



**NLS-NVH300**

**Industrial Handheld Barcode Scanner**

**User Guide**

# Disclaimer

© 2019 Fujian Newland Auto-ID Tech. Co., Ltd. All rights reserved.

Please read through the manual carefully before using the product and operate it according to the manual. It is advised that you should keep this manual for future reference.

Do not disassemble the device or remove the seal label from the device, doing so will void the product warranty provided by Fujian Newland Auto-ID Tech. Co., Ltd.

All pictures in this manual are for reference only and actual product may differ. Regarding to the product modification and update, Fujian Newland Auto-ID Tech. Co., Ltd. reserves the right to make changes to any software or hardware to improve reliability, function, or design at any time without notice. The information contained herein is subject to change without prior notice.

The products depicted in this manual may include software copyrighted by Fujian Newland Auto-ID Tech. Co., Ltd or third party. The user, corporation or individual, shall not duplicate, in whole or in part, distribute, modify, decompile, disassemble, decode, reverse engineer, rent, transfer or sublicense such software without prior written consent from the copyright holders.

This manual is copyrighted. No part of this publication may be reproduced, distributed or used in any form without written permission from Newland.

Fujian Newland Auto-ID Tech. Co., Ltd. reserves the right to make final interpretation of the statement above.

# Revision History

Version	Description	Date
V1.0.0	Initial release. This Manual is based on the firmware NVH300_V1.00.003.	June19th, 2020

# Table of Contents

<b>Revision History.....</b>	<b>- 3 -</b>
<b>Preface.....</b>	<b>1</b>
Introduction.....	1
Chapter Description.....	1
Explanation of Icons.....	2
<b>Chapter 1 Getting Started.....</b>	<b>3</b>
Introduction.....	3
Features of the NVH300.....	3
Accessories .....	4
Unpacking.....	4
NVH300 Scanner.....	5
Dimensions of the Scanner (unit: mm).....	6
Front View.....	7
Top View.....	7
Data Port.....	8
Connecting the NVH300 to a Host.....	9
Using USB Cable.....	9
Using RS-232 Cable.....	10
Connect the NVH300-DP Communication Data Cable.....	11
Remove the NVH300-DP Communication Data Cable.....	13
Remove the Cable of NVH300-HD.....	15
Power On, Power Off, Reboot.....	15
Maintenance.....	16
Scanning Instructions.....	16
<b>Chapter 2 NSet.....</b>	<b>18</b>
Introduction.....	18
Main Features.....	18
<b>Chapter 3 System Settings.....</b>	<b>19</b>
Introduction.....	19
Barcode Programming.....	19
Command Programming.....	19
NSet Programming.....	19
Programming Barcode/ Programming Command/Function.....	20



Use of Programming Command.....	21
Command Syntax.....	21
Query Commands.....	21
Responses.....	22
Examples.....	22
Use of Programming Barcode.....	23
Illumination.....	24
Illumination Color.....	24
Illumination Rotation Interval.....	25
Aiming.....	25
Good Read LED.....	27
Good Read LED Duration.....	28
Power On Beep.....	29
Good Read Beep.....	29
Good Read Beep Volume.....	30
Good Read Beep Duration.....	31
Good Read Beep Frequency.....	32
Number of Good Read Beeps.....	33
Good Read Beep Interval Time.....	34
Good Read Vibration.....	34
Good Read Vibration Duration.....	35
Scan Mode.....	35
One Reading Timeout.....	38
Reading Interval.....	38
Image Stabilization Timeout (Sense Mode).....	39
Good Read Delay.....	39
Image Decoding Timeout.....	41
Scanning Preference.....	42
Sensitivity.....	43
Read Barcode On/Off.....	44
Decode Area.....	44
Image Flipping.....	49
Image Resize.....	50
Transmit Not Good Read Message.....	51
Edit NGR Message.....	51
Image Filter.....	51
Default Settings.....	52
Factory Defaults.....	52

Custom Defaults.....	53
Query Scanner Information.....	54
Query Product Name.....	54
Query Firmware Version.....	54
Query Decoder Version.....	55
Query Hardware Version.....	55
Query Serial Number.....	55
Query Manufacturing Date.....	55
<b>Chapter 4 RS-232 Interface.....</b>	<b>56</b>
Introduction.....	56
Baud Rate.....	57
Parity Check.....	58
Data Bit.....	60
Stop Bit.....	60
Hardware Auto Flow Control.....	61
<b>Chapter 5 USB Interface.....</b>	<b>62</b>
Introduction.....	62
USB HID Keyboard.....	63
USB Country Keyboard Types.....	64
Beep on Unknown Character.....	68
Emulate ALT+Keypad.....	69
Code Page.....	70
Unicode Encoding.....	73
Function Key Mapping.....	74
ASCII Function Key Mapping Table.....	75
ASCII Function Key Mapping Table (Continued).....	76
Inter-Keystroke Delay.....	77
Caps Lock.....	77
Convert Case.....	78
Emulate Numeric Keypad.....	78
Polling Rate.....	81
USB CDC.....	83
USB HID-POS.....	83
Introduction.....	83
Access the Scanner with Your Program.....	84
Acquire Scanned Data.....	84
Send Command to the Scanner.....	84

VID/PID.....	84
<b>Chapter 6 Symbolologies.....</b>	<b>86</b>
Introduction.....	86
Global Settings.....	86
Enable/Disable All Symbolologies.....	86
Enable/Disable 1D Symbolologies.....	87
Enable/Disable 2D Symbolologies.....	87
Image Brightness Adaptation.....	89
1D Decoding Complexity.....	89
1D Inverse.....	90
Code 128.....	91
Restore Factory Defaults.....	91
Enable/Disable Code 128.....	91
Set Length Range for Code 128.....	92
EAN-8.....	93
Restore Factory Defaults.....	93
Enable/Disable EAN-8.....	93
Transmit Check Character.....	94
2-Digit Add-On Code.....	94
5-Digit Add-On Code.....	96
Add-On Code Required.....	96
Add-On Code Separator.....	97
Convert EAN-8 to EAN-13.....	97
EAN-13.....	99
Restore Factory Defaults.....	99
Enable/Disable EAN-13.....	99
Transmit Check Character.....	100
2-Digit Add-On Code.....	100
5-Digit Add-On Code.....	102
Add-On Code Required.....	102
Add-On Code Separator.....	103
EAN-13 Beginning with 290 Add-On Code Required.....	103
EAN-13 Beginning with 378/379 Add-On Code Required.....	104
EAN-13 Beginning with 414/419 Add-On Code Required.....	104
EAN-13 Beginning with 434/439 Add-On Code Required.....	105
EAN-13 Beginning with 977 Add-On Code Required.....	106
EAN-13 Beginning with 978 Add-On Code Required.....	106

EAN-13 Beginning with 979 Add-On Code Required.....	107
UPC-E.....	108
Restore Factory Defaults.....	108
Enable/Disable UPC-E.....	108
Transmit Check Character.....	109
2-Digit Add-On Code.....	109
5-Digit Add-On Code.....	110
Add-On Code Required.....	111
Add-On Code Separator.....	111
Transmit Preamble Character.....	112
Convert UPC-E to UPC-A.....	112
UPC-A.....	113
Restore Factory Defaults.....	113
Enable/Disable UPC-A.....	113
Transmit Check Character.....	114
2-Digit Add-On Code.....	114
5-Digit Add-On Code.....	116
Add-On Code Required.....	116
Add-On Code Separator.....	117
Transmit Preamble Character.....	117
Interleaved 2 of 5.....	119
Restore Factory Defaults.....	119
Enable/Disable Interleaved 2 of 5.....	119
Set Length Range for Interleaved 2 of 5.....	120
Check Character Verification.....	121
ITF-14.....	122
Restore Factory Defaults.....	122
Enable/Disable ITF-14.....	122
ITF-6.....	124
Restore Factory Defaults.....	124
Enable/Disable ITF-6.....	124
Matrix 2 of 5.....	126
Restore Factory Defaults.....	126
Enable/Disable Matrix 2 of 5.....	126
Set Length Range for Matrix 2 of 5.....	127
Check Character Verification.....	128
Code 39.....	130
Restore Factory Defaults.....	130

Enable/Disable Code 39.....	130
Set Length Range for Code 39.....	131
Check Character Verification.....	132
Transmit Start/Stop Character.....	133
Enable/Disable Code 39 Full ASCII.....	133
Codabar.....	134
Restore Factory Defaults.....	134
Enable/Disable Codabar.....	134
Set Length Range for Codabar.....	135
Check Character Verification.....	136
Start/Stop Character.....	137
Code 93.....	139
Restore Factory Defaults.....	139
Enable/Disable Code 93.....	139
Set Length Range for Code 93.....	140
Check Character Verification.....	141
GS1-128 (UCC/EAN-128).....	142
Restore Factory Defaults.....	142
Enable/Disable GS1-128.....	142
AIM 128.....	143
Restore Factory Defaults.....	143
Enable/Disable AIM 128.....	143
Set Length Range for AIM 128.....	144
ISBT 128.....	145
Restore Factory Defaults.....	145
Enable/Disable ISBT 128.....	145
Code 11.....	146
Restore Factory Defaults.....	146
Enable/Disable Code 11.....	146
Set Length Range for Code 11.....	147
Check Character Verification.....	148
Transmit Check Character.....	150
ISBN.....	151
Restore Factory Defaults.....	151
Enable/Disable ISBN.....	151
Set ISBN Format.....	152
ISSN.....	153
Restore Factory Defaults.....	153

Enable/Disable ISSN.....	153
Industrial 2/5.....	154
Restore Factory Defaults.....	154
Enable/Disable Industrial 2/5.....	154
Set Length Range for Industrial 25.....	155
Check Character Verification.....	156
Standard 2/5.....	157
Restore Factory Defaults.....	157
Enable/Disable Standard 2/5.....	157
Set Length Range for Standard 25.....	158
Check Character Verification.....	159
COOP 25.....	160
Restore Factory Defaults.....	160
Enable/Disable COOP 25.....	160
Set Length Range for COOP 25.....	161
Check Character Verification.....	162
Deutsche 14.....	163
Restore Factory Defaults.....	163
Enable/Disable Deutsche 14.....	163
Deutsche 12.....	164
Restore Factory Defaults.....	164
Enable/Disable Deutsche 12.....	164
Plessey.....	165
Restore Factory Defaults.....	165
Enable/Disable Plessey.....	165
Set Length Range for Plessey.....	166
Check Character Verification.....	167
MSI-Plessey.....	168
Restore Factory Defaults.....	168
Enable/Disable MSI-Plessey.....	168
Set Length Range for MSI-Plessey.....	169
Check Character Verification.....	170
Transmit Check Character.....	171
QR Code.....	172
Restore Factory Defaults.....	172
Enable/Disable QR Code.....	172
Set Length Range for QR Code.....	173
QR Inverse.....	174

Character Encoding.....	174
QR ECI Output.....	175
QR Decoding Complexity.....	175
Continuous Code.....	176
Dot-peen Code.....	176
Mirror Code.....	177
QR Model 1.....	177
QR Model 2.....	178
Data Matrix.....	179
Restore Factory Defaults.....	179
Enable/Disable Data Matrix.....	179
Set Length Range for Data Matrix.....	181
Rectangular Barcode.....	182
Data Matrix Inverse.....	182
Character Encoding.....	183
Data Matrix ECI Output.....	183
ECC200.....	184
ECC0TO140.....	184
DM Decoding Complexity.....	185
Continuous Code.....	186
Dot-peen Code.....	186
Mirror Code.....	187
PDF417.....	188
Restore Factory Defaults.....	188
Enable/Disable PDF417.....	188
Set Length Range for PDF417.....	189
PDF417 Inverse.....	190
Character Encoding.....	190
<b>Chapter 8 Data Formatter.....</b>	<b>191</b>
Introduction.....	191
Add a Data Format.....	191
Programming with Barcodes.....	191
Programming with Serial Commands.....	194
Enable/Disable Data Formatter.....	195
Non-Match Error Beep.....	196
Data Format Selection.....	198
Change Data Format for a Single Scan.....	199

Clear Data Format.....	200
Query Data Formats.....	200
Formatter Command Type 6.....	202
Send Commands.....	202
Move Commands.....	205
Search Commands.....	207
Miscellaneous Commands.....	210
<b>Chapter 9 Prefix &amp; Suffix.....</b>	<b>215</b>
Introduction.....	215
Global Settings.....	216
Enable/Disable All Prefixes/Suffixes.....	216
Prefix Sequence.....	216
Custom Prefix.....	217
Enable/Disable Custom Prefix.....	217
Set Custom Prefix.....	217
AIM ID Prefix.....	218
Code ID Prefix.....	219
Restore All Default Code IDs.....	219
Modify Code ID.....	219
Custom Suffix.....	227
Enable/Disable Custom Suffix.....	227
Set Custom Suffix.....	227
Terminating Character Suffix.....	228
Enable/Disable Terminating Character Suffix.....	228
Set Terminating Character Suffix.....	228
<b>Chapter 10 Batch Programming.....</b>	<b>230</b>
Introduction.....	230
Create a Batch Command.....	231
Create a Batch Barcode.....	231
Use Batch Barcode.....	232
<b>Appendix.....</b>	<b>233</b>
Digit Barcodes.....	233
Save/Cancel Barcodes.....	236
Factory Defaults Table.....	237
AIM ID Table.....	244
Code ID Table.....	245



Symbology ID Number.....	246
ASCII Table.....	247



# Preface

## Introduction

This manual provides detailed instructions for setting up and using the NLS-NVH300 Industrial Handheld Barcode Scanner (hereinafter referred to as “**the NVH300**” or “**the scanner**”).

## Chapter Description

<i>Chapter 1 Getting Started</i>	: Gives a general description of NVH300 scanner.
<i>Chapter 2 NSet</i>	: Introduces a useful tool you can use to set up NVH300 scanner and develop new applications.
<i>Chapter 3 System Settings</i>	: Introduces three configuration methods and describes how to configure general parameters of NVH300 scanner.
<i>Chapter 4 RS-232 Interface</i>	: Describes how to configure RS-232 communication parameters.
<i>Chapter 5 USB Interface</i>	: Describes how to configure USB communication parameters.
<i>Chapter 6 Symbologies</i>	: Lists all compatible symbologies and describes how to configure the relevant parameters.
<i>Chapter 7 Data Formatter</i>	: Explains how to customize scanned data with the data formatter.
<i>Chapter 8 Prefix &amp; Suffix</i>	: Describes how to use prefix and suffix to customize scanned data.
<i>Chapter 9 Batch Programming</i>	: Explains how to integrate a complex programming task into a single barcode.
<i>Appendix</i>	: Provides factory defaults table and a bunch of frequently used programming barcodes.

---

## Explanation of Icons



Tools

This icon indicates something relevant to this manual.



This icon indicates this information requires extra attention from the reader.



This icon indicates handy tips that can help you use or configure the scanner with ease.



This icon indicates practical examples that can help you to acquaint yourself with operations.

# Chapter 1 Getting Started

## Introduction

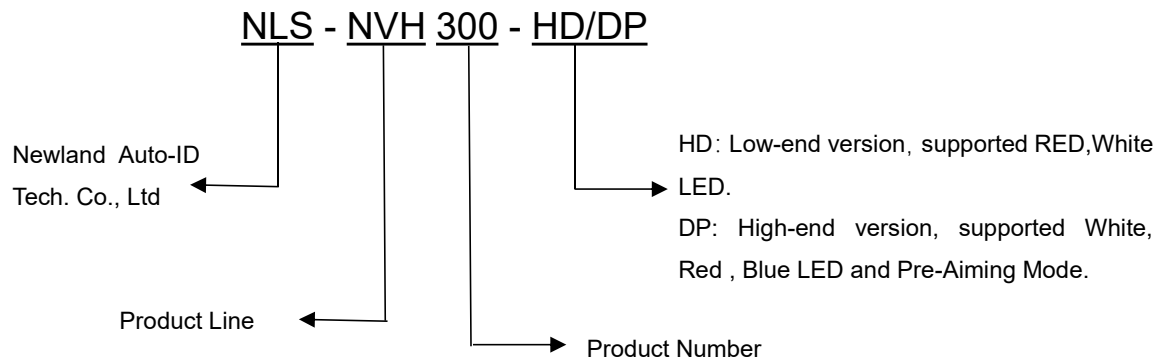
The NVH300 reads a 1D or 2D barcode by capturing its image. Adopting the advanced **UIMC**<sup>®</sup> technology independently developed by Newland Auto-ID Tech, it provides several scan modes, including Manual mode, Pulse mode, Aim mode, Continuous mode and Batch mode, tailored to different scanning needs.

An illustrated introduction to the NVH300 is included in this chapter. If you have a scanner at hand, make good use of it to develop a better understanding of this manual. This chapter is written for normal users, maintenance staff and software developers.

## Features of the NVH300

- Excellent DPM Decoding Capability
- Advanced Illumination system
- Fast Intelligent Learning and Rich Scene Configuration
- High Protection Industrial Structure Design

## Model description



---

## Accessories

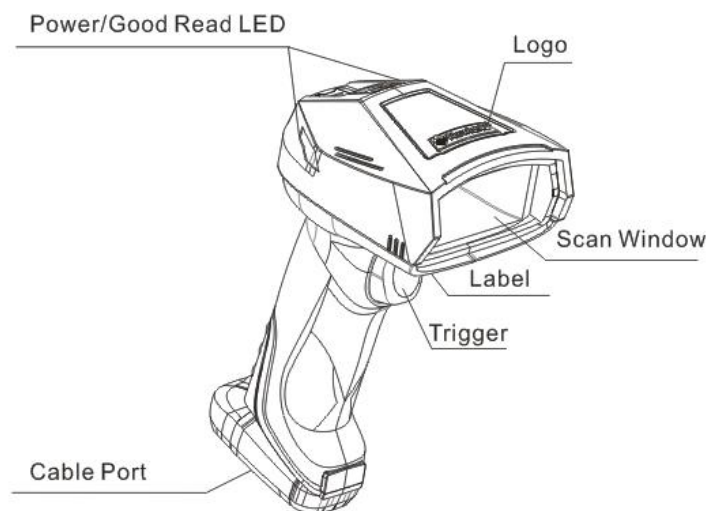
Name	Specification	Number
NLS-NVH300 Scanner	NLS-NVH300-DP/HD	1
USB or RS232 Cable	RJ45 to USB or RS232	1
Power Adapter	Output: DC 5V,1.5A	1
Quick Guide	Paper	1
Qualification Certificate /Warranty Card	Paper	1

## Unpacking

Open the package and take out the scanner and its accessories. Check to make sure everything on the packing list is present and intact. If any contents are damaged or missing, please keep the original package and contact your dealer immediately for after-sales service.

## NVH300 Scanner

### NVH300-DP

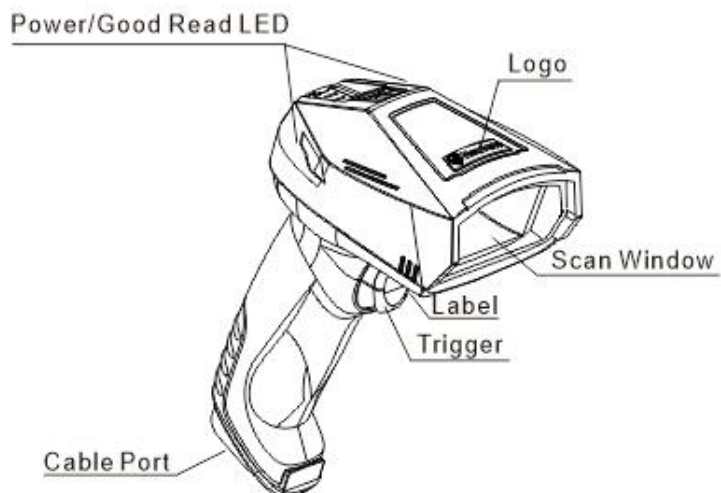


#### Good Read/Network LED:

Red LED: Device is started successfully.

Green LED: Barcode is decoded successfully.

### NVH300-HD



#### Good Read/Network LED:

Red LED: Device is started successfully.

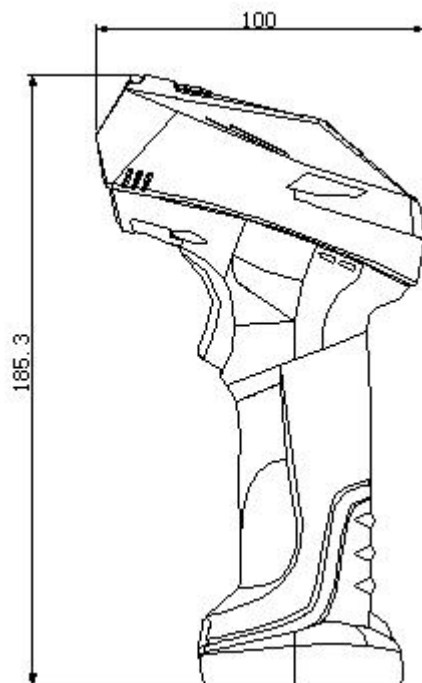
Green LED: Barcode is decoded successfully.

---

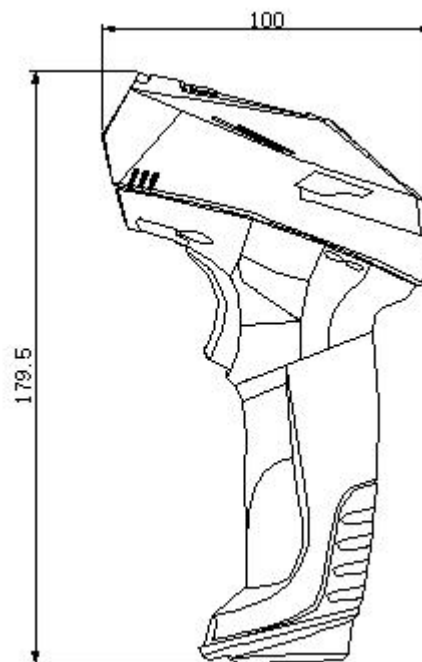
**Dimensions of the Scanner (unit: mm)**

**NVH300-DP**

**Left View**



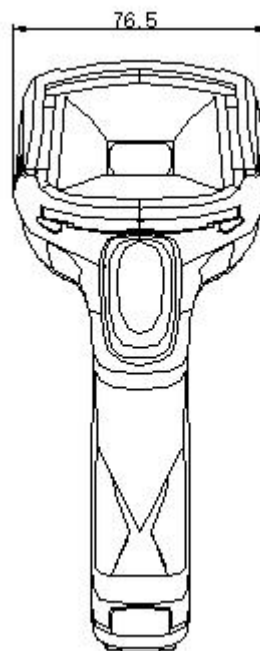
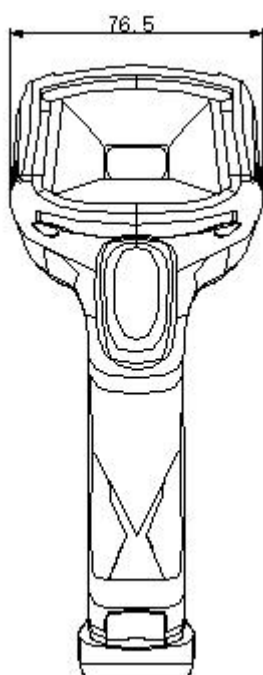
**NVH300-HD**



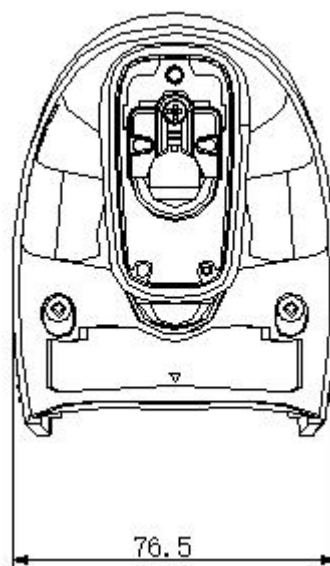
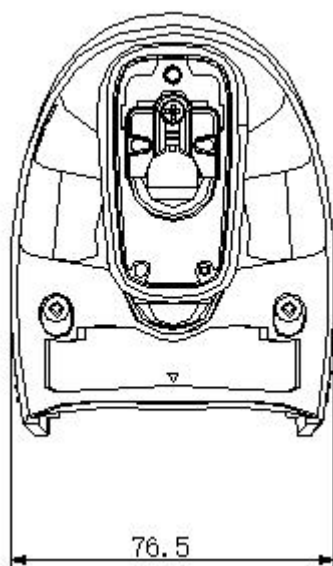


---

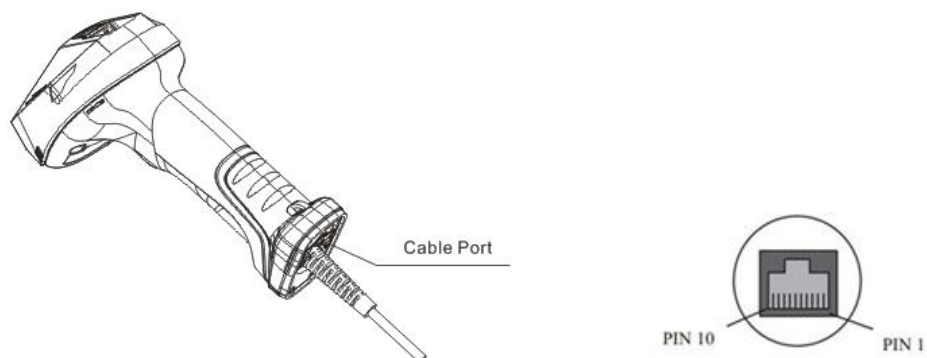
**Front View**



**Top View**



## Data Port



Data Port Pinout:

PIN	Signal	Type	Function
1	KB_CLK	I/O	Keyboard Clock Signal
2	KB_DATA	I/O	Keyboard Data Signal
3	VCC	P	Power, +5V
4	TXD	O	RS-232 Output
5	RXD	I	RS-232 Input
6	CTS	I	Clear To Send
7	RTS	O	Request To Send
8	GND	P	Ground
9	D-	I/O	USB Signal
10	D+	I/O	

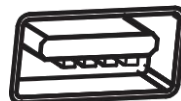
---

## Connecting the NVH300 to a Host

The scanner must be connected to a host device in actual application, such as PC, POS or any intelligent terminal with USB or RS-232, using a communication cable (USB or RS-232 cable).

### USB

USB port on the host device

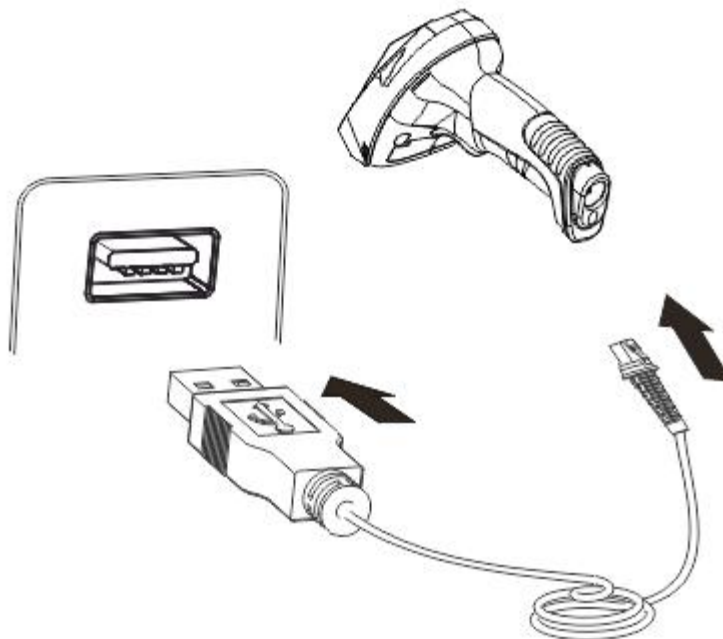


### RS-232

RS-232 port on the host device



## Using USB Cable

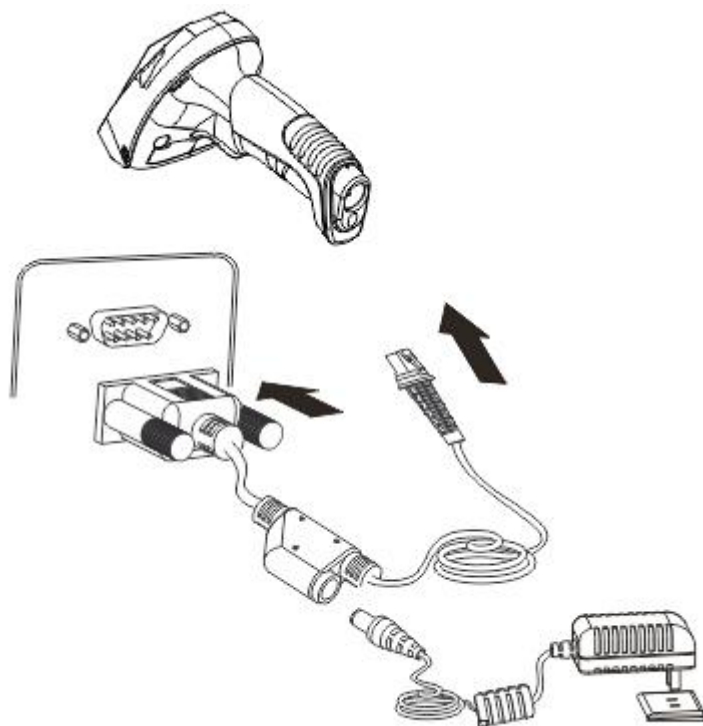


Connect the scanner to a host device through a USB cable with RJ45 and USB connectors:

1. Plug the RJ45 connector into the data port on the scanner.
2. Plug the USB connector into the USB port on the host device.

---

## Using RS-232 Cable



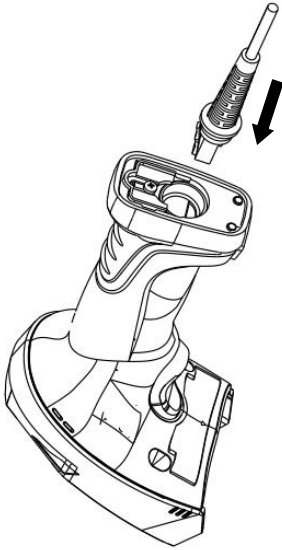
Connect the scanner to a host device through a RS-232 cable with RJ45 and RS-232 connectors and a power jack:

1. Plug the RJ45 connector into the data port on the scanner.
2. Plug the RS-232 connector into the RS-232 port on the host device.
3. Plug the supplied power adapter into the power jack on the RS-232 cable.

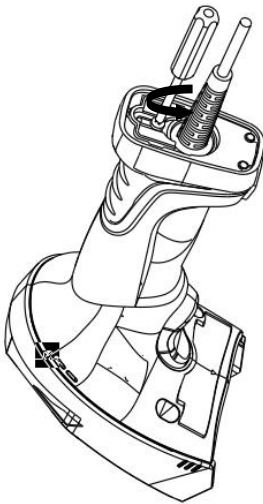
---

## Connect the NVH300-DP Communication Data Cable

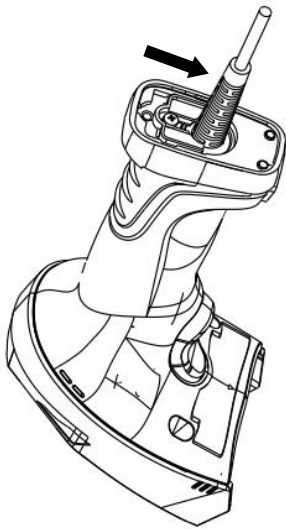
1. Insert cable fully so that the connector is flush with the scanner surface.



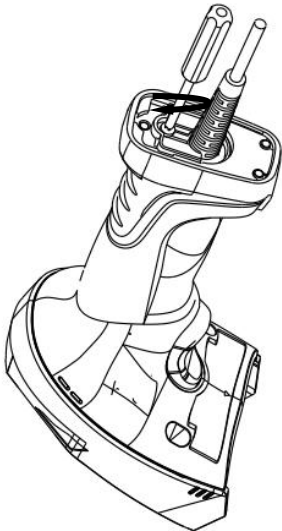
2. Loosen metal lock plate screw using a PH1 driver.



3. Slide lock plate to fully locked position.



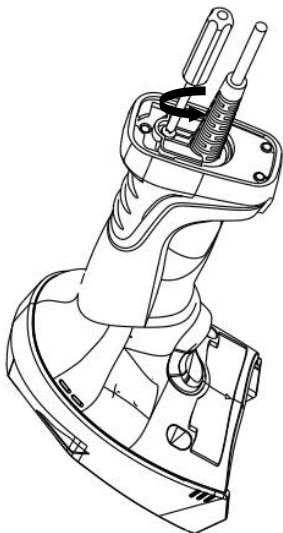
4. Tighten lock plate screw using a PH1 driver.



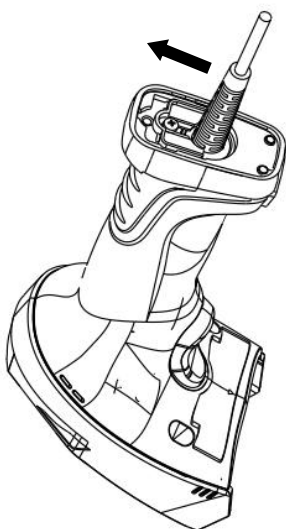
---

## Remove the NVH300-DP Communication Data Cable

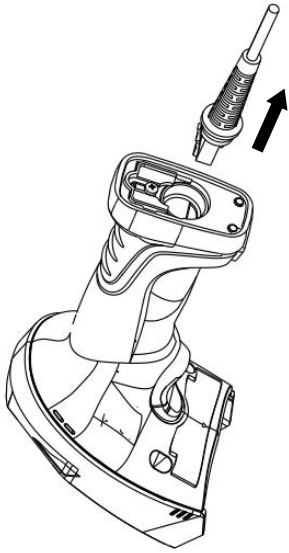
1. Loosen lock plate screw using a PH1 driver.



2. Slide lock plate to fully unlocked position.



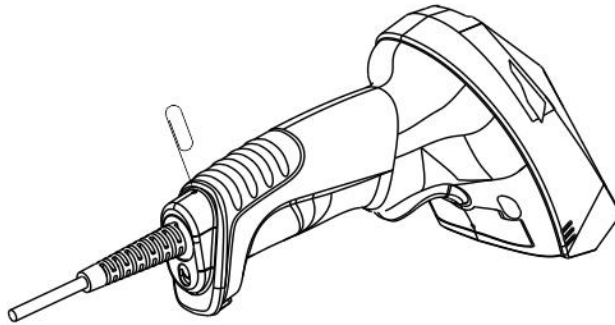
3. Remove cable.





---

## Remove the Cable of NVH300-HD



Get an appropriate needle or a straightened paper clip and then follow the steps below:

1. Disconnect the power adapter from mains if there is one.
2. Insert the needle into the hole.
3. Pull out the cable slowly from the scanner while pressing the needle in.
4. Remove the needle.
5. Disconnect the cable from the host device.

## Power On, Power Off, Reboot

### Power on the scanner

Connect the scanner to a host device. Then the scanner will be turned on and automatically enter the sleep mode.

### Power off the scanner

Remove the cable from the scanner; or remove the cable from the host device; or disconnect the power adapter from mains.

### Reboot the scanner

If the scanner stops responding to input or runs abnormally, turn off the scanner and then turn it back on.

---

## Maintenance

- ✧ The scan window should be kept clean.
- ✧ Do not scratch the scan window.
- ✧ Use soft brush to remove the stain from the scan window.
- ✧ Use the soft cloth to clean the window, such as eyeglass cleaning cloth.
- ✧ Do not spray any liquid on the scan window.
- ✧ Do not use any detergent to clean other parts of the scanner except for water.



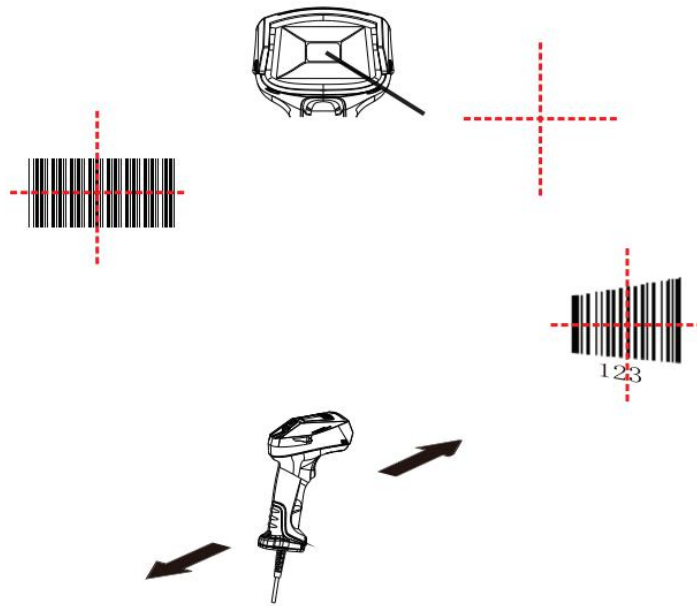
The warranty **DOES NOT** cover damages caused by inappropriate care and maintenance.

## Scanning Instructions

When the scanner is in the Level scan mode, you can follow the steps below to scan a barcode:

1. Press and hold the trigger. Then the scanner will project a red cross focusing dashed line .
2. Center the aiming pattern on the barcode, as shown in the figure below.
3. Release the trigger when the aiming pattern goes off. If the barcode is decoded successfully, the scanner will emit a good read beep and the decoded data will be sent to the host device.

**Note:** For barcodes of the same batch, the scanner keeps a high success ratio in certain distance which is regarded as the optimal scanning distance.



## Chapter 2 NSet

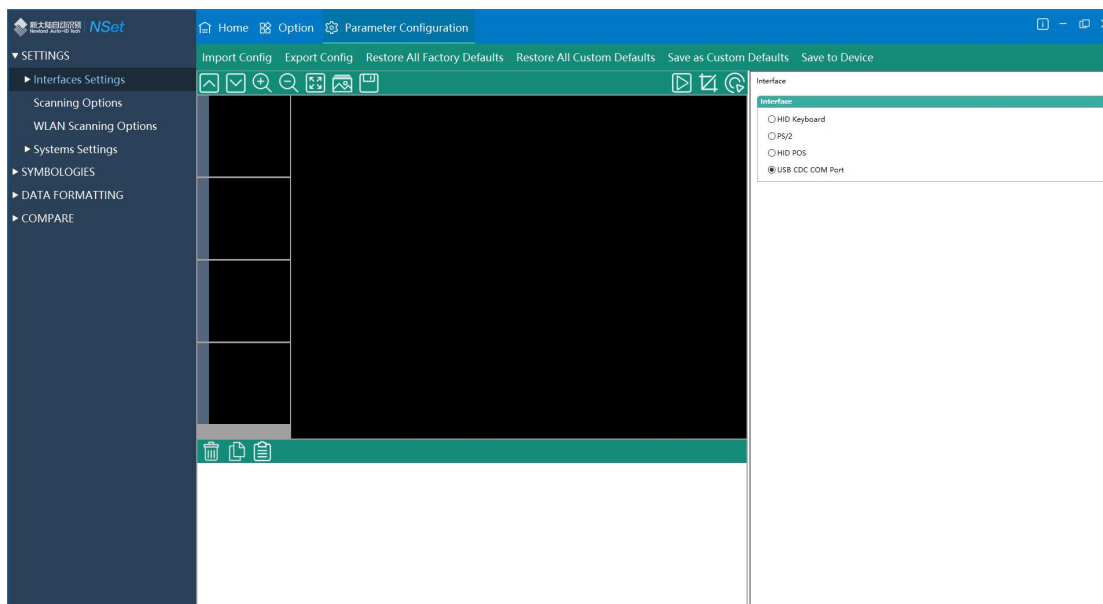
### Introduction

NSet Setup Tool is a User Interface client application developed by Fujian Newland Auto-ID Tech. Co., Ltd. That simplifies initial reader setup and changing parameters of the scanners you use.

The NSet is a common platform across all models. It is installed in and runs on Windows-based PCs (usually laptops), and connection takes place through several different interface.

### Main Features

- View and modify the information and setting parameters of online scanner.
- View the general configurations of the offline devices.
- Upgrade the firmware, compare the configurations, restore to the factory default ,etc in batch.
- Create/print/save setting barcodes to a PDF or Word file.
- Support to see how different options affect the reader in real time.
- Real-time refresh of device status.
- Support image playback.



## Chapter 3 System Settings

### Introduction

There are three ways to configure the NVH300: Barcode programming, command programming and NSet programming.

#### Barcode Programming

The NVH300 can be configured by scanning programming barcodes. All user programmable features/options are described along with their programming barcodes/commands in the following sections.

This programming method is most straightforward. However, it requires manually scanning barcodes. As a result, errors are more likely to occur.

#### Command Programming

The NVH300 can also be configured by serial commands (HEX) sent from the host device.

Users can design an application program to send those command strings to the scanners to perform device configuration.

#### NSet Programming

Besides the two methods mentioned above, you can conveniently perform scanner configuration through NSet too. NSet is a Windows-based configuration tool particularly designed for Newland products, enabling users to gain access to decoded data and captured images and to configure scanners. For more information about this tool, refer to the *NSet User Guide*.



Before programming the scanner, you need to pair it to a cradle first. When programming the scanner with serial commands or NSet, you need to insert it in the paired cradle.

---

## Programming Barcode/ Programming Command/Function



The figure above is an example that shows you the programming barcode and command for the Level Mode function:

1. The **Level Mode** barcode.
2. The **Level Mode** command.
3. The description of feature/option.
4. \*\* indicates factory default settings.

Note: "@" included in the programming command indicates permanent setting which means the setting will not be lost by removing power from the scanner or turning off or rebooting it; whereas "#" included in the programming command indicates temporary setting which means the setting will be lost by removing power from the scanner or turning off or rebooting it.

---

## Use of Programming Command

Besides the barcode programming method, the scanner can also be configured by serial commands (HEX) sent from the host device. **All commands must be entered in uppercase letters.**

### Command Syntax

*Prefix StorageType Tag SubTag {Data} [,SubTag {Data}] [:Tag SubTag {Data}] [...] Suffix*

**Prefix:** "~<SOH>0000" (HEX: **7E 01 30 30 30 30**), 6 characters.

**StorageType:** "@" (HEX: **40**) or "#" (HEX: **23**), 1 character. "@" means permanent setting which will not be lost by removing power from the scanner or rebooting it; "#" means temporary setting which will be lost by removing power from the scanner or rebooting it.

**Tag:** A 3-character case-sensitive field that identifies the desired command group. For example, all USB HID-KBW configuration settings are identified with a Tag of KBW.

**SubTag:** A 3-character case-sensitive field that identifies the desired parameter within the tag group. For example, the SubTag for the keyboard layout is CTY.

**Data:** The value for a feature or parameter setting, identified by the Tag and SubTag.

**Suffix:** "<ETX>" (HEX: **3B 03**), 2 characters.

Multiple commands can be issued within one Prefix/Suffix sequence. For configuration commands, only the **Tag**, **SubTag**, and **Data** fields must be repeated for each command in sequence. If an additional command is to be applied to the same Tag, then the command is separated with a comma (,) and only the **SubTag** and **Data** fields of the additional commands are issued. If the additional command requires a different **Tag** field, the command is separated from previous command by a semicolon (;).

### Query Commands

For query commands, the entry in the **Data** field in the syntax above is one of the following characters means:

- \* (HEX: **2A**)      What is the scanner's current value for the setting(s).
- & (HEX: **26**)      What is the factory default value for the setting(s).
- ^ (HEX: **5E**)      What is the range of possible values for the setting(s).

---

The value of the **StoreType** field in a query command can be either “@” (HEX: **40**) or “#” (HEX: **23**).

A query command with the **SubTag** field omitted means to query all the settings concerning a tag. For example, to query all the current settings about Code 11, you should enter **7E 01 30 30 30 30 40 43 31 31 2A 3B 03** (i.e. ~<SOH>0000@C11\*;<ETX>).

## Responses

Different from command sequence, the prefix of a response consists of the six characters of “<STX><SOH>0000” (HEX: **02 01 30 30 30 30**).

The scanner responds to serial commands with one of the following three responses:

<ACK> (HEX: **06**)      Indicates a good command which has been processed.

<NAK> (HEX: **15**)      Indicates a good configuration command with its **Data** field entry out of the allowable range for this Tag and SubTag combination (e.g. an entry for an inter-keystroke delay of 100 when the field will only allow 2 digits), or an invalid query command.

<ENQ> (HEX: **05**)      Indicates an invalid Tag or SubTag command.

When responding, the scanner echoes back the command sequence with the status character above inserted directly before each of the punctuation marks (the comma or semicolon) in the command.

## Examples

**Example 1: Enable Code 11, set the minimum and maximum lengths to 12 and 22 respectively.**

Enter:      **7E 01 30 30 30 30 40 43 31 31 45 4E 41 31 2C 4D 49 4E 31 32 2C 4D 41 58 32 32 3B 03**  
(~<SOH>0000@C11ENA1,MIN12,MAX22;<ETX>)

Response: **02 01 30 30 30 30 40 43 31 31 45 4E 41 31 06 2C 4D 49 4E 31 32 06 2C 4D 41 58 32 32 06 3B 03**  
(<STX><SOH>0000@C11ENA1<ACK>,MIN12<ACK>,MAX22<ACK>;<ETX>)

**Example 2: Query the current minimum and maximum lengths of Code 11.**

Enter:      **7E 01 30 30 30 30 40 43 31 31 4D 49 4E 2A 2C 4D 41 58 2A 3B 03**  
(~<SOH>0000@C11MIN\*,MAX\*;<ETX>)

Response: **02 01 30 30 30 30 40 43 31 31 4D 49 4E 31 32 06 2C 4D 41 58 32 32 06 3B 03**  
(<STX><SOH>0000@C11MIN12<ACK>,MAX22<ACK>;<ETX>)



---

## Use of Programming Barcode

Programming barcode data (i.e. the characters under programming barcode) can be transmitted to the host device. Scan the appropriate barcode below to enable or disable the transmission of programming barcode data to the host device.



@SETUPT0

**\*\* Do Not Transmit Programming Barcode Data**



@SETUPT1

**Transmit Programming Barcode Data**

---

## Illumination



@ILLSCN1

**\*\* On**



@ILLSCN0

**Off**

## Illumination Color



@ILLCLR0

**Red LED**



@ILLCLR1

**White LED**



@ILLCLR2

**\*\* Red LED and White LED**



@ILLCLR4

**Blue LED**



@ILLCLR5

**Red LED,White LED and Blue LED**



@ILLCLR6

**White LED and Blue LED**



@ILLCLR7

**Red LED and Blue LED**

### **Illumination Rotation Interval**

This parameter sets the Illumination Rotation Interval. The default value is 2000ms.



@ILLSCT

**Illumination Rotation Interval**

## **Aiming**

When scanning/capturing image, the scanner projects an aiming pattern which allows positioning the target barcode within its field of view and thus makes decoding easier.

**Normal:** The scanner projects an aiming pattern only during barcode scanning/capture.

**Always On:** Aiming pattern is constantly on after the scanner is powered on.

**Off:** Aiming pattern is off all the time.



@AMLENA1

**\*\* Normal**



@AMLENA0

**Off**



@AMLENA2

**Always On**

---

## Good Read LED

The green LED can be programmed to be On or Off to indicate good read.



@GRLENA1

**\*\* On**



@GRLENA0

**Off**

---

## Good Read LED Duration

This parameter sets the amount of time the Good Read LED remains on following a good read. It is programmable in 10ms increments from 20ms to 10,000ms.



@GRLDUR20

**\*\* Short (20ms)**



@GRLDUR120

**Medium (120ms)**



@GRLDUR220

**Long (220ms)**



@GRLDUR320

**Prolonged (320ms)**



@GRLDUR

**Custom (20 - 10,000ms)**

**E**  
*xample*

**Set the Good Read LED duration to 800ms:**

1. Scan the **Custom** barcode.
2. Scan the numeric barcodes “8”, “0” and “0” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.

---

## Power On Beep

The scanner can be programmed to beep when it is powered on. Scan the **Off** barcode if you do not want a power on beep.



@PWBENA1

**\*\* On**



@PWBENA0

**Off**

## Good Read Beep

Scanning the **Off** barcode can turn off the beep that indicates successful decode; scanning the **On** barcode can turn it back on.



@GRBENA1

**\*\* On**



@GRBENA0

**Off**

---

## Good Read Beep Volume

Scanning the below barcode can set the Good Read Beep Volume to low, medium and high.



@GRBVLL20

**Low Volume**



@GRBVLL12

**Medium Volume**



@GRBVLL5

**\*\*High Volume**



@GRBVLL

**Custom (1-10)**



## Good Read Beep Duration

This parameter sets the length of the beep the scanner emits on a good read. It is programmable in 10ms increments from 20ms to 300ms.



@GRBDUR40

**Short (40ms)**



@GRBDUR80

**\*\* Medium (80ms)**



@GRBDUR120

**Long (120ms)**



@GRBDUR

**Custom (20 - 300ms)**

**E**  
*xample*

**Set the Good Read Beep duration to 200ms:**

1. Scan the **Custom** barcode.
2. Scan the numeric barcodes “2”, “0” and “0” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.

---

## Good Read Beep Frequency

This parameter is programmable in 1Hz increments from 20Hz to 20,000Hz.



@GRBFRQ800

**Extra Low (800Hz)**



@GRBFRQ1600

**Low (1600Hz)**



@GRBFRQ2730

**\*\* Medium (2730Hz)**



@GRBFRQ4200

**High (4200Hz)**



@GRBFRQ

**Custom (20 - 20,000Hz)**

**E**  
*sample*

**Set the Good Read Beep frequency to 2,000Hz:**

1. Scan the **Custom** barcode.
2. Scan the numeric barcodes “2”, “0”, “0” and “0” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.

---

## Number of Good Read Beeps

The number of beeps of a good read can be programmed from 1 to 9. To change the number of beeps, scan the barcode below and then scan a numeric barcode (1-9) and the **Save** barcode. The default value is 1.



@GRBNUM

**Number of Good Read Beeps**

---

## Good Read Beep Interval Time

Good Read Beep Interval Time modifies the interval time between beeps, and it becomes effective when the Number of Good Read Beeps is greater than one.



@GRBITV0

**\*\* Short**



@GRBITV1

**Medium**

## Good Read Vibration

You may choose whether or not to let the scanner vibrate when a barcode is successfully read by scanning the appropriate barcode below.



@GRVENA0

**\*\* Off**



@GRVENA1

**On**

---

## Good Read Vibration Duration

This parameter sets the length for the Good Read Vibration. It is programmable in 1ms increments from 100ms to 2,000ms. The default value is 300ms.



@GRVDUR

### Set Good Read Vibration Duration

**E**  
*xample*

#### Set the Good Read Vibration duration to 800ms:

1. Scan the Set Good Read Vibration Duration barcode.
2. Scan the numeric barcodes “8”, “0” and “0” from the “Digit Barcodes” section in Appendix.
3. Scan the Save barcode from the “Save/Cancel Barcodes” section in Appendix.

## Scan Mode

**Level Mode:** A trigger pull activates a decode session. The decode session continues until a barcode is decoded or the trigger is released.

**Trigger Mode:** A trigger pull activates a decode session. The decode session continues until a barcode is decoded or reach the decode session timeout.

**Sense Mode:** The scanner activates a decode session every time when it detects a change in ambient illumination. The decode session continues until the barcode is decoded or the decode session timeout expires. Pressing the trigger can also activate a decode session.

**Continuous Mode:** The scanner automatically starts one decode session after another. To suspend/resume barcode reading, simply press the trigger. **Reread Timeout** can avoid undesired rereading of same barcode in a given period of time.

**Pulse Mode:** When the trigger is pulled and released, scanning is activated until a barcode is decoded or the decode session timeout expires (The decode session timeout begins when the trigger is released).

**Batch Mode:** A trigger pull activates a round of multiple decode sessions. This round of multiple scans continues until the trigger is released. Rereading the same barcode is not allowed in the same round.

---

**Pre-Aiming Mode:**When the user holds the scanner , the scanner lights up the aiming light to indicate the current position of the scanner before the user presses the button. After pressing the button, the scanner starts to read the bar code( the aimer is turned on or off according to the configuration of aimer mode. When the button is released or the barcode is decooded successfully, the code is stopped. When the user drops down the scanner for 3 seconds, the aiming lamp will turn off.



@SCNMOD0

**Level Mode**



@SCNMOD1

**Trigger Mode**



@SCNMOD2

**Sense Mode**



@SCNMOD3

**Continuous Mode**



@SCNMOD4

**Pulse Mode**



@SCNMOD7

**Batch Mode**



@SCNMOD13

**\*\* Pre-Aiming Mode**

---

## One Reading Timeout

This parameter sets the maximum time decode session continues during a scan attempt. It is programmable in 1ms increments from 1ms to 3,600,000ms. When it is set to 0, the timeout is infinite. The default setting is 3,000ms.



@ORTSET

**One Reading Timeout**

**E**  
*xample*

### Set the One Reading Timeout to 1,500ms:

1. Scan the **One Reading Timeout** barcode.
2. Scan the numeric barcodes “1”, “5”, “0” and “0” from the “Digit Barcodes” section in Appendix.
3. Scan the Save barcode from the “Save/Cancel Barcodes” section in Appendix.

## Reading Interval

Reading Interval sets the time period between the end of one decode session and the start of next session. It is programmable in 1ms increments from 1ms to 10,000ms. The default timeout is 500ms. This feature is only applicable to the Continuous modes.



@SCNINV

**Reading Interval**

**E**  
*xample*

### Set the timeout between decodes to 800ms:

1. Scan the **Reading Interval** barcode.
2. Scan the numeric barcodes “8”, “0” and “0” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.



---

## Image Stabilization Timeout (Sense Mode)

In the course of the sense mode, when the scanner stop reading, it will enter a new process of read-adaptation to the reading environment (image).After the Image Stabilization Timeout will the scanner enter the sense mode and wait for the barcode to present.Set the parameters of Image Stabilization Timeout could change the time to adapt the reading environment. It is programmable in 1ms increments from 1ms to 3,000ms. The default setting is 200ms.



@SENIST

**Image Stabilization Timeout (Sense Mode)**

## Good Read Delay

Good Read Delay sets the minimum amount of time before the scanner can read another barcode. This parameter is programmable in 1ms increments from 1ms to 3,600,000ms. The default setting is 500ms. Scan the appropriate barcode below to enable or disable the delay.



@GRDENA1

**Enable Good Read Delay**



@GRDENA0

**\*\* Disable Good Read Delay**

To set the good read delay, scan the barcode below, then set the delay (from 1 to 3,600,000ms) by scanning the digit barcode(s) then scanning the **Save** barcode from the Appendix.



@GRDDUR

### Good Read Delay

**E**  
*example*

#### Set the good read delay to 1,000ms:

1. Scan the Good Read Delay barcode.
2. Scan the numeric barcodes “1”, “0”, “0” and “0” from the “Digit Barcodes” section in Appendix.
3. Scan the Save barcode from the “Save/Cancel Barcodes” section in Appendix.

---

## Image Decoding Timeout

Image Decoding Timeout specifies the maximum time the scanner will spend decoding an image. This parameter is programmable in 1ms increments from 1ms to 3,000ms. The default timeout is 500ms.



@DETSET

**Image Decoding Timeout**

**E**  
*xample*

**Set the image decoding timeout to 800ms:**

1. Scan the **Image Decoding Timeout** barcode.
2. Scan the numeric barcodes “8”, “0” and “0” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.

---

## Scanning Preference

**Normal Mode:** Select this mode when reading barcodes on paper.

**Screen Mode:** Select this mode when reading barcodes on the screen.

**High Motion Tolerance Mode:** Select this mode when reading barcodes on moving objects.

**Wide Adaptive Mode:** Select this mode when application scenarios and code types are diversified.



@EXPLVL0

**\*\* Normal Mode**



@EXPLVL4

**High Motion Tolerance Mode**



@EXPLVL5

**Screen Mode**



@EXPLVL6

**Wide Adaptive Mode**

---

# Sensitivity

Sensitivity specifies the degree of acuteness of the scanner’s response to changes in images captured. There are 20 levels to choose from. The smaller the value, the higher the sensitivity and the lower requirement in image change to trigger the scanner. You can select an appropriate degree of sensitivity that fits the application environment. The default setting is Level 1. This feature is only applicable to the Sense mode.



@SENLVL14

**Low Sensitivity**



@SENLVL11

**Medium Sensitivity**



@SENLVL8

**High Sensitivity**



@SENLVL5

**Enhance Sensitivity**



@SENLVL

**\*\*Custom Sensitivity**

---

## Example

### Set the Sensitivity to 10:

1. Scan the **Custom Sensitivity** barcode.
2. Scan the numeric barcodes "1", "0" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

## Read Barcode On/Off

Sending the Read Barcode Off command **@NGRENA0** to the scanner can disable it from reading barcode, and the scanner is unable to scan barcode unless you send the Read Barcode On command **@NGRENA1** to it or power cycle it. By default, Read Barcode is On.

## Decode Area

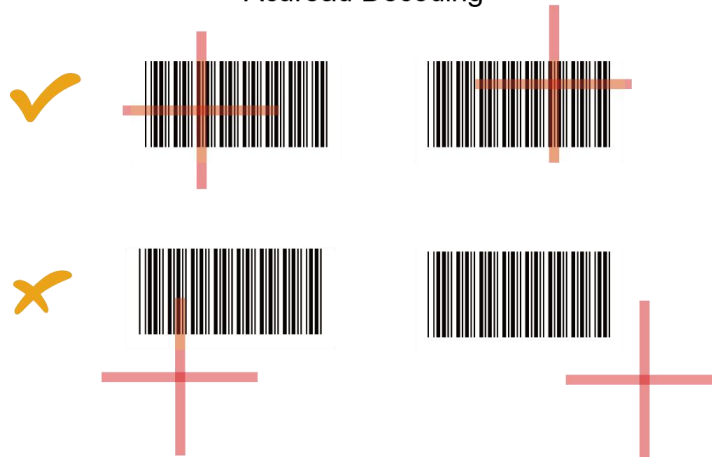
**Whole Area Decoding:** The scanner attempts to decode barcode(s) within its field of view, from the center to the periphery, and transmits the barcode that has been first decoded.

**Specific Area Decoding:** The scanner attempts to read barcode(s) within a specified decoding area and transmits the barcode that has been first decoded. This option allows the scanner to narrow its field of view to make sure it reads only those barcodes intended by the user. For instance, if multiple barcodes are placed closely together, specific area decoding in conjunction with appropriate pre-defined decoding area will insure that only the desired barcode is read.

**Acuscan:** The scanner only decodes the barcode aimed squarely by the aiming pattern. For those using a crosshair aiming pattern, only the barcode aimed by the center of crosshair will be decoded.

**Priori On Aimed Barcode:** The scanner attempts to decode priority the aimed barcode(s). If there is no barcode in the aimed area, the scanner would decode the barcode which is close to the aiming area first.

## Acuread Decoding





@CADENA0

**Whole Area Decoding**



@CADENA1

**Specific Area Decoding**



@CADENA2

**Acuscan**



@CADENA4;

**\*\*Priori On Aimed Barcode**



---

If **Specific Area Decoding** is enabled, the scanner only reads barcodes that intersect the predefined decoding area.

The default decoding area is an area of 40% top, 60% bottom, 40% left and 60% right of the scanner's field of view

You can define the decoding area using the **Top of Decoding Area**, **Bottom of Decoding Area**, **Left of Decoding Area** and **Right of Decoding Area** barcodes as well as numeric barcode(s) that represent(s) a desired percentage (0-100). The value of Bottom must be greater than that of Top; the value of Right must be greater than that of Left.



@CADTOP

**Top of Decoding Area**



@CADBOT

**Bottom of Decoding Area**



@CADLEF

**Left of Decoding Area**



@CADRIG

**Right of Decoding Area**



## Example

**Program the scanner to only read Barcode 1 in the figure above by setting the decoding area to 10% top, 45% bottom, 15% left and 30% right:**

1. Scan the **Top of Decoding Area** barcode.
2. Scan the numeric barcode "0" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Bottom of Decoding Area** barcode.
5. Scan the numeric barcodes "4" and "5" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
7. Scan the **Top of Decoding Area** barcode.
8. Scan the numeric barcodes "1" and "0" from the "Digit Barcodes" section in Appendix.
9. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
10. Scan the **Left of Decoding Area** barcode.
11. Scan the numeric barcode "0" from the "Digit Barcodes" section in Appendix.
12. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
13. Scan the **Right of Decoding Area** barcode.
14. Scan the numeric barcodes "3" and "0" from the "Digit Barcodes" section in Appendix.
15. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
16. Scan the **Left of Decoding Area** barcode.
17. Scan the numeric barcodes "1" and "5" from the "Digit Barcodes" section in Appendix.
18. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## Image Flipping



@MIRROR0

**\*\* Do Not Flip**

Example of image not flipped



@MIRROR1

**Flip Horizontally**

Example of image flipped horizontally



---

## Image Resize

This option alters image resolution before compression. Multiple pixels are combined to one pixel, resulting in a smaller image containing the original content with reduced resolution. You can set to original image or resize by 50% on both horizontal and vertical.



@IMGRSE0

**\*\* Original Image**



@IMGRSE1

**Resize By 50% On Both Horizontal And  
Vertical**

---

## Transmit Not Good Read Message

- ✧ Enable: When the code failed to decoded, and you release the trigger or Decode Session Timeout , Not Good Read Message will be sent to the host.
- ✧ Disable: When the code failed to decoded, Not Good Read Message will not be sent to the host.



@NGRENA0

\*\* Disable



@NGRENA1

Enable

## Edit NGR Message

Not Good Read Message supports up to 7 characters, from 0 to 0xff. (Default: NG)



@NGRSET

Edit NGR Message

## Image Filter

Sets the filter to be applied to the image before being processed. This parameter can be used to successfully decode

---

particular ink-spread printed codes (ex. direct part mark codes). Different filters can be applied to a single code. See the NSet User's Manual for examples of Image Filter application.

## Default Settings

### Factory Defaults

Scanning the following barcode can restore the scanner to the factory defaults.

You may need to reset all parameters to the factory defaults when:

- ✧ scanner is not properly configured so that it fails to decode barcodes.
- ✧ you forget previous configuration and want to avoid its impact.



@FACDEF

**Restore All Factory Defaults**

---

## Custom Defaults

Scanning the **Restore All Custom Defaults** barcode can reset all parameters on the scanner to the custom defaults. Scanning the **Save as Custom Defaults** barcode can set the current settings as custom defaults.

Custom defaults are stored in the non-volatile memory.



@CUSSAV

**Save as Custom Defaults**



@CUSDEF

**Restore All Custom Defaults**



Restoring the scanner to the factory defaults will not remove the custom defaults from the scanner.

---

## Query Scanner Information

After scanning the barcode below, the scanner information (including product name, firmware version, decoder version, hardware version, product serial number, OEM serial number and manufacturing date) will be sent to the host device.



@QRYSYS

**Query Scanner Information**

## Query Product Name



@QRYPDN

**Query Product Name**

## Query Firmware Version



@QRYFWV

**Query Firmware Version**



---

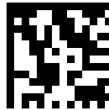
### Query Decoder Version



@QRYDCV

**Query Decoder Version**

### Query Hardware Version



@QRYHWV

**Query Hardware Version**

### Query Serial Number



@QRYPSN

**Query Serial Number**

### Query Manufacturing Date



@QRYDAT

**Query Manufacturing Date**

---

## Chapter 4 RS-232 Interface

### Introduction

When the cradle is connected to the RS-232 port of a host device, you need to set communication parameters (including baud rate, parity check, data bit and stop bit) on the cradle to match the host device so that two devices can communicate with each other.



@INTERF0

**RS-232**

---

## Baud Rate

Baud rate is the number of bits of data transmitted per second. Set the baud rate to match the host requirements.



@232BAD8

**115200**



@232BAD7

**57600**



@232BAD6

**38400**



@232BAD5

**19200**



@232BAD4

**14400**



@232BAD2

**4800**



@232BAD0

**1200**



@232BAD3

**\*\*9600**



@232BAD1

**2400**

## Parity Check

Set the parity type to match the host requirements.

**Odd Parity:** If the data contains an odd number of 1 bits, the parity bit value is set to 0.

**Even Parity:** If the data contains an even number of 1 bits, the parity bit value is set to 0.

**None:** Select this option when no parity bit is required.



@232PAR0

**\*\* None**



@232PAR2

**Odd Parity**



@232PAR1

**Even Parity**

---

## Data Bit

Set the number of data bits to match the host requirements.



@232DAT1

**7 Data Bits**



@232DAT0

**\*\* 8 Data Bits**

## Stop Bit

The stop bit(s) at the end of each transmitted character marks the end of transmission of one character and prepares the receiving device for the next character in the serial data stream. Set the number of stop bits to match the host requirements.



@232STP0

**\*\* 1 Stop Bit**



@232STP1

**2 Stop Bits**

---

## Hardware Auto Flow Control

If this feature is enabled, the cradle determines whether to transmit data based on CTS signal level. When CTS signal is at a low level which means the serial port's cache memory of receiving device (such as PC) is full, the cradle sends data through RS-232 port until CTS signal is set to high level by receiving device. When the cradle is not ready for receiving, it will set RTS signal to low level. When sending device (such as PC) detects it, it will not send data to the cradle any more to prevent data loss.

If this feature is disabled, reception/transmission of serial data will not be influenced by RTS/CTS signal.



@232AFL0

**\*\* Disable Hardware Auto Flow Control**



@232AFL1

**Enable Hardware Auto Flow Control**



Before enabling this feature, make sure that RTS/CTS signal lines are contained in RS-232 cable. Without the signal lines, RS-232 communication errors will occur.

---

## Chapter 5 USB Interface

### Introduction

There are two options for USB connection between the cradle and host device:

- ✧ USB HID Keyboard: The cradle's transmission is simulated as USB keyboard input with no need for command configuration or a driver. Barcode data could be entered by the virtual keyboard directly and it is also convenient for the host device to receive data.
- ✧ USB CDC: It is compliant with the standard USB CDC class specifications defined by the USB-IF and allows the host device to receive data in the way as a serial port does. A driver is needed when using this feature.

When the scanner is connected to both USB and RS-232 ports on a host device, it will select the USB connection by default.



---

## USB HID Keyboard

When the cradle is connected to the USB port on a host device, you can enable the USB HID Keyboard feature by scanning the barcode below. The cradle's transmission will be simulated as USB keyboard input. The Host receives keystrokes on the virtual keyboard. It works on a Plug and Play basis and no driver is required.



@INTERF3

**\*\* USB HID Keyboard**



If the host device allows keyboard input, then no extra software is needed for HID Keyboard input.

---

## USB Country Keyboard Types

Keyboard layouts vary from country to country. The default setting is U.S. keyboard.



@KBWCTY0

**\*\* U.S. (English)**



@KBWCTY1

**Belgium**



@KBWCTY2

**Brazil**



@KBWCTY3

**Canada (French)**



@KBWCTY4

**Czechoslovakia**



@KBWCTY5

**Denmark**



@KBWCTY6

**Finland (Swedish)**



@KBWCTY7

**France**



@KBWCTY8

**Germany/ Austria**



@KBWCTY9

**Greece**



@KBWCTY10

**Hungary**



@KBWCTY11

**Israel (Hebrew)**



@KBWCTY12

**Italy**



@KBWCTY13

**Latin America/ South America**



@KBWCTY14

**Netherlands (Dutch)**



@KBWCTY15

**Norway**



@KBWCTY16

**Poland**



@KBWCTY17

**Portugal**



@KBWCTY18

**Romania**



@KBWCTY19

**Russia**



@KBWCTY21

**Slovakia**



@KBWCTY22

**Spain**



@KBWCTY23

**Sweden**



@KBWCTY24

**Switzerland (German)**



@KBWCTY25

**Turkey\_F**



@KBWCTY26

**Turkey\_Q**



@KBWCTY27

**UK**



@KBWCTY28

**Japan**

### **Beep on Unknown Character**

Due to the differences in keyboard layouts, some characters contained in barcode data may be unavailable on the selected keyboard. As a result, the scanner fails to transmit the unknown characters.

Scan the appropriate barcode below to enable or disable the emission of beep when an unknown character is detected.



@KBWBUC0

**\*\* Do Not Beep on Unknown Character**



@KBWBUC1

**Beep on Unknown Character**

---

## Emulate ALT+Keypad

When **Emulate ALT+Keypad** is turned on, any ASCII character (0x00 - 0xFF) is sent over the numeric keypad no matter which keyboard type is selected.

1. ALT Make
2. Enter the number corresponding to a desired character on the keypad.
3. ALT Break

After **Emulate ALT+Keypad ON** is selected, you need to choose the code page with which the barcodes were created and to turn **Unicode Encoding** On or Off depending on the encoding used by the application software.



@KBWALT0

**\*\* Emulate ALT+Keypad OFF**



@KBWALT1

**Emulate ALT+Keypad ON**



Since sending a character involves multiple keystroke emulations, this method appears less efficient.



Supposing **Emulate ALT+Keypad** is ON, **Unicode Encoding** is Off, and **Code Page 1252 (West European Latin)** is selected, barcode data "ADF" (65/208/70) is sent as below:

"A" -- "ALT Make" + "065" + "ALT Break"

"D" -- "ALT Make" + "208" + "ALT Break"

"F" -- "ALT Make" + "070" + "ALT Break"

---

## Code Page

Code pages define the mapping of character codes to characters. If the data received does not display with the proper characters, it may be because the barcode being scanned was created using a code page that is different from the one the host program is expecting. If this is the case, select the code page with which the barcodes were created by scanning the appropriate barcode below. For PDF417, QR Code, Aztec and Data Matrix, besides setting the code page, you also need to set the character encoding in the “Character Encoding” section in Chapter 7. This feature is only effective when **Emulate ALT+Keypad** is turned on.



@KBWCPCG0

**\*\* Code Page 1252 (West European Latin)**



@KBWCPCG1

**Code Page 1251 (Cyrillic)**



@KBWCPCG2

**Code Page 1250 (Central and East European Latin)**



@KBWCPCG3

**Code Page 1253 (Greek)**



@KBWCPCG4

**Code Page 1254 (Turkish)**





@KBWCPG5

**Code Page 1255 (Hebrew)**



@KBWCPG6

**Code Page 1256 (Arabic)**



@KBWCPG7

**Code Page 1257 (Baltic)**



@KBWCPG8

**Code Page 1258 (Vietnamese)**



@KBWCPG9

**Code Page 936 (Simplified Chinese, GB2312,GBK)**



@KBWCPG10

**Code Page 950 (Traditional Chinese, Big5)**



@KBWCPG11

**Code Page 874(Thai)**



@KBWC PG12

**Code Page 932 (Japanese, Shift-JIS)**

---

## Unicode Encoding

Different host program may use different character encodings for handling incoming barcode data. For instance, Microsoft Office Word uses Unicode encoding and therefore you should turn **Unicode Encoding** on, whereas Microsoft Office Excel or Notepad uses Code Page encoding and therefore you should turn **Unicode Encoding** off. This feature is only effective when **Emulate ALT+Keypad** is turned on.



@KBWCPU0

**\*\* Off**



@KBWCPU1

**On**

---

## Function Key Mapping

When **Ctrl+ASCII Mode** is selected, function characters (0x00 - 0x1F) are sent as ASCII sequences.

1. CTRL Make
2. Press function key
3. CTRL Break



@KBWFKM0

**\*\* Disable**



@KBWFKM1

**Ctrl+ASCII Mode**

**E**  
*xample*

If **Ctrl+ASCII Mode** is selected and other parameters of USB HID Keyboard adopt factory defaults, barcode data "A<HT>(i.e. Horizontal Tab)F" (0x41/0x09/0x46) is sent as below:

"A" - Keystroke "A".

<HT> - "Ctrl Make" + Keystroke "I" + "Ctrl Break"

"F" - Keystroke "F"

For some text editors, "Ctrl I" means italic convert. So the output may be "AF".



**Emulate ALT+Keypad ON** prevails over **Ctrl+ASCII Mode**.

## ASCII Function Key Mapping Table

ASCII Function	ASCII Value (HEX)	Function Key Mapping Disabled	Ctrl+ASCII
NUL	00	Null	Ctrl+@
SOH	01	Keypad Enter	Ctrl+A
STX	02	Caps Lock	Ctrl+B
ETX	03	ALT	Ctrl+C
EOT	04	Null	Ctrl+D
ENQ	05	CTRL	Ctrl+E
ACK	06	Null	Ctrl+F
BEL	07	Enter	Ctrl+G
BS	08	Left Arrow	Ctrl+H
HT	09	Horizontal Tab	Ctrl+I
LF	0A	Down Arrow	Ctrl+J
VT	0B	Vertical Tab	Ctrl+K
FF	0C	Delete	Ctrl+L
CR	0D	Enter	Ctrl+M
SO	0E	Insert	Ctrl+N
SI	0F	Esc	Ctrl+O
DLE	10	F11	Ctrl+P
DC1	11	Home	Ctrl+Q
DC2	12	PrintScreen	Ctrl+R
DC3	13	Backspace	Ctrl+S
DC4	14	tab+shift	Ctrl+T
NAK	15	F12	Ctrl+U
SYN	16	F1	Ctrl+V
ETB	17	F2	Ctrl+W
CAN	18	F3	Ctrl+X
EM	19	F4	Ctrl+Y
SUB	1A	F5	Ctrl+Z
ESC	11	F6	Ctrl+[
FS	1C	F7	Ctrl+\
GS	1D	F8	Ctrl+]
RS	1E	F9	Ctrl+6
US	1F	F10	Ctrl+-

## ASCII Function Key Mapping Table (Continued)

The last five characters (0x1B~0x1F) in the table above apply to US keyboard layout only. The following chart provides the equivalents of these five characters for other countries.

Country	Ctrl+ASCII					
United States	Ctrl+[	Ctrl+\	Ctrl+]	Ctrl+6	Ctrl+-	
Belgium	Ctrl+[	Ctrl+<	Ctrl+]	Ctrl+6	Ctrl+-	
Scandinavia	Ctrl+8	Ctrl+<	Ctrl+9	Ctrl+6	Ctrl+-	
France	Ctrl+^	Ctrl+8	Ctrl+\$	Ctrl+6	Ctrl+=	
Germany		Ctrl+Ã	Ctrl++	Ctrl+6	Ctrl+-	
Italy		Ctrl+\	Ctrl++	Ctrl+6	Ctrl+-	
Switzerland		Ctrl+<	Ctrl+..	Ctrl+6	Ctrl+-	
United Kingdom	Ctrl+[	Ctrl+ ¢	Ctrl+]	Ctrl+6	Ctrl+-	
Denmark	Ctrl+8	Ctrl+\	Ctrl+9	Ctrl+6	Ctrl+-	
Norway	Ctrl+8	Ctrl+\	Ctrl+9	Ctrl+6	Ctrl+-	
Spain	Ctrl+[	Ctrl+\	Ctrl+]	Ctrl+6	Ctrl+-	

---

## Inter-Keystroke Delay

This parameter specifies the delay between emulated keystrokes.



@KBWDLY0

**\*\* No Delay**



@KBWDLY40

**Long Delay (40ms)**



@KBWDLY20

**Short Delay (20ms)**

## Caps Lock

The **Caps Lock ON** option can invert upper and lower case characters contained in barcode data. This inversion occurs regardless of the state of Caps Lock key on the host device's keyboard.



@KBWCAP0

**\*\* Caps Lock OFF**



@KBWCAP1

**Caps Lock ON**



**Emulate ALT+Keypad ON/ Convert All to Upper Case/ Convert All to Lower Case** prevails over **Caps Lock ON**.



When the **Caps Lock ON** feature is selected, barcode data "AbC" is transmitted as "aBc".

---

## Convert Case

You may scan the appropriate barcode below to convert all barcode data to your desired case.



@KBWCAS0

**\*\* No Case Conversion**



@KBWCAS1

**Convert All to Upper Case**



@KBWCAS2

**Convert All to Lower Case**

**E**  
*sample*

When the **Convert All to Lower Case** feature is enabled, barcode data “AbC” is transmitted as “abc”.



If **Emulate ALT+Keypad ON** is selected, **Convert All to Lower Case** and **Convert All to Upper Case** do not function.

## Emulate Numeric Keypad





---

**Do Not Emulate Numeric Keypad 1:** Sending a number (0-9) is emulated as keystroke(s) on main keyboard.

**Emulate Numeric Keypad 1:** Sending a number (0-9) is emulated as keystroke(s) on numeric keypad. The state of Num Lock on the simulated numeric keypad is determined by its equivalent on the host device. If Num Lock on the host device is turned off, the output of simulated numeric keypad is function key instead of number.

**Do Not Emulate Numeric Keypad 2:** Sending "+", "-", "\*\*" and "/" is emulated as keystroke(s) on main keyboard.

**Emulate Numeric Keypad 2:** Sending "+", "-", "\*\*" and "/" is emulated as keystroke(s) on numeric keypad.



@KBWNUM0

**\*\* Do Not Emulate Numeric Keypad 1**



@KBWNUM1

**Emulate Numeric Keypad 1**



@KBWNCH0

**\*\* Do Not Emulate Numeric Keypad 2**



@KBWNCH1

**Emulate Numeric Keypad 2**



**Emulate ALT+Keypad ON** prevails over **Emulate Numeric Keypad**.

## **E** *example*

Supposing the **Emulate Numeric Keypad 1** feature is enabled:

if Num Lock on the host device is ON, "A4.5" is transmitted as "A4.5";

if Num Lock on the host device is OFF, "A4.5" is transmitted as ".A":

1. "A" is sent on main keyboard;
2. "4" is sent as the function key "Cursor Move to Left";
3. "." is sent on main keyboard;
4. "5" is not sent as it does not correspond to any function key.

---

## Polling Rate

This parameter specifies the polling rate for a USB keyboard. If the Host drops characters, change the polling rate to a bigger value.



@KBWPOR0

**1ms**



@KBWPOR1

**2ms**



@KBWPOR2

**3ms**



@KBWPOR3

**\*\* 4ms**



@KBWPOR4

**5ms**



@KBWPOR5

**6ms**



@KBWPOR6

**7ms**



@KBWPOR7

**8ms**



@KBWPOR8

**9ms**



@KBWPOR9

**10ms**

---

## USB CDC

If your cradle is connected to the USB port on a host device, the USB CDC feature allows the host device to receive data in the way as a serial port does. A driver is needed when using this feature. You may download it from our website at [www.newlandaidc.com](http://www.newlandaidc.com).



@INTERF8

**USB CDC**

## USB HID-POS

### Introduction

The HID-POS interface is recommended for new application programs. It can send up to 56 characters in a single USB report and appears more efficient than keyboard emulation.

Features:

- ✧ HID based, no custom driver required.
- ✧ Way more efficient in communication than keyboard emulation and traditional RS-232 interface.

**Note:** HID-POS does not require a custom driver. However, a HID interface on Windows 98 does. All HID interfaces employ standard driver provided by the operating system. Use defaults when installing the driver.



@INTERF8

**USB HID-POS**

---

## Access the Scanner with Your Program

Use CreateFile to access the scanner as a HID device and then use ReadFile to deliver the scanned data to the application program. Use WriteFile to send data to the scanner.

For detailed information about USB and HID interfaces, go to [www.USB.org](http://www.USB.org).

## Acquire Scanned Data

After a barcode is decoded, the scanner sends an input report as below:

	Bit							
Byte	7	6	5	4	3	2	1	0
0	Report ID = 0x02							
1	Barcode Length							
2-57	Decoded Data (1-56)							
58-61	Reserved (1-4)							
62	Newland Symbology Identifier or N/C: 0x00							
63	-	-	-	-	-	-	-	Decoded data continued

## Send Command to the Scanner

This output report is used to send commands to the scanner. All programming commands can be used.

	Bit							
Byte	7	6	5	4	3	2	1	0
0	Report ID = 0x04							
1	Length of command							
2-62	Command							
63	-	-	-	-	-	-	-	Command continued

## VID/PID

USB uses VID (Vendor ID) and PID (Product ID) to identify and locate a device. The VID is assigned by USB Implementers Forum. Newland's vendor ID is 1EAB (Hex). A range of PIDs are used for each Newland product family. Every PID contains a base number and interface type (keyboard, COM port, etc.).

---

Product	Interface	PID (Hex)	PID (Dec)
NVH300	USB HID Keyboard	2E22	11810
	USB CDC	2E06	11782
	USB HID-POS	2E10	11792

---

## Chapter 6 Symbologies

### Introduction

Every symbology (barcode type) has its own unique attributes. This chapter provides programming barcodes for configuring the scanner so that it can identify various symbologies. It is recommended to disable those that are rarely used to increase the efficiency of the scanner.

### Global Settings

#### Enable/Disable All Symbologies

If the **Disable All Symbologies** feature is enabled, the scanner will not be able to read any non-programming barcodes except the programming barcodes.



@ALLENA1

**Enable All Symbologies**



@ALLENA0

**Disable All Symbologies**



---

## Enable/Disable 1D Symbolologies



@ALL1DC1

**Enable 1D Symbolologies**



@ALL1DC0

**Disable 1D Symbolologies**

## Enable/Disable 2D Symbolologies



@ALL2DC1

**Enable 2D Symbolologies**



@ALL2DC0

**Disable 2D Symbolologies**



Enable Disable 2D Symbologies, the Programming Barcode of 2D Data Martrix in this User Guide will be invalid. It is recommended to use NSet to reactive Data Matrix or scan these barcodes below.



#SETUPE1

**Enter Setup**



@DMCENA1

**Enable Data Matrix**



#SETUPE0

**Exit Setup**

---

## Image Brightness Adaptation

The two parameters of image brightness adaptation 1 and 2, which are linkage relations, are set to adapt to the brightness of the image. When the image is too dark, two parameters can be adjusted down at the same time; When the image is too bright, you can try to raise both parameters at the same time.



@CCFCTV

**Image Brightness Adaptation 1**



@CCFMTV

**Image Brightness Adaptation 2**

## 1D Decoding Complexity

The Decoding Complexity parameter selects the decoding algorithm according to the printing/marking technique used to create the symbol and on the overall printing/marking quality.

- ✧ Low: For no stain, wrinkle and other high-quality barcodes;
- ✧ Medium: Bar code with general quality such as certain dirt, fold, etc;
- ✧ High: For Bar code with poor printing quality.



@CCF1DC0

**Low**



@CCF1DC1

**\*\*Medium**



@CCF1DC2

**High**

## 1D Inverse



@CCF1IV0

**Regular**



@CCF1IV2

**\*\*Regular and Inverse**

---

## Code 128

### Restore Factory Defaults



@128DEF

**Restore the Factory Defaults of Code 128**

### Enable/Disable Code 128



@128ENA1

**\*\* Enable Code 128**



@128ENA0

**Disable Code 128**



If the scanner fails to identify Code 128 barcodes, you may first try this solution by scanning the **Enable Code 128** barcode.

---

## Set Length Range for Code 128

The scanner can be configured to only decode Code 128 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@128MIN

**Set the Minimum Length (Default: 1)**



@128MAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes Code 128 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Code 128 barcodes with that length are to be decoded.



**Set the scanner to decode Code 128 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## EAN-8

### Restore Factory Defaults



@EA8DEF

Restore the Factory Defaults of EAN-8

### Enable/Disable EAN-8



@EA8ENA1

**\*\* Enable EAN-8**



@EA8ENA0

**Disable EAN-8**



If the scanner fails to identify EAN-8 barcodes, you may first try this solution by scanning the **Enable EAN-8** barcode.

---

## Transmit Check Character

EAN-8 is 8 digits in length with the last one as its check character used to verify the integrity of the data.



@EA8CHK2

**\*\* Transmit EAN-8 Check Character**



@EA8CHK1

**Do Not Transmit EAN-8 Check Character**

## 2-Digit Add-On Code

An EAN-8 barcode can be augmented with a two-digit add-on code to form a new one. In the example below, the part surrounded by blue dotted line is an EAN-8 barcode while the part circled by red dotted line is a two-digit add-on code.



@EA8AD20

**\*\* Disable 2-Digit Add-On Code**



@EA8AD21

**Enable 2-Digit Add-On Code**





**Disable 2-Digit Add-On Code:** The scanner decodes EAN-8 and ignores the add-on code when presented with an EAN-8 plus 2-digit add-on barcode. It can also decode EAN-8 barcodes without 2-digit add-on codes.

**Enable 2-Digit Add-On Code:** The scanner decodes a mix of EAN-8 barcodes with and without 2-digit add-on codes.

---

## 5-Digit Add-On Code

An EAN-8 barcode can be augmented with a five-digit add-on code to form a new one. In the example below, the part surrounded by blue dotted line is an EAN-8 barcode while the part circled by red dotted line is a five-digit add-on code.



@EA8AD50

**\*\* Disable 5-Digit Add-On Code**



@EA8AD51

**Enable 5-Digit Add-On Code**



**Disable 5-Digit Add-On Code:** The scanner decodes EAN-8 and ignores the add-on code when presented with an EAN-8 plus 5-digit add-on barcode. It can also decode EAN-8 barcodes without 5-digit add-on codes.

**Enable 5-Digit Add-On Code:** The scanner decodes a mix of EAN-8 barcodes with and without 5-digit add-on codes.

## Add-On Code Required

When **EAN-8 Add-On Code Required** is selected, the scanner will only read EAN-8 barcodes that contain add-on codes.



@EA8REQ1

**EAN-8 Add-On Code Required**



@EA8REQ0

**\*\* EAN-8 Add-On Code Not Required**

### **Add-On Code Separator**

When this feature is on, there is a space between the data from the barcode and the data from the add-on code. When turned off, there is no space.



@EA8SEP0

**\*\* Off**



@EA8SEP1

**On**

### **Convert EAN-8 to EAN-13**

**Convert EAN-8 to EAN-13:** Convert EAN-8 decoded data to EAN-13 format before transmission. After conversion, the data follows EAN-13 format and is affected by EAN-13 programming selections (e.g., Check Character).

**Do Not Convert EAN-8 to EAN-13:** EAN-8 decoded data is transmitted as EAN-8 data, without conversion.



@EA8EXP0

**\*\* Do Not Convert EAN-8 to EAN-13**



@EA8EXP1

**Convert EAN-8 to EAN-13**

---

## EAN-13

### Restore Factory Defaults



@E13DEF

**Restore the Factory Defaults of EAN-13**

### Enable/Disable EAN-13



@E13ENA1

**\*\* Enable EAN-13**



@E13ENA0

**Disable EAN-13**



If the scanner fails to identify EAN-13 barcodes, you may first try this solution by scanning the **Enable EAN-13** barcode.

---

## Transmit Check Character



@E13CHK2

**\*\* Transmit EAN-13 Check Character**



@E13CHK1

**Do Not Transmit EAN-13 Check Character**

## 2-Digit Add-On Code

An EAN-13 barcode can be augmented with a two-digit add-on code to form a new one. In the example below, the part surrounded by blue dotted line is an EAN-13 barcode while the part circled by red dotted line is a two-digit add-on code.



@E13AD20

**\*\* Disable 2-Digit Add-On Code**



@E13AD21

**Enable 2-Digit Add-On Code**



**Disable 2-Digit Add-On Code:** The scanner decodes EAN-13 and ignores the add-on code when

---

presented with an EAN-13 plus 2-digit add-on barcode. It can also decode EAN-13 barcodes without 2-digit add-on codes.

**Enable 2-Digit Add-On Code:** The scanner decodes a mix of EAN-13 barcodes with and without 2-digit add-on codes.

---

## 5-Digit Add-On Code

An EAN-13 barcode can be augmented with a five-digit add-on code to form a new one. In the example below, the part surrounded by blue dotted line is an EAN-13 barcode while the part circled by red dotted line is a five-digit add-on code.



@E13AD50

**\*\* Disable 5-Digit Add-On Code**



@E13AD51

**Enable 5-Digit Add-On Code**



**Disable 5-Digit Add-On Code:** The scanner decodes EAN-13 and ignores the add-on code when presented with an EAN-13 plus 5-digit add-on barcode. It can also decode EAN-13 barcodes without 5-digit add-on codes.

**Enable 5-Digit Add-On Code:** The scanner decodes a mix of EAN-13 barcodes with and without 5-digit add-on codes.

## Add-On Code Required

When **EAN-13 Add-On Code Required** is selected, the scanner will only read EAN-13 barcodes that contain add-on codes.



@E13REQ0

**\*\* EAN-13 Add-On Code Not Required**





@E13REQ1

### EAN-13 Add-On Code Required

#### Add-On Code Separator

When this feature is on, there is a space between the data from the barcode and the data from the add-on code. When turned off, there is no space.



@E13SEP0

**\*\* Off**



@E13SEP1

**On**

#### EAN-13 Beginning with 290 Add-On Code Required

This setting programs the scanner to require an add-on code (2-digit or 5-digit) on EAN-13 barcodes that begin with “290”. The following settings can be programmed:

**Require Add-On Code:** All EAN-13 barcodes that begin with “290” must have a 2-digit or 5-digit add-on code. The EAN-13 barcode with the add-on code is then transmitted. If the required add-on code is not found, the EAN-13 barcode is discarded.

**Do Not Require Add-On Code:** If you have selected **Require Add-On Code**, and you want to disable this feature, scan **Do Not Require Add-On Code**. EAN-13 barcodes are handled, depending on your selection for the “Add-On Code Required” feature.



@E132900

**\*\* Do Not Require Add-On Code**



@E132901

**Require Add-On Code**

### **EAN-13 Beginning with 378/379 Add-On Code Required**

This setting programs the scanner to require an add-on code (2-digit or 5-digit) on EAN-13 barcodes that begin with a “378” or “379”. The following settings can be programmed:

**Require Add-On Code:** All EAN-13 barcodes that begin with a “378” or “379” must have a 2-digit or 5-digit add-on code. The EAN-13 barcode with the add-on code is then transmitted. If the required add-on code is not found, the EAN-13 barcode is discarded.

**Do Not Require Add-On Code:** If you have selected **Require Add-On Code**, and you want to disable this feature, scan **Do Not Require Add-On Code**. EAN-13 barcodes are handled, depending on your selection for the “Add-On Code Required” feature.



@E133780

**\*\* Do Not Require Add-On Code**



@E133781

**Require Add-On Code**

### **EAN-13 Beginning with 414/419 Add-On Code Required**

This setting programs the scanner to require an add-on code (2-digit or 5-digit) on EAN-13 barcodes that begin with a “414”

---

or “419”. The following settings can be programmed:

**Require Add-On Code:** All EAN-13 barcodes that begin with a “414” or “419” must have a 2-digit or 5-digit add-on code. The EAN-13 barcode with the add-on code is then transmitted. If the required add-on code is not found, the EAN-13 barcode is discarded.

**Do Not Require Add-On Code:** If you have selected **Require Add-On Code**, and you want to disable this feature, scan **Do Not Require Add-On Code**. EAN-13 barcodes are handled, depending on your selection for the “Add-On Code Required” feature.



@E134140

**\*\* Do Not Require Add-On Code**



@E134141

**Require Add-On Code**

### **EAN-13 Beginning with 434/439 Add-On Code Required**

This setting programs the scanner to require an add-on code (2-digit or 5-digit) on EAN-13 barcodes that begin with a “434” or “439”. The following settings can be programmed:

**Require Add-On Code:** All EAN-13 barcodes that begin with a “434” or “439” must have a 2-digit or 5-digit add-on code. The EAN-13 barcode with the add-on code is then transmitted. If the required add-on code is not found, the EAN-13 barcode is discarded.

**Do Not Require Add-On Code:** If you have selected **Require Add-On Code**, and you want to disable this feature, scan **Do Not Require Add-On Code**. EAN-13 barcodes are handled, depending on your selection for the “Add-On Code Required” feature.



@E134340

**\*\* Do Not Require Add-On Code**



@E134341

**Require Add-On Code**

### **EAN-13 Beginning with 977 Add-On Code Required**

This setting programs the scanner to require an add-on code (2-digit or 5-digit) on EAN-13 barcodes that begin with “977”. The following settings can be programmed:

**Require Add-On Code:** All EAN-13 barcodes that begin with “977” must have a 2-digit or 5-digit add-on code. The EAN-13 barcode with the add-on code is then transmitted. If the required add-on code is not found, the EAN-13 barcode is discarded.

**Do Not Require Add-On Code:** If you have selected **Require Add-On Code**, and you want to disable this feature, scan **Do Not Require Add-On Code**. EAN-13 barcodes are handled, depending on your selection for the “Add-On Code Required” feature.



@E139770

**\*\* Do Not Require Add-On Code**



@E139771

**Require Add-On Code**

### **EAN-13 Beginning with 978 Add-On Code Required**

This setting programs the scanner to require an add-on code (2-digit or 5-digit) on EAN-13 barcodes that begin with “978”. The following settings can be programmed:

**Require Add-On Code:** All EAN-13 barcodes that begin with “978” must have a 2-digit or 5-digit add-on code. The EAN-13 barcode with the add-on code is then transmitted. If the required add-on code is not found, the EAN-13 barcode is discarded.

**Do Not Require Add-On Code:** If you have selected **Require Add-On Code**, and you want to disable this feature, scan **Do Not Require Add-On Code**. EAN-13 barcodes are handled, depending on your selection for the “Add-On Code Required” feature.



@E139780

**\*\* Do Not Require Add-On Code**



@E139781

**Require Add-On Code**

### **EAN-13 Beginning with 979 Add-On Code Required**

This setting programs the scanner to require an add-on code (2-digit or 5-digit) on EAN-13 barcodes that begin with “979”. The following settings can be programmed:

**Require Add-On Code:** All EAN-13 barcodes that begin with “979” must have a 2-digit or 5-digit add-on code. The EAN-13 barcode with the add-on code is then transmitted. If the required add-on code is not found, the EAN-13 barcode is discarded.

**Do Not Require Add-On Code:** If you have selected **Require Add-On Code**, and you want to disable this feature, scan **Do Not Require Add-On Code**. EAN-13 barcodes are handled, depending on your selection for the “Add-On Code Required” feature.



@E139790

**\*\* Do Not Require Add-On Code**



@E139791

**Require Add-On Code**

---

## UPC-E

### Restore Factory Defaults



@UPEDEF

**Restore the Factory Defaults of UPC-E**

### Enable/Disable UPC-E



@UPEENA1

**\*\* Enable UPC-E**



@UPEENA0

**Disable UPC-E**

---

## Transmit Check Character

UPC-E is 8 digits in length with the last one as its check character used to verify the integrity of the data.



@UPECHK2

**\*\* Transmit UPC-E Check Character**



@UPECHK1

**Do Not Transmit UPC-E Check Character**

## 2-Digit Add-On Code

A UPC-E barcode can be augmented with a two-digit add-on code to form a new one. In the example below, the part surrounded by blue dotted line is a UPC-E barcode while the part circled by red dotted line is a two-digit add-on code.



@UPEAD20

**\*\* Disable 2-Digit Add-On Code**



@UPEAD21

**Enable 2-Digit Add-On Code**



**Disable 2-Digit Add-On Code:** The scanner decodes UPC-E and ignores the add-on code when presented with a UPC-E plus 2-digit add-on barcode. It can also decode UPC-E barcodes without 2-digit add-on codes.

**Enable 2-Digit Add-On Code:** The scanner decodes a mix of UPC-E barcodes with and without 2-digit add-on codes.

## 5-Digit Add-On Code

A UPC-E barcode can be augmented with a five-digit add-on code to form a new one. In the example below, the part surrounded by blue dotted line is a UPC-E barcode while the part circled by red dotted line is a five-digit add-on code.



@UPEAD50

**\*\* Disable 5-Digit Add-On Code**



@UPEAD51

**Enable 5-Digit Add-On Code**



**Disable 5-Digit Add-On Code:** The scanner decodes UPC-E and ignores the add-on code when presented with a UPC-E plus 5-digit add-on barcode. It can also decode UPC-E barcodes without 5-digit add-on codes.

**Enable 5-Digit Add-On Code:** The scanner decodes a mix of UPC-E barcodes with and without 5-digit add-on codes.



---

## Add-On Code Required

When **UPC-E Add-On Code Required** is selected, the scanner will only read UPC-E barcodes that contain add-on codes.



@UPEREQ1

**UPC-E Add-On Code Required**



@UPEREQ0

**\*\* UPC-E Add-On Code Not Required**

## Add-On Code Separator

When this feature is on, there is a space between the data from the barcode and the data from the add-on code. When turned off, there is no space.



@UPESEP0

**\*\* Off**



@UPESEP1

**On**

---

## Transmit Preamble Character

Preamble characters (Country Code and System Character) can be transmitted as part of a UPC-E barcode. Select one of the following options for transmitting UPC-E preamble to the host device: transmit system character only, transmit system character and country code ("0" for USA), or transmit no preamble.



@UPEPRE1

**\*\* System Character**



@UPEPRE0

**No Preamble**



@UPEPRE2

**System Character & Country Code**

## Convert UPC-E to UPC-A

**Convert UPC-E to UPC-A:** Convert UPC-E decoded data to UPC-A format before transmission. After conversion, the data follows UPC-A format and is affected by UPC-A programming selections (e.g., Preamble, Check Character).

**Do Not Convert UPC-E to UPC-A:** UPC-E decoded data is transmitted as UPC-E data, without conversion.



@UPEEXP0

**\*\* Do Not Convert UPC-E to UPC-A**



@UPEEXP1

**Convert UPC-E to UPC-A**

## UPC-A

**Restore Factory Defaults**



@UPADEF

**Restore the Factory Defaults of UPC-A**

**Enable/Disable UPC-A**



@UPAENA1

**\*\* Enable UPC-A**



@UPAENA0

**Disable UPC-A**



If the scanner fails to identify UPC-A barcodes, you may first try this solution by scanning the **Enable UPC-A** barcode.

---

## Transmit Check Character

UPC-A is 13 digits in length with the last one as its check character used to verify the integrity of the data.



@UPACHK2

**\*\* Transmit UPC-A Check Character**



@UPACHK1

**Do Not Transmit UPC-A Check Character**

## 2-Digit Add-On Code

A UPC-A barcode can be augmented with a two-digit add-on code to form a new one. In the example below, the part surrounded by blue dotted line is a UPC-A barcode while the part circled by red dotted line is a two-digit add-on code.



@UPAAD20

**\*\* Disable 2-Digit Add-On Code**



@UPAAD21

**Enable 2-Digit Add-On Code**



**Disable 2-Digit Add-On Code:** The scanner decodes UPC-A and ignores the add-on code when presented with a UPC-A plus 2-digit add-on barcode. It can also decode UPC-A barcodes without 2-digit add-on codes.

**Enable 2-Digit Add-On Code:** The scanner decodes a mix of UPC-A barcodes with and without 2-digit add-on codes.

---

## 5-Digit Add-On Code

A UPC-A barcode can be augmented with a five-digit add-on code to form a new one. In the example below, the part surrounded by blue dotted line is a UPC-A barcode while the part circled by red dotted line is a five-digit add-on code.



@UPAAD50

**\*\* Disable 5-Digit Add-On Code**



@UPAAD51

**Enable 5-Digit Add-On Code**



**Disable 5-Digit Add-On Code:** The scanner decodes UPC-A and ignores the add-on code when presented with a UPC-A plus 5-digit add-on barcode. It can also decode UPC-A barcodes without 5-digit add-on codes.

**Enable 5-Digit Add-On Code:** The scanner decodes a mix of UPC-A barcodes with and without 5-digit add-on codes.

## Add-On Code Required

When **UPC-A Add-On Code Required** is selected, the scanner will only read UPC-A barcodes that contain add-on codes.



@UPAREQ1

**UPC-A Add-On Code Required**



@UPAREQ0

**\*\* UPC-A Add-On Code Not Required**

### **Add-On Code Separator**

When this feature is on, there is a space between the data from the barcode and the data from the add-on code. When turned off, there is no space.



@UPASEP0

**\*\* Off**



@UPASEP1

**On**

### **Transmit Preamble Character**

Preamble characters (Country Code and System Character) can be transmitted as part of a UPC-A barcode. Select one of the following options for transmitting UPC-A preamble to the host device: transmit system character only, transmit system character and country code ("0" for USA), or transmit no preamble.



@UPAPRE0

**No Preamble**



@UPAPRE2

**System Character & Country Code**



@UPAPRE1

**\*\* System Character**



---

## Interleaved 2 of 5

### Restore Factory Defaults



@I25DEF

**Restore the Factory Defaults of Interleaved 2 of 5**

### Enable/Disable Interleaved 2 of 5



@I25ENA1

**\*\* Enable Interleaved 2 of 5**



@I25ENA0

**Disable Interleaved 2 of 5**



If the scanner fails to identify Interleaved 2 of 5 barcodes, you may first try this solution by scanning the **Enable Interleaved 2 of 5** barcode.

---

## Set Length Range for Interleaved 2 of 5

The scanner can be configured to only decode Interleaved 2 of 5 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@I25MIN

**Set the Minimum Length (Default: 6)**



@I25MAX

**Set the Maximum Length (Default: 80)**



If minimum length is set to be greater than maximum length, the scanner only decodes Interleaved 2 of 5 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Interleaved 2 of 5 barcodes with that length are to be decoded.

## Example

**Set the scanner to decode Interleaved 2 of 5 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode “8” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes “1” and “2” from the “Digit Barcodes” section in Appendix.
6. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.

---

## Check Character Verification

A check character is optional for Interleaved 2 of 5 and can be added as the last character. It is a calculated value used to verify the integrity of the data.

- ✧ **Disable:** The scanner transmits Interleaved 2 of 5 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The scanner checks the integrity of all Interleaved 2 of 5 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted except the last digit, whereas those failing it will not be transmitted.
- ✧ **Transmit Check Character After Verification:** The scanner checks the integrity of all Interleaved 2 of 5 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted, whereas those failing it will not be transmitted.

Since Interleaved 2 of 5 must always have an even number of digits, a zero may need to be added as the first digit when the check character is added. The check character is automatically generated when making Interleaved 2 of 5 barcodes.



@I25CHK0

**\*\* Disable**



@I25CHK1

**Do Not Transmit Check Character After Verification**



@I25CHK2

**Transmit Check Character After Verification**



If the **Do Not Transmit Check Character After Verification** option is enabled, Interleaved 2 of 5 barcodes with a length that is less than the configured minimum length after having the check character excluded will not be decoded. (For example, when the **Do Not Transmit Check Character After Verification** option is enabled and the minimum length is set to 4, Interleaved 2 of 5 barcodes with a total length of 4 characters including the check character cannot be read.)

---

## ITF-14

ITF-14 is a special kind of Interleaved 2 of 5 with a length of 14 characters and the last character as the check character.

ITF-14 priority principle: For the Interleaved 2 of 5 barcodes with a length of 14 characters and the last character as the check character, the ITF-14 configurations shall take precedence over the Interleaved 2 of 5 settings.

### Restore Factory Defaults



@I14DEF

**Restore the Factory Defaults of ITF-14**

### Enable/Disable ITF-14



@I14ENA0

**\*\* Disable ITF-14**



@I14ENA1

**Enable ITF-14 But Do Not Transmit Check Character**



@I14ENA2

**Enable ITF-14 and Transmit Check Character**



An example of the ITF-14 priority principle: when ITF-14 is enabled and Interleaved 2 of 5 is disabled, the scanner only decodes Interleaved 2 of 5 barcodes with a length of 14 characters and the last character as the check character.

---

## ITF-6

ITF-6 is a special kind of Interleaved 2 of 5 with a length of 6 characters and the last character as the check character.

ITF-6 priority principle: For the Interleaved 2 of 5 barcodes with a length of 6 characters and the last character as the check character, the ITF-6 configurations shall take precedence over the Interleaved 2 of 5 settings.

### Restore Factory Defaults



@IT6DEF

**Restore the Factory Defaults of ITF-6**

### Enable/Disable ITF-6



@IT6ENA0

**\*\* Disable ITF-6**



@IT6ENA1

**Enable ITF-6 But Do Not Transmit Check Character**



@IT6ENA2

**Enable ITF-6 and Transmit Check Character**



An example of the ITF-6 priority principle: when ITF-6 is enabled and Interleaved 2 of 5 is disabled, the scanner only decodes Interleaved 2 of 5 barcodes with a length of 6 characters and the last character as the check character.

---

## Matrix 2 of 5

### Restore Factory Defaults



@M25DEF

**Restore the Factory Defaults of Matrix 2 of 5**

### Enable/Disable Matrix 2 of 5



@M25ENA1

**\*\* Enable Matrix 2 of 5**



@M25ENA0

**Disable Matrix 2 of 5**



If the scanner fails to identify Matrix 2 of 5 barcodes, you may first try this solution by scanning the **Enable Matrix 2 of 5** barcode.



---

## Set Length Range for Matrix 2 of 5

The scanner can be configured to only decode Matrix 2 of 5 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@M25MIN

**Set the Minimum Length (Default: 4)**



@M25MAX

**Set the Maximum Length (Default: 80)**



If minimum length is set to be greater than maximum length, the scanner only decodes Matrix 2 of 5 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Matrix 2 of 5 barcodes with that length are to be decoded.



**Set the scanner to decode Matrix 2 of 5 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## Check Character Verification

A check character is optional for Matrix 2 of 5 and can be added as the last character. It is a calculated value used to verify the integrity of the data.

- ✧ **Disable:** The scanner transmits Matrix 2 of 5 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The scanner checks the integrity of all Matrix 2 of 5 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted except the last digit, whereas those failing it will not be transmitted.
- ✧ **Transmit Check Character After Verification:** The scanner checks the integrity of all Matrix 2 of 5 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted, whereas those failing it will not be transmitted.

Since Matrix 2 of 5 must always have an even number of digits, a zero may need to be added as the first digit when the check character is added. The check character is automatically generated when making Matrix 2 of 5 barcodes.



@M25CHK0

**\*\* Disable**



@M25CHK1

**Do Not Transmit Check Character After Verification**



@M25CHK2

**Transmit Check Character After Verification**



If the **Do Not Transmit Check Character After Verification** option is enabled, Matrix 2 of 5 barcodes with a length that is less than the configured minimum length after having the check character excluded will not be decoded. (For example, when the **Do Not Transmit Check Character After Verification**

---

option is enabled and the minimum length is set to 4, Matrix 2 of 5 barcodes with a total length of 4 characters including the check character cannot be read.)

---

## Code 39

### Restore Factory Defaults



@C39DEF

**Restore the Factory Defaults of Code 39**

### Enable/Disable Code 39



@C39ENA1

**\*\* Enable Code 39**



@C39ENA0

**Disable Code 39**



If the scanner fails to identify Code 39 barcodes, you may first try this solution by scanning the **Enable Code 39** barcode.

---

## Set Length Range for Code 39

The scanner can be configured to only decode Code 39 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@C39MIN

**Set the Minimum Length (Default: 1)**



@C39MAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes Code 39 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Code 39 barcodes with that length are to be decoded.



**Set the scanner to decode Code 39 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## Check Character Verification

A check character is optional for Code 39 and can be added as the last character. It is a calculated value used to verify the integrity of the data.

- ✧ **Disable:** The scanner transmits Code 39 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The scanner checks the integrity of all Code 39 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted except the last digit, whereas those failing it will not be transmitted.
- ✧ **Transmit Check Character After Verification:** The scanner checks the integrity of all Code 39 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted, whereas those failing it will not be transmitted.



@C39CHK0

**\*\* Disable**



@C39CHK1

**Do Not Transmit Check Character After Verification**



@C39CHK2

**Transmit Check Character After Verification**



If the **Do Not Transmit Check Character After Verification** option is enabled, Code 39 barcodes with a length that is less than the configured minimum length after having the check character excluded will not be decoded. (For example, when the **Do Not Transmit Check Character After Verification** option is enabled and the minimum length is set to 4, Code 39 barcodes with a total length of 4 characters including the check character cannot be read.)

---

## Transmit Start/Stop Character

Code 39 uses an asterisk (\*) for both the start and the stop characters. You can choose whether or not to transmit the start/stop characters by scanning the appropriate barcode below.



@C39TSC0

**\*\* Do Not Transmit Start/Stop Character**



@C39TSC1

**Transmit Start/Stop Character**

## Enable/Disable Code 39 Full ASCII

The scanner can be configured to identify all ASCII characters by scanning the appropriate barcode below.



@C39ASC0

**\*\* Disable Code 39 Full ASCII**



@C39ASC1

**Enable Code 39 Full ASCII**

---

## Codabar

### Restore Factory Defaults



@CBADEF

**Restore the Factory Defaults of Codabar**

### Enable/Disable Codabar



@CBAENA1

**\*\* Enable Codabar**



@CBAENA0

**Disable Codabar**



If the scanner fails to identify Codabar barcodes, you may first try this solution by scanning the **Enable Codabar** barcode.



---

## Set Length Range for Codabar

The scanner can be configured to only decode Codabar barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@CBAMIN

### Set the Minimum Length (Default: 2)



@CBAMAX

### Set the Maximum Length (Default: 60)



If minimum length is set to be greater than maximum length, the scanner only decodes Codabar barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Codabar barcodes with that length are to be decoded.

## Example

### Set the scanner to decode Codabar barcodes containing between 8 and 12 characters:

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## Check Character Verification

A check character is optional for Codabar and can be added as the last character. It is a calculated value used to verify the integrity of the data.

- ✧ **Disable:** The scanner transmits Codabar barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The scanner checks the integrity of all Codabar barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted except the last digit, whereas those failing it will not be transmitted.
- ✧ **Transmit Check Character After Verification:** The scanner checks the integrity of all Codabar barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted, whereas those failing it will not be transmitted.



@CBACHK0

**\*\* Disable**



@CBACHK1

**Do Not Transmit Check Character After Verification**



@CBACHK2

**Transmit Check Character After Verification**



If the **Do Not Transmit Check Character After Verification** option is enabled, Codabar barcodes with a length that is less than the configured minimum length after having the check character excluded will not be decoded. (For example, when the **Do Not Transmit Check Character After Verification** option is enabled and the minimum length is set to 4, Codabar barcodes with a total length of 4 characters including the check character cannot be read.)

---

## Start/Stop Character

You can set the start/stop characters and choose whether or not to transmit the start/stop characters by scanning the appropriate barcode below.



@CBATSC0

**\*\* Do Not Transmit Start/Stop Character**



@CBATSC1

**Transmit Start/Stop Character**



@CBASCF0

**\*\* ABCD/ABCD as the Start/Stop Character**



@CBASCF1

**ABCD/TN\*E as the Start/Stop Character**



@CBASCF2

**abcd/abcd as the Start/Stop Character**



@CBASCF3

**abcd/tn\*e as the Start/Stop Character**

---

## Code 93

### Restore Factory Defaults



@C93DEF

**Restore the Factory Defaults of Code 93**

### Enable/Disable Code 93



@C93ENA1

**Enable Code 93**



@C93ENA0

**\*\* Disable Code 93**



If the scanner fails to identify Code 93 barcodes, you may first try this solution by scanning the **Enable Code 93** barcode.

---

## Set Length Range for Code 93

The scanner can be configured to only decode Code 93 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@C93MIN

**Set the Minimum Length (Default: 1)**



@C93MAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes Code 93 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Code 93 barcodes with that length are to be decoded.



**Set the scanner to decode Code 93 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## Check Character Verification

Check characters are optional for Code 93 and can be added as the last two characters, which are calculated values used to verify the integrity of the data.

- ✧ **Disable:** The scanner transmits Code 93 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The scanner checks the integrity of all Code 93 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the checks will be transmitted except the last two digits, whereas those failing them will not be transmitted.
- ✧ **Transmit Check Character After Verification:** The scanner checks the integrity of all Code 93 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the checks will be transmitted, whereas those failing them will not be transmitted.



@C93CHK0

**Disable**



@C93CHK1

**\*\* Do Not Transmit Check Character After Verification**



@C93CHK2

**Transmit Check Character After Verification**



If the **Do Not Transmit Check Character After Verification** option is enabled, Code 93 barcodes with a length that is less than the configured minimum length after having the two check characters excluded will not be decoded. (For example, when the **Do Not Transmit Check Character After Verification** option is enabled and the minimum length is set to 4, Code 93 barcodes with a total length of 4 characters including the two check characters cannot be read.)

---

## GS1-128 (UCC/EAN-128)

### Restore Factory Defaults



@GS1DEF

**Restore the Factory Defaults of GS1-128**

### Enable/Disable GS1-128



@GS1ENA1

**\*\* Enable GS1-128**



@GS1ENA0

**Disable GS1-128**



If the scanner fails to identify GS1-128 barcodes, you may first try this solution by scanning the **Enable GS1-128** barcode.



---

## AIM 128

### Restore Factory Defaults



@AIMDEF

**Restore the Factory Defaults of AIM 128**

### Enable/Disable AIM 128



@AIMENA1

**Enable AIM128**



@AIMENA0

**\*\* Disable AIM 128**



If the scanner fails to identify AIM 128 barcodes, you may first try this solution by scanning the **Enable AIM 128** barcode.

---

## Set Length Range for AIM 128

The scanner can be configured to only decode AIM 128 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@AIMMIN

**Set the Minimum Length (Default: 1)**



@AIMMAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes GS1-128 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only GS1-128 barcodes with that length are to be decoded.

**E**  
*sample*

**Set the scanner to decode AIM 128 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## ISBT 128

### Restore Factory Defaults



@IBTDEF

**Restore the Factory Defaults of ISBT 128**

### Enable/Disable ISBT 128



@IBTENA1

**Enable ISBT 128**



@IBTENA0

**\*\* Disable ISBT 128**



If the scanner fails to identify ISBT-128 barcodes, you may first try this solution by scanning the **Enable ISBT-128** barcode.

---

## Code 11

### Restore Factory Defaults



@C11DEF

**Restore the Factory Defaults of Code 11**

### Enable/Disable Code 11



@C11ENA1

**Enable Code 11**



@C11ENA0

**\*\* Disable Code 11**



If the scanner fails to identify Code 11 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable Code 11** barcode.

---

## Set Length Range for Code 11

The scanner can be configured to only decode Code 11 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@C11MIN

**Set the Minimum Length (Default: 4)**



@C11MAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes Code 11 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Code 11 barcodes with that length are to be decoded.



**Set the scanner to decode Code 11 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## Check Character Verification

Check characters are optional for Code 11 and can be added as the last one or two characters, which are calculated values used to verify the integrity of the data.

If the **Disable** option is enabled, the scanner transmits Code 11 barcodes as is.



@C11CHK0

**Disable**



@C11CHK1

**\*\* One Check Character, MOD11**



@C11CHK2

**Two Check Characters, MOD11/MOD11**



@C11CHK3

**Two Check Characters, MOD11/MOD9**



@C11CHK4

**One Check Character, MOD11 (Len<=10)**

**Two Check Characters, MOD11/MOD11(Len>10)**



@C11CHK5

**One Check Character, MOD11 (Len≤10)**

**Two Check Characters, MOD11/MOD9 (Len>10)**

---

## Transmit Check Character



@C11TCK0

## Do Not Transmit Code 11 Check Character



@C11TCK1

## \*\* Transmit Code 11 Check Character



If you select a check character algorithm and the **Do Not Transmit Check Character** option, Code 11 barcodes with a length that is less than the configured minimum length after having the check character(s) excluded will not be decoded. (For example, when the **One Check Character, MOD11** and **Do Not Transmit Check Character** options are enabled and the minimum length is set to 4, Code 11 barcodes with a total length of 4 characters including the check character cannot be read.)



---

## ISBN

### Restore Factory Defaults



@ISBDEF

**Restore the Factory Defaults of ISBN**

### Enable/Disable ISBN



@ISBENA1

**Enable ISBN**



@ISBENA0

**\*\* Disable ISBN**

---

## Set ISBN Format



@ISBT101

**\*\* ISBN-10**



@ISBT100

**ISBN-13**

---

## ISSN

### Restore Factory Defaults



@ISSDEF

**Restore the Factory Defaults of ISSN**

### Enable/Disable ISSN



@ISSENA1

**Enable ISSN**



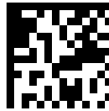
@ISSENA0

**\*\* Disable ISSN**

---

## Industrial 2/5

### Restore Factory Defaults



@L25DEF

**Restore the Factory Defaults of Industrial 25**

### Enable/Disable Industrial 2/5



@L25ENA1

**Enable Industrial 2/5**



@L25ENA0

**\*\* Disable Industrial 2/5**



If the scanner fails to identify Industrial 2/5 barcodes, you may first try this solution by scanning the **Enable Industrial 2/5** barcode.

---

## Set Length Range for Industrial 25

The scanner can be configured to only decode Industrial 25 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@L25MIN

**Set the Minimum Length (Default: 6)**



@L25MAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes Industrial 25 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Industrial 25 barcodes with that length are to be decoded.



**Set the scanner to decode Industrial 25 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## Check Character Verification

A check character is optional for Industrial 25 and can be added as the last character. It is a calculated value used to verify the integrity of the data.

- ✧ **Disable:** The scanner transmits Industrial 25 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The scanner checks the integrity of all Industrial 25 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted except the last digit, whereas those failing it will not be transmitted.
- ✧ **Transmit Check Character After Verification:** The scanner checks the integrity of all Industrial 25 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted, whereas those failing it will not be transmitted.



@L25CHK0

**\*\* Disable**



@L25CHK1

**Do Not Transmit Check Character After Verification**



@L25CHK2

**Transmit Check Character After Verification**



If the **Do Not Transmit Check Character After Verification** option is enabled, Industrial 25 barcodes with a length that is less than the configured minimum length after having the check character excluded will not be decoded. (For example, when the **Do Not Transmit Check Character After Verification** option is enabled and the minimum length is set to 4, Industrial 25 barcodes with a total length of 4 characters including the check character cannot be read.)

---

## Standard 2/5

### Restore Factory Defaults



@S25DEF

**Restore the Factory Defaults of Standard 25**

### Enable/Disable Standard 2/5



@S25ENA1

**Enable Standard 25**



@S25ENA0

**\*\* Disable Standard 25**



If the scanner fails to identify Standard 25 barcodes, you may first try this solution by scanning the **Enable Standard 25** barcode.

---

## Set Length Range for Standard 25

The scanner can be configured to only decode Standard 25 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@S25MIN

**Set the Minimum Length (Default: 6)**



@S25MAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes Standard 25 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Standard 25 barcodes with that length are to be decoded.



**Set the scanner to decode Standard 25 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.



---

## Check Character Verification

A check character is optional for Standard 25 and can be added as the last character. It is a calculated value used to verify the integrity of the data.

- ✧ **Disable:** The scanner transmits Standard 25 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The scanner checks the integrity of all Standard 25 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted except the last digit, whereas those failing it will not be transmitted.
- ✧ **Transmit Check Character After Verification:** The scanner checks the integrity of all Standard 25 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted, whereas those failing it will not be transmitted.



@S25CHK0

**\*\* Disable**



@S25CHK1

**Do Not Transmit Check Character After Verification**



@S25CHK2

**Transmit Check Character After Verification**

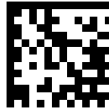


If the **Do Not Transmit Check Character After Verification** option is enabled, Standard 25 barcodes with a length that is less than the configured minimum length after having the check character excluded will not be decoded. (For example, when the **Do Not Transmit Check Character After Verification** option is enabled and the minimum length is set to 4, Standard 25 barcodes with a total length of 4 characters including the check character cannot be read.)

---

## COOP 25

### Restore Factory Defaults



@COPDEF

**Restore the Factory Defaults of COOP 25**

### Enable/Disable COOP 25



@COPENA1

**Enable COOP 25**



@COPENA0

**\*\*Disable COOP 25**

---

## Set Length Range for COOP 25

The scanner can be configured to only decode COOP 25 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@COPMIN

**Set the Minimum Length (Default: 1)**



@PLYMAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes COOP 25 barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only COOP 25 barcodes with that length are to be decoded.



**Set the scanner to decode COOP 25 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## Check Character Verification

A check character is optional for COOP 25 and can be added as the last character. It is a calculated value used to verify the integrity of the data.

- ✧ **Disable:** The scanner transmits COOP 25 barcodes as is.
- ✧ **Enable But Do Not Transmit Check Character:** The scanner checks the integrity of all COOP 25 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted except the last digit, whereas those failing it will not be transmitted.
- ✧ **Enable And Transmit Check Character:** The scanner checks the integrity of all COOP 25 barcodes to verify that the data complies with the check character algorithm. Barcodes passing the check will be transmitted, whereas those failing it will not be transmitted.



@COPCHK0

**\*\* Disable**



@COPCHK1

**Enable But Do Not Transmit Check Character**



@COPCHK2

**Transmit Check Character After Verification**

---

## Deutsche 14

### Restore Factory Defaults



@D14DEF

**Restore the Factory Defaults of Deutsche 14**

### Enable/Disable Deutsche 14



@D14ENA1

**Enable Deutsche 14 But Do Not Transmit  
Check Digit**



@D14ENA2

**Enable Deutsche 14 and Transmit Check  
Digit**



@D14ENA0

**\*\*Disable**

---

## Deutsche 12

### Restore Factory Defaults



@D12DEF

**Restore the Factory Defaults of Deutsche 12**

### Enable/Disable Deutsche 12



@D12ENA1

**Enable Deutsche 12 But Do Not Transmit**

**Check Digit**



@D12ENA2

**Enable Deutsche 12 and Transmit Check**

**Digit**



@D12ENA0

**\*\*Disable**

---

## Plessey

### Restore Factory Defaults



@PLYDEF

**Restore the Factory Defaults of Plessey**

### Enable/Disable Plessey



@PLYENA1

**Enable Plessey**



@PLYENA0

**\*\* Disable Plessey**



If the scanner fails to identify Plessey barcodes, you may first try this solution by scanning the **Enable Plessey** barcode.

---

## Set Length Range for Plessey

The scanner can be configured to only decode Plessey barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@PLYMIN

**Set the Minimum Length (Default: 4)**



@PLYMAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes Plessey barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only Plessey barcodes with that length are to be decoded.



**Set the scanner to decode Plessey barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.



---

## Check Character Verification

Check characters are optional for Plessey and can be added as the last two characters, which are calculated values used to verify the integrity of the data.

- ✧ **Disable:** The scanner transmits Plessey barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The scanner checks the integrity of all Plessey barcodes to verify that the data complies with the check character algorithm. Barcodes passing the checks will be transmitted except the last two digits, whereas those failing them will not be transmitted.
- ✧ **Transmit Check Character After Verification:** The scanner checks the integrity of all Plessey barcodes to verify that the data complies with the check character algorithm. Barcodes passing the checks will be transmitted, whereas those failing them will not be transmitted.



@PLYCHK0

**\*\* Disable**



@PLYCHK1

**Do Not Transmit Check Character After Verification**



@PLYCHK2

**Transmit Check Character After Verification**



If the **Do Not Transmit Check Character After Verification** option is enabled, Plessey barcodes with a length that is less than the configured minimum length after having the check characters excluded will not be decoded. (For example, when the **Do Not Transmit Check Character After Verification** option is enabled and the minimum length is set to 4, Plessey barcodes with a total length of 4 characters including the check characters cannot be read.)

---

## MSI-Plessey

### Restore Factory Defaults



@MSIDEF

**Restore the Factory Defaults of MSI-Plessey**

### Enable/Disable MSI-Plessey



@MSIENA1

**Enable MSI-Plessey**



@MSIENA0

**\*\* Disable MSI-Plessey**



If the scanner fails to identify MSI-Plessey barcodes, you may first try this solution by scanning the **Enable MSI-Plessey** barcode.

---

## Set Length Range for MSI-Plessey

The scanner can be configured to only decode MSI-Plessey barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@MSIMIN

**Set the Minimum Length (Default: 4)**



@MSIMAX

**Set the Maximum Length (Default: 48)**



If minimum length is set to be greater than maximum length, the scanner only decodes MSI-Plessey barcodes with either the minimum or maximum length. If minimum length is same as maximum length, only MSI-Plessey barcodes with that length are to be decoded.



**Set the scanner to decode MSI-Plessey barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## Check Character Verification

Check characters are optional for MSI-Plessey and can be added as the last one or two characters, which are calculated values used to verify the integrity of the data.

If the **Disable** option is enabled, the scanner transmits MSI-Plessey barcodes as is.



@MSICHK0

**Disable**



@MSICHK1

**\*\* One Check Character, MOD10**



@MSICHK2

**Two Check Characters, MOD10/MOD10**



@MSICHK3

**Two Check Characters, MOD10/MOD11**

---

## Transmit Check Character



@MSITCK1

### \*\* Transmit MSI-Plessey Check Character



@MSITCK0

### Do Not Transmit MSI-Plessey Check Character



If you select a check character algorithm and the **Do Not Transmit Check Character** option, MSI-Plessey barcodes with a length that is less than the configured minimum length after having the check character(s) excluded will not be decoded. (For example, when the One Check Character, **MOD10** and **Do Not Transmit Check Character** options are enabled and the minimum length is set to 4, MSI-Plessey barcodes with a total length of 4 characters including the check character cannot be read.)

---

## QR Code

### Restore Factory Defaults



@QRCDEF

**Restore the Factory Defaults of QR Code**

### Enable/Disable QR Code



@QRCENA1

**\*\* Enable QR Code**



@QRCENA0

**Disable QR Code**



If the scanner fails to identify QR Code barcodes, you may first try this solution by scanning the **Enable QR Code** barcode.

---

## Set Length Range for QR Code

The scanner can be configured to only decode QR Code barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@QRCMIN

**Set the Minimum Length (Default: 1)**



@QRCMAX

**Set the Maximum Length (Default: 7089)**



Minimum length is not allowed to be greater than maximum length. If you only want to read QR Code barcodes with a specific length, set both minimum and maximum lengths to be that desired length.



**Set the scanner to decode QR Code barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode “8” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes “1” and “2” from the “Digit Barcodes” section in Appendix.
6. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.

---

## QR Inverse

Regular barcode: Dark bars on a bright background.

Inverse barcode: Bright bars on a dark background.



@QRCINV0

**Decode Regular QR Barcodes Only**



@QRCINV2

**\*\* Decode Both**

## Character Encoding



@QRCENC0

**\*\* Default Character Encoding**



@QRCENC1

**UTF-8**



---

## QR ECI Output



@QRCEC10

**Disable QR ECI Output**



@QRCEC11

**\*\* Enable QR ECI Output**

## QR Decoding Complexity

The Decoding Complexity parameter selects the decoding algorithm according to the printing/marketing technique used to create the symbol and on the overall printing/marketing quality.

- ✧ Low: For no stain, wrinkle and other high-quality barcodes;
- ✧ Medium: Bar code with general quality such as certain dirt, fold, etc;
- ✧ High: For Bar code with poor printing quality.



@CCFQRC0

**Low**



@CCFQRC1

**\*\*Medium**



@CCFQRC2

High

## Continuous Code



@QRCMFE1

\*\*On



@QRCMFE0

Off

## Dot-peen Code



@QRCDFE1

\*\*On



@QRCDFE0

Off

---

## Mirror Code

For QR, when you have enabled Image Mirror, processes reverse printed or marked codes (mirrored images).



@QRCMRE0

Off



@QRCMRE1

\*\*On

## QR Model 1

You can set disable and enable QR model 1 through the following setting code.



@QRCM1E0

Off



@QRCM1E1

\*\*On

---

## QR Model 2

Model 2 of QR code is an improved version of model 1, which can also read data smoothly when the QR code is deformed. You can set disable and enable QR model 2 through the following setting code.



@QRCM2E0

**\*\*On**



@QRCM2E1

**Off**

---

## Data Matrix

### Restore Factory Defaults



@DMCDEF

**Restore the Factory Defaults of Data Matrix**

### Enable/Disable Data Matrix



@DMCENA1

**\*\* Enable Data Matrix**



@DMCENA0

**Disable Data Matrix**



If the scanner fails to identify Data Matrix barcodes, you may first try this solution by scanning the **Enable Data Matrix** barcode.

If the **Disable 2D Symbologies** feature is enabled, the Programming Barcode of 2D Data Matrix in this user guide will lose efficacy. It is recommended to use Setup Tool to reactive Data Matrix or scan these barcodes below.



#SETUPE1

**Enter Setup**



#SETUPE0

**Exit Setup**



@DMCENA1

**Enable Data Matrix**

---

## Set Length Range for Data Matrix

The scanner can be configured to only decode Data Matrix barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@DMCMIN

**Set the Minimum Length (Default: 1)**



@DMCMAX

**Set the Maximum Length (Default: 3116)**



Minimum length is not allowed to be greater than maximum length. If you only want to read Data Matrix barcodes with a specific length, set both minimum and maximum lengths to be that desired length.

## Example

**Set the scanner to decode Data Matrix barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode “8” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes “1” and “2” from the “Digit Barcodes” section in Appendix.
6. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.

---

## Rectangular Barcode

Data Matrix has two formats:

Square barcodes having the same amount of modules in length and width: 10\*10, 12\*12.... 144\*144.

Rectangular barcodes having different amounts of modules in length and width: 6\*16, 6\*14...14\*22.



@DMCREC1

**\*\* Enable Rectangular Barcode**



@DMCREC0

**Disable Rectangular Barcode**

## Data Matrix Inverse

Regular barcode: Dark bars on a bright background.

Inverse barcode: Bright bars on a dark background.



@DMCINV0

**Decode Regular Data Matrix Barcodes Only**



@DMCINV2

**\*\* Decode Both**



---

## Character Encoding



@DMCENC0

**\*\* Default Character Encoding**



@DMCENC1

**UTF-8**

## Data Matrix ECI Output



@DMCEC10

**Disable Data Matrix ECI Output**



@DMCEC11

**\*\* Enable Data Matrix ECI Output**

---

## ECC200



@DMCE2E0

**Off**



@DMCE2E1

**\*\* On**

## ECC0TO140



@DMCE0E0

**Off**



@DMCE0E1

**\*\* On**

---

## DM Decoding Complexity

The Decoding Complexity parameter selects the decoding algorithm according to the printing/marking technique used to create the symbol and on the overall printing/marking quality.

- ✧ Low: For no stain, wrinkle and other high-quality barcodes;
- ✧ Medium: Bar code with general quality such as certain dirt, fold, etc;
- ✧ High: For Bar code with poor printing quality.



@CCFDMC0

**Low**



@CCFDMC1

**Medium**



@CCFDMC2

**\*\*High**

---

## Continuous Code



@DMCMFE1

**\*\*On**



@DMCMFE0

**Off**

## Dot-peen Code



@DMCDFE1

**\*\*On**



@DMCDFE0

**Off**

---

## Mirror Code

For Data Matrix, when you have enabled Image Mirror, processes reverse printed or marked codes (mirrored images).



@DMCMRE0

**Off**



@DMCMRE1

**\*\*On**

---

## PDF417

### Restore Factory Defaults



@PDFDEF

**Restore the Factory Defaults of PDF417**

### Enable/Disable PDF417



@PDFENA1

**\*\* Enable PDF417**



@PDFENA0

**Disable PDF417**



If the scanner fails to identify PDF417 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable PDF417** barcode.

---

## Set Length Range for PDF417

The scanner can be configured to only decode PDF417 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



@PDFMIN

**Set the Minimum Length (Default: 1)**



@PDFMAX

**Set the Maximum Length (Default: 2710)**



Minimum length is not allowed to be greater than maximum length. If you only want to read PDF417 barcodes with a specific length, set both minimum and maximum lengths to be that desired length.



**Set the scanner to decode PDF417 barcodes containing between 8 and 12 characters:**

1. Scan the **Set the Minimum Length** barcode.
2. Scan the numeric barcode "8" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Set the Maximum Length** barcode.
5. Scan the numeric barcodes "1" and "2" from the "Digit Barcodes" section in Appendix.
6. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.

---

## PDF417 Inverse

Regular barcode: Dark bars on a bright background.

Inverse barcode: Bright bars on a dark background.



@PDFINV0

**Decode Regular PDF417 Barcodes Only**



@PDFINV1

**Decode Inverse PDF417 Barcodes Only**



@PDFINV2

**\*\*Decode Both**

## Character Encoding



@PDFENC0

**\*\* Default Character Encoding**



@PDFENC1

**UTF-8**



---

## Chapter 8 Data Formatter

### Introduction

You may use the Data Formatter to modify the scanner's output. For example, you can use the Data Formatter to insert characters at certain points in barcode data or to suppress/ replace/ send certain characters in barcode data as it is scanned.

Normally, when you scan a barcode, it gets outputted automatically; however, when you create a format, you must use a "send" command (see the "Send Commands" section in this chapter) within the format programming to output data. Multiple data formats can be programmed into the scanner. The maximum size of all data formats created is 2048 characters. By default, the data formatter is disabled. Enable it when required. If you have changed data format settings, and wish to clear all formats and return to the factory defaults, scan the **Default Data Format** code below.



@DFMDEF

**Default Data Format**

### Add a Data Format

Data format is used to edit barcode data. When you create a data format, you must select one of the four labels (Format\_0, Format\_1, Format\_2 and Format\_3) for your data format, specify the application scope of data format (such as barcode type and data length) and include formatter commands. Multiple data formats may be created using the same label. When scanned data does not match your data format requirements, you will hear the non-match error beep (if the non-match error beep is ON).

There are two methods to program a data format: Programming with barcodes and programming with serial commands.

### Programming with Barcodes

The following explains how to program a data format by scanning the specific barcodes. Scanning any irrelevant barcode or failing to follow the setting procedure will result in programming failure. To find the alphanumeric barcodes needed to

---

create a data format, see the “Digit Barcodes” section in Appendix.

**Step 1:** Scan the **Add Data Format** barcode.



@DFMADD

**Add Data Format**

**Step 2:** Select a label (Format\_0 or Format\_1 or Format\_2 or Format\_3).

Scan a numeric barcode **0** or **1** or **2** or **3** to label this data format Format\_0 or Format\_1 or Format\_2 or Format\_3.

**Step 3:** Select formatter command type.

Specify what type of formatter commands will be used. Scan a numeric barcode “6” to select formatter command type 6. (See the “Formatter Command Type 6” section in this chapter for more information)

**Step 4:** Set interface type

Scan **999** for any interface type.

**Step 5:** Set Symbology ID Number

Refer to the “Symbology ID Number” section in Appendix and find the ID number of the symbology to which you want to apply the data format. Scan three numeric barcodes for the symbology ID number. If you wish to create a data format for all symbologies, scan **999**.

**Step 8:** Set barcode data length

Specify what length of data will be acceptable for this symbology. Scan the four numeric barcodes that represent the data length. 9999 is a universal number, indicating all lengths. For example, 32 characters should be entered as 0032.

**Step 9:** Enter formatter command

Refer to the “Formatter Command Type 6” section in this chapter. Scan the alphanumeric barcodes that represent the command you need to edit data. For example, when a command is F141, you should scan F141.

**Step 10:** Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix to save your data format.

---

**Example:** Program a Format\_0 data format using formatter command type 6, Code 128 containing 10 characters applicable, send all characters followed by “A”.

- |   |   |
|---|---|
| 1. Scan the <b>Add Data Format</b> barcode    | Add a data format                             |
| 2. Scan the <b>0</b> barcode                  | Select Format_0 as the label                  |
| 3. Scan the <b>6</b> barcode                  | Select formatter command type 6               |
| 4. Scan the <b>9</b> barcode three times      | All interface types applicable                |
| 5. Scan the barcodes <b>002</b>               | Only Code 128 applicable                      |
| 6. Scan the barcodes <b>0010</b>              | Only a length of 10 characters applicable     |
| 7. Scan the alphanumeric barcodes <b>F141</b> | Send all characters followed by “A” (HEX: 41) |
| 8. Scan the <b>Save</b> barcode               | Save the data format                          |

To streamline the programming process, you may as well generate a batch barcode by inputting the command (e.g. **@DFMADD069990020010F141;**) used to create a data format. See the “Use Batch Barcode” section in Chapter 10 to learn how to put a batch barcode into use.

When creating multiple data formats sharing a label, the formats are separated from each other by a vertical bar (|) in the batch command, e.g. **@DFMADD069990029999F141|069990039999F142|069990049999F143;**.

---

## Programming with Serial Commands

A data format can also be created by serial commands (HEX) sent from the host device. **All commands must be entered in uppercase letters.**

The syntax consists of the following elements:

**Prefix:** "~<SOH>0000" (HEX: **7E 01 30 30 30 30**), 6 characters.

**Storage type:** "@" (HEX: **40**) or "#" (HEX: **23**), 1 character. "@" means permanent setting which will not be lost by removing power from the scanner or rebooting it; "#" means temporary setting which will be lost by removing power from the scanner or rebooting it.

**Add Data Format Command:** "DFMADD" (HEX: **44 46 4D 41 44 44**), 6 characters.

**Data format label:** "0" (HEX: **30**) or "1" (HEX: **31**) or "2" (HEX: **32**) or "3" (HEX: **33**), 1 character. "0", "1", "2" and "3" represent Format\_0, Format\_1, Format\_2 and Format\_3 respectively.

**Formatter command type:** "6" (HEX: **36**), 1 character.

**Interface type:** "999" (HEX: **39 39 39**), 3 characters.

**Symbology ID Number:** The ID number of the symbology to which you want to apply the data format, 3 characters. 999 indicates all symbologies.

**Data length:** The length of data that will be acceptable for this symbology, 4 characters. 9999 indicates all lengths. For example, 32 characters should be entered as 0032.

**Formatter commands:** The command string used to edit data. For more information, see the "Formatter Command Type 6" section in this chapter.

**Suffix:** "<ETX>" (HEX: **3B 03**), 2 characters.

**Example:** Program a Format\_0 data format using formatter command type 6, Code 128 containing 10 characters applicable, send all characters followed by "A".

Enter: **7E 01 30 30 30 30 40 44 46 4D 41 44 44 30 36 39 39 39 30 30 33 39 39 39 39 46 31 34 31 3B 03**  
(~<SOH>0000@DFMADD069990020010F141;<ETX>)

Response: **02 01 30 30 30 30 40 44 46 4D 41 44 44 30 36 39 39 39 30 30 33 39 39 39 39 46 31 34 31 06 3B 03**  
(<STX><SOH>0000@DFMADD069990020010F141<ACK>;<ETX>)

When creating multiple data formats sharing a label, the formats are separated from each other by a vertical bar (|) in the serial command.

**Example:** ~<SOH>0000@DFMADD069990020010F141|069990039999F142|069990049999F143;<ETX>

---

## Enable/Disable Data Formatter

When Data Formatter is disabled, the barcode data is outputted to the host as read, including prefixes and suffixes.



@DFMENA0

### **\*\* Disable Data Formatter**

You may wish to require the data to conform to a data format you have created. The following settings can be applied to your data format:

**Enable Data Formatter, Required, Keep Prefix/Suffix:** Scanned data that meets your data format requirements is modified accordingly and gets outputted along with prefixes and suffixes (if prefix and suffix are enabled). Any data that does not match your data format requirements generates an error beep (if Non-Match Error Beep is turned ON) and the data in that barcode is not transmitted.

**Enable Data Formatter, Required, Drop Prefix/Suffix:** Scanned data that meets your data format requirements is modified accordingly and gets outputted without prefixes and suffixes (even if prefix and suffix are enabled). Any data that does not match your data format requirements generates an error beep (if Non-Match Error Beep is turned ON) and the data in that barcode is not transmitted.

**Enable Data Formatter, Not Required, Keep Prefix/Suffix:** Scanned data that meets your data format requirements is modified accordingly and gets outputted along with prefixes and suffixes (if prefix and suffix are enabled). Barcode data that does not match your data format requirements is transmitted as read along with prefixes and suffixes (if prefix and suffix are enabled).

**Enable Data Formatter, Not Required, Drop Prefix/Suffix:** Scanned data that meets your data format requirements is modified accordingly and gets outputted without prefixes and suffixes (even if prefix and suffix are enabled). Barcode data that does not match your data format requirements is transmitted as read along with prefixes and suffixes (if prefix and suffix are enabled).



@DFMENA1

**Enable Data Formatter, Required, Keep Prefix/Suffix**



@DFMENA2

**Enable Data Formatter, Required, Drop Prefix/Suffix**



@DFMENA3

**Enable Data Formatter, Not Required, Keep Prefix/Suffix**



@DFMENA4

**Enable Data Formatter, Not Required, Drop Prefix/Suffix**

## **Non-Match Error Beep**

If Non-Match Error Beep is turned ON, the scanner generates an error beep when a barcode is encountered that does not match your required data format.



@DFMTON0

**Non-Match Error Beep Off**



@DFMTON1

**\*\* Non-Match Error Beep On**

---

## Data Format Selection

After enabling the Data Formatter, you can select a data format you want to use by scanning the appropriate barcode below.



@DFMUSE0

**\*\* Format\_0**



@DFMUSE1

**Format\_1**



@DFMUSE2

**Format\_2**



@DFMUSE3

**Format\_3**



---

## Change Data Format for a Single Scan

You can switch between data formats for a single scan. The next barcode is scanned using the data format selected here, then reverts to the format you have selected above. For example, you may have set your scanner to the data format you saved as Format\_3. You can switch to Format\_1 for a single trigger pull by scanning the **Single Scan – Format\_1** barcode below. The next barcode that is scanned uses Format\_1, then reverts back to Format\_3.

Note: This setting will be lost by removing power from the scanner, or turning off/ rebooting the device.



@DFMSIN0

**Single Scan – Format\_0**



@DFMSIN1

**Single Scan – Format\_1**



@DFMSIN2

**Single Scan – Format\_2**



@DFMSIN3

**Single Scan – Format\_3**

---

## Clear Data Format

There are two methods to remove data format created from your scanner:

Delete one data format: Scan the **Clear One** barcode, a numeric barcode (0-3) and the **Save** barcode. For example, to delete Format\_2, you should scan the **Clear One** barcode, the **2** barcode and the **Save** barcode

Delete all data formats: Scan the **Clear All** barcode.



@DFMCAL

**Clear All**



@DFMCLR

**Clear One**

## Query Data Formats

You may scan the appropriate barcode below to get the information of data format(s) created by you or preset by manufacturer. For instance, if you have added Format\_0 as per the example in the “Add a Data Format” section in this chapter, scanning the **Query Current Data Formats** barcode, you will get the result: **Data Format0:069990020010F141;**.



@DFMQCU

**Query Current Data Formats**



@DFMQFA

**Query Preset Data Formats**

---

## Formatter Command Type 6

When working with the Data Formatter, a virtual cursor is moved along your input data string. The following commands are used to both move this cursor to different positions, and to select, replace, and insert data into the final output. For the hex value of ASCII characters involved in the commands, refer to the “ASCII Table” in Appendix.

### Send Commands

#### F1 Send all characters

Syntax=F1xx (xx: The insert character’s hex value)

Include in the output message all of the characters from the input message, starting from current cursor position, followed by an insert character.

#### F2 Send a number of characters

Syntax=F2nnxx (nn: The numeric value (00-99) for the number of characters; xx: The insert character’s hex value)

Include in the output message a number of characters followed by an insert character. Start from the current cursor position and continue for “nn” characters or through the last character in the input message, followed by character “xx.”

#### F2 Example: Send a number of characters



Send the first 10 characters from the barcode above, followed by a carriage return.

Command string: **F2100D**

F2 is the “Send a number of characters” command

10 is the number of characters to send

0D is the hex value for a CR

The data is output as: **1234567890**

**<CR>**

---

### **F3 Send all characters up to a particular character**

Syntax=F3ssxx (ss: The particular character's hex value; xx: The insert character's hex value)

Include in the output message all characters from the input message, starting with the character at the current cursor position and continuing to, but not including, the particular character "ss," followed by character "xx." The cursor is moved forward to the "ss" character.

#### **F3 Example: Send all characters up to a particular character**



Using the barcode above, send all characters up to but not including "D," followed by a carriage return.

Command string: **F3440D**

F3 is the "Send all characters up to a particular character" command

44 is the hex value for a "D"

0D is the hex value for a CR

The data is output as: **1234567890ABC**  
**<CR>**

### **E9 Send all but the last characters**

Syntax=E9nn (nn: The numeric value (00-99) for the number of characters that will not be sent at the end of the message)

Include in the output message all but the last "nn" characters, starting from the current cursor position. The cursor is moved forward to one position past the last input message character included.

### **F4 Insert a character multiple times**

Syntax=F4xxnn (xx: The insert character's hex value; nn: The numeric value (00-99) for the number of times it should be sent)

Send "xx" character "nn" times in the output message, leaving the cursor in the current position.

---

### E9 and F4 Example: Send all but the last characters, followed by 2 tabs



Send all characters except for the last 8 from the barcode above, followed by 2 tabs.

Command string: **E908F40902**

E9 is the "Send all but the last characters" command

08 is the number of characters at the end to ignore

F4 is the "Insert a character multiple times" command

09 is the hex value for a horizontal tab

02 is the number of time the tab character is sent

The data is output as: **1234567890AB<tab><tab>**

### B3 Insert symbology name

Insert the name of the barcode's symbology in the output message, without moving the cursor.

### B4 Insert barcode length

Insert the barcode's length in the output message, without moving the cursor. The length is expressed as a numeric string and does not include leading zeros.

---

### B3 and B4 Example: Insert the symbology name and length



Send the symbology name and length before the barcode data from the barcode above. Break up these insertions with spaces. End with a carriage return.

Command string: **B3F42001B4F42001F10D**

B3 is the "Insert symbology name" command

F4 is the "Insert a character multiple times" command

20 is the hex value for a space

01 is the number of time the space character is sent

B4 is the "Insert barcode length" command

F4 is the "Insert a character multiple times" command

20 is the hex value for a space

01 is the number of time the space character is sent

F1 is the "Send all characters" command

0D is the hex value for a CR

The data is output as: **Code128 20 1234567890ABCDEFGHIJ**  
**<CR>**

### Move Commands

#### F5 Move the cursor forward a number of characters

Syntax=F5nn (nn: The numeric value (00-99) for the number of characters the cursor should be moved ahead)

Move the cursor ahead "nn" characters from current cursor position.

---

#### **F5 Example: Move the cursor forward and send the data**



Move the cursor forward 3 characters, then send the rest of the barcode data from the barcode above. End with a carriage return.

Command string: **F503F10D**

F5 is the "Move the cursor forward a number of characters" command

03 is the number of characters to move the cursor

F1 is the "Send all characters" command

0D is the hex value for a CR

The data is output as: **4567890ABCDEFGHIJ**

**<CR>**

#### **F6 Move the cursor backward a number of characters**

Syntax=F6nn (nn: The numeric value (00-99) for the number of characters the cursor should be moved back)

Move the cursor back "nn" characters from current cursor position.

#### **F7 Move the cursor to the beginning**

Syntax=F7

Move the cursor to the first character in the input message.

#### **EA Move the cursor to the end**

Syntax=EA

Move the cursor to the last character in the input message.



---

## Search Commands

### F8 Search forward for a character

Syntax=F8xx (xx: The search character's hex value)

Search the input message forward for "xx" character from the current cursor position, leaving the cursor pointing to the "xx" character.

### F8 Example: Send barcode data that starts after a particular character



Search for the letter "D" in barcodes and send all the data that follows, including the "D". Using the barcode above:

Command string: **F844F10D**

F8 is the "Search forward for a character" command

44 is the hex value for "D"

F1 is the "Send all characters" command

0D is the hex value for a CR

The data is output as: **DEFGHIJ**

**<CR>**

### F9 Search backward for a character

Syntax=F9xx(xx: The search character's hex value)

Search the input message backward for "xx" character from the current cursor position, leaving the cursor pointing to the "xx" character.

---

### **B0 Search forward for a string**

Syntax=B0nnnnS (nnnn: The string length (up to 9999); S: The ASCII hex value of each character in the string)

Search forward for “S” string from the current cursor position, leaving cursor pointing to “S” string. For example, B0000454657374 will search forward for the first occurrence of the 4-character string “Test.”

### **B0 Example: Send barcode data that starts after a string of characters**



Search for the letters “FGH” in barcodes and send all the data that follows, including “FGH.” Using the barcode above:

Command string: **B00003464748F10D**

B0 is the “Search forward for a string” command

0003 is the string length (3 characters)

46 is the hex value for “F”

47 is the hex value for “G”

48 is the hex value for “H”

F1 is the “Send all characters” command

0D is the hex value for a CR

The data is output as: **FGHIJ**

**<CR>**

### **B1 Search backward for a string**

Syntax=B1nnnnS (nnnn: The string length (up to 9999); S: The ASCII hex value of each character in the string)

Search backward for “S” string from the current cursor position, leaving cursor pointing to “S” string. For example, B1000454657374 will search backward for the first occurrence of the 4-character string “Test.”

---

## E6 Search forward for a non-matching character

Syntax=E6xx (xx: The search character's hex value)

Search the input message forward for the first non-"xx" character from the current cursor position, leaving the cursor pointing to the non-"xx" character.

### E6 Example: Remove zeros at the beginning of barcode data



This example shows a barcode that has been zero filled. You may want to ignore the zeros and send all the data that follows. E6 searches forward for the first character that is not zero, then sends all the data after, followed by a carriage return. Using the barcode above:

Command string: **E630F10D**

E6 is the "Search forward for a non-matching character" command

30 is the hex value for 0

F1 is the "Send all characters" command

0D is the hex value for a CR

The data is output as: **123abc**

**<CR>**

## E7 Search backward for a non-matching character

Syntax=E7xx(xx: The search character's hex value)

Search the input message backward for the first non-"xx" character from the current cursor position, leaving the cursor pointing to the non-"xx" character.

---

## Miscellaneous Commands

### FB Suppress characters

Syntax=FBnnxyy..zz (nn: The numeric value (00-15) for the number of suppressed characters; xyy..zz: The hex value of the characters to be suppressed)

Suppress all occurrences of up to 15 different characters, starting at the current cursor position, as the cursor is advanced by other commands.

#### FB Example: Remove spaces in barcode data



12 34\_5\*6 78

This example shows a barcode that has spaces in the data. You may want to remove the spaces before sending the data. Using the barcode above:

Command string: **FB0120F10D**

FB is the “Suppress characters” command

01 is the number of the characters to be suppressed

20 is the hex value for a space

F1 is the “Send all characters” command

0D is the hex value for a CR

The data is output as: **1234\_5\*678**

**<CR>**

### E4 Replace characters

Syntax = E4nnxx<sub>1</sub>xx<sub>2</sub>yy<sub>1</sub>yy<sub>2</sub>...zz<sub>1</sub>zz<sub>2</sub>(nn: The total count of the number of characters (characters to be replaced plus replacement characters; xx<sub>1</sub>: The characters to be replaced, xx<sub>2</sub>: The replacement characters, continuing through zz<sub>1</sub> and zz<sub>2</sub>)

Replace up to 15 characters in the output message, without moving the cursor.

---

#### E4 Example: Replace zeros with CRs in barcode data



If the barcode has characters that the host application does not want included, you can use the E4 command to replace those characters with something else. In this example, you will replace the zeros in the barcode above with carriage returns.

Command string: **E402300DF10D**

E4 is the "Replace characters" command

02 is the total count of characters to be replaced, plus the replacement characters (0 is replaced by CR, so total characters=2)

30 is the hex value for 0

0D is the hex value for a CR (the character that will replace the 0)

F1 is the "Send all characters" command

0D is the hex value for a CR

The data is output as: **123**

**456**

**78**

**AB**

**<CR>**

---

## BA Replace a string with another

Syntax=BA $nn$  $NN_1$  $SS_1$  $NN_2$  $SS_2$

$nn$ : The count of replacements to be made, if  $nn=00$  or  $nn\geq$  the number of occurrences of a string to be replaced, then replace all occurrences of that string.

$NN_1$ : The length of the string to be replaced,  $NN_1>0$ .

$SS_1$ : The ASCII hex value of each character in the string to be replaced.

$NN_2$ : The length of replacement string,  $NN_2\geq 0$ . To replace string " $SS_1$ " with NUL (i.e. delete string " $SS_1$ "), you should set  $NN_2$  to 00 and leave out  $SS_2$ .

$SS_2$ : The ASCII hex value of each character in the replacement string.

From the current cursor position, search forward for the occurrence of " $SS_1$ " string (of length " $NN_1$ ") and replace the string with " $SS_2$ " string (of length " $NN_2$ ") in the output message until every " $SS_1$ " string is replaced or the count of replacements made reaches " $nn$ " times, without moving the cursor.

### BA Example: Replace "23"s with "ABC"s in barcode data



If the barcode has a string of characters that the host application does not want included, you can use the BA command to replace the string with something else. In this example, you will replace the "23"s in the barcode above with "ABC"s.

Command string: **BA0002323303414243F100**

BA is the "Replace a string with another" command

00 is the count of replacements to be made, 00 means to replace all occurrences of that string

02 is the length of the string to be replaced

32 is the hex value for 2 (character in the string to be replaced)

33 is the hex value for 3 (character in the string to be replaced)

03 is the length of the replacement string

---

41 is the hex value for A (character in the replacement string)

42 is the hex value for B (character in the replacement string)

43 is the hex value for C (character in the replacement string)

F1 is the "Send all characters" command

00 is the hex value for a NUL

The data is output as: **1ABC4AbcABCR01ABCU**

#### **BA Example: Remove only the first occurrence of "23"s in barcode data**

If the barcode has a string of characters that the host application wants removed, you can use the BA command to replace the string with NUL. In this example, you will remove the first occurrence of "23" in the barcode above.

Command string: **BA0102323300F100**

BA is the "Replace a string with another" command

01 is the count of replacements to be made

02 is the length of the string to be replaced

32 is the hex value for 2 (character in the string to be replaced)

33 is the hex value for 3 (character in the string to be replaced)

00 is the length of the replacement string, 00 means to replace the string to be replaced with NUL

F1 is the "Send all characters" command

00 is the hex value for a NUL

The data is output as: **14Abc23R0123U**

---

## **EF Insert a delay**

Syntax=EFnnnn (nnnn: The delay in 5ms increments, up to 9999)

Inserts a delay of up to 49,995 milliseconds (in multiples of 5), starting from the current cursor position. This command can only be used with USB HID Keyboard.

### **EF Example: Insert a delay of 1s between the 5<sup>th</sup> and 6<sup>th</sup> character**

Send the first 5 characters in a barcode, wait for 1s, then send the rest of the barcode data.

Command string: **F20500EF0200E900**

F2 is the "Send a number of characters" command

05 is the number of characters to send

00 is the hex value for a Null character

EF is the "Insert a delay" command

0200 is the delay value (5msX200=1000ms=1s)

E9 is the "Send all but the last characters" command

00 is the number of characters that will not be sent at the end of the message



## Chapter 9 Prefix & Suffix

### Introduction

A 1D barcode could contain digits, letters, symbols, etc. A 2D barcode could contain more data, such as Chinese characters and other multi-byte characters. However, in real applications, they do not and should not have all information we need, such as barcode type, data acquisition time and delimiter, in order to keep the barcodes short and flexible.

Prefix and suffix are how to fulfill the needs mentioned above. They can be added, removed and modified while the original barcode data remains intact.



Barcode processing procedure:

1. Edit data with Data Formatter
2. Append prefix/suffix
3. Pack data
4. Append terminating character

---

## Global Settings

### Enable/Disable All Prefixes/Suffixes

**Disable All Prefixes/Suffixes:** Transmit barcode data with no prefix/suffix.

**Enable All Prefixes/Suffixes:** Allow to append Code ID prefix, AIM ID prefix, custom prefix/suffix and terminating character to the barcode data before the transmission.



@APSENA0

**Disable All Prefixes/Suffixes**



@APSENA1

**Enable All Prefixes/Suffixes**

## Prefix Sequence



@PRESEQ0

**\*\* Code ID+ Custom +AIM ID**



@PRESEQ1

**Custom + Code ID + AIM ID**

---

## Custom Prefix

### Enable/Disable Custom Prefix

If custom prefix is enabled, you are allowed to append to the data a user-defined prefix that cannot exceed 10 characters. For example, if the custom prefix is “AB” and the barcode data is “123”, the Host will receive “AB123”.



@CPRENA0

**\*\* Disable Custom Prefix**



@CPRENA1

**Enable Custom Prefix**

### Set Custom Prefix

To set a custom prefix, scan the **Set Custom Prefix** barcode then the numeric barcodes corresponding to the hexadecimal value of a desired prefix then the **Save** barcode.

**Note:** A custom prefix cannot exceed 10 characters.



@CPRSET

**Set Custom Prefix**

**E**  
*xample*

**Set the custom prefix to “CODE” (HEX: 0x43/0x4F/0x44/0x45):**

1. Scan the **Set Custom Prefix** barcode.
2. Scan the numeric barcodes “4”, “3”, “4”, “F”, “4”, “4”, “4” and “5” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
4. Scan the **Enable Custom Prefix** barcode.

---

## AIM ID Prefix

AIM (Automatic Identification Manufacturers) ID defines symbology identifier (For the details, see the “AIM ID Table” section in Appendix). If AIM ID prefix is enabled, the scanner will add the symbology identifier before the scanned data after decoding.



@AIDENA0

**\*\* Disable AIM ID Prefix**



@AIDENA1

**Enable AIM ID Prefix**



AIM ID is not user programmable.

---

## Code ID Prefix

Code ID can also be used to identify barcode type. Unlike AIM ID, Code ID is user programmable. Code ID can only consist of one or two English letters.



@CIDENA0

**\*\* Disable Code ID Prefix**



@CIDENA1

**Enable Code ID Prefix**

## Restore All Default Code IDs

For the information of default Code IDs, see the “Code ID Table” section in Appendix.



@CIDDEF

**Restore All Default Code IDs**

## Modify Code ID

See the examples below to learn how to modify a Code ID and restore the default Code IDs of all symbologies.

---

## **E** *Example*

### **Modify PDF417 Code ID to be “p” (HEX: 0x70):**

1. Scan the **Modify PDF417 Code ID** barcode.
2. Scan the numeric barcodes “7” and “0” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.

### **Restore the default Code IDs of all symbologies:**

1. Scan the **Restore All Default Code IDs** barcode.

---

1D symbologies:



@CID002

**Modify Code 128 Code ID**



@CID003

**Modify GS1-128 Code ID**



@CID004

**Modify EAN-8 Code ID**



@CID005

**Modify EAN-13 Code ID**



@CID006

**Modify UPC-E Code ID**



@CID007

**Modify UPC-A Code ID**



@CID008

**Modify Interleaved 2 of 5 Code ID**



@CID009

**Modify ITF-14 Code ID**



@CID010

**Modify ITF-6 Code ID**



@CID011

**Modify Matrix 2 of 5 Code ID**



@CID013

**Modify Code 39 Code ID**



@CID015

**Modify Codabar Code ID**





@CID017

**Modify Code 93 Code ID**



@CID020

**Modify AIM 128 Code ID**



@CID021

**Modify ISBT 128 Code ID**



@CID022

**Modify COOP 25 Code ID**



@CID023

**Modify ISSN Code ID**



@CID024

**Modify ISBN Code ID**



@CID025

**Modify Industrial 25 Code ID**



@CID026

**Modify Standard 25 Code ID**



@CID027

**Modify Plessey Code ID**



@CID028

**Modify Code 11 Code ID**



@CID029

**Modify MSI-Plessey Code ID**



@CID031

**Modify GS1 Databar Code ID**



@CID030

**Modify GS1 Composite Code ID**

---

**2D symbologies:**



@CID032

**Modify PDF417 Code ID**



@CID033

**Modify QR Code ID**



@CID035

**Modify Data Matrix Code ID**

---

## Custom Suffix

### Enable/Disable Custom Suffix

If custom suffix is enabled, you are allowed to append to the data a user-defined suffix that cannot exceed 10 characters. For example, if the custom suffix is “AB” and the barcode data is “123”, the Host will receive “123AB”.



@CSUENA0

**\*\* Disable Custom Suffix**



@CSUENA1

**Enable Custom Suffix**

### Set Custom Suffix

To set a custom suffix, scan the **Set Custom Suffix** barcode then the numeric barcodes corresponding to the hexadecimal value of a desired suffix then the **Save** barcode.

**Note:** A custom suffix cannot exceed 10 characters.



@CSUSET

**Set Custom Suffix**

**E**  
*xample*

**Set the custom suffix to “CODE” (HEX: 0x43/0x4F/0x44/0x45):**

1. Scan the **Set Custom Suffix** barcode.
2. Scan the numeric barcodes “4”, “3”, “4”, “F”, “4”, “4”, “4” and “5” from the “Digit Barcodes” section in Appendix.
3. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.

- 
4. Scan the **Enable Custom Suffix** barcode.

## Terminating Character Suffix

### Enable/Disable Terminating Character Suffix

A terminating character such as carriage return (CR) or carriage return/line feed pair (CRLF) can only be used to mark the end of data, which means nothing can be added after it.



@TSUENA0

### Disable Terminating Character Suffix



@TSUENA1

### \*\* Enable Terminating Character Suffix

### Set Terminating Character Suffix

To set a terminating character suffix, scan the **Set Terminating Character Suffix** barcode then the numeric barcodes corresponding to the hexadecimal value of a desired terminating character then the **Save** barcode.

**Note:** A terminating character suffix cannot exceed 2 characters.



@TSUSET

### Set Terminating Character Suffix



@TSUSET0D

### \*\* Set Terminating Character to CR (0x0D)



@TSUSET0D0A

**Set Terminating Character to CRLF (0x0D,0x0A)**

**Set the terminating character suffix to 0x0A:**

1. Scan the **Set Terminating Character Suffix** barcode.
2. Scan the numeric barcodes "0" and "A" from the "Digit Barcodes" section in Appendix.
3. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
4. Scan the **Enable Terminating Character Suffix** barcode.

---

## Chapter 10 Batch Programming

### Introduction

Batch programming enables users to integrate a batch of commands into a single batch barcode.

Listed below are batch programming rules:

1. Command format: Command + Parameter Value.
2. Each command is terminated by a semicolon (;). Note that there is no space between a command and its terminator semicolon.
3. Use the barcode generator software to generate a 2D batch barcode.

Example: Create a batch barcode for **Illumination On, Sense Mode, Decode Session Timeout = 2s**:

1. Input the commands:

```
@ILLSCN1;SCNMOD2;ORTSET2000;
```

2. Generate a batch barcode.

When setting up a scanner with the above configuration, scan the **Enable Batch Barcode** barcode and then the batch barcode generated.



@BATCHS

**Enable Batch Barcode**



---

## Create a Batch Command

A batch command may contain a number of individual commands each of which is terminated by a semicolon (;).

For more information, refer to the “Use of Programming Command” section in Chapter 3.

## Create a Batch Barcode

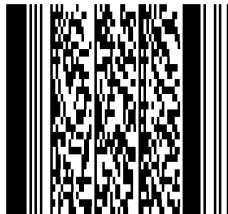
Batch barcodes can be produced in the format of PDF417, QR Code or Data Matrix.

Example: Create a batch barcode for **Illumination On, Sense Mode, Decode Session Timeout = 2s**:

1. Input the following commands:

```
@ILLSCN1;SCNMOD2;ORTSET2000;
```

2. Generate a PDF417 batch barcode.



---

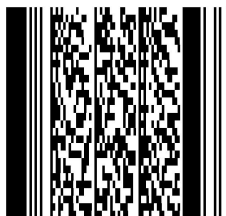
## Use Batch Barcode

To put a batch barcode into use, scan the following barcodes. (Use the example above.)



@BATCHS

**Enable Batch Barcode**



**Batch Barcode**

# Appendix

## Digit Barcodes

0~9



@DIGIT0

0



@DIGIT1

1



@DIGIT2

2



@DIGIT3

3



@DIGIT4

4



@DIGIT6

6



@DIGIT8

8



@DIGIT5

5



@DIGIT7

7



@DIGIT9

9

---

A~F



@DIGITA

A



@DIGITB

B



@DIGITC

C



@DIGITD

D



@DIGITE

E



@DIGITF

F

---

## Save/Cancel Barcodes

After reading numeric barcode(s), you need to scan the **Save** barcode to save the data. If you scan the wrong digit(s), you can either scan the **Cancel** barcode and then start the configuration all over again, or scan the **Delete the Last Digit** barcode and then the correct digit, or scan the **Delete All Digits** barcode and then the digits you want.

For instance, after reading the **Maximum Length** barcode and numeric barcodes “1”, “2” and “3”, you scan:

- ✧ **Delete the Last Digit:** The last digit “3” will be removed.
- ✧ **Delete All Digits:** All digits “123” will be removed.
- ✧ **Cancel:** The maximum length configuration will be cancelled. And the scanner is still in the setup mode.



@DIGSAV

**Save**



@DIGCAN

**Cancel**



@DIGDEL

**Delete the Last Digit**



@DIGDAL

**Delete All Digits**

## Factory Defaults Table

Parameter	Factory Default	Remark
<b>System Settings</b>		
Barcode Programming	Disabled (Exit Setup)	
Programming Barcode Data	Do not transmit	
Illumination	On	
Illumination Color	Red LED and White LED	
Interval Time of Illumination switching	2000ms	
Aiming	Normal	
Good Read LED	On	
Good Read LED Duration	Short (20ms)	
Power On Beep	On	
Good Read Beep	On	
Good Read Beep Duration	Medium (80ms)	
Good Read Beep Frequency	Medium (2730Hz)	
Good Read Beep Volume	High	
Number of Good Read Beeps	1	
Good Read Beep Interval Time	Short	
Good Read Vibration	Off	
Good Read Vibration Duration	300ms	
Default Scan Mode	Level Mode	
Decode Session Timeout	3,000ms	1-3,600,000ms; 0: infinite.
Timeout between Decodes	500ms	1-10,000ms
Reread Timeout	Enabled, 15,000ms	1-3,600,000ms
Good Read Delay	Disabled, 500ms	1-3,600,000ms
Image Decoding Timeout	500ms	1-3,000ms
Sensitivity	Level 1	Level 1-20
Scanning Preference	Normal Mode	
Read Barcode	On	
Decode Area	Priori on Aimed Barcode	
Specify Decoding Area	40% top, 60% bottom, 40% left, 60% right	
Image Flipping	Do Not Flip	
Image Resize	Original Image	
Default Interface	USB HID Keyboard	
<b>RS-232 Interface</b>		
Baud Rate	9600	

Parity Check	None	
Data Bits	8	
Stop Bits	1	
Hardware Auto Flow Control	Disabled	
<b>USB Interface</b>		
USB Country Keyboard	US keyboard	USB HID Keyboard
Beep on Unknown Character	Off	USB HID Keyboard
Emulate ALT+Keypad	Off	USB HID Keyboard
Code Page	Code Page 1252 (West European Latin)	USB HID Keyboard
Unicode Encoding	Off	USB HID Keyboard
Function Key Mapping	Disable	USB HID Keyboard
Inter-Keystroke Delay	No Delay	USB HID Keyboard
Convert Case	No Case Conversion	USB HID Keyboard
Emulate Numeric Keypad 1	Disabled	USB HID Keyboard
Emulate Numeric Keypad 2	Disabled	USB HID Keyboard
Polling Rate	4ms	USB HID Keyboard
Leading Zero	On	
<b>Wireless Communication</b>		
Batch Mode Option	Off	
Prevent Same Barcode Storage	Off	
Batch Mode Transmit Delay	No Transmit Delay (0ms)	
Auto Clear Flash Memory	Off	
Set End of Transmission Message	Off	
Wireless Scanner Sleep Timeout	15s	
Wireless Scanner Shutdown Timeout	30min	
<b>Symbologies</b>		
<b>Global Settings</b>		
Image Brightness Adaptation 1	15	
Image Brightness Adaptation 2	10	
1D Decoding Complexity	Medium	
1D Inverse	Regular and Inverse	
<b>Code 128</b>		
Code 128	Enabled	
Maximum Length	48	
Minimum Length	1	
<b>EAN-8</b>		
EAN-8	Enabled	
Check Character	Transmit	



2-Digit Add-On Code	Disabled	
5-Digit Add-On Code	Disabled	
Add-On Code	Not Required	
Add-On Code Separator	Off	
Convert EAN-8 to EAN-13	Disabled	
<b><i>EAN-13</i></b>		
EAN-13	Enabled	
Check Character	Transmit	
2-Digit Add-On Code	Disabled	
5-Digit Add-On Code	Disabled	
Add-On Code	Not required	
Add-On Code Separator	Off	
EAN-13 Beginning with 290 Add-On Code Required	Do Not Require Add-On Code	
EAN-13 Beginning with 378/379 Add-On Code Required	Do Not Require Add-On Code	
EAN-13 Beginning with 414/419 Add-On Code Required	Do Not Require Add-On Code	
EAN-13 Beginning with 434/439 Add-On Code Required	Do Not Require Add-On Code	
EAN-13 Beginning with 977 Add-On Code Required	Do Not Require Add-On Code	
EAN-13 Beginning with 978 Add-On Code Required	Do Not Require Add-On Code	
EAN-13 Beginning with 979 Add-On Code Required	Do Not Require Add-On Code	
<b><i>UPC-E</i></b>		
UPC-E0	Enabled	
UPC-E1	Disabled	
Check Character	Transmit	
2-Digit Add-On Code	Disabled	
5-Digit Add-On Code	Disabled	
Add-On Code	Not required	
Add-On Code Separator	Off	
Transmit Preamble Character	System Character	
Convert UPC-E to UPC-A	Disabled	
<b><i>UPC-A</i></b>		
UPC-A	Enabled	

Check Character	Transmit	
2-Digit Add-On Code	Disabled	
5-Digit Add-On Code	Disabled	
Add-On Code	Not required	
Add-On Code Separator	Off	
Transmit Preamble Character	System Character	
<b>Interleaved 2 of 5</b>		
Interleaved 2 of 5	Enabled	
Check Character Verification	Disabled	
Maximum Length	80	
Minimum Length	6	
<b>ITF-14</b>		
ITF-14	Disabled	
<b>ITF-6</b>		
ITF-6	Disabled	
<b>Matrix 2 of 5</b>		
Matrix 2 of 5	Enabled	
Check Character Verification	Disabled	
Maximum Length	80	
Minimum Length	4	
<b>Code 39</b>		
Code 39	Enabled	
Check Character Verification	Disabled	
Start/Stop Character	Do not transmit	
Code 39 Full ASCII	Disabled	
Maximum Length	48	
Minimum Length	1	
<b>Codabar</b>		
Codabar	Enabled	
Check Character Verification	Disabled	
Start/Stop Character	Do not transmit ABCD/ABCD as Start/Stop Character	
Maximum Length	60	
Minimum Length	2	
<b>Code 93</b>		
Code 93	Disabled	
Check Character	Do not transmit	
Check Character Verification	Enabled	

Maximum Length	48	
Minimum Length	1	
<b>GS1-128 (UCC/EAN-128)</b>		
GS1-128	Enabled	
Maximum Length	48	
Minimum Length	1	
<b>AIM 128</b>		
AIM 128	Disabled	
Maximum Length	48	
Minimum Length	1	
Check Character	Do not transmit	
<b>ISBT 128</b>		
ISBT 128	Disabled	
<b>Code 11</b>		
Code 11	Disabled	
Check Character	Transmit	
Check Character Verification	One Check Character, MOD11	
Maximum Length	48	
Minimum Length	4	
<b>ISBN</b>		
ISBN	Disabled	
ISBN Format	ISBN-10	
<b>ISSN</b>		
ISSN	Disabled	
<b>Industrial 25</b>		
Industrial 25	Disabled	
Check Character Verification	Disabled	
Maximum Length	48	
Minimum Length	6	
<b>Standard 25</b>		
Standard 25	Disabled	
Check Character Verification	Disabled	
Maximum Length	48	
Minimum Length	6	
<b>COOP 25</b>		
COOP 25	Disabled	
Check Character Verification	Disabled	
Maximum Length	48	

Minimum Length	1	
<b>Deutsche 14</b>		
Deutsche 14	Disabled	
<b>Deutsche 12</b>		
Deutsche 12	Disabled	
<b>Plessey</b>		
Plessey	Disabled	
Check Character Verification	Disabled	
Maximum Length	48	
Minimum Length	4	
<b>MSI-Plessey</b>		
MSI-Plessey	Disabled	
Check Character	Transmit	
Check Character Verification	One Check Character, MOD10	
Maximum Length	48	
Minimum Length	4	
<b>PDF417</b>		
PDF417	Enabled	
PDF417 Inverse	Decode Both	
Character Encoding	Default Character Encoding	
Maximum Length	2710	
Minimum Length	1	
<b>QR Code</b>		
QR Code	Enabled	
Read Single QR Only	Enabled	
QR Inverse	Read regular & inverse barcodes	
Character Encoding	Default Character Encoding	
QR ECI Output	Enabled	
Maximum Length	7089	
Minimum Length	1	
QR Decoding Complexity	Medium	
Continuous Code	Enabled	
Dot-peen Code	Enabled	
Mirror Code	Enabled	
QR Model 1	Enabled	
QR Model 2	Enabled	
<b>Data Matrix</b>		
Data Matrix	Enabled	

Rectangular Barcode	Enabled	
Data Matrix Inverse	Read regular & inverse barcodes	
Character Encoding	Default Character Encoding	
Data Matrix ECI Output	Enabled	
Maximum Length	3116	
Minimum Length	1	
Data Matrix Decoding Complexity	Medium	
ECC200	Enabled	
ECCTO140	Enabled	
Continuous Mode	Enabled	
Dot-peen Code	Enabled	
Mirror Code	Enabled	
<b>Data Formatter</b>		
Data Formatter	Disabled	
Non-Match Error Beep	On	
Data Format Selection	Format_0	
<b>Prefix &amp; Suffix</b>		
Prefix Sequence	Code ID+ Custom +AIM ID	
Custom Prefix	Disabled	
AIM ID Prefix	Disabled	
Code ID Prefix	Disabled	
Custom Suffix	Disabled	
Terminating Character Suffix	Enabled	

## AIM ID Table

Symbology	AIM ID	Possible AIM ID Modifiers (m)
Code 128	JC0	
GS1-128 (UCC/EAN-128)	JC1	
EAN-8	JE4	
EAN-8 with Addon	JE3	
EAN-13	JE0	
EAN-13 with Addon	JE3	
UPC-E	JE0	
UPC-E with Addon	JE3	
UPC-A	JE0	
UPC-A with Addon	JE3	
Interleaved 2 of 5	JIm	0, 1, 3
ITF-14	JIm	1, 3
ITF-6	JIm	1, 3
Matrix 2 of 5	JX0	
Code 39	JAm	0, 1, 3, 4, 5, 7
Codabar	JFm	0, 2, 4
Code 93	JG0	
ISSN	JX0	
ISBN	JX0	
Industrial 25	JS0	
Standard 25	JR0	
COOP 25	JX0	
Deutsche 12	JX0	
Deutsche 14	JX0	
AIM 128	JC2	
Plessey	JP0	
Code 11	JHm	0, 1, 3
MSI Plessey	JMm	0, 1
PDF417	JLm	0-2
QR Code	JQm	0-6
Data Matrix	Jdm	0-6

**Note:** “m” represents the AIM modifier character. Refer to ISO/IEC 15424:2008 Information technology – Automatic identification and data capture techniques – Data Carrier Identifiers (including Symbology Identifiers) for AIM modifier character details.

## Code ID Table

Symbology	Code ID
Code128	j
GS1-128 (UCC/EAN-128)	j
EAN-8	d
EAN-13	d
UPC-E	c
UPC-A	c
Interleaved 2 of 5	e
ITF-14	e
ITF-6	e
Matrix 2 of 5	v
Code 39	b
Codabar	a
Code 93	i
AIM 128	X
COOP 25	X
Deutsche 12	X
Deutsche 14	X
ISSN	g
ISBN	B
Industrial 2/5	l
Standard 2/5	f
Plessey	n
Code 11	H
MSI Plessey	m
PDF417	r
QR Code	s
Data Matrix	u

---

## Symbology ID Number

Symbology	ID Number
Code 128	002
GS1-128 (UCC/EAN-128)	003
EAN-8	004
EAN-13	005
UPC-E	006
UPC-A	007
Interleaved 2 OF 5	008
ITF-14	009
ITF-6	010
Matrix 2 of 5	011
Code 39	013
Codabar	015
Code 93	017
AIM 128	020
COOP 25	022
ISSN	023
ISBN	024
Industrial25	025
Standard25	026
Plessey	027
Code11	028
MSI-Plessey	029
PDF417	032
QR Code	033
Data Matrix	035
Deutsche 14	128
Deutsche 12	129



## ASCII Table

Hex	Dec	Char
00	0	NUL (Null char.)
01	1	SOH (Start of Header)
02	2	STX (Start of Text)
03	3	ETX (End of Text)
04	4	EOT (End of Transmission)
05	5	ENQ (Enquiry)
06	6	ACK (Acknowledgment)
07	7	BEL (Bell)
08	8	BS (Backspace)
09	9	HT (Horizontal Tab)
0a	10	LF (Line Feed)
0b	11	VT (Vertical Tab)
0c	12	FF (Form Feed)
0d	13	CR (Carriage Return)
0e	14	SO (Shift Out)
0f	15	SI (Shift In)
10	16	DLE (Data Link Escape)
11	17	DC1 (XON) (Device Control 1)
12	18	DC2 (Device Control 2)
13	19	DC3 (XOFF) (Device Control 3)
14	20	DC4 (Device Control 4)
15	21	NAK (Negative Acknowledgment)
16	22	SYN (Synchronous Idle)
17	23	ETB (End of Trans. Block)
18	24	CAN (Cancel)
19	25	EM (End of Medium)
1a	26	SUB (Substitute)
1b	27	ESC (Escape)
1c	28	FS (File Separator)
1d	29	GS (Group Separator)

Hex	Dec	Char
1e	30	RS (Request to Send)
1f	31	US (Unit Separator)
20	32	SP (Space)
21	33	! (Exclamation Mark)
22	34	" (Double Quote)
23	35	# (Number Sign)
24	36	\$ (Dollar Sign)
25	37	% (Percent)
26	38	& (Ampersand)
27	39	` (Single Quote)
28	40	( (Left/ Opening Parenthesis)
29	41	) (Right/ Closing Parenthesis)
2a	42	* (Asterisk)
2b	43	+ (Plus)
2c	44	, (Comma)
2d	45	- (Minus/ Dash)
2e	46	. (Dot)
2f	47	/ (Forward Slash)
30	48	0
31	49	1
32	50	2
33	51	3
34	52	4
35	53	5
36	54	6
37	55	7
38	56	8
39	57	9
3a	58	: (Colon)
3b	59	; (Semi-colon)
3c	60	< (Less Than)
3d	61	= (Equal Sign)

Hex	Dec	Char
3e	62	> (Greater Than)
3f	63	? (Question Mark)
40	64	@ (AT Symbol)
41	65	A
42	66	B
43	67	C
44	68	D
45	69	E
46	70	F
47	71	G
48	72	H
49	73	I
4a	74	J
4b	75	K
4c	76	L
4d	77	M
4e	78	N
4f	79	O
50	80	P
51	81	Q
52	82	R
53	83	S
54	84	T
55	85	U
56	86	V
57	87	W
58	88	X
59	89	Y
5a	90	Z
5b	91	[ (Left/ Opening Bracket)
5c	92	\ (Back Slash)
5d	93	] (Right/ Closing Bracket)

Hex	Dec	Char
5e	94	^ (Caret/ Circumflex)
5f	95	_ (Underscore)
60	96	' (Grave Accent)
61	97	a
62	98	b
63	99	c
64	100	d
65	101	e
66	102	f
67	103	g
68	104	h
69	105	i
6a	106	j
6b	107	k
6c	108	l
6d	109	m
6e	110	n
6f	111	o
70	112	p
71	113	q
72	114	r
73	115	s
74	116	t
75	117	u
76	118	v
77	119	w
78	120	x
79	121	y
7a	122	z
7b	123	{ (Left/ Opening Brace)
7c	124	(Vertical Bar)
7d	125	} (Right/ Closing Brace)
7e	126	~ (Tilde)
7f	127	DEL (Delete)

## Newland Auto-ID Tech. Co., Ltd. (Headquarter)

Add: No.1, Rujiang West Rd., Mawei, Fuzhou, Fujian 350001, China  
Tel: +86 (0) 591 8397 9500  
Fax: +86 (0) 591 8397 9216  
E-mail: [info@nlscan.com](mailto:info@nlscan.com)  
Web: [www.newlandaidc.com](http://www.newlandaidc.com)

### Newland APAC

#### Newland Taiwan Inc.

Add: 7F-6, No. 268, Liancheng Rd., Jhonghe Dist. 235, New Taipei City, Taiwan  
Tel: +886 2 7731 5388  
Fax: +886 2 7731 5389  
Email: [info@newland-id.com.tw](mailto:info@newland-id.com.tw)  
Web: [www.newland-id.com.tw](http://www.newland-id.com.tw)

#### Newland Japan

Add: REID-C Meguro Fudomae-Bldg.7F, 3-15-6 Nishigotanda Shinagawa-ku, Tokyo, Japan  
Tel: +886 2 7731 5388 ext. 71  
Email: [info@nlscan.com](mailto:info@nlscan.com)  
Web: [www.newlandaidc.com/jp/](http://www.newlandaidc.com/jp/)

#### Newland Korea

Add: Biz. Center Best-one, Jang-eun Medical Plaza 6F, Bojeong-dong 1261-4, Kihung-gu, Yongin-City, Kyunggi-do, South Korea  
Tel: +82 10 8990 4838  
Fax: +82 70 4369 0009  
Email: [info@nlscan.com](mailto:info@nlscan.com)  
Web: [www.newlandaidc.com/kor/](http://www.newlandaidc.com/kor/)

#### Newland India

Add: 814, Tower B, NOIDA ONE business park B-8, Sector 62, Noida, Uttar Pradesh-201301  
Tel: +91 120 7964266  
Email: [info@nlscan.com](mailto:info@nlscan.com)  
Web: [www.newlandaidc.com](http://www.newlandaidc.com)

### Newland EMEA

#### Newland Europe BV

Add: Rolweg 25, 4104 AV Culemborg, The Netherlands  
Tel: +31 (0) 345 87 00 33  
Fax: +31 (0) 345 87 00 39  
Email: [sales@newland-id.com](mailto:sales@newland-id.com)  
Tech Support: [tech-support@newland-id.com](mailto:tech-support@newland-id.com)  
Web: [www.newland-id.com](http://www.newland-id.com)

### Newland NALA

#### Newland North America Inc.

Add: 46559 Fremont Blvd., Fremont, CA 94538, USA  
Tel: +1 510 490 3888  
Fax: +1 510 490 3887  
Email: [info@nlscan.com](mailto:info@nlscan.com)  
Web: [www.newlandamerica.com](http://www.newlandamerica.com)

#### Newland Latin America

Tel: +1 (239) 598 0068  
Fax: +1 (239) 280 1238  
Email: [info@newlandla.com](mailto:info@newlandla.com)  
Web: [www.newlandamerica.com](http://www.newlandamerica.com)



**Newland Auto-ID Tech. Co., Ltd.**