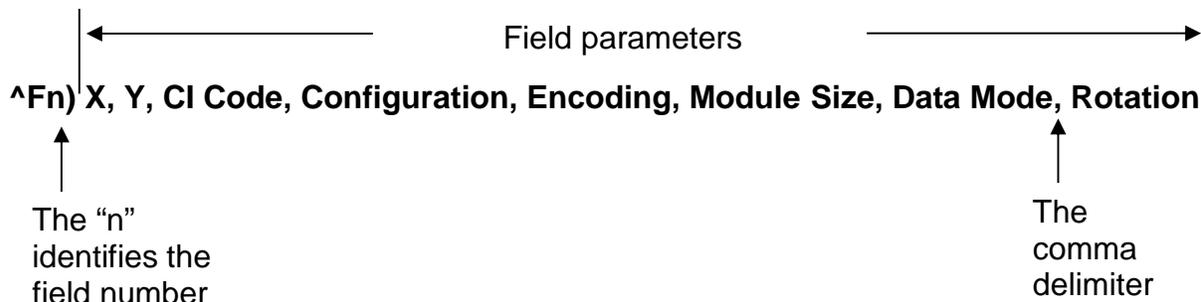


Figure 15 Data Matrix Format Structure

X Parameter

This parameter is used to set the X coordinate start position of the Data Matrix symbol.

Y Parameter

This parameter is used to set the Y coordinate start position of the Data Matrix symbol.

CI Code

This is the Text Equivalent Code that is used to generate the Data Matrix symbol associated with the Text Entry command line ("`^Tn`"). The valid codes are `datamatrix`, `data`, or `dm`.

Configuration

The configuration parameter used to identify the size and capacity of the Data Matrix symbol. The ECC 200 symbols have an even number of rows and an even number of columns. The symbols can be square, with sizes ranging from 10 x 10 to 144 x 144, or rectangular, with sizes 8 x 18 to 16 x 48. The default is set to "auto", which means that the printer will automatically generate the smallest Data Matrix symbol possible for the given data. Refer to the Table below for additional information regarding the valid choices for the configuration parameter.

Symbol Size	Numeric Capacity	Alphanumeric Capacity	Binary Byte Capacity
Auto	---	---	---
10 x 10	6	3	1
12 x 12	10	6	3
14 x 14	16	10	6
16 x 16	24	16	10
18 x 18	36	25	16
20 x 20	44	31	20
22 x 22	60	43	28
24 x 24	72	52	34
26 x 26	88	64	42
32 x 32	124	91	60
36 x 36	172	127	84
40 x 40	228	169	112
44 x 44	288	214	142
48 x 48	348	259	172
52 x 52	408	304	202
64 x 64	560	418	278
72 x 72	736	550	366
80 x 80	912	682	454
88 x 88	1152	862	574
96 x 96	1392	1042	694
104 x 104	1632	1222	814
120 x 120	2100	1573	1048
132 x 132	2608	1954	1302
144 x 144	3116	2335	1556
8 x 18	10	6	3
8 x 32	20	13	8
12 x 26	32	22	14
12 x 36	44	31	20
16 x 36	64	46	30
16 x 48	98	72	47

Table 10 Data Matrix Configuration Parameters

Encoding

The Encoding parameter used to set the encoding scheme used for the supplied data. The printer default is set to "Auto", which will automatically switch to the most efficient encoding

scheme for the data provided. Refer to the table below for the valid choices for the encoding parameter. Consult the Data Matrix Specification (11/96) ANSI/AIM BC11 for additional information.

Encoding Scheme	Characters	Bits per Character
ASCII	Double digit numeric	4
	ACSII values 0 – 127	8
	Extended ASCII values 128 – 255	16
C40	Primarily Uppercase Alphanumeric	5.33
Text	Primarily Lowercase Alphanumeric	5.33
Base256	All byte values 0 – 255	8
Auto	Automatically switches to the most efficient scheme.	NA

Table 11 Data Matrix Encoding Schemes

ACSII Encoding Scheme

ASCII encoding is the basic scheme that encodes ASCII data, double density numeric data and symbology control characters. Which means it encodes one alphabetic or two numeric characters per byte. All other encoding schemes are invoked from ASCII and will then return to this scheme through the use of code words, refer to Table 9.

C40 Encoding Scheme

C40 encoding scheme used to encode uppercase alphabetic, numerical and space characters. C40 encodes three alphanumeric data characters into two bytes.

Text Encoding Scheme

Text encoding primarily used to encode lowercase alphabetic and numerical characters. Text encodes three alphanumeric data characters into two bytes.

Base 256 Encoding Scheme

The Base 256 encoding scheme used to encode any 8-bit byte data, including extended channel interpretations (ECI's) and binary data.

Codeword	Data and/or Function
1 - 128	ASCII data (ASCII value +1)
129	Pad Character
130 - 229	2 digit data 00 – 99 (Numeric Value + 130)
230	Latch to C40 Encoding Scheme
231	Latch to Base256 Encoding Scheme
232	FNC1
233	Structured Append
234	Reader Programming
235	Upper Shift (shift to Extended ASCII)
236	05 Macro
237	06 Macro
239	Latch to Text Encoding Scheme
241	ECI Character

Table 12 ASCII Codeword Values

Module Size

This parameter used to adjust the size of the individual square modules used to create the Data Matrix symbol. This parameter uses dots as its unit of measure with four (4 dots by 4 dots) being the default value.

Data Mode

The Data Mode parameter used to determine whether the resulting barcode is a standard or GS1 Data Matrix. The default value of “0” will generate a standard Data Matrix while a value of “1” will generate a GS1 version. Please note that to generate a proper GS1 barcode, the appropriate AI’s need to be sent surrounded by “[“ and ”]” (bracket) characters.

GS1 Data Example: [10]1234567890123[11]210621

```

^A)
^D564)1
^D200)4,4,0.155,0,4,0,0,0
^F1)1.56,2.0,@datamatrix,20X20,ASCII,15,1,0
^T1)[10]1234567890123[11]210621
^D300)1
^Z)

```

Rotation

This parameter controls the rotation of the generated Data Matrix. The valid values are 0, 90, 180, and 270.

Special Characters

Encode Special characters by sending the “~” tilde character followed by the desired effect. The tilde allows control codes and ASCII characters to be encoded. Refer to the table below for more detail.

Special Character	Description
~X	Used to represent decimal characters 0 to 26 where “X” is the equivalent control code. Example: ~@ = character 0, ~A = character 1, ~B = character 2, etc...
~dNNN	Represents the ASCII encoded by the “NNN” value. For example: “~d065” would represent the ASCII character “A”.

Table 13 Data Matrix Special Characters

CI text equivalent codes: @datamatrix, @data

5.3.2 PDF-417

The Portable Data File 417 or PDF-417 is a two-dimensional stacked symbology that is capable of encoding the entire ASCII character set, 2710 digits, and up to 1108 bytes in a single bar code. The requirements for generating a PDF-417 symbol are the subject data, placement information, size configuration, rotation, and Error Correcting Code or ECC. These parameters are set using a PDF-417 unique Format Entry Command that is used in place of the standard Format Entry Command, which is listed below.

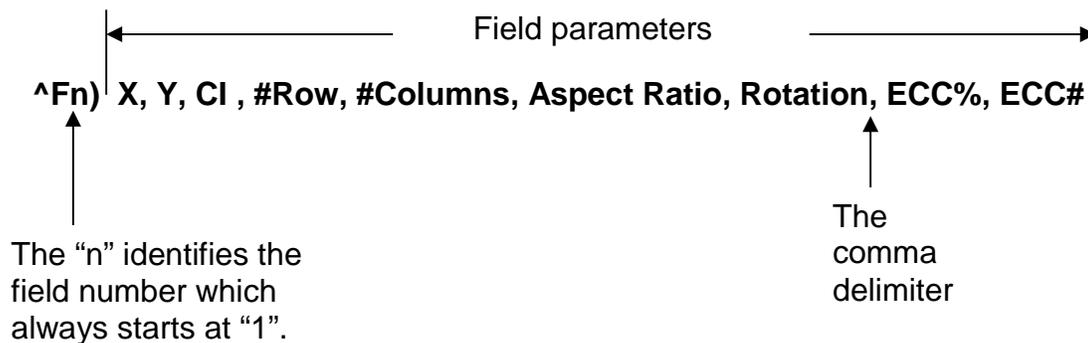


Figure 16 PDF-417 Format Structure

X Parameter

This parameter is used to set the X coordinate start position of the PDF-417 symbol and has a default value of 0.

Y Parameter

This parameter is used to set the Y coordinate start position of the PDF-417 symbol and has a default value of 0.

CI Code

This is the Text Equivalent Code that is used to generate the PDF-417 symbol associated with the Text Entry command line (“^Tn”). The valid codes are pdf417, pdf, or 417.

#Rows

This parameter is used to specify the number of rows used for rendering the PDF-417 symbol. The PDF-417 symbol may have as few as 3 or as many as 90 rows with a default setting of 90.

#Columns

This parameter is used to specify the number of columns used for rendering the PDF-417 symbol. The PDF-417 symbol may have 1 to 30 columns with a default setting of 30.

Aspect Ratio

This parameter is used to adjust the aspect ratio of the PDF-417 symbol. The value is entered in ratio format (height: width) and has a default value of 2:1.

Rotation

This parameter controls the rotation of the rendered PDF-417 symbol in 90-degree increments. The valid arguments for this parameter would be 0, 90, 180 or 270 with the default being set to 0.

ECC%

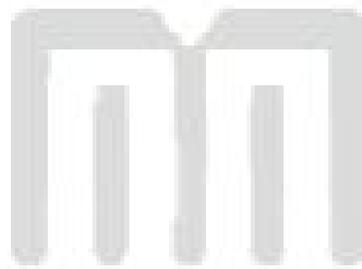
The ECC% parameter is used to set the percentage of overall data dedicated to the Error Correction Control (ECC) of the PDF-417 symbol. This parameter has to be set to 0 if the ECC# method is desired. The default ECC% is set to 0.

ECC#

The ECC# parameter is used to set the level of error correction of the PDF-417 symbol with a default value set to 0. This parameter can only be used if the ECC% parameter is set to 0. Refer to Table 18 for the valid ECC# and their equivalent correction codewords.

ECC#	Number of Error Correction Codewords
0	2
1	4
2	8
3	16
4	32
5	64
6	128
7	256
8	512

Table 14 PDF-417 Error Correction Level



Chapter 6: Status Responses

6.1 Status Responses

Enquiry responses provide a method of determining the current status of the printer as well as identifying different error conditions. This is possible by sending either the “Ctrl+E” combination, issuing the “^D327)” command, or by enabling any of the auto-reporting commands; Refer to the ^D72x series commands. Fatal error responses are triggered from the printer to the host when they occur and halt the printers operation until resolved. Status responses do not halt the printers operation but are simply a status of the printer’s current state.

Refer to the table below and the following sections for more information about the various types of responses.

Byte Mode Responses (ASCII Hex)	Text and Verbose Mode Responses	Definition
0x01	>STOCK OUT<	Out of Media
0x03	>STOCK JAMMED<	Media is jammed.
0x06	>OK<	ACK message used with the ^D726)1 command.
0x10	>CUTTER JAMMED<	The cutter has jammed.
0x11	>CUTTER INOPERATIVE<	The cutter is not detected
0x21	>INPUT 1<	The Input 1 sensor has been actuated.
0x30	>SCRIPT ERROR<	An error with the script has been detected.
0x31	>INVALID PARAMETER<	The printer has detected an invalid parameter.
0x32	>PARSER ERROR<	An error with the printer’s parser has occurred.
0x40	>COMM ERROR<	A Communication error has been detected.
0x51	>FLASH ERROR<	The flash memory has encountered an error.
0x80	>RESTARTED<	The printer has been powered on or the power has been reset.

Byte Mode Responses (ASCII Hex)	Text and Verbose Mode Responses	Definition
0x81	>READY<	The printer is in ready to receive data.
0x87	>STOCK ADVANCED<	The media has been advanced.
0x8F	>STOCK NOT LOADED<	The printer was powered up with out media installed.
0xD1	>NONEXISTENT COMMAND<	The received command does not exist.
0xD2	>INCOMPLETE COMMAND<	The received command is not complete.
0xD3	>ZIP FILE ERROR<	The received Zipped file has an error.
0xD4	>FILE CONVERSION ERROR<	The file was not converted properly.
0xD5	>UNSUPPORTED FILE TYPE<	The received file is not recognized.
0xD6	>FILE NOT FOUND<	The requested file does not exist or cannot be found.
0xD9	>FILE IS READ ONLY<	File is read only and cannot be deleted.
0xDA	>INADEQUATE FLASH SPACE<	Cannot save an item to the flash memory due to lack of space.
0xDB	>I/O FAILURE<	The Input/Output circuitry has failed.
0xDC	>FONT/GRAPHIC NOT FOUND<	The requested file does not exist or cannot be found.
0xF0	>FIELD OFF IMAGE<	An image placement coordinates exceed the print head or form size.
0xF2	>GRAPHIC TOO LARGE<	The sent graphic exceeds the available memory.
0xFF	>POWERED DOWN<	The printer's power has been removed.

Table 15 Enquiry Responses

6.2 Fatal Error Responses

Fatal error responses are enquiry responses that halt the printer's current operation and cause the indicator light or LED to flash RED in color. A fatal error condition persists until the condition causing the error is removed and the error is cleared. These enquiry responses are sent from the printer to the host, when the appropriate condition occurs, by issuing the "Ctrl+E" (^E) combination or by issuing the queued "^D327)1" command. The printer may be configured to automatically reports errors and status messages by using the ^D72x series commands. The following sections provide further explanations of the various fatal error responses. Refer to the ^D550) command regarding the types of enquiry response sets.

NOTE: The following section titles are the responses that the printer will generate once the associating condition occurs and the printer processes an enquiry command. The first response (between and including the ">" and "<" characters) is generated if text and verbose responses have been enabled using the ^D550) command. The ASCII hex value listed between the parenthesis characters, indicate the enquiry response if byte mode has been enabled.

6.2.1 >STOCK OUT< (01)

The ">STOCK OUT<" error response is generated when the printer runs out of stock or media while in text and verbose modes. If the printer is configured for byte responses, the code ASCII 0x01 will be issued from the printer in response to an enquiry command.

6.2.2 >STOCK JAMMED< (03)

The ">STOCK JAMMED<" error response indicates that media is jammed and requires correction before the printer can continue printing while in text and verbose modes. A stock jam condition may be detected by two separate methods depending is the printer is configured with a Label Present Detector or LPD. If the printer is configured for byte responses, the code ASCII 0x03 will be issued from the printer in response to an enquiry command.

6.2.3 >CUTTER JAMMED< (10)

The ">CUTTER JAMMED<" error response is generated when the cutter is in a jammed condition and cannot complete a full cycle while in text and verbose modes. If the printer is configured for byte responses, the code ASCII 0x10 will be issued from the printer in response to an enquiry command.

6.2.4 >CUTTER INOPERATIVE< (11)

This error response is typically generated if the cutter is not detected by the printer while in text and verbose modes. If the printer is configured for byte responses, the code ASCII 0x11 will be issued from the printer in response to an enquiry command.

6.2.5 >SCRIPT ERROR< (30)

The ">SCRIPT ERROR<" response is generated when a script contains an error and cannot be processed while in text and verbose response sets. If the printer is configured for byte responses, the code ASCII 0x30 will be issued from the printer in response to an enquiry command.

6.2.6 >INVALID PARAMETER< (31)

This error response indicates that a received command contains an invalid parameter or illegal structure while in text and verbose response sets. If the printer is configured for byte responses, the code ASCII 0x31 will be issued from the printer in response to an enquiry command.

6.2.7 >PARSER ERROR< (32)

The ">PARSER ERROR<" indicates that the printer has detected an internal error with the parser while in text and verbose response sets. If the printer is configured for byte responses, the code ASCII 0x32 will be issued from the printer in response to an enquiry command.

6.2.8 >COMM ERROR< (40)

The ">COMM ERROR<" indicates that the printer has detected a communication error while in text or verbose response modes. If the printer is configured for byte responses, the code ASCII 0x40 is issued from the printer in response to an enquiry command. This error response is generated if the serial communication timeout period of 10 seconds plus 100 times the transfer has occurred.

6.2.9 >FLASH ERROR< (51)

The ">FLASH ERROR<" indicates that an error has occurred with the flash memory while in text and verbose response sets. The printer power should be cycled in the event that this error message has been received. If the printer is configured for byte responses, the code ASCII 0x51 will be issued from the printer in response to an enquiry command.

6.3 Basic Status Responses

Basic status responses are enquiry responses that are not immediate. These enquiry responses are sent from the printer to the host, when the appropriate condition occurs, by issuing the "Ctrl+E" (^E) combination or by issuing the "^D327)" command. The following sections provide further explanations of the non-immediate basic status responses. Refer to the ^D550) command regarding the types of enquiry response sets.

NOTE: The following section titles are the responses that the printer will generate once the associating condition occurs. The first response (between and including the ">" and "<" characters) is generated if text and verbose responses have been enabled using the ^D550) command. The ASCII hex value listed between the parenthesis characters, indicate the enquiry response if byte mode has been enabled.

6.3.1 >OK< (06)

The ">OK<" response is generated in conjunction with the "^D726)1" command and indicates that a command completed successfully while in text or verbose response modes. If the printer is configured for byte responses, the code ASCII 0x06 is issued from the printer in response to an enquiry command.

6.3.2 >INPUT 1< (21)

The ">INPUT 1<" response indicates that the INPUT 1 sensor option is tripped while in text or verbose response modes. If the printer is configured for byte responses, the code ASCII 0x21 is issued from the printer in response to an enquiry command. This sensor and response is intended to provide the user with the flexibility for custom configurations.

6.3.3 >RESTARTED< (80)

The “>RESTARTED<” response is issued the first time the printer’s status is queried after a power up or reset condition while in verbose and/or semi byte responses. If the printer is configured for byte responses, the code ASCII 0x80 will be issued from the printer in response to an enquiry command.

6.3.4 >READY< (81)

The “>READY<” response indicates that the printer is in the normal operating state and that it is ready to accept data while in text and verbose response sets. If the printer is configured for byte responses, the code ASCII 0x81 will be issued from the printer in response to an enquiry command.

6.3.5 >STOCK ADVANCED< (87)

The “>STOCK ADVANCED<” response indicates that the media has been advanced after printing when a Dispense Mode (^D597) command) is being used while in text and verbose response sets. If the printer is configured for byte responses, the code ASCII 0x87 will be issued from the printer in response to an enquiry command.

6.3.6 >STOCK NOT LOADED< (8F)

The “>STOCK NOT LOADED<” response indicates that media is not being detected in the printer on power up while in text and verbose response sets. If the printer is configured for byte responses, the code ASCII 0x8F is issued from the printer in response to an enquiry command. This message persists until media is detected at the registration mark sensors but does not prevent commands from being processed.

6.3.7 >FLASH MEMORY LOW< (B0)

The “>FLASH MEMORY LOW<” response indicates that the user storage area of the flash memory is 95% full while in text and verbose response sets. This message will continue to be received until files are removed from flash memory or until the printer performs garbage collecting and defragmenting. If the printer is configured for byte responses, the code ASCII 0xB0 will be issued from the printer in response to an enquiry command.

6.3.8 >TEMPERATURE OUT OF RANGE< (D0)

The “>TEMPERATURE OUT OF RANGE<” response indicates that the printer’s print head is not within the proper operating range while in text and verbose response sets. This response will not allow the printer to print until the print head returns to the proper operating range. If the printer is configured for byte responses, the code ASCII 0xD0 will be issued from the printer in response to an enquiry command.

6.3.9 >NONEXISTENT COMMAND< (D1)

The “>NONEXISTENT COMMAND<” response indicates that the printer has received data in the 438M command structure that does not exist or is yet to be implemented while in text and verbose response sets. If the printer is configured for byte responses, the code ASCII 0xD1 will be issued from the printer in response to an enquiry command.

6.3.10 >ZIP FILE ERROR< (D3)

The “>ZIP FILE ERROR<” response indicates that the printer has received a zip file that is either not complete or that the file contains errors while in text and verbose response sets. If the printer is configured for byte responses, the code ASCII 0xD3 will be issued from the printer in response to an enquiry command.

6.3.11 >UNSUPPORTED FILE TYPE< (D5)

This enquiry response indicates that the printer does not recognize the type of the received file while in text and verbose modes. If the printer is configured for byte responses, the code ASCII 0xD5 will be issued from the printer in response to an enquiry command.

6.3.12 >FILE NOT FOUND< (D6)

The “>FILE NOT FOUND<” response typically indicates that the printer is being instructed to recall and process a file that does not exist in the printer’s memory while in text and verbose response sets. If the printer is configured for byte responses, the code ASCII 0xD6 will be issued from the printer in response to an enquiry command.

6.3.13 >FILE IS READ ONLY< (D9)

This enquiry response typically indicates that a save or delete command is being issued to the printer and that the file is a protected file stored in the printer. If the printer is configured for byte responses, the code ASCII 0xD9 will be issued from the printer in response to an enquiry command.

6.3.14 >INADEQUATE FLASH SPACE< (DA)

This enquiry response indicates that the available flash memory is less than the total size of a file being saved to flash memory. If the printer is configured for byte responses, the code ASCII 0xDA will be issued from the printer in response to an enquiry command.

6.3.15 >FONT/GRAPHIC NOT FOUND< (DC)

This enquiry response typically indicates that a font being specified a script file does not exist in flash memory. If the printer is configured for byte responses, the code ASCII 0xDC will be issued from the printer in response to an enquiry command.

Chapter 7: Communications

7.1 RS-232 Serial

The printer provides a 9 pin female D-Sub connector, on the rear panel, for RS-232 serial communication. The printer is configured as DCE (Data Circuit-terminating Equipment) so a NULL modem cable is not necessary. A serial data cable is required to use serial communications, refer to the table below for correct cable configurations.

The factory settings, unless otherwise requested, are set at 9600 baud, 8 data bits, 1 stop bit, and no parity with both hardware and software handshaking. The baud rate is user selectable from 1200 to 230,400 and uses software XON/XOFF flow control and/or hardware CTS handshaking.

9 to 9 Pin Cable				9 to 25 Pin Cable			
State	426 DE-9 Pin #	Signals / Direction	PC DE-9 Pin #	State	426 DE-9 Pin #	Signals / Directions	PC DB-25 Pin #
XX	1	DCD → DCD	1	HI	1	DCD → DCD	8
XX	2	TXD → RXD	2	XX	2	TXD → RXD	3
XX	3	RXD ← TXD	3	XX	3	RXD ← TXD	2
HI	4	DTR → DSR	4	HI	4	DTR → DSR	20
LO	5	GND ↔ GND	5	LO	5	GND ↔ GND	7
HI	6	DSR → DTR	6	HI	6	DSR → DTR	6
DC	7	RTS ← CTS	7	DC	7	RTS ← CTS	4
XX	8	CTS → RTS	8	XX	8	CTS → RTS	5
HI	9	+5V ↔ +5V	9	HI	9	+5V	-
DC = DO NOT CARE				XX = INDETERMINATE			

Table 16 RS-232 Cable Configurations

If XON/XOFF handshaking is used, only signals RXD, TXD, and GND are required for proper operation. If the hardware handshaking is used the CTS and RTS signals are required. The other signals are offered in the event that the host computer requires them.

7.1.1 RS-232 Printer Cables

The printer uses standard cables that may be purchased through Microcom Corporation or a local computer supply company. A NULL-modem cable is not required because the printer is configured as DCE.



CAUTION: Connection of a serial port to a parallel port may result in damage to the printer and / or computer.

7.2 USB Port

The printer provides the capability of a Universal Serial Bus (USB) port that supports version 1.1. The USB Type 1 port is a virtual serial communications port and requires the installation of the Microcom Corporation supplied virtual serial communication USB driver to be installed to properly function. Refer to the “^D570)” command for additional information regarding the available communication ports the printer offers.

Appendix A – ASCII Chart

DEC	HEX	ASCII	KEY	DEC	HEX	ASCII	DEC	HEX	ASCII	DEC	HEX	ASCII
0	00	NUL	Ctrl @	32	20	Space	64	40	@	96	60	`
1	01	SOH	Ctrl A	33	21	!	65	41	A	97	61	a
2	02	STX	Ctrl B	34	22	"	66	42	B	98	62	b
3	03	ETX	Ctrl C	35	23	#	67	43	C	99	63	c
4	04	EOT	Ctrl D	36	24	\$	68	44	D	100	64	d
5	05	ENQ	Ctrl E	37	25	%	69	45	E	101	65	e
6	06	ACK	Ctrl F	38	26	&	70	46	F	102	66	f
7	07	BEL	Ctrl G	39	27	'	71	47	G	103	67	g
8	08	BS	Ctrl H	40	28	(72	48	H	104	68	h
9	09	HT	Ctrl I	41	29)	73	49	I	105	69	i
10	0A	LF	Ctrl J	42	2A	*	74	4A	J	106	6A	j
11	0B	VT	Ctrl K	43	2B	+	75	4B	K	107	6B	k
12	0C	FF	Ctrl L	44	2C	,	76	4C	L	108	6C	l
13	0D	CR	Ctrl M	45	2D	-	77	4D	M	109	6D	m
14	0E	SO	Ctrl N	46	2E	.	78	4E	N	110	6E	n
15	0F	SI	Ctrl O	47	2F	/	79	4F	O	111	6F	o
16	10	DLE	Ctrl P	48	30	0	80	50	P	112	70	p
17	11	DC1	Ctrl Q	49	31	1	81	51	Q	113	71	q
18	12	DC2	Ctrl R	50	32	2	82	52	R	114	72	r
19	13	DC3	Ctrl S	51	33	3	83	53	S	115	73	s
20	14	DC4	Ctrl T	52	34	4	84	54	T	116	74	t
21	15	NAK	Ctrl U	53	35	5	85	55	U	117	75	u
22	16	SYN	Ctrl V	54	36	6	86	56	V	118	76	v
23	17	ETB	Ctrl W	55	37	7	87	57	W	119	77	w
24	18	CAN	Ctrl X	56	38	8	88	58	X	120	78	y
25	19	EM	Ctrl Y	57	39	9	89	59	Y	121	79	x
26	1A	SUB	Ctrl Z	58	3A	:	90	5A	Z	122	7A	z
27	1B	ESC	Ctrl [59	3B	;	91	5B	[123	7B	{
28	1C	FS	Ctrl \	60	3C	<	92	5C	\	124	7C	
29	1D	GS	Ctrl]	61	3D	=	93	5D]	125	7D	}
30	1E	RS	Ctrl ^	62	3E	>	94	5E	^	126	7E	~
31	1F	US	Ctrl _	63	3F	?	95	5F	_	127	7F	DEL

ASCII = American Standard Code for Information Interchange

Glossary

Alphanumeric - A term used to identify characters consisting of letters, numbers, and other symbols such as punctuation marks and mathematical symbols.

Bar code - Alphanumeric characters that are represented using a system of printed lines, which can be read by a computer.

Blackline - A registration mark, typically a black line on the underside of certain media that is used to identify the starting position of a label.

Blowhole - A registration mark, typically a notched or cutout in certain media's that is used to identify the starting position of a label.

Continuous media - Media that contains no gap and continues as one piece of media.

CTS (Clear to Send) - A handshaking signal used to provide information when a DCE or modem is ready to receive data in an RS-232 communication link. Used in conjunction with RTS.

Cutter - An optional device, mounted on the printer, which is used to cut media into custom sized labels after printing

DCE (Data-Circuit-terminating Equipment) - The modem or printer end of an RS-232 serial communication link. Modem serial ports and other peripherals are typically configured as DCE.

Die-cut media - Media that is cut using a die to form individual labels that have a start and stop point. Typically the excess material surrounding the label is removed from the backing material.

Direct thermal - Method of printing with a special heat-sensitive media that directly contacts heat elements of a thermal print head. (See *Thermal print head*)

DPI (Dots Per Inch) - A unit of measure term used to identify the print resolution capability.

Drive roller - Platen roller that is located below the print head that is used to drive media through the printer.

DSR (Data Set Ready) - A handshaking signal used to provide information about the status of an RS-232 communication link. Used in conjunction with DTR.

DTE (Data Terminal Equipment) - The terminal or computer end of an RS-232 serial communication link. Serial ports on computers are typically configured as DTE.

DTR (Data Terminal Ready) - A handshaking signal used to provide information about the status of the communication line connected to the DCE end of an RS-232 link. Used in conjunction with DSR.

Ethernet - A fast and capable serial interface used by many networks for connecting host computers to various peripherals.

Fanfold media - Die-cut or tag media that is folded into stacks.

Flash memory - Memory that is used to store font, graphics and label formats for faster access. This is a non-volatile type of memory, which means that the information stored is saved even when the printer power is turned off.

GND (Signal Ground) - The ground terminal of a power supply's output, and all points that connect to it. This is an essential signal in an RS-232 communication link that is used to reduce noise spikes, which can cause a receiver to misread logic levels. Also known as SG and SGND.

Guillotine Cutter – A cutter that has a blade that cuts perpendicular to the media path.

LDSII (Label Design Software II) - This is the resident control software language that is used to develop label formats and controls the full function of the printer.

LPS (Label Present Sensor) - This sensor, when enabled, detects when media is present. Also referred to as LTS (Label Taken Sensor).

Platen - Roller that is located below the print head, which the media rests upon. (See *Drive roller*)

Rolled media - Media that has been rolled onto a core, which is typically cardboard in nature.

Rotary Cutter – A cutter that has multiple angled blades mounted to a shaft that cut media by rotating the shaft.

RS-232 - A popular and inexpensive asynchronous serial interface used for connecting host computers to various peripherals.

RTS (Request To Send) - A handshaking signal used to provide information when a DTE or terminal has data to send in an RS-232 communication link. Used in conjunction with CTS.

RXD (Receive Data) - An essential signal that carries data from the DCE to the DTE for 2-way RS-232 serial communication. Also known as RX and RD.

SDRAM (Synchronous Dynamic Random Access Memory) - Memory that is used to store formats while printing. This is volatile memory, which means that the information stored in this memory is not saved when the printer power is turned off.

Tag stock - Media that typically has no adhesive backing, thicker in nature and generally is a fanfold type media.

Thermal print head - A print head that uses thermal technology, to heat individual elements or dots to produce images. Also referred to as print head in this manual.

TXD (Transmit Data) - An essential signal that carries data from the DTE to the DCE for 2-way RS-232 serial communication. Also known as TX and TD.

USB (Universal Serial Bus) - A high-speed asynchronous serial interface typically transfers up to 12M bits/second, used for connecting host computers to various peripherals.

Index

A

AI Parameter	35, 39, 40, 41, 42
AN Parameter	15, 16, 31, 32

B

Bar Codes	12, 13, 39, 42, 59
Box Creation	38

C

CI Parameter	9, 11, 12, 13, 14, 17, 20, 22, 36, 39, 40, 41, 42, 43, 47, 48
Codabar Symbol	12, 39
Code 128 Symbol	12, 40
Code 39 Symbol	41
Code 93 Symbol	12, 42
Communication	
RS-232 Serial	57
Control Codes	3, 40, 41, 47

D

Data Matrix Symbol	12, 43, 44, 45, 46, 47
DN Parameter	15
DPG Parameter	25

E

Enablement Commands	30, 31, 32
Encoding	44, 45, 46
Enquiry	51
Ethernet	61

F

File Storage	22
Graphics	12
Scripts	6, 19
Flash Memory	33, 61
Flow Control	57
Format Entry Command	9, 20, 39, 42, 43, 47
Formatting Commands	20
FW Parameter	16

G

Glossary	60
Graphics	12

I

Interleaved 2 of 5 Symbol	42
---------------------------------	----

L

Lines	9, 11, 13, 14, 15, 17, 20, 35, 36, 37, 38, 43, 48, 60
LSX Parameter	7
LSY Parameter	7, 8

M

Modified Plessey Symbol42

O

ON Parameter 15
Operation Commands 21, 22

P

PDF-417 Symbol47, 48, 49
Printhead..... 60, 61

R

Reverse Video 35, 36
RS-232 Cable.....57

S

Saving Files 6, 12, 19, 22
Script Header 3, 4, 5, 6, 15, 20, 25, 26, 47
Selection Commands 23, 26
Serial..... 1, 33, 60, 61, 62
Setting Commands..... 25, 26, 28, 30, 60
SH Parameter 11, 13, 14, 36, 39
SW Parameter 11, 13, 36, 39

T

Test and Factory Commands 32
Text Entry Command 17

X

XB Parameter 9, 11, 15, 36

Y

YB Parameter 9, 11, 15, 36

