

N1-W

OEM Scan Engines

Disclaimer

© 2023 Newland Europe BV. All rights reserved.

Please read the manual carefully before using the product and operate it according to the manual. It is advised that you 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 Newland Europe BV.

All pictures in this manual are for reference only, and the actual product may differ.

Regarding product modification and update, Newland Europe BV 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 Newland Europe BV or a 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 Newland Europe BV's written permission.

Risk Warning Regarding Unauthorized System Updates:

You should use the Newland-provided tool to update this product's system. Modifying system files by installing a third-party ROM system or using any cracking method may result in product malfunction or data loss and void your warranty.

Newland Europe BV reserves the right to make a final interpretation of the statement above.

Newland Europe BV

Rolweg 25, 4104 AV, Culemborg,
The Netherlands
www.newland-id.com

Newland Europe BV is a subsidiary of Newland Digital Technology Co., Ltd. Our general conditions of Purchase, Sale and Delivery are filed with the Record Office of the Chamber of Commerce of Utrecht, The Netherlands.

K.v.K. H.R. Utrecht / Chamber of
Commerce Utrecht:
Reg. nr. 17109876

Revision History

Version	Description	Date
V1.0.0	Initial release.	December 09, 2022
V1.0.1	Deleted the 1D Twin Code in Chapter 10.	July 19, 2023

Table of Contents

Revision History	3
Preface	1
Introduction.....	1
Chapter Description.....	1
Explanation of Symbols.....	2
Explanation of Icons.....	2
Chapter 1 Getting Started	3
Introduction.....	3
Symbologies.....	3
N1-W Scan Engine.....	4
Chapter 2 Installation	5
Introduction.....	5
Dimensions (unit: mm).....	5
Mounting.....	6
13-PIN FPC Connector.....	7
Connector Specifications	7
FPC Cable.....	9
ESD.....	10
Dust and Dirt.....	10
Ambient Environment.....	10
Thermal Considerations.....	10
Maintenance.....	11
Chapter 3 Optics	12
Introduction.....	12
Sensor.....	12
Illumination.....	12
Aimer.....	12
Window Placement.....	13
Window Material and Color.....	14
Coatings and Scratch Resistance.....	14
Window Size.....	15
Ambient Light.....	18
Eye Safety.....	18
Depth of Field.....	18
Chapter 4 Electrical Specifications	19
Power Supply.....	19

Ripple Noise	19
Interface Pinouts.....	20
DC Characteristics.....	21
Operating Voltage.....	21
Operating Current.....	22
I/O Voltage.....	22
Timing Sequence.....	23
Power Up Timing Sequence.....	23
Chapter 5 External Reference Circuit	24
External Circuit Design.....	24
Good Read LED Circuit.....	24
Beeper Circuit.....	25
Trigger Circuit.....	25
External Light Control Circuit.....	27
Reset Circuit.....	28
Chapter 6 Auxiliary Tools	29
EVK	29
EasySet	29
UFCOM	30
Chapter 7 Configuration.....	31
Introduction.....	31
Barcode Programming	31
Command Programming	31
EasySet Programming	31
Programming Barcode/ Programming Command/Function.....	32
Use of Programming Barcodes	33
Default Settings	34
Factory Defaults	34
Custom Defaults.....	34
Query Product Information	35
Query Product Name.....	35
Query Firmware Version.....	35
Query Hardware Version	36
Query Product Serial Number	36
Query Manufacturing Date	37
Query OEM Serial Number	37
Query Data Formatter Version	37
Chapter 8 Communication Interface	38

Introduction	38
Adaptive Wired Communication	39
RS-232 Interface	40
Baud Rate	41
Parity Check	42
Data Bit	43
Stop Bit	43
USB HID Keyboard	44
USB Country Keyboard Types	45
Beep on Unknown Character	49
Emulate ALT+Keypad	50
Code Page	51
Unicode Encoding	53
Emulate Keypad with Leading Zero	53
Function Key Mapping	54
ASCII Function Key Mapping Table	55
ASCII Function Key Mapping Table (Continued)	56
Inter-Keystroke Delay	57
Caps Lock	58
Convert Case	59
Emulate Numeric Keypad	60
Fast Mode	62
Polling Rate	63
USB CDC	65
HID POS (POS HID Barcode Scanner)	66
Introduction	66
Access the Engine with Your Program	66
Acquire Scanned Data	67
Send Command to the Engine	67
IBM SurePOS (Tabletop)	68
IBM SurePOS (Handheld)	68
VID/PID	68
Chapter 9 System Settings	69
Scan Mode	69
Decode Session Timeout	70
Image Stabilization Timeout (Sense Mode)	72
Image Change Trigger Sensitivity	73
Scanning Interval (Continuous Mode)	74
Reread Timeout	74
Good Read Delay	76

Scanning Preference	77
Image Flipping	78
Bad Read Message	79
Set Bad Read Message	80
Trigger Commands	80
Modify Start Scanning Command.....	81
Modify Stop Scanning Command.....	81
Illumination	83
External Illumination	84
Aiming.....	85
Good Read LED	86
Good Read LED Duration	86
Power On Beep	88
Good Read Beep.....	88
Good Read Beep Duration	89
Good Read Beep Frequency.....	90
Good Read Beep Volume.....	91
Auto Sleep.....	92

Chapter 10 Symbologies 93

Introduction.....	93
Global Settings	93
Enable/Disable All Symbologies.....	93
Enable/Disable 1D Symbologies	93
Enable/Disable 2D Symbologies	94
Surround GS1 Application Identifiers (AI's) with Parentheses	94
Code 128	96
Restore Factory Defaults.....	96
Enable/Disable Code 128.....	96
Set Length Range for Code 128.....	97
Transmit Check Character.....	98
FNC1 Output	98
EAN-8	99
Restore Factory Defaults.....	99
Enable/Disable EAN-8.....	99
Transmit Check Character.....	99
2-Digit Add-On Code	100
5-Digit Add-On Code	101
Add-On Code Required	102
Convert EAN-8 to EAN-13.....	102
EAN-13.....	103

Restore Factory Defaults.....	103
Enable/Disable EAN-13.....	103
Transmit Check Character.....	104
2-Digit Add-On Code	104
5-Digit Add-On Code	105
Add-On Code Required.....	105
EAN-13 Beginning with 290 Add-On Code Required.....	106
EAN-13 Beginning with 378/379 Add-On Code Required.....	106
EAN-13 Beginning with 414/419 Add-On Code Required.....	107
EAN-13 Beginning with 434/439 Add-On Code Required.....	107
EAN-13 Beginning with 977 Add-On Code Required.....	108
EAN-13 Beginning with 978 Add-On Code Required.....	108
EAN-13 Beginning with 979 Add-On Code Required.....	109
UPC-E	110
Restore Factory Defaults.....	110
Enable/Disable UPC-E.....	110
Transmit Check Character.....	111
2-Digit Add-On Code	111
5-Digit Add-On Code	112
Add-On Code Required.....	112
Transmit Preamble Character	113
Convert UPC-E to UPC-A	113
UPC-A	114
Restore Factory Defaults.....	114
Enable/Disable UPC-A.....	114
Transmit Check Character.....	114
2-Digit Add-On Code	115
5-Digit Add-On Code	116
Add-On Code Required.....	117
Transmit Preamble Character	117
Interleaved 2 of 5.....	118
Restore Factory Defaults.....	118
Enable/Disable Interleaved 2 of 5	118
Set Length Range for Interleaved 2 of 5	119
Check Character Verification.....	120
Febraban	121
Disable/Enable Febraban.....	121
Transmit Delay per Character	121
Transmit Delay per 12 Characters.....	124
ITF-14.....	126

Restore Factory Defaults.....	126
Enable/Disable ITF-14.....	126
ITF-6	127
Restore Factory Defaults.....	127
Enable/Disable ITF-6.....	127
Matrix 2 of 5	128
Restore Factory Defaults.....	128
Enable/Disable Matrix 2 of 5	128
Set Length Range for Matrix 2 of 5	129
Check Character Verification.....	130
Code 39	131
Restore Factory Defaults.....	131
Enable/Disable Code 39.....	131
Set Length Range for Code 39.....	132
Check Character Verification.....	133
Transmit Start/Stop Character.....	134
Enable/Disable Code 39 Full ASCII.....	134
Enable/Disable Code 32 (Italian Pharma Code)	135
Code 32 Prefix.....	135
Transmit Code 32 Start/Stop Character	136
Transmit Code 32 Check Character	136
Codabar	137
Restore Factory Defaults.....	137
Enable/Disable Codabar	137
Set Length Range for Codabar	138
Check Character Verification.....	139
Start/Stop Character.....	140
Code 93	141
Restore Factory Defaults.....	141
Enable/Disable Code 93.....	141
Set Length Range for Code 93.....	142
Check Character Verification.....	143
GS1-128 (UCC/EAN-128)	144
Restore Factory Defaults.....	144
Enable/Disable GS1-128.....	144
Set Length Range for GS1-128.....	145
Transmit Check Character.....	146
FNC1 Output	146
GS1 Databar (RSS).....	147
Restore Factory Defaults.....	147

Enable/Disable GS1 Databar	147
Transmit Application Identifier "01"	147
Code 11	148
Restore Factory Defaults.....	148
Enable/Disable Code 11	148
Set Length Range for Code 11	149
Check Character Verification.....	150
Transmit Check Character.....	151
ISBN	152
Restore Factory Defaults.....	152
Enable/Disable ISBN.....	152
Set ISBN Format	153
ISSN	153
Restore Factory Defaults.....	153
Enable/Disable ISSN.....	154
Industrial 25	155
Restore Factory Defaults.....	155
Enable/Disable Industrial 25.....	155
Set Length Range for Industrial 25.....	156
Check Character Verification.....	157
Standard 25	158
Restore Factory Defaults.....	158
Enable/Disable Standard 25.....	158
Set Length Range for Standard 25.....	159
Check Character Verification.....	160
Plessey	161
Restore Factory Defaults.....	161
Enable/Disable Plessey.....	161
Set Length Range for Plessey.....	162
Check Character Verification.....	163
MSI-Plessey	164
Restore Factory Defaults.....	164
Enable/Disable MSI-Plessey	164
Set Length Range for MSI-Plessey	165
Check Character Verification.....	166
Transmit Check Character.....	167
AIM 128	168
Restore Factory Defaults.....	168
Enable/Disable AIM 128.....	168
Set Length Range for AIM 128.....	169

Transmit Check Character.....	170
FNC1 Output	170
PDF417	171
Restore Factory Defaults.....	171
Enable/Disable PDF417	171
Set Length Range for PDF417	172
PDF417 Twin Code	173
Character Encoding.....	174
PDF417 ECI Output	174
QR Code.....	175
Restore Factory Defaults.....	175
Enable/Disable QR Code	175
Set Length Range for QR Code	176
QR Twin Code	177
Character Encoding.....	178
QR ECI Output	179
URL QR	179
Micro QR Code.....	179
Restore Factory Defaults.....	179
Enable/Disable Micro QR	180
Set Length Range for Micro QR	181
Aztec.....	182
Restore Factory Defaults.....	182
Enable/Disable Aztec Code.....	182
Set Length Range for Aztec Code.....	183
Character Encoding.....	184
Aztec ECI Output.....	184
Data Matrix	185
Restore Factory Defaults.....	185
Enable/Disable Data Matrix.....	185
Set Length Range for Data Matrix.....	186
Data Matrix Twin Code	187
Rectangular Barcode.....	188
Character Encoding.....	188
Data Matrix ECI Output	189
Chapter 11 Data Formatter	190
Introduction.....	190
Add a Data Format	190
Programming with Barcodes	190
Programming with Serial Commands.....	193

Enable/Disable Data Formatter	194
Non-Match Error Beep.....	195
Data Format Selection.....	196
Change Data Format for a Single Scan.....	197
Clear Data Format.....	198
Query Data Formats	198
Formatter Command Type 6.....	199
Send Commands.....	199
Move Commands	202
Search Commands.....	204
Miscellaneous Commands	207
Chapter 12 Prefix & Suffix.....	213
Introduction.....	213
Global Settings	214
Enable/Disable All Prefixes/Suffixes.....	214
Prefix Sequence	214
Custom Prefix.....	215
Enable/Disable Custom Prefix.....	215
Set Custom Prefix	215
AIM ID Prefix	216
Code ID Prefix	217
Restore All Default Code IDs.....	217
Modify Code ID.....	218
Custom Suffix	223
Enable/Disable Custom Suffix.....	223
Set Custom Suffix.....	223
Data Packing	224
Introduction.....	224
Data Packing Options.....	224
Terminating Character Suffix	226
Enable/Disable Terminating Character Suffix.....	226
Set Terminating Character Suffix.....	226
Chapter 13 Programming Commands	228
Use of Programming Command.....	228
Command Syntax.....	228
Query Commands	228
Responses	229
Examples.....	229
Read Barcode On/Off	230

Make a Beeping Sound	230
Turn On Good Read LED	230
Turn On Illumination LED	231
Chapter 14 Batch Programming	232
Introduction	232
Create a Batch Command	233
Create a Batch Barcode	233
Use Batch Barcode.....	234
Appendix.....	235
Digit Barcodes	235
Save/Cancel Barcodes	238
Factory Defaults Table (ST.G01.1).....	239
AIM ID Table(V2022.6)	246
Code ID Table(V1.00.0)	247
Symbology ID Number(V1.00.0).....	248
ASCII Table.....	249
Keyboard Key References.....	253

Preface

Introduction

This manual provides installation, optics, electrical specifications as well as detailed instructions for setting up and using the NLS-N1-W engine (hereinafter referred to as “the N1-W” or “the engine”).

This guide provides programming instructions for the N1-W. Users can configure the N1-W by scanning the programming barcodes included in this manual.

The N1-W has been properly configured for most applications and can be put into use without further configuration. Users may check Appendix: Factory Defaults Table for reference.

Chapter Description

- ✧ *Chapter 1, Getting Started* : Gives a general description of the N1-W.
- ✧ *Chapter 2, Installation* : Describes how to install the engine, including installation information, connector, cable, ESD, and environmental considerations.
- ✧ *Chapter 3, Optics* : Provides parameters for optics and illumination.
- ✧ *Chapter 4 Electrical Specifications* : Includes the electrical characteristics for the engine and timing sequences.
- ✧ *Chapter 5 External Reference Circuit* : Provide external driver circuit diagrams.
- ✧ *Chapter 6, Auxiliary Tools* : Introduces useful tools you can use to set up the N1-W.
- ✧ *Chapter 7 Configuration* : Introduces the use of programming barcodes and product information query.
- ✧ *Chapter 8 Communication Interface* : Describes how to configure RS-232 and USB communication parameters.
- ✧ *Chapter 9, System Settings* : Describes how to configure general parameters of the N1-W.
- ✧ *Chapter 10, Symbologies* : Lists all compatible symbologies and describes how to configure the relevant parameters.
- ✧ *Chapter 11, Data Formatter* : Explains how to customize scanned data with the advanced data formatter.
- ✧ *Chapter 12, Prefix & Suffix* : Describes how to use prefix and suffix to customize scanned data.
- ✧ *Chapter 13 Programming Commands* : Introduces how to configure the N1-W by serial commands sent from the





host.

- ✧ *Chapter 14, Batch Programming* : Explains how to integrate a complex programming task into a single barcode.
- ✧ *Appendix* : Provides factory defaults table and a bunch of frequently used programming barcodes.

Explanation of Symbols

- This symbol indicates lists of required steps.
- ※ This symbol indicates notes of some parameters.

Explanation of Icons

	This icon indicates auxiliary tools that help user to refer to the 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 engine.
	This icon indicates practical examples that can help you to acquaint yourself with operations.

Chapter 1 Getting Started

Introduction

The N1-W OEM scan engines are armed with CMOS image capturer and the Newland patented **UIMG**, a computerized image recognition system-on-chip, featuring fast scanning and accurate decoding on barcodes on virtually any medium - paper, magnetic card, mobile phones and LCD displays.



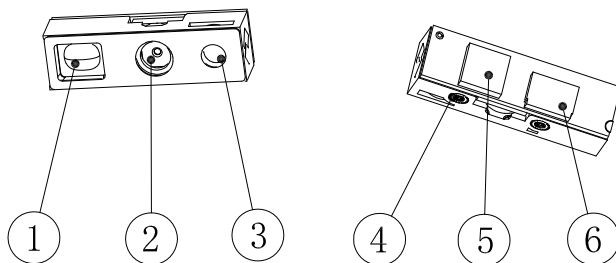
Note: This guide provides general instructions for the installation. Newland EMEA recommends an opto-mechanical engineer should conduct an opto-mechanical analysis before design.

Symbologies

The N1-W can easily read printed barcodes and on-screen barcodes, including:

1D	Code 128, EAN-8, EAN-13, UPC-E, UPC-A, Interleaved 2/5, ITF-14, ITF-6, Matrix 2/5, Code 39, Codabar, Code 93, UCC/EAN-128, GS1 Databar, Code 11, ISBN, ISSN, Industrial 2/5, Standard 2/5, Plessey, MSI-Plessey, AIM 128
2D	PDF 417, QR Code, Micro QR, Data Matrix, Aztec

N1-W Scan Engine



1. Illumination System
2. Lens
3. Aiming System
4. Mounting Hole
5. Label
6. External Interface

Figure 1-1

Chapter 2 Installation

Introduction

This chapter explains how to install the N1-W, including general requirements, housing design, and physical and optical information.



Caution: Do not touch the imaging lens when installing the engine. Be careful not to leave fingerprints on the lens.



Caution: Do not touch the illumination LED during handling. Improper handling may damage the LED.

Dimensions (unit: mm)

23.8(W)×7.5(D)×7.0(H) (max.)

Mounting

The illustrations below show the mechanical mounting dimensions (unit: mm) for the N1-W.

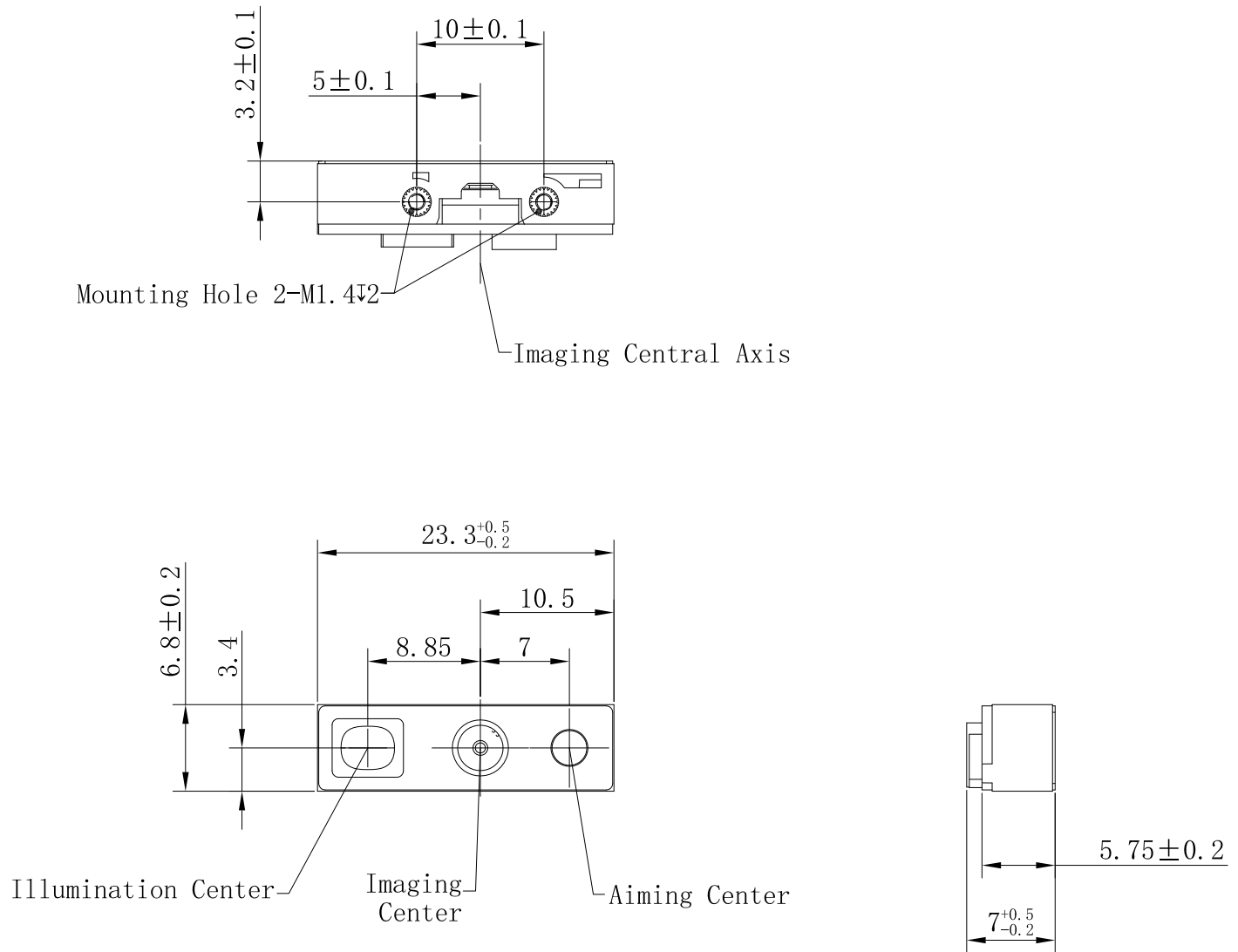


Figure 2-1

13-PIN FPC Connector

The physical interface of the N1-W consists of a 13-pin FPC connector:

- 13-pin FPC connector can be used as TTL-232 interface or USB interface.

The figure below illustrates the position of 13-pin FPC connector on the N1-W decoder board, as well as the pin 1 location.

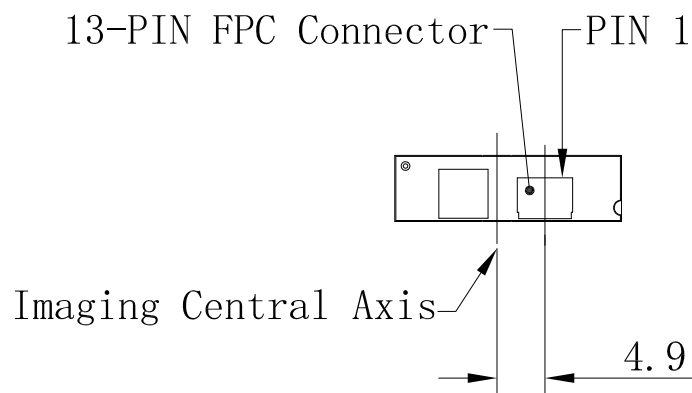


Figure 2-2

Connector Specifications

The N1-W is equipped with a 13-pin FPC connector. The connector is supplied by Hirose Electric Co.,Ltd. Model No.: FH35C-13S-0.3SHW(50). For more information about the connector, please visit the website at <https://www.hirose.com/product/p/CL0580-2925-2-50>.

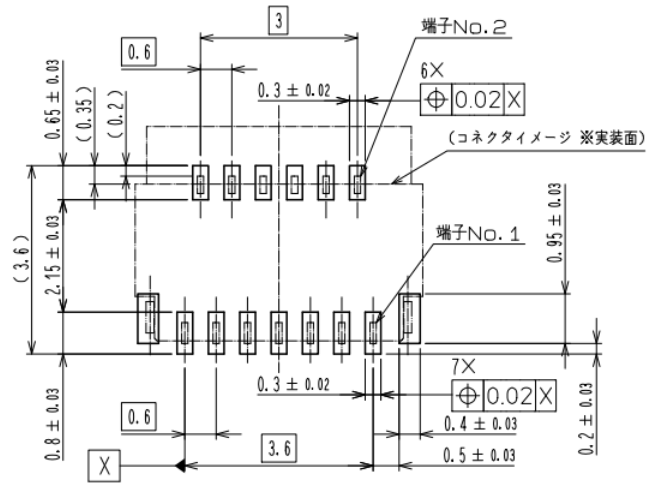


Figure 2-3



Warnings:

1. In order to improve the connecting stability, using the downward connection priority.
2. Do not close the connector before inserting the FPC cable.
3. Avoid opening/closing the connector multiple times.

ESD

ESD protection has been taken into account when designing the N1-W. The engine is shipped in ESD safe packaging. Always exercise care when handling the engine outside its package. Be sure grounding wrist straps and properly grounded work areas are used.

Dust and Dirt

The N1-W must be sufficiently enclosed to prevent dust particles from gathering on the lens and circuit board. Dust and other external contaminants will eventually degrade the engine's performance.

Ambient Environment

The following environmental requirements should be met to ensure good performance of the N1-W.

Table 2-1

Operating Temperature	-20°C to 55°C
Storage Temperature	-40°C to 70°C
Humidity	5%~95% (non-condensing)

Thermal Considerations

Electronic components in the N1-W will generate heat during the course of their operation. Operating the N1-W in continuous mode for an extended period may cause temperatures to rise on CPU, LEDs, DC-DC, etc. Overheating can degrade image quality and affect scanning performance. Given that, the following precautions should be taken into consideration when integrating the N1-W.

- ✧ Reserve sufficient space for good air circulation in the design.
- ✧ Avoid wrapping the N1-W with thermal insulation materials such as rubber.

Maintenance

The scan window should be kept clean.

Avoid scratching the scan window.

Use a soft lint free cloth to clean the window, e.g. a microfiber cloth.

Do not spray any liquid on the scan window.

Do not use any detergent to clean other parts of the device except for water.

Note: The warranty DOES NOT cover the damage caused by inappropriate care and maintenance.

Chapter 3 Optics

Introduction

The N1-W contains:

- a CMOS image sensor and its lens
- a white LED based illumination system and a lens
- a red LED based aiming system and a lens

Sensor

Pixel: 640×480 CMOS

Frame rate: 60fps

Illumination

The N1-W has a white LED for supplementary lighting, making it possible to scan barcodes even in complete darkness. The illumination can be programmed On or Off. Customers can add the external illumination system if needed. The spectral range should be within the visible light.

Aimer

The N1-W contains a red LED aimer to assist the user to position the target barcode within the engine's field of view. The aimer can be enabled or disabled. For most applications, it is helpful to leave the aimer enabled. Additionally, the aimer can be used as good read indicator. In this mode of operation, the aimer will be off during the decode session and come on to indicate a good read.

Note: when the engine comes closer to the barcodes, aiming center and imaging center may deviate.

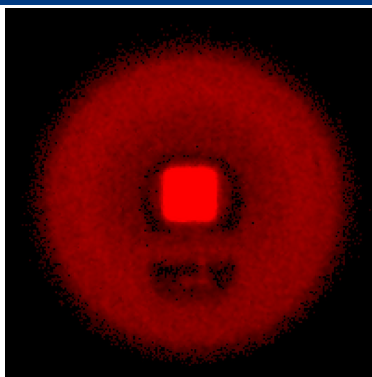


Figure 3-1

Window Placement

The window should be positioned properly to let the illumination and aiming beams pass through as much as possible and no reflections back into the engine (reflections can degrade the reading performance of the engine).

There are two window placement options.

Parallel window – Primary option for imager engines. The following window distance requirements should be satisfied: The maximum distance is measured from the front of the engine housing to the furthest surface of the window. In order to reach better reading performance, the distance from the front of the engine housing to the furthest surface of the window should not exceed $a+b$ ($a+b \leq 3\text{mm}$), as shown in **Figure 3-2**.

Tilted window. For the tilted window distance requirements, please see **Table 3-1**.

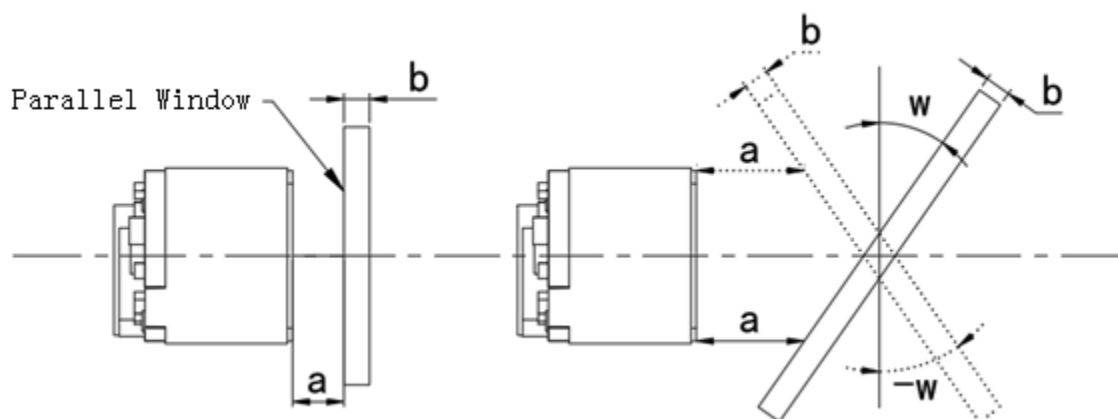


Figure 3-2

Table 3-1

Distance from the front of the engine housing (a)		Thickness of window (b)		
		b=1mm	b=2mm	b=3mm
Angle of tilted window (w)	W=0°	a≤2mm	a≤1mm	a=0mm
	W=±10°	a≤1mm	a≤1mm	a=0mm
	W=±20°	a≤1mm	a≤1mm	a=0mm
	W=±30°	a≤1mm	a≤1mm	a=0mm
	W=±40°	a≤20mm	a≤20mm	a≤20mm

Window Material and Color

Window material must be clear. PMMA and optical glass are recommended. Window material selected for the engine should meet or exceed the specifications specified in **Table 3-2**. When using a clear plastic window, it is recommended to apply anti-reflection (AR) coating on it.

Table 3-2

Specification	Description
Spectral Transmittance	≥92%
Light Wavelength	400-780nm

Coatings and Scratch Resistance

Scratches and abrasions of the scan window can significantly reduce the performance of the N1-W. It is suggested that abrasion resistant materials and/or scratch resistant coatings are employed.

Two commonly used coatings:

Anti-reflection (AR) coatings: AR Coatings can be applied to one or both window surfaces to reduce reflected light from the scan engine's illumination LED. Coating both surfaces reduces the required window angle to avoid reflections.

Scratch resistance coatings: Scratch resistance coatings are recommended to have a hardness of 5h or greater. It is suggested that scan windows are recessed to reduce the risk of physical damage to the scan window

Window Size

The window must not block the field of view and should be sized to accommodate the illumination and FOV envelopes shown below.

Front View:

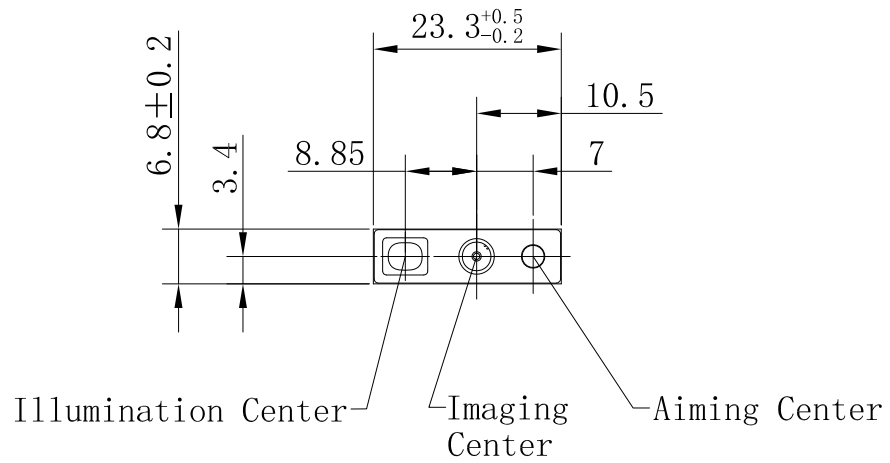


Figure 3-3

Horizontal:

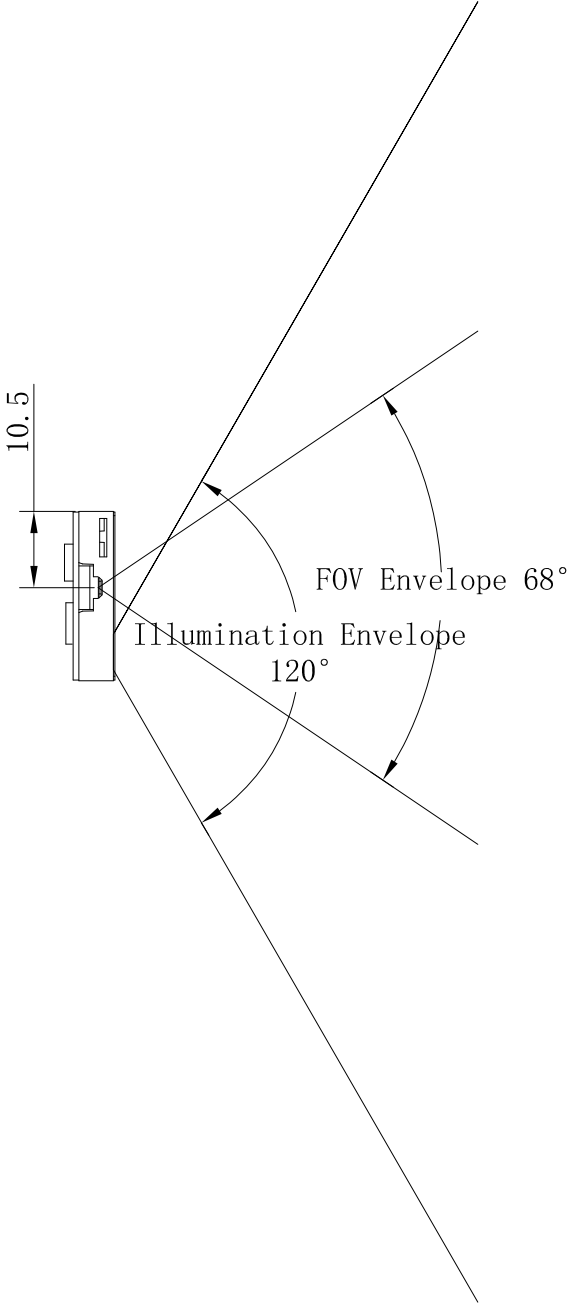


Figure 3-4

Vertical:

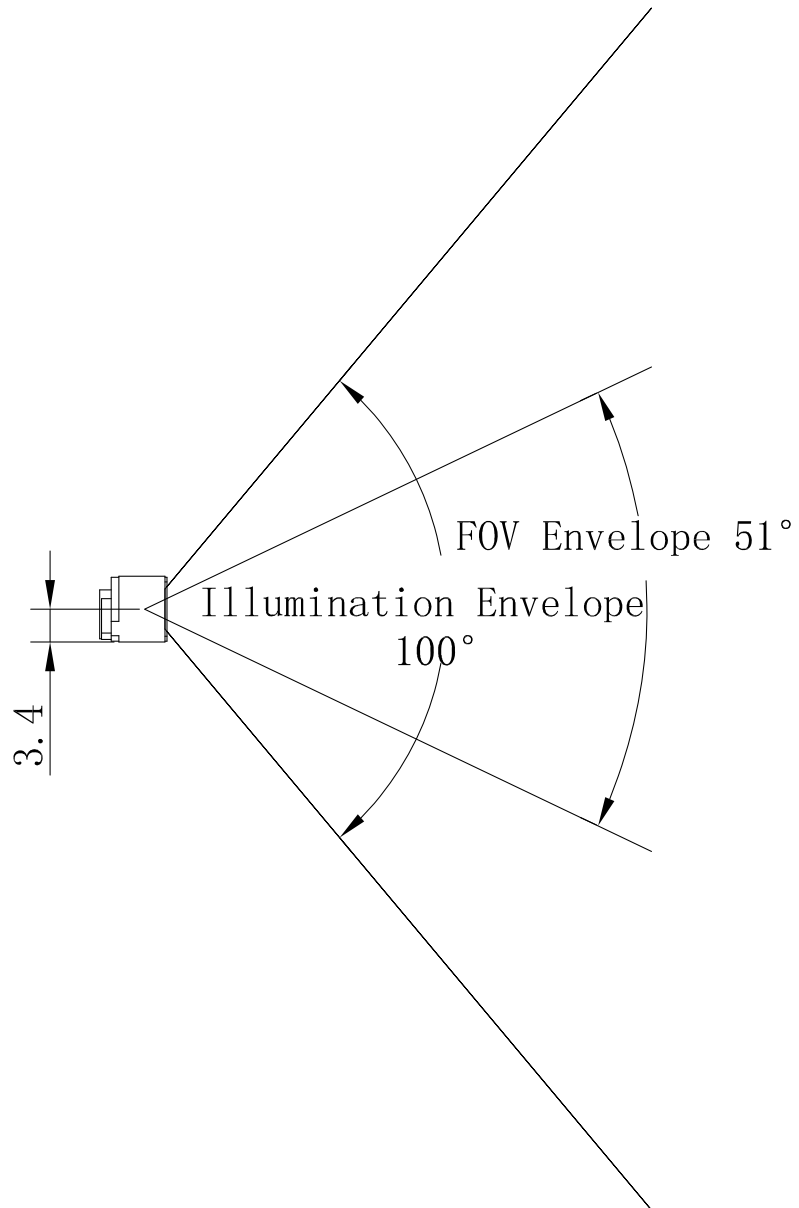


Figure 3-5

Ambient Light

The N1-W shows better performance with ambient light. However, high-frequency pulsed light can result in performance degradation.

Eye Safety

The N1-W uses LED's to project general illumination and the aiming pattern. The N1-W complies with IEC 62471:2006 pertaining to LED safety, demonstrating the N1-W is safe under normal usage conditions. The user should, however, avoid looking into the LEDs directly, as is recommended with any bright source of light.

Depth of Field

The tables below list the depth of view tested in the 0lx and 300lx natural light.

Table 3-3

Ambient light: 0lux natural light

Symbology	Near	Far
PDF417 (6.7mil)	25mm	45mm
Code39 (5mil)	25mm	50mm
Data Matrix (10mil)	25mm	45mm
QR Code (15mil)	20mm	60mm

Table 3-4

Ambient light: 300lux natural light

Symbology	Near	Far
PDF417 (6.7mil)	25mm	45mm
Code39 (5mil)	25mm	50mm
Data Matrix (10mil)	25mm	45mm
QR Code (15mil)	20mm	60mm

Chapter 4 Electrical Specifications

Power Supply

Do not apply power to the scan engine host connector until the N1-W is securely connected. Ensure that power is disconnected before disconnecting the N1-W from the host connect/FPC cable.

An unstable power supply or poor regulation/rapid voltage changes will lead to instability of the scan engine and poor performance.



1. During the integration phase, Vcc should be decoupled using suitable decoupling capacitors. 22uF and 100nF capacitors are recommended.
2. Ensure that Vcc drops below 0.5v before re applying power to ensure complete power down and power up processes within the engine.

Ripple Noise

To ensure the image quality, a power supply with low ripple noise is needed.

Acceptable ripple range (peak-to-peak) $\leq 100\text{mV}$

Interface Pinouts

The following table lists the pin functions of the 13-PIN FPC Connector.

Table 4-1

PIN#	Signal	I/O	Function	Note
1	GND	-	Power-supply ground	
2	EXT_TRIG#	I	Trigger signal input	100k Pull-up See Note 1
3	EXT_RST#	I	Reset signal input	See Note 1
4	EXT_DSF	O	Good Read LED output	100k Pull-down See Note 3
5	EXT_BUZ	O	Beeper output	100k Pull-down See Note 4
6	EXT_LIGHT	O	External illumination control signal	See Note 5
7	USB_D+	-	USB_D+ signal	
8	USB_D-	-	USB_D- signal	
9	TTL232_TX	O	TTL level 232 transmit data	
10	TTL232_RX	I	TTL level 232 receive data	
11	GND	-	Power-supply ground	
12	VCC	-	3.3V power input	
13	VCC	-	3.3V power input	

※ I = Input; O = Output; od = Open Drain;

※ 1 The EXT_TRIG# signal is not directly from the MCU pin, so high and low levels are different from other input and output interfaces. When designing the external control module, please refer to I/O Voltage in Table 4-4.

This external trigger signal can be either level trigger or pulse trigger.

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

Pulse trigger: When the trigger is pulled and released (pulse width : 50ms), scanning is activated until a barcode is decoded or the decode session timeout expires. The duration can be programmed. To learn how to program the parameter, please see the N1-W user guide.

For the external trigger circuit, please see the “Trigger Circuit” section in Chapter 5.

※ 2 Keep level low above 10ms to reset the engine. If this pin is not used, leave it unconnected.

-
- ※ 3 This output signal can be used by an external LED to indicate good read status.

The EXT_DSF pin produces a high output (default duration: 20ms, user-programmable) when a good read occurs. The Good Read LED can be programmed On or Off. To learn how to program these parameters, please see the N1-W user guide.

For the external LED circuit design, please see the “Good Read LED Circuit” section in Chapter 5. If the EXT_DSF pin is not used, leave it unconnected.

- ※ 4 This output signal can be used by an external beeper circuit to generate audible feedback to the user to indicate power-on and good read statuses.

Power On beep: The EXT_BUZ pin produces a PWM output (duration: 400ms; frequency: 2090Hz, 2630Hz, 3130Hz and 4260Hz.) 215ms after power-on. The beep can be programmed On or Off. To learn how to program the parameter, please see the N1-W user guide.

Good Read beep: The EXT_BUZ pin produces a PWM output (default duration: 80ms; default frequency: 2710Hz) when a good read occurs. The beep can be programmed On or Off. To learn how to program these parameters, please see the N1-W user guide.

For the external beeper circuit design, please see the “Beeper Circuit” section in Chapter 5. If the EXT_BUZ pin is not used, leave it unconnected.

- ※ 5 Pin 6 is the external illumination control signal. For the external illumination control circuit design, please see the “External Illumination Control Circuit” section in Chapter 5. If the external illumination control signal pin is not used, leave it unconnected.

DC Characteristics

Operating Voltage

Table 4-2

T=25°C

Parameter	Description	Minimum	Typical	Maximum	Unit
VCC	Input Voltage $\pm 5\%$ (3.3V)	3.14	3.3	3.47	V

Operating Current

Table 4-3

T=25°C

Mode		State	Typical	Maximum	Unit
Working Current	RMS ¹	VCC=3.3V	68	150	mA
	PEAK ²		-	440	mA
Idle Current	RMS ¹		11.8	-	mA
Sleep Current	RMS ¹		-	0.3	mA

1. RMS indicates the RMS value of the current under the stable working state.
2. PEAK indicates the peak current the device reaches.

I/O Voltage

Table 4-4

VCC=3.3 V, GND =0 V, T=25°C

Parameter	Description	Condition	Minimum	Maximum	Unit
VIL	input low level	-	-	0.8	V
VIH	input high level	-	2	-	V
VOL	output low level	I _{ol} = 4mA~16mA	-	0.4	V
VOH	output high level	I _{oh} = 4mA~16mA	2.4	-	V

Timing Sequence

Power Up Timing Sequence

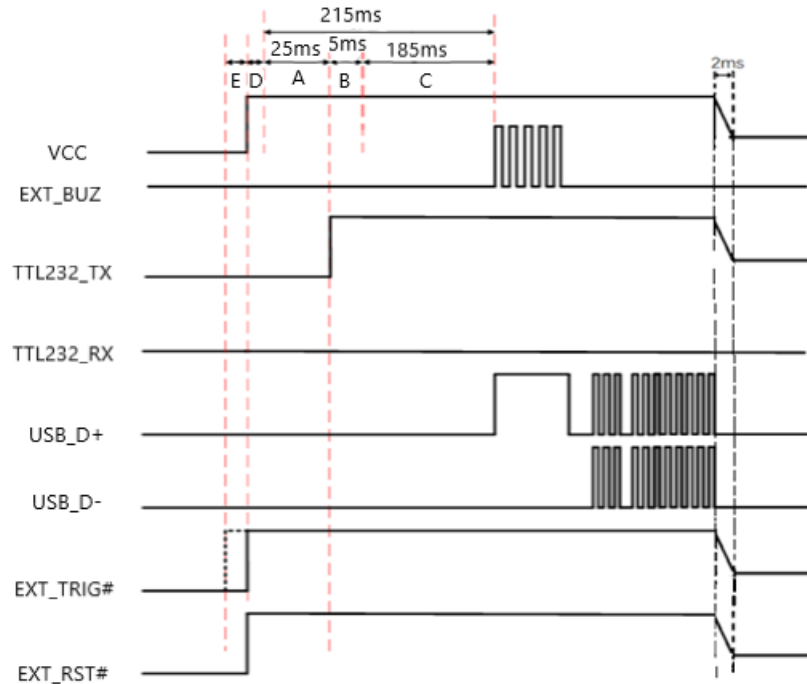


Figure 4-1

1. In the diagram above, it takes **A+B+C** (about 215ms) for the engine to power up: **A** is bootloader execution time (25ms), **B** is kernel boot time (5ms) and **C** is decoding chip initialization time(185ms).
2. D is reset time (300us). If the EXT_RST# signal is not operated when powered on, the startup time should be calculated after VCC_3V3 reaches 3.3V.
3. Ensure that all communication interface data has been transmitted before powering off.
4. The N1-W has a 100k pull-up on the EXT_TRIG# signal. During the period between power-on and bootloader execution, the user should not lower the EXT_TRIG# signal. If EXT_TRIG# signal is set high before power-on, it should meet the duration of E (E=0~1ms) as shown in Figure 4-1. Levels of other signals are kept low during power-on to prevent abnormal function.
5. To ensure that the engine works at a stable level, the minimum interval between removing and resupplying the power must exceed 2s.

Chapter 5 External Reference Circuit

External Circuit Design

Good Read LED Circuit

The circuit below is used to drive an external LED for indicating good read. The EXT_DSF signal is from PIN 4 of the 13-pin FPC connector.

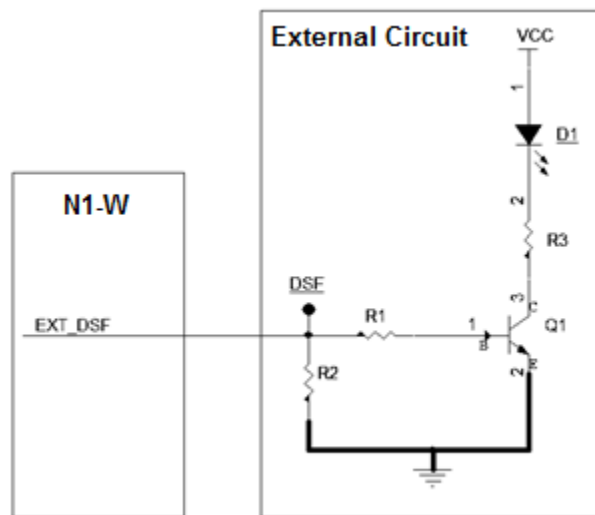


Figure 5-1

Beeper Circuit

The circuit below is used to drive an external beeper. The EXT_BUZ signal is from PIN 5 of the 13-pin FPC connector.

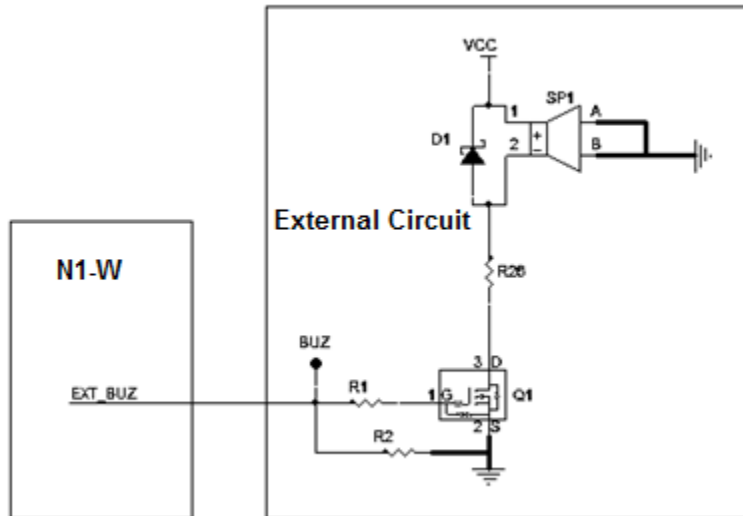


Figure 5-2

Trigger Circuit

The circuit below is used to provide the engine with a signal to trigger a scan and decode session. The EXT_TRIG# signal is from PIN 2 of the 13-pin FPC connector.

The host can adjust the external circuit and its functions based on actual applications. R1 is recommended as 10K-100K and R2 as 33Ω. C1 is used to eliminate the vibration of mechanical keys. Generally, 1nF-10nF ceramic capacitors are recommended. When ESD protection is required, ESD protector can be added to the external circuit like ED1 shown as below.

When using the external IO port as a trigger output, note that high and low levels must meet the requirements provided in Table 4-4. It is recommended to use the default floating or default pull-up IO port as the trigger pin. If you can only use the default pull-down IO port as the trigger pin, refer to the power-on timing sequence in Figure 4-1. When not triggered, ensure that the pin meets the high level requirements in Table 4-4.

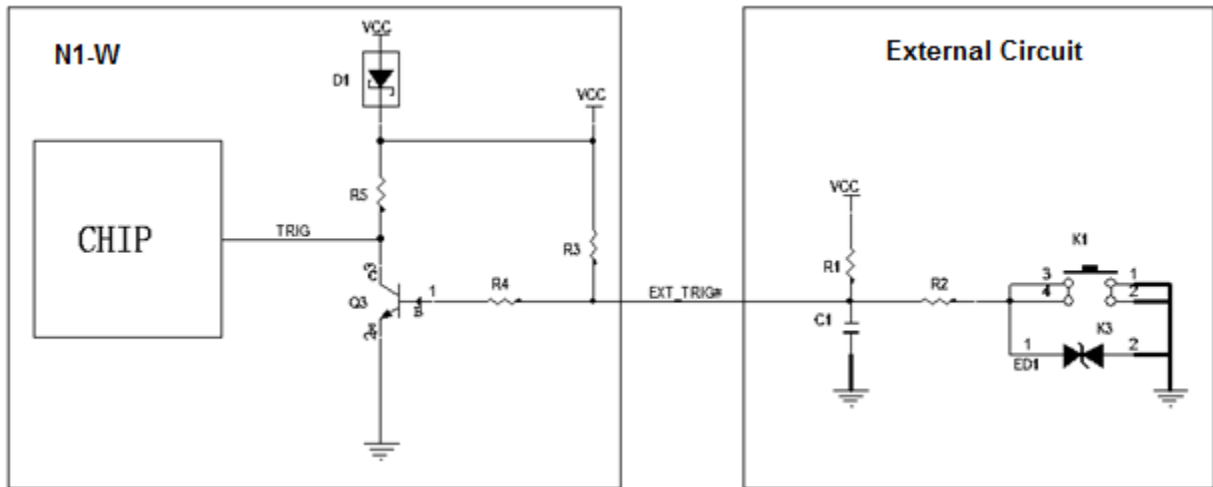


Figure 5-3

External Light Control Circuit

Pin 6 (EXT_LIGHT) on the 13-pin FPC connector is the external illumination control signal. I/O is in the Floating state before initialization. When the external illumination is not enabled, I/O is configured as the input pin. When enabled, I/O is the output pin in working mode and input pin in the sleep mode. For external illumination, please refer to the user guide for software configuration. The duration from the EXT_LIGHT signal output to the external illumination should not exceed 100us.

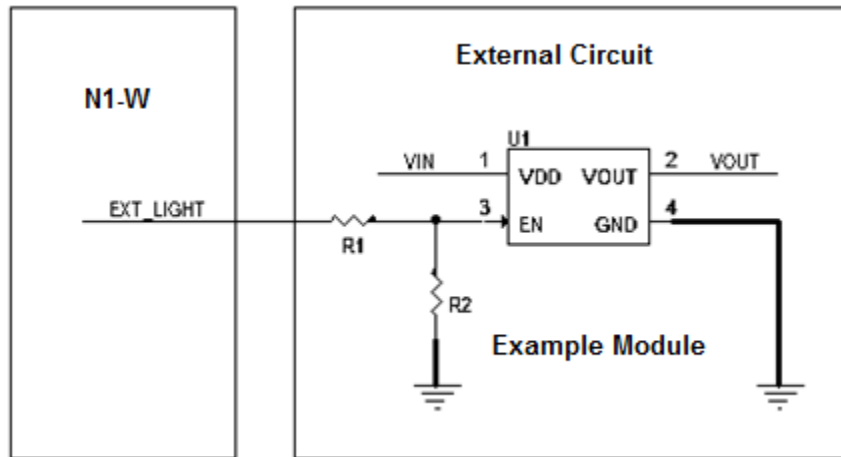


Figure 5-4

Reset Circuit

The circuit below is used to provide a signal to reset the engine. The EXT_RST# signal is from PIN 3 of the 13-pin FPC connector.

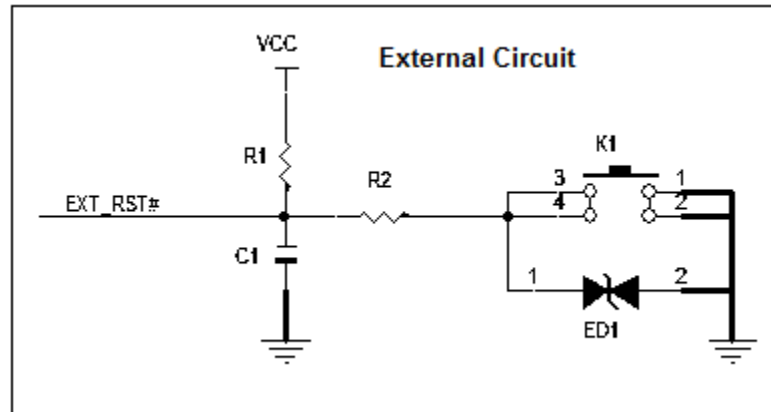


Figure 5-5

Chapter 6 Auxiliary Tools

EVK

The EVK is provided to help users to test and evaluate the N1-W, which contains beeper & beeper driver circuit, LED & LED driver circuit, and trigger, TTL-232 to RS-232 converter, RS-232 & USB interfaces, reserved signal debugging interface, etc. The N1-W can be connected to the EVK via a 13-pin FPC cable to 12-PIN FPC cable.

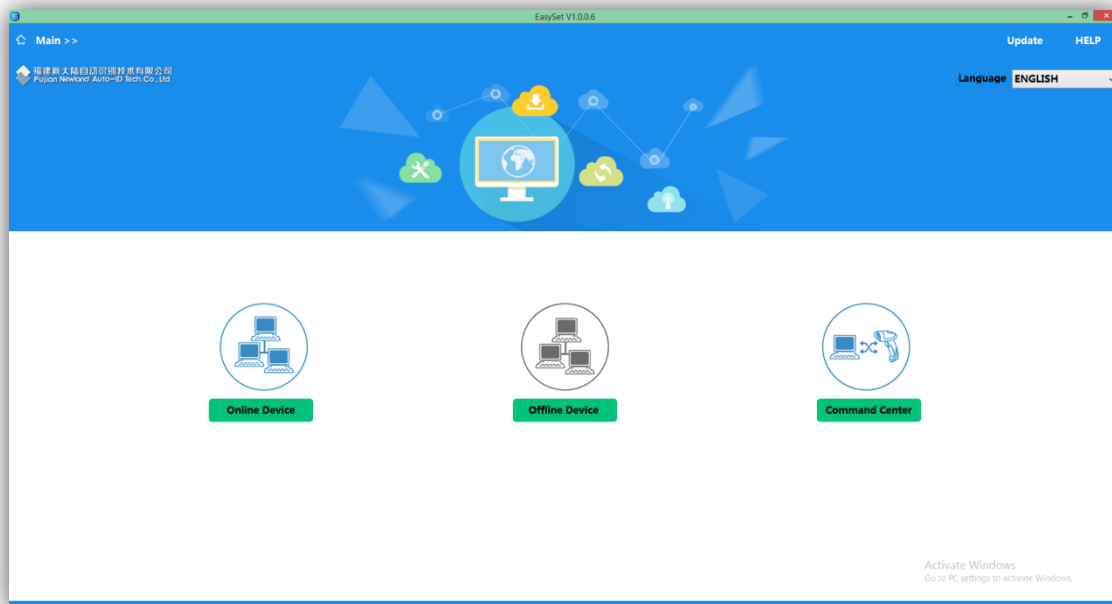
EasySet

EasySet, developed by Fujian Newland Auto-ID Tech. Co., Ltd., is a configuration tool for Newland's 1D/2D handheld barcode scanner, fixed mount barcode scanners and OEM scan engines. Its main features include:

- ✧ View device & configuration information of online device
- ✧ Configure device
- ✧ Update firmware of online device
- ✧ Load/modify existing XML configuration file; save current settings to an XML file
- ✧ Create/print/save programming barcodes to a PDF or Word file
- ✧ View/edit/save image stored on online device in the original image/BMP/JPG/TIFF format
- ✧ Send serial commands to online device and receive device response
- ✧ Supported languages: Chinese and English

EasySet supports 32-bit/64-bit Microsoft WinXP/Win7/Win 8/Win 8.1/Win 10 operating systems.

EasySet can communicate with device via one of the following interface: TTL-232, USB COM Port Emulation (UFCOM driver required), USB CDC (UFCOM driver required), USB DataPipe (UFCOM driver required), USB HID-POS.



UFCOM

UFCOM, developed by Fujian Newland Auto-ID Tech. Co., Ltd., is a virtual serial driver. It is used in conjunction with a USB engine or a scan engine configured as virtual serial port to provide two-way communication between the device and the host. UFCOM can run on all versions of Windows XP ~ Windows 10 x86 & x64, including the contemporary versions of Windows Server. Users can download the driver from the website at: <http://down.nlscan.com:82/Release/UFCOM/>.



@SETUPE1
Enter Setup

Chapter 7 Configuration

Introduction

There are three ways to configure the N1-W: Barcode programming, command programming and Easysset programming.

Barcode Programming

The N1-W 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 N1-W can also be configured by serial commands sent from the host device.

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

EasySet Programming

Besides the two methods mentioned above, you can conveniently perform engine configuration through EasySet too. EasySet 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 engines. For more information about this tool, refer to the *EasySet User Guide*.



@SETUPE0
** Exit Setup



@SETUPE1
Enter Setup

Programming Barcode/ Programming Command/Function



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

1. The **No Case Conversion** barcode.
2. The **No Case Conversion** command.
3. The description of feature/option.



@SETUPE0
** Exit Setup



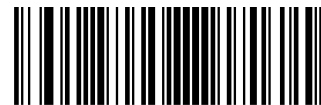
@SETUPE1
Enter Setup

Use of Programming Barcodes

Scanning the **Enter Setup** barcode can enable the engine to enter the setup mode. Then you can scan a number of programming barcodes to configure your engine. To exit the setup mode, scan the **Exit Setup** barcode or a non-programming barcode, or reboot the engine.

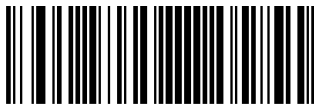


@SETUPE0
Exit Setup



@SETUPE1
Enter Setup

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



@SETUPE0
**** Exit Setup**



@SETUPE1
Enter Setup

Default Settings

Factory Defaults

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

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

- ✧ engine 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 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 engine to the factory defaults will not remove the custom defaults from the engine.



@SETUPE0
**** Exit Setup**



@SETUPE1
Enter Setup

Query Product Information

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



@QRYSYS
Query Product Information

Query Product Name



@QRYPDN
Query Product Name

Query Firmware Version



@QRYFWV
Query Firmware Version



@SETUPE0
**** Exit Setup**



@SETUPE1
Enter Setup

Query Hardware Version



@QRYHWW
Query Hardware Version

Query Product Serial Number



@QRYPSN
Query Product Serial Number



@SETUPE0
**** Exit Setup**



@SETUPE1
Enter Setup

Query Manufacturing Date



@QRYDAT
Query Manufacturing Date

Query OEM Serial Number

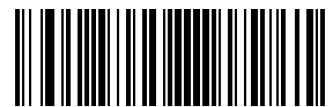


@QRYESN
Query OEM Serial Number

Query Data Formatter Version



@QRYDFM
Query Data Formatter Version



@SETUPE0
**** Exit Setup**



@SETUPE1
Enter Setup

Chapter 8 Communication Interface

Introduction

- ✧ Serial communication interface is usually used when connecting the engine to a host device (like PC, POS). You need to set communication parameters to match the host device.
- ✧ USB HID Keyboard: The engine'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.
- ✧ HID POS (POS HID Barcode Scanner): It is based on the HID interface, with no need for a custom driver. It excels virtual keyboard and traditional TTL-232 interface in transmission speed.
- ✧ IBM SurePOS: It conforms to IBM (now Toshiba Global Commerce Solutions) 4698 USB engine interface specifications.

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



@SETUPE0
** Exit Setup

Adaptive Wired Communication

When this feature is on, the engine can automatically adapt its communication configuration to the way it is connected to the host device: Automatically enable USB/serial communication when connected to the host device via USB/serial port, respectively.

Note: You must restart the engine before this setting will take effect.





@SETUPE1

Enter Setup

RS-232 Interface

Serial communication interface is usually used when connecting the engine to a host device (like PC, POS). However, to ensure smooth communication and accuracy of data, you need to set communication parameters (including baud rate, parity check, data bit and stop bit) to match the host device.



@INTERF0

RS-232



@SETUPE0

**** Exit Setup**

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



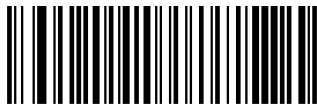
@232BAD6

38400



@232BAD5

19200



@232BAD4

14400



@232BAD3

9600



@232BAD2

4800



@SETUPE1
Enter Setup



@232BAD1
2400



@232BAD0
1200

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



@232PAR1
Even Parity



@232PAR2
Odd Parity



@SETUPE0
**** Exit Setup**

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



@SETUPE1

Enter Setup

USB HID Keyboard

When the engine is connected to the USB port on a host device, you can enable the USB HID Keyboard feature by scanning the barcode below. Then engine'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.



@SETUPE0

**** Exit Setup**

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)



@SETUPE1
Enter Setup



@KBWCTY8
Germany/ Austria



@KBWCTY10
Hungary



@KBWCTY12
Italy



@KBWCTY14
Netherlands (Dutch)



@KBWCTY7
France



@KBWCTY9
Greece



@KBWCTY11
Israel (Hebrew)



@KBWCTY13
Latin America/ South America



@SETUPE0
**** Exit Setup**



@KBWCTY16

Poland



@KBWCTY18

Romania



@KBWCTY21

Slovakia



@KBWCTY23

Sweden



@KBWCTY15

Norway



@KBWCTY17

Portugal



@KBWCTY19

Russia



@KBWCTY22

Spain



@SETUPE1
Enter Setup



@KBWCTY24
Switzerland (German)



@KBWCTY25
Turkey_F



@KBWCTY26
Turkey_Q



@KBWCTY27
UK



@KBWCTY28
Japan



@SETUPE0
**** Exit Setup**

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 engine 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.



Do Not Beep on Unknown Character



Beep on Unknown Character



Supposing French keyboard (Country Code: 7) is selected and barcode data "ADF" is being dealt with, the keyboard will fail to locate the "Đ" (0xD0) character and the engine will ignore the character and continue to process the next one.

Do Not Beep on Unknown Character: The engine does not beep and the Host receives "AF".

Beep on Unknown Character: The engine beeps and the Host still receives "AF".



If **Emulate ALT+Keypad ON** is selected, **Beep on Unknown Character** does not function.



@SETUPE1
Enter Setup

Emulate ALT+Keypad

When **Emulate ALT+Keypad** is turned on, ASCII characters (0x20 - 0xFF) are 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.



@KBWALTO
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, **Code Page 1252 (West European Latin)** is selected, and **Emulate Keypad with Leading Zero** is Off, 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"



@SETUPE0
**** Exit Setup**

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 and Data Matrix, besides setting the code page, you also need to set the character encoding in the “Character Encoding” section in Chapter 6. This feature is only effective when **Emulate ALT+Keypad** is turned on.

Note: Code Page 932, Code Page 936 and Code Page 950 are selectable and respectively supported by different software versions.



@KBWCPG0

Code Page 1252 (West European Latin)



@KBWCPG1

Code Page 1251 (Cyrillic)



@KBWCPG2

Code Page 1250 (Central and East European Latin)



@KBWCPG3

Code Page 1253 (Greek)



@KBWCPG4

Code Page 1254 (Turkish)



@KBWCPG5

Code Page 1255 (Hebrew)



@SETUPE1

Enter Setup



@KBWCPCG6

Code Page 1256 (Arabic)



@KBWCPCG8

Code Page 1258 (Vietnamese)



@KBWCPCG10

Code Page 950 (Traditional Chinese, Big5)



@KBWCPCG12

Code Page 932 (Japanese, Shift-JIS)



@KBWCPCG7

Code Page 1257 (Baltic)



@KBWCPCG9

Code Page 936 (Simplified Chinese, GB2312,GBK)



@KBWCPCG11

Code Page 874 (Thai)



@SETUPE0

**** Exit Setup**

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.



Emulate Keypad with Leading Zero

You may turn this feature on to send character sequences sent over the numeric keypad as ISO characters which have a leading zero. For example, ASCII A transmits as "ALT MAKE" 0065 "ALT BREAK". This feature is only effective when **Emulate ALT+Keypad** is enabled.





@SETUPE1
Enter Setup

Function Key Mapping

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



@KBWFKM0
Disable



@KBWFKM1
Ctrl+ASCII Mode



@KBWFKM2
Alt+Keypad Mode



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”.

If **Alt+Keypad Mode** is selected and other parameters of USB HID Keyboard adopt factory defaults, the data above is sent as below:

“A” - Keystroke “A”.

<HT> - “Alt Make” + Keystrokes “009” + “Alt Break”

“F” - Keystroke “F”



@SETUPE0
** Exit Setup

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	Print Screen	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	1B	F6	Ctrl+[
FS	1C	F7	Ctrl+\
GS	1D	F8	Ctrl+]
RS	1E	F9	Ctrl+6
US	1F	F10	Ctrl+-



@SETUPE1
Enter Setup

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+-	



@SETUPE0
** Exit Setup

Inter-Keystroke Delay

This parameter specifies the delay between emulated keystrokes.



@KBWDLY0

No Delay



@KBWDLY40

Long Delay (40ms)



@KBWDLY20

Short Delay (20ms)



@SETUPE1
Enter Setup

Caps Lock

The **Caps Lock On** options 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. To disable this feature, scan the appropriate **Caps Lock OFF** barcode below based on your keyboard.



@KBWCAP0

Caps Lock OFF, Non-Japanese Keyboard



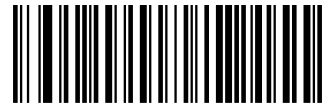
@KBWCAP1

Caps Lock ON, Non-Japanese Keyboard



@KBWCAP2

Caps Lock OFF, Japanese Keyboard



@KBWCAP3

Caps Lock ON, Japanese Keyboard



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”.



@SETUPE0
**** Exit Setup**

Convert Case

Scan the appropriate barcode below to convert all bar code data to your desired case.



@KBWCAS0

No Case Conversion



@KBWCAS1

Convert All to Upper Case



@KBWCAS2

Convert All to Lower Case



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.



@SETUPE1
Enter Setup

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.

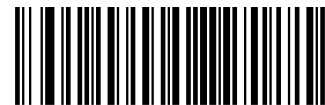
Do Not Emulate Numeric Keypad 2: Sending “+”, “-”, “*” and “/” is emulated as keystroke(s) on main keyboard.

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



@SETUPE0
** Exit Setup



@KBWNCH0

Do Not Emulate Numeric Keypad 2



@KBWNCH1

Emulate Numeric Keypad 2



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



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.



@SETUPE1

Enter Setup

Fast Mode

When **Fast Mode On** is selected, the engine sends characters to the Host faster. If the Host drops characters, turn the Fast Mode off or change the polling rate to a bigger value.



@KBWFAS0

Fast Mode Off



@KBWFAS1

Fast Mode On



@SETUPE0

**** Exit Setup**

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.





@SETUPE1

Enter Setup



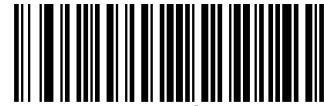
@KBWPOR7

8ms



@KBWPOR8

9ms



@KBWPOR9

10ms



@SETUPE0

**** Exit Setup**

USB CDC

If your engine 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.





@SETUPE1
Enter Setup

HID POS (POS HID Barcode Scanner)

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 TTL-232 interface.



@INTERF5
USB HID-POS

Access the Engine with Your Program

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

For detailed information about USB and HID interfaces, go to www.USB.org.



@SETUPE0
** Exit Setup

Acquire Scanned Data

After a barcode is decoded, the engine 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							
62	Newland Symbology Identifier or N/C: 0x00							
63	-	-	-	-	-	-	-	Decoded data continued

Send Command to the Engine

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

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



@SETUPE1
Enter Setup

IBM SurePOS (Tabletop)



@INTERF6
IBM SurePOS (Tabletop)

IBM SurePOS (Handheld)



@INTERF7
IBM SurePOS (Handheld)

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)
N1-W	USB HID Keyboard	0x1EAB	0x0022
	USB CDC	0x1EAB	0x0006
	HID POS	0x1EAB	0X0010
	IBM SurePOS (Tabletop)	0x1EAB	0x0020
	IBM SurePOS(Handheld)	0x1EAB	0x0021



@SETUPE0
** Exit Setup

Chapter 9 System Settings

Scan Mode

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

Sense Mode: The engine activates a decode session every time it detects a barcode presented to it. The decode session continues until a barcode is decoded or the decode session timeout expires. **Reread Timeout** can avoid undesired rereading of same barcode in a given period of time. **Sensitivity** can change the Sense Mode's sensibility to changes in images captured. **Image Stabilization Timeout** gives the engine time to adapt to ambient environment after it decodes a barcode and "looks" for another.

Continuous Mode: The engine 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).

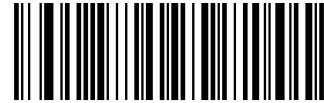
Batch Mode: A trigger pull activates a round of multiple decode sessions. This round of multiple scans continues until you release the trigger. Rereading the same barcode is not allowed in the same round.



@SETUPE1
Enter Setup



@SCNMOD0
Level Mode



@SCNMOD2
Sense Mode



@SCNMOD3
Continuous Mode



@SCNMOD4
Pulse Mode



@SCNMOD7
Batch Mode

Decode Session 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. This feature is only applicable to the Pulse, Sense and Level modes.



@ORTSET
Decode Session Timeout



@SETUPE0
**** Exit Setup**



Set the decode session timeout to 1,500ms:

1. Scan the **Enter Setup** barcode.
2. Scan the **Decode Session Timeout** barcode.
3. Scan the numeric barcodes “1”, “5”, “0” and “0” from the “Digit Barcodes” section in Appendix.
4. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
5. Scan the **Exit Setup** barcode.



@SETUPE1
Enter Setup

Image Stabilization Timeout (Sense Mode)

This parameter defines the amount of time the engine will spend adapting to ambient environment after it decodes a barcode and “looks” for another. It is programmable in 1ms increments from 0ms to 3,000ms.



@SENIST
Image Stabilization Timeout



Set the image stabilization timeout to 800ms:

1. Scan the **Enter Setup** barcode.
2. Scan the **Image Stabilization Timeout** barcode.
3. Scan the numeric barcodes “8”, “0” and “0” from the “Digit Barcodes” section in Appendix.
4. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
5. Scan the **Exit Setup** barcode.



@SETUPE0
**** Exit Setup**

Image Change Trigger Sensitivity

This specifies the degree of acuteness of the engine'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 engine. You can select an appropriate degree of sensitivity that fits your application environment. This feature is only applicable to the Sense mode.



Low Sensitivity



Medium Sensitivity



High Sensitivity



Enhanced Sensitivity



Custom Sensitivity (Level 1-20)



Set the image change trigger sensitivity to Level 10:

1. Scan the **Enter Setup** barcode.
2. Scan the **Custom Sensitivity** barcode.
3. Scan the numeric barcodes "1" and "0" from the "Digit Barcodes" section in Appendix.
4. Scan the **Save** barcode from the "Save/Cancel Barcodes" section in Appendix.
5. Scan the **Exit Setup** barcode.



@SETUPE1
Enter Setup

Scanning Interval (Continuous Mode)

This parameter sets the duration the engine will stop decoding an image before restarting scanning after a good read.



@SCNINV
Scanning Interval (Continuous Mode)

Reread Timeout

Reread Timeout can avoid undesired rereading of same barcode in a given period of time. This feature is only applicable to the Sense and Continuous modes.

To enable/disable the Reread Timeout, scan the appropriate barcode below.

Enable Reread Timeout: Do not allow the engine to re-read same barcode before the reread timeout expires.

Disable Reread Timeout: Allow the engine to re-read same barcode.



@RRDENA1
Enable Reread Timeout



@RRDENA0
Disable Reread Timeout

The following parameter sets the time interval between two successive reads on same barcode. It is programmable in 1ms increments from 0ms to 3,600,000ms. When it is set to a value greater than 3,000, the timeout for rereading same programming barcode is limited to 3,000ms.



@SETUPE0
** Exit Setup



@RRDDUR

Reread Timeout



Set the reread timeout to 1,000ms:

1. Scan the **Enter Setup** barcode.
2. Scan the **Reread Timeout** barcode.
3. Scan the numeric barcodes “1”, “0”, “0” and “0” from the “Digit Barcodes” section in Appendix.
4. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
5. Scan the **Exit Setup** barcode.

You may wish to restart the reread timeout when the engine encounters the same barcode that was decoded in the last scan session before the reread timeout expires. To enable this feature, scan the **Reread Timeout Reset On** barcode. This feature is only effective when **Reread Timeout** is enabled.



@RRDREN1

Reread Timeout Reset On



@RRDRENO

Reread Timeout Reset Off



@SETUPE1
Enter Setup

Good Read Delay

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

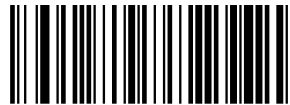


@GRDENA1
Enable Good Read Delay



@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



Set the good read delay to 1,000ms:

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



@SETUPE0
**** Exit Setup**

Scanning Preference

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

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

Barcode Pay Mode: Select this mode when reading barcodes to perform payment transactions.



@EXPLVL0

Normal Mode



@EXPLVL2

Screen Mode



@EXPLVL5

Barcode Pay Mode



@SETUPE1
Enter Setup

Image Flipping



@MIRROR0
Do Not Flip



@MIRROR2
Flip Vertically



@MIRROR1
Flip Horizontally



@MIRROR3
Flip Horizontally & Vertically



@SETUPE0
** Exit Setup

Example of image not flipped



Example of image flipped horizontally



Example of image flipped vertically



Example of image flipped horizontally & vertically



Bad Read Message

Scan the appropriate barcode below to select whether or not to send a bad read message (user-programmable) when a good read does not occur before trigger release, or the decode session timeout expires, or the engine receives the **Stop Scanning** command (For more information, see the “Serial Trigger Command” section in this Chapter).



@NGRENA0

Bad Read Message OFF



@NGRENA1

Bad Read Message ON



@SETUPE1
Enter Setup

Set Bad Read Message

A bad read message can contain up to 7 characters (HEX values from 0x00 to 0xFF). To set a bad read message, scan the **Set Bad Read Message** barcode, the numeric barcodes representing the hexadecimal values of desired character(s) and the **Save** barcode.



@NGRSET
Set Bad Read Message



Set the bad read message to “F” (HEX: 0x46):

1. Scan the **Enter Setup** barcode.
2. Scan the **Set Bad Read Message** barcode.
3. Scan the numeric barcodes “4” and “6” from the “Digit Barcodes” section in Appendix.
4. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
5. Scan the **Exit Setup** barcode.

Trigger Commands

When **Enable Trigger Commands** is selected, you can activate and deactivate the engine in the Level mode with serial trigger commands. Sending the **Start Scanning** command (default: **<SOH> T <EOT>**, user-programmable) to the engine in the Level mode activates a decode session. The decode session continues until a barcode is decoded or the decode session timeout expires or the engine receives the **Stop Scanning** command (default: **<SOH> P <EOT>**, user-programmable).



@SETUPE0
**** Exit Setup**



@SCNTCE0

Disable Trigger Commands



@SCNTCE1

Enable Trigger Commands

Modify Start Scanning Command

The **Start Scanning** command can consist of 1-10 characters (HEX values from 0x01 to 0xFF). In this command, the character “?” (HEX: 0x3F) cannot be the first character. The default **Start Scanning** command is **<SOH> T <EOT>**.



@SCNTCT

Modify Start Scanning Command



Set the Start Scanning command to “*T”:

1. Scan the **Enter Setup** barcode.
2. Scan the **Modify Start Scanning Command** barcode.
3. Scan the numeric barcodes “2”, “A”, “5” and “4” from the “Digit Barcodes” section in Appendix.
4. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
5. Scan the **Exit Setup** barcode.

Modify Stop Scanning Command

The **Stop Scanning** command can consist of 1-10 characters (HEX values from 0x01 to 0xFF). In this command, the character “?” (HEX: 0x3F) cannot be the first character. The default **Stop Scanning** command is **<SOH> P <EOT>**.



@SETUPE1
Enter Setup



@SCNTCP
Modify Stop Scanning Command



Set the Stop Scanning command to “*P”:

1. Scan the **Enter Setup** barcode.
2. Scan the **Modify Stop Scanning Command** barcode.
3. Scan the numeric barcodes “2”, “A”, “5” and “0” from the “Digit Barcodes” section in Appendix.
4. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
5. Scan the **Exit Setup** barcode.



@SETUPE0
**** Exit Setup**

Illumination

A couple of illumination options are provided to improve the lighting conditions during every image capture:

Normal: Illumination LEDs are turned on during image capture.

Always On: Illumination LEDs on the engine keep on after the engine is powered on.

Off: Illumination LEDs are off all the time.



@ILLSCN1

Normal



@ILLSCN0

Off



@ILLSCN2

Always On



@SETUPE1
Enter Setup

External Illumination

Pin 6 (EXT_LIGHT) on the 13-pin FPC connector is the external illumination control signal. Users can design the external illumination LEDs based on actual needs.

Normal: External Illumination LEDs are turned on when starting a decode session and off when stopping barcode reading.

Always On: External Illumination LEDs keep on.

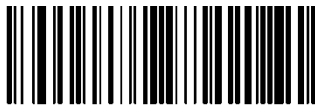
Off: External Illumination LEDs are off when a decode session starts.



@EILSCN0
Off



@EILSCN1
Normal



@EILSCN2
Always On



@SETUPE0
** Exit Setup

Aiming

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

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

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

Off: Aiming pattern is off all the time.





@SETUPE1
Enter Setup

Good Read LED

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



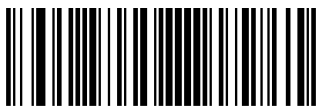
@GRLENA1
On



@GRLENA0
Off

Good Read LED Duration

This parameter sets the amount of time that the Good Read LED to remain on following a good read. It is programmable in 1ms increments from 1ms to 2,500ms.



@SETUPE0
**** Exit Setup**



@GRLDUR20
Short (20ms)



@GRLDUR120
Medium (120ms)



@GRLDUR220
Long (220ms)



@GRLDUR320
Prolonged (320ms)



@GRLDUR
Custom (1 - 2,500ms)



Set the Good Read LED duration to 800ms:

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



@SETUPE1
Enter Setup

Power On Beep

The engine 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



@SETUPE0
**** Exit Setup**

Good Read Beep Duration

This parameter sets the length of the beep the engine emits on a good read. It is programmable in 1ms increments from 20ms to 300ms.



@GRBDUR40
Short (40ms)



@GRBDUR80
Medium (80ms)



@GRBDUR120
Long (120ms)



@GRBDUR
Custom (20 – 300ms)



Set the Good Read Beep duration to 200ms:

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



@SETUPE1
Enter Setup

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)



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

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



@SETUPE0
**** Exit Setup**

Good Read Beep Volume

There are 20 volume levels to choose from. The bigger the value, the louder the Good Read Beep.



@GRBVLL20

Loud



@GRBVLL12

Medium



@GRBVLL5

Low



@GRBVLL

Custom Volume (Level 1-20)



Set the Good Read Beep volume to Level 8:

1. Scan the **Enter Setup** barcode.
2. Scan the **Custom Volume** barcode.
3. Scan the numeric barcode “8” from the “Digit Barcodes” section in Appendix.
4. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
5. Scan the **Exit Setup** barcode.



@SETUPE1
Enter Setup

Auto Sleep

Auto Sleep allows the engine to automatically enter the sleep mode if no operation or communication is performed for a time period (user programmable). When the engine is in the sleep mode, receiving trigger signal or communication from the Host can awake the engine.



@ATSENA0
Disable Auto Sleep



@ATSENA1
Enable Auto Sleep

The parameter below specifies how long the engine remains idle (no operation or communication occurs) before it is put into sleep mode.



@ATSDUR
Time Period from Idle to Sleep



Set the Time Period from Idle to Sleep duration to 5s:

6. Scan the **Enter Setup** barcode.
7. Scan the **Time Period from Idle to Sleep** barcode.
8. Scan the numeric barcode “5” from the “Digit Barcodes” section in Appendix.
9. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
10. Scan the **Exit Setup** barcode.



@SETUPE0
**** Exit Setup**

Chapter 10 Symbologies

Introduction

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

Global Settings

Enable/Disable All Symbologies

If the **Disable All Symbologies** feature is enabled, the engine 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 Symbologies



@ALL1DC1

Enable 1D Symbologies



@ALL1DC0

Disable 1D Symbologies



@SETUPE1
Enter Setup

Enable/Disable 2D Symbologies



@ALL2DC1
Enable 2D Symbologies



@ALL2DC0
Disable 2D Symbologies

Surround GS1 Application Identifiers (AI's) with Parentheses

When **Surround GS1 AI's with Parentheses** is selected, each application identifier (AI) contained in scanned data will be enclosed in parentheses in the output message.



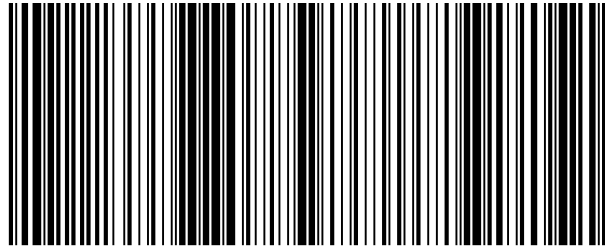
@GS1AIP0
Do Not Surround GS1 AI's with Parentheses



@GS1AIP1
Surround GS1 AI's with Parentheses



@SETUPE0
** Exit Setup



(01) 0 0614141 99999 6 (10) 10ABCEDF123456

If **Surround GS1 AI's with Parentheses** is selected, the barcode above is output as (01)00614141999996(10)10ABCEDF123456.

If **Do Not Surround GS1 AI's with Parentheses** is selected, the barcode above is output as 01006141419999961010ABCEDF123456.



@SETUPE1
Enter Setup

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 engine fails to identify Code 128 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable Code 128** barcode.



@SETUPE0
** Exit Setup

Set Length Range for Code 128

The engine 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.



Set the Minimum Length



Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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 engine to decode Code 128 barcodes containing between 8 and 12 characters:

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



@SETUPE1
Enter Setup

Transmit Check Character



@128CHK2
Transmit Code 128 Check Character



@128CHK1
Do Not Transmit Code 128 Check Character

FNC1 Output



@128FNC0
Off



@128FNC1
On



@SETUPE0
**** Exit Setup**

EAN-8

Restore Factory Defaults



Restore the Factory Defaults of EAN-8

Enable/Disable EAN-8



Enable EAN-8



Disable EAN-8



If the engine fails to identify EAN-8 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **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.



Transmit EAN-8 Check Character



Do Not Transmit EAN-8 Check Character



@SETUPE1
Enter Setup

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 engine 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 engine decodes a mix of EAN-8 barcodes with and without 2-digit add-on codes.



@SETUPE0
** Exit Setup

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.



Disable 5-Digit Add-On Code



Enable 5-Digit Add-On Code



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



@SETUPE1
Enter Setup

Add-On Code Required

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



@EA8REQ0
EAN-8 Add-On Code Not Required



@EA8REQ1
EAN-8 Add-On Code Required

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



@SETUPE0
**** Exit Setup**

EAN-13

Restore Factory Defaults



Restore the Factory Defaults of EAN-13

Enable/Disable EAN-13



Enable EAN-13



Disable EAN-13



If the engine fails to identify EAN-13 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable EAN-13** barcode.



@SETUPE1
Enter Setup

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 engine 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 engine decodes a mix of EAN-13 barcodes with and without 2-digit add-on codes.



@SETUPE0
** Exit Setup

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.



Disable 5-Digit Add-On Code



Enable 5-Digit Add-On Code



Disable 5-Digit Add-On Code: The engine 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 engine 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 engine will only read EAN-13 barcodes that contain add-on codes.



EAN-13 Add-On Code Not Required



EAN-13 Add-On Code Required



@SETUPE1
Enter Setup

EAN-13 Beginning with 290 Add-On Code Required

This setting programs the engine 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 engine 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



@SETUPE0

**** Exit Setup**

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

This setting programs the engine 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.



Do Not Require Add-On Code



Require Add-On Code

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

This setting programs the engine 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.



Do Not Require Add-On Code



Require Add-On Code



@SETUPE1
Enter Setup

EAN-13 Beginning with 977 Add-On Code Required

This setting programs the engine 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 engine 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



@SETUPE0
** Exit Setup

EAN-13 Beginning with 979 Add-On Code Required

This setting programs the engine 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.



Do Not Require Add-On Code



Require Add-On Code



@SETUPE1
Enter Setup

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



If the engine fails to identify UPC-E barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable UPC-E** barcode.



@SETUPE0
**** Exit Setup**

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.



Transmit UPC-E Check Character



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.



Disable 2-Digit Add-On Code



Enable 2-Digit Add-On Code



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



@SETUPE1
Enter Setup

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 engine 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 engine decodes a mix of UPC-E barcodes with and without 5-digit add-on codes.

Add-On Code Required

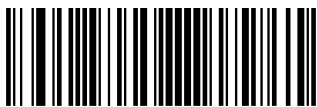
When **UPC-E Add-On Code Required** is selected, the engine will only read UPC-E barcodes that contain add-on codes.



@UPEREQ0
UPC-E Add-On Code Not Required



@UPEREQ1
UPC-E Add-On Code Required



@SETUPE0
** Exit Setup

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 (zero suppressed) 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



@SETUPE1
Enter Setup

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 engine fails to identify UPC-A barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **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



@SETUPE0
** Exit Setup

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.



Disable 2-Digit Add-On Code



Enable 2-Digit Add-On Code



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



@SETUPE1
Enter Setup

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 engine 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 engine decodes a mix of UPC-A barcodes with and without 5-digit add-on codes.



@SETUPE0
** Exit Setup

Add-On Code Required

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



@UPAREQ0

UPC-A Add-On Code Not Required



@UPAREQ1

UPC-A Add-On Code Required

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



@UPAPRE1

System Character



@UPAPRE2

System Character & Country Code



@SETUPE1
Enter Setup

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 engine fails to identify Interleaved 2 of 5 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable Interleaved 2 of 5** barcode.



@SETUPE0
**** Exit Setup**

Set Length Range for Interleaved 2 of 5

The engine 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.



Set the Minimum Length



Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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.



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

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



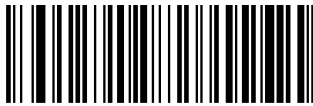
@SETUPE1
Enter Setup

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 engine transmits Interleaved 2 of 5 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The engine 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 engine 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.)



@SETUPE0
**** Exit Setup**

Febraban

Disable/Enable Febraban



@I25FBB0

Disable Febraban



@I25FBB1

Enable Febraban, Do Not Expand



@I25FBB2

Enable Febraban, Expand

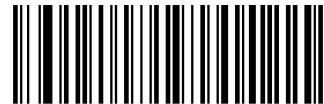
Transmit Delay per Character

Transmit Delay per Character applies to both Expanded and Unexpanded Febraban. This feature is available only when USB HID Keyboard is enabled.



@FEBSEN0

Disable Transmit Delay per Character



@FEBSEN1

Enable Transmit Delay per Character



@SETUPE1
Enter Setup

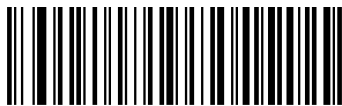
You may select an appropriate delay value from the options below as per your actual needs.



@FEBSDT0
0ms



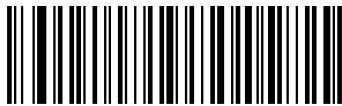
@FEBSDT5
5ms



@FEBSDT10
10ms



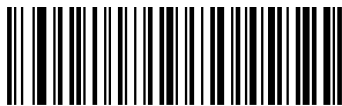
@FEBSDT15
15ms



@FEBSDT20
20ms



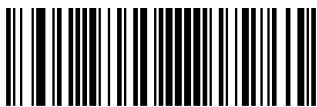
@FEBSDT25
25ms



@FEBSDT30
30ms



@FEBSDT35
35ms



@SETUPE0
**** Exit Setup**



@FEBSDT40

40ms



@FEBSDT45

45ms



@FEBSDT50

50ms



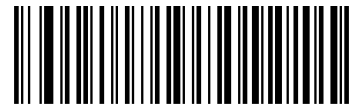
@FEBSDT55

55ms



@FEBSDT60

60ms



@FEBSDT65

65ms



@FEBSDT70

70ms



@FEBSDT75

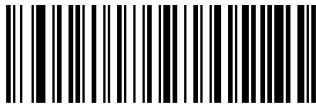
75ms



@SETUPE1
Enter Setup

Transmit Delay per 12 Characters

Transmit Delay per 12 Characters applies to Expanded Febraban only. This feature is available only when USB HID Keyboard is enabled.



@FEBMENO
Disable Transmit Delay per 12 Characters



@FEBMEN1
Enable Transmit Delay per 12 Characters

You may select an appropriate delay value from the options below as per your actual needs.



@FEBMDT0
0ms



@FEBMDT1
300ms



@FEBMDT2
400ms



@SETUPE0
**** Exit Setup**



@FEBMDT4

600ms



@FEBMDT6

800ms



@FEBMDT3

500ms



@FEBMDT5

700ms



@FEBMDT7

900ms



@SETUPE1
Enter Setup

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 engine only decodes Interleaved 2 of 5 barcodes with a length of 14 characters and the last character as the check character.



@SETUPE0
** Exit Setup

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



Restore the Factory Defaults of ITF-6

Enable/Disable ITF-6



Disable ITF-6



Enable ITF-6 But Do Not Transmit Check Character



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 engine only decodes Interleaved 2 of 5 barcodes with a length of 6 characters and the last character as the check character.



@SETUPE1
Enter Setup

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 engine fails to identify Matrix 2 of 5 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable Matrix 2 of 5** barcode.



@SETUPE0
**** Exit Setup**

Set Length Range for Matrix 2 of 5

The engine 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.



Set the Minimum Length



Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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 engine to decode Matrix 2 of 5 barcodes containing between 8 and 12 characters:

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



@SETUPE1
Enter Setup

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 engine transmits Matrix 2 of 5 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The engine 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 engine 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.)



@SETUPE0
** Exit Setup

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 engine fails to identify Code 39 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable Code 39** barcode.



@SETUPE1
Enter Setup

Set Length Range for Code 39

The engine 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



@C39MAX
Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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 engine to decode Code 39 barcodes containing between 8 and 12 characters:

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



@SETUPE0
**** Exit Setup**

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 engine transmits Code 39 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The engine 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 engine 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.



Disable



Do Not Transmit Check Character After Verification



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.)



@SETUPE1
Enter Setup

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 engine 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



@SETUPE0
**** Exit Setup**

Enable/Disable Code 32 (Italian Pharma Code)

Code 32 is a variant of Code 39 used by the Italian pharmaceutical industry. Scan the appropriate bar code below to enable or disable Code 32. Code 39 must be enabled and Code 39 check character verification must be disabled for this parameter to function.



@C39E320

Disable Code 32



@C39E321

Enable Code 32

Code 32 Prefix

Scan the appropriate barcode below to enable or disable adding the prefix character "A" to all Code 32 barcodes. Code 32 must be enabled for this parameter to function.



@C39S320

Disable Code 32 Prefix



@C39S321

Enable Code 32 Prefix



@SETUPE1
Enter Setup

Transmit Code 32 Start/Stop Character

Code 32 must be enabled for this parameter to function.



@C39T320

Do Not Transmit Code 32 Start/Stop Character



@C39T321

Transmit Code 32 Start/Stop Character

Transmit Code 32 Check Character

Code 32 must be enabled for this parameter to function.



@C39C320

Do Not Transmit Code 32 Check Character



@C39C321

Transmit Code 32 Check Character



@SETUPE0
**** Exit Setup**

Codabar

Restore Factory Defaults



Restore the Factory Defaults of Codabar

Enable/Disable Codabar



Enable Codabar



Disable Codabar



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



@SETUPE1
Enter Setup

Set Length Range for Codabar

The engine 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



@CBAMAX
Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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.



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

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

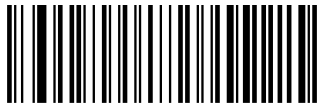


@SETUPE0
**** Exit Setup**

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 engine transmits Codabar barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The engine 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 engine 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.



Disable



Do Not Transmit Check Character After Verification



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.)



@SETUPE1
Enter Setup

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



@SETUPE0
**** Exit Setup**

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 engine fails to identify Code 93 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable Code 93** barcode.



@SETUPE1
Enter Setup

Set Length Range for Code 93

The engine 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



@C93MAX
Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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 engine to decode Code 93 barcodes containing between 8 and 12 characters:

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



@SETUPE0
**** Exit Setup**

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 engine transmits Code 93 barcodes as is.

Do Not Transmit Check Character After Verification: The engine 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 engine 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.



Disable



Do Not Transmit Check Character After Verification



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.)



@SETUPE1
Enter Setup

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 engine fails to identify GS1-128 barcodes, you may first try this solution by scanning the **EnterSetup** barcode and then **Enable GS1-128** barcode.



@SETUPE0
** Exit Setup

Set Length Range for GS1-128

The engine can be configured to only decode GS1-128 barcodes with lengths that fall between (inclusive) the minimum and maximum lengths. To accomplish it, you need to set the minimum and maximum lengths.



Set the Minimum Length



Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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.



Set the engine to decode GS1-128 barcodes containing between 8 and 12 characters:

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



@SETUPE1
Enter Setup

Transmit Check Character

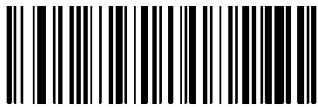


@GS1CHK2
Transmit GS1-128 Check Character



@GS1CHK1
Do Not Transmit GS1-128 Check Character

FNC1 Output



@GS1FNC0
Off



@GS1FNC1
On



@SETUPE0
**** Exit Setup**

GS1 Databar (RSS)

Restore Factory Defaults



Restore the Factory Defaults of GS1 Databar

Enable/Disable GS1 Databar



Enable GS1 Databar



Disable GS1 Databar



If the engine fails to identify GS1 Databar barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable GS1 Databar** barcode.

Transmit Application Identifier “01”



Transmit Application Identifier “01”



Do Not Transmit Application Identifier “01”



@SETUPE1
Enter Setup

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



@SETUPE0
**** Exit Setup**

Set Length Range for Code 11

The engine 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.



Set the Minimum Length



Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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 engine to decode Code 11 barcodes containing between 8 and 12 characters:

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



@SETUPE1
Enter Setup

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 engine transmits Code 11 barcodes as is.



@C11CHK0
Disable



@C11CHK1
One Check Character, MOD11



@C11CHK2
Two Check Characters, MOD11/MOD11



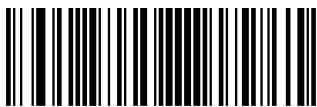
@C11CHK3
Two Check Characters, MOD11/MOD9



@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)



@SETUPE0
**** Exit Setup**

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.)



@SETUPE1
Enter Setup

ISBN

Restore Factory Defaults



@ISBDEF
Restore the Factory Defaults of ISBN

Enable/Disable ISBN



@ISBENA1
Enable ISBN



@ISBENA0
Disable ISBN



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



@SETUPE0
**** Exit Setup**

Set ISBN Format



@ISBT101
ISBN-10



@ISBT100
ISBN-13

ISSN

Restore Factory Defaults



@ISSDEF

Restore the Factory Defaults of ISSN



@SETUPE1

Enter Setup

Enable/Disable ISSN



@ISSENA1

Enable ISSN

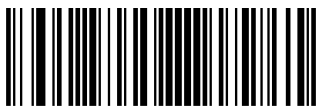


@ISSENA0

Disable ISSN



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



@SETUPE0

**** Exit Setup**

Industrial 25

Restore Factory Defaults



@L25DEF

Restore the Factory Defaults of Industrial 25

Enable/Disable Industrial 25



@L25ENA1

Enable Industrial 25



@L25ENA0

Disable Industrial 25



If the engine fails to identify Industrial 25 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable Industrial 25** barcode.



@SETUPE1
Enter Setup

Set Length Range for Industrial 25

The engine 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



@L25MAX
Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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 engine to decode Industrial 25 barcodes containing between 8 and 12 characters:

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



@SETUPE0
**** Exit Setup**

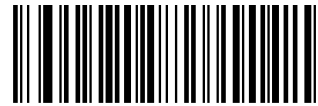
Check Character Verification

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 engine transmits Industrial 25 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The engine 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 engine 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.



Disable



Do Not Transmit Check Character After Verification



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.)



@SETUPE1
Enter Setup

Standard 25

Restore Factory Defaults



@S25DEF
Restore the Factory Defaults of Standard 25

Enable/Disable Standard 25



@S25ENA1
Enable Standard 25



@S25ENA0
Disable Standard 25



If the engine fails to identify Standard 25 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable Standard 25** barcode.



@SETUPE0
**** Exit Setup**

Set Length Range for Standard 25

The engine 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.



Set the Minimum Length



Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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 engine to decode Standard 25 barcodes containing between 8 and 12 characters:

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



@SETUPE1
Enter Setup

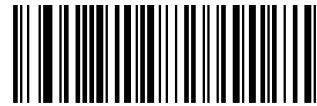
Check Character Verification

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 engine transmits Standard 25 barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The engine 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 engine 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



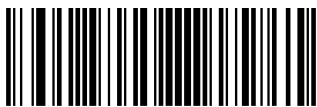
@S25CHK1
Do Not Transmit Check Character After Verification



@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.)



@SETUPE0
** Exit Setup

Plessey

Restore Factory Defaults



@PLYDEF

Restore the Factory Defaults of Plessey

Enable/Disable Plessey



@PLYENA1

Enable Plessey



@PLYENA0

Disable Plessey



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



@SETUPE1
Enter Setup

Set Length Range for Plessey

The engine 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



@PLYMAX
Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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 engine to decode Plessey barcodes containing between 8 and 12 characters:

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



@SETUPE0
**** Exit Setup**

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 engine transmits Plessey barcodes as is.
- ✧ **Do Not Transmit Check Character After Verification:** The engine 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 engine 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.



Disable



Do Not Transmit Check Character After Verification



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.)



@SETUPE1
Enter Setup

MSI-Plessey

Restore Factory Defaults



@MSIDF
Restore the Factory Defaults of MSI-Plessey

Enable/Disable MSI-Plessey



@MSIENA1
Enable MSI-Plessey



@MSIENA0
Disable MSI-Plessey



If the engine fails to identify MSI-Plessey barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable MSI-Plessey** barcode.



@SETUPE0
**** Exit Setup**

Set Length Range for MSI-Plessey

The engine 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.



Set the Minimum Length



Set the Maximum Length



If minimum length is set to be greater than maximum length, the engine 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 engine to decode MSI-Plessey barcodes containing between 8 and 12 characters:

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

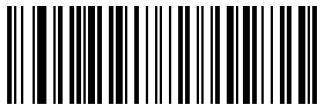


@SETUPE1
Enter Setup

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.

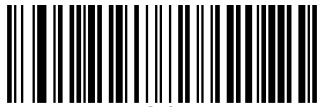
If the **Disable** option is enabled, the engine transmits MSI-Plessey barcodes as is.



@MSICLK0
Disable



@MSICLK1
One Check Character, MOD10



@MSICLK2
Two Check Characters, MOD10/MOD10



@MSICLK3
Two Check Characters, MOD10/MOD11



@SETUPE0
**** Exit Setup**

Transmit Check Character



Transmit MSI-Plessey Check Character



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.)



@SETUPE1
Enter Setup

AIM 128

Restore Factory Defaults



@AIMDEF
Restore the Factory Defaults of AIM 128

Enable/Disable AIM 128



@AIMENA1
Enable AIM 128



@AIMENA0
Disable AIM 128



If the engine fails to identify AIM 128 barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable AIM 128** barcode.



@SETUPE0
**** Exit Setup**

Set Length Range for AIM 128

The engine 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.



Set the Minimum Length



Set the Maximum Length



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



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

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

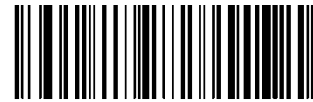


@SETUPE1
Enter Setup

Transmit Check Character



@AIMCHK2
Transmit AIM 128 Check Character



@AIMCHK1
Do Not Transmit AIM 128 Check Character

FNC1 Output



@AIMFNC0
Off



@AIMFNC1
On



@SETUPE0
** Exit Setup

PDF417

Restore Factory Defaults



@PDFDEF

Restore the Factory Defaults of PDF417

Enable/Disable PDF417



@PDFENA1

Enable PDF417



@PDFENA0

Disable PDF417



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



@SETUPE1
Enter Setup

Set Length Range for PDF417

The engine 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



@PDFMAX
Set the Maximum Length



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 engine to decode PDF417 barcodes containing between 8 and 12 characters:

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



@SETUPE0
**** Exit Setup**

PDF417 Twin Code

PDF417 twin code is 2 PDF417 barcodes paralleled vertically or horizontally. They must both be either regular or inverse barcodes. They must have similar specifications and be placed closely together.

There are 3 options for reading PDF417 twin codes:

- ◇ **Single PDF417 Only:** Read either PDF417 code.
- ◇ **Twin PDF417 Only:** Read both PDF417 codes.
- ◇ **Both Single & Twin:** Read both PDF417 codes. If successful, transmit as twin PDF417 only. Otherwise, try single PDF417 only.



Single PDF417 Only



Twin PDF417 Only



Both Single & Twin



@SETUPE1

Enter Setup

Character Encoding



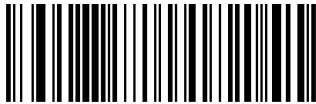
@PDFENC0

Default Character Encoding



@PDFENC1

UTF-8



@PDFENC2

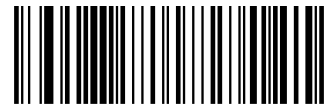
Automatically Select UTF-8 or Code Page

PDF417 ECI Output



@PDFECI0

Disable PDF417 ECI Output



@PDFECI1

Enable PDF417 ECI Output



@SETUPE0

**** Exit Setup**

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



@SETUPE1
Enter Setup

Set Length Range for QR Code

The engine 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.



@QRCPMIN
Set the Minimum Length



@QRCPMAX
Set the Maximum Length



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 engine to decode QR Code barcodes containing between 8 and 12 characters:

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



@SETUPE0
**** Exit Setup**

QR Twin Code

QR twin code is 2 QR barcodes paralleled vertically or horizontally. They must both be either regular or inverse barcodes. They must have similar specifications and be placed closely together.

There are 3 options for reading QR twin codes:

- ✧ **Single QR Only:** Read either QR code.
- ✧ **Twin QR Only:** Read both QR codes. Transmission sequence: left (upper) QR code followed by right (lower) QR code.
- ✧ **Both Single & Twin:** Read both QR codes. If successful, transmit as twin QR only. Otherwise, try single QR only.



@QRCD0U0

Single QR Only



@QRCD0U1

Twin QR Only



@QRCD0U2

Both Single & Twin



@SETUPE1

Enter Setup

Character Encoding



@QRCENC0

Default Character Encoding



@QRCENC1

UTF-8



@QRCENC3

Automatically Select UTF-8 or Code Page



@SETUPE0

**** Exit Setup**

QR ECI Output



@QRCECI0

Disable QR ECI Output



@QRCECI1

Enable QR ECI Output

URL QR

URL QR code refers to QR code whose barcode data begins with the http or HTTP.



@QRCURL0

Disable URL QR



@QRCURL1

Enable URL QR

Micro QR Code

Restore Factory Defaults



@SETUPE1

Enter Setup



@MQRDEF

Restore the Factory Defaults of Micro QR

Enable/Disable Micro QR



@MQRENA1

Enable Micro QR



@MQRENA0

Disable Micro QR



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



@SETUPE0

**** Exit Setup**

Set Length Range for Micro QR

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



Set the Minimum Length



Set the Maximum Length



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



Set the engine to decode Micro QR Code barcodes containing between 8 and 12 characters:

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



@SETUPE1
Enter Setup

Aztec

Restore Factory Defaults



@AZTDEF
Restore the Factory Defaults of Aztec Code

Enable/Disable Aztec Code



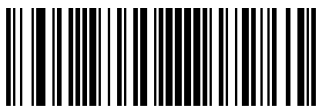
@AZTENA1
Enable Aztec Code



@AZTENA0
Disable Aztec Code



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



@SETUPE0
**** Exit Setup**

Set Length Range for Aztec Code

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



Set the Minimum Length



Set the Maximum Length



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



Set the engine to decode Aztec barcodes containing between 8 and 12 characters:

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



@SETUPE1
Enter Setup

Character Encoding



@AZTENC0
Default Character Encoding



@AZTENC1
UTF-8



@AZTENC2
Automatically Select UTF-8 or Code Page

Aztec ECI Output



@AZTECI0
Disable Aztec ECI Output



@AZTECI1
Enable Aztec ECI Output



@SETUPE0
**** Exit Setup**

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 engine fails to identify Data Matrix barcodes, you may first try this solution by scanning the **Enter Setup** barcode and then **Enable Data Matrix** barcode.



@SETUPE1
Enter Setup

Set Length Range for Data Matrix

The engine 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



@DMCMAX
Set the Maximum Length



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.



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

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



@SETUPE0
**** Exit Setup**

Data Matrix Twin Code

Data Matrix twin code is 2 Data Matrix barcodes paralleled vertically or horizontally. They must both be either regular or inverse barcodes. They must have similar specifications and be placed closely together.

There are 3 options for reading Data Matrix twin codes:

- ❖ **Single Data Matrix Only:** Read either Data Matrix code.
- ❖ **Twin Data Matrix Only:** Read both Data Matrix codes. Transmission sequence: left (upper) Data Matrix code followed by right (lower) Data Matrix code.
- ❖ **Both Single & Twin:** Read both Data Matrix codes. If successful, transmit as twin Data Matrix only. Otherwise, try single Data Matrix only.



Single Data Matrix Only



Twin Data Matrix Only



Both Single & Twin



@SETUPE1
Enter Setup

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 models in length and width: 6*16, 6*14...14*22.



@DMCREC1
Enable Rectangular Barcode



@DMCREC0
Disable Rectangular Barcode

Character Encoding



@DMCENC0
Default Character Encoding



@DMCENC1
UTF-8



@DMCENC2
Automatically Select UTF-8 or Code Page



@SETUPE0
**** Exit Setup**

Data Matrix ECI Output



Disable Data Matrix ECI Output



Enable Data Matrix ECI Output



@SETUPE1
Enter Setup

Chapter 11 Data Formatter

Introduction

You may use the Data Formatter to modify the engine'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 engine. 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

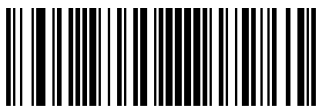
Add a Data Format

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.



@SETUPE0
** Exit Setup

Step 1: Scan the **Enter Setup** barcode.

Step 2: Scan the **Add Data Format** barcode.



Add Data Format

Step 3: 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 4: 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 5: Set interface type

Scan **999** for any interface type.

Step 6: 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 7: 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 8: 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 9: Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix to save your data format.



@SETUPE1
Enter Setup

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 Enter Setup barcode | Enter the Setup mode |
| 2. Scan the Add Data Format barcode | Add a data format |
| 3. Scan the 0 barcode | Select Format_0 as the label |
| 4. Scan the 6 barcode | Select formatter command type 6 |
| 5. Scan the 9 barcode three times | All interface types applicable |
| 6. Scan the barcodes 002 | Only Code 128 applicable |
| 7. Scan the barcodes 0010 | Only a length of 10 characters applicable |
| 8. Scan the alphanumeric barcodes F141 | Send all characters followed by "A" (HEX: 41) |
| 9. Scan the Save 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 9 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|169990049999F143;**.



@SETUPE0
**** Exit Setup**

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 engine or rebooting it; “#” means temporary setting which will be lost by removing power from the engine 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>



@SETUPE1
Enter Setup

Enable/Disable Data Formatter

When Data Formatter is disabled, the data format you have enabled becomes invalid.



@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).



@SETUPE0
** Exit Setup



@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

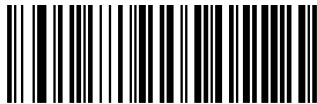


@DFMENA4

Enable Data Formatter, Not Required, Drop Prefix/Suffix

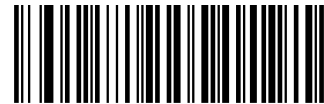
Non-Match Error Beep

If Non-Match Error Beep is turned ON, the engine 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



@SETUPE1
Enter Setup

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



@SETUPE0
**** Exit Setup**

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 engine to use 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 engine, or turning off/ rebooting the device.



@DFMSIN0

Single Scan – Format_0



@DFMSIN2

Single Scan – Format_2



@DFMSIN1

Single Scan – Format_1



@DFMSIN3

Single Scan – Format_3



@SETUPE1
Enter Setup

Clear Data Format

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

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



@SETUPE0
** Exit Setup

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=F2nxx (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>



@SETUPE1
Enter Setup

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



1234567890ABCDEFGHIJ

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.



@SETUPE0
**** Exit Setup**

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.



@SETUPE1
Enter Setup

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.



@SETUPE0
**** Exit Setup**

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.



@SETUPE1
Enter Setup

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.



@SETUPE0
** Exit Setup

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.”



@SETUPE1
Enter Setup

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

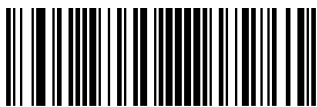
The data is output as: **37692**

<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.



@SETUPE0
** Exit Setup

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



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: **34567890**

<CR>

E4 Replace characters

Syntax = E4nnxx₁xx₂yy₁yy₂...zz₁zz₂(nn: The total count of the number of characters (characters to be replaced plus replacement characters; xx₁: The characters to be replaced, xx₂: The replacement characters, continuing through zz₁ and zz₂)

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



@SETUPE1
Enter Setup

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: **1234**

5678

ABC

<CR>



@SETUPE0
** Exit Setup

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



cd123abc23bc12ab232

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



@SETUPE1
Enter Setup

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: **cd1ABCabcABCbc12abABC2**

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: **cd1abc23bc12ab232**



@SETUPE0
**** Exit Setup**

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 5th and 6th 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



@SETUPE1
Enter Setup

B5 Insert key strokes

Syntax=B5nnssxx (nn: The number of keys pressed (without key modifiers); ss: the key modifier from the table below; xx: the key number from the “Unicode Key Maps” in Appendix.)

Insert a key stroke or combination of key strokes. Key strokes are dependent on your keyboard (see the “Unicode Key Maps” in Appendix). This command can only be used with USB HID Keyboard.

Key Modifiers	
No Key Modifier	00
Shift Left	01
Shift Right	02
Alt Left	04
Alt Right	08
Control Left	10
Control Right	20

For example, B501001F inserts an “a” on a U.S. style keyboard. B5 = the command, 01 = number of keys pressed (without the key modifier), 00 is No Key Modifier, and 1F is the “a” key. If an “A” were to be inserted, B501011F or B501021F would be entered.

If there are two keystrokes, the syntax would change from Syntax=B5nnssxx for one keystroke to Syntax=B5nnssxxssxx. An example that would insert “aA” is as follows: B502001F011F.

Note: Key modifiers can be added together when needed. Example: Shift Left + Alt Left + Control Left =15.



@SETUPE0
** Exit Setup

Chapter 12 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



@SETUPE1
Enter Setup

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



@SETUPE0
** Exit Setup

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”.



Disable Custom Prefix



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.



Set Custom Prefix



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

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



@SETUPE1

Enter Setup

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 engine 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.



@SETUPE0

**** Exit Setup**

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



@SETUPE1
Enter Setup

Modify Code ID

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



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

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

Restore the default Code IDs of all symbologies:

1. Scan the **Enter Setup** barcode.
2. Scan the **Restore All Default Code IDs** barcode.
3. Scan the **Exit Setup** barcode.



@SETUPE0
**** Exit Setup**

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



@SETUPE1
Enter Setup



@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



@SETUPE0
**** Exit Setup**



@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



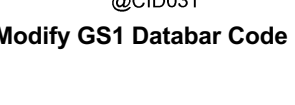
@CID028

Modify Code 11 Code ID



@CID029

Modify MSI-Plessey Code ID



@CID031

Modify GS1 Databar Code ID



@SETUPE1

Enter Setup

2D symbologies:



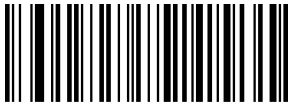
@CID032

Modify PDF417 Code ID



@CID033

Modify QR Code ID



@CID034

Modify Aztec Code ID



@CID035

Modify Data Matrix Code ID



@CID043

Modify Micro QR Code ID



@SETUPE0

**** Exit Setup**

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”.



Disable Custom Suffix



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.



Set Custom Suffix



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

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



@SETUPE1
Enter Setup

Data Packing

Introduction

Data packing is designed for a specific group of users who want to have the data packed before transmission. Data packing influences data format, so it is advised to disable this feature when it is not required.

Data Packing Options

- ✧ **Disable Data Packing:** Transmit decoded data in raw format (unpacked).
- ✧ **Enable Data Packing, Format 1:** Transmit decoded data with the packet format 1 defined below.
Packet format 1: [STX + ATTR + LEN] + [AL_TYPE + DATA] + [LRC]
STX: 0x02
ATTR: 0x00
LEN: Barcode data length is expressed in 2 bytes ranging from 0x0000 (0) to 0xFFFF (65535).
AL_TYPE: 0x36
DATA: Raw barcode data.
LRC: Check digit.
LRC calculation algorithm: computation sequence: 0xFF+LEN+AL_TYPE+DATA; computation method is XOR, byte by byte.
- ✧ **Enable Data Packing, Format 2:** Transmit decoded data with the packet format 2 defined below.
Packet format 2: [STX + ATTR + LEN] + [AL_TYPE] + [Symbology_ID + DATA] + [LRC]
STX: 0x02
ATTR: 0x00
LEN: Barcode data length is expressed in 2 bytes ranging from 0x0000 (0) to 0xFFFF (65535).
AL_TYPE: 0x3B
Symbology_ID: The ID number of symbology, 1 byte.
DATA: Raw barcode data.
LRC: Check digit.
LRC calculation algorithm: computation sequence: 0xFF+LEN+AL_TYPE+Symbology_ID+DATA; computation method is XOR, byte by byte.



@SETUPE0
** Exit Setup



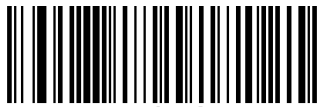
@PACKAG0

Disable Data Packing



@PACKAG1

Enable Data Packing, Format 1



@PACKAG2

Enable Data Packing, Format 2

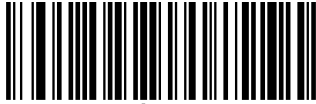


@SETUPE1
Enter Setup

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)



@SETUPE0
** Exit Setup



Set the terminating character suffix to 0x0A:

1. Scan the **Enter Setup** barcode.
2. Scan the **Set Terminating Character Suffix** barcode.
3. Scan the numeric barcodes “0” and “A” from the “Digit Barcodes” section in Appendix.
4. Scan the **Save** barcode from the “Save/Cancel Barcodes” section in Appendix.
5. Scan the **Enable Terminating Character Suffix** barcode.
6. Scan the **Exit Setup** barcode.



@SETUPE1
Enter Setup

Chapter 13 Programming Commands

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 Keyboard 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).



@SETUPE0
** Exit Setup

^ (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>)



@SETUPE1
Enter Setup

Read Barcode On/Off

Sending the Read Barcode Off command `~<SOH>0000#SCNENA0;<ETX>` to the engine can disable it from reading barcode, and the engine is unable to scan barcode unless you send the Read Barcode On command `~<SOH>0000#SCNENA1;<ETX>` to it or power cycle it. By default, Read Barcode is On.

Make a Beeping Sound

You may wish to force the engine to beep upon a command sent from the host. A beeping sound is made to gain a user's attention to an error or other important event.

BEEPONxxxFyyyTnnV (xxx: The desired frequency, 1-20,000Hz; yyy: The desired duration, 1-10,000ms; nn: The desired volume level, 1-20 (lowest-loudest))

Example: Make a 50ms beep at 2,000Hz with volume level set to 20

Enter: `~<SOH>0000#BEEPON2000F50T20V;<ETX>`

Response: `<STX><SOH>0000#BEEPON2000F50T20V<ACK>;<ETX>`

Turn On Good Read LED

You may turn on the external Good Read LED of the engine for a certain period of time with a command sent from the host. Note that the engine **cannot** scan barcodes when it is executing this command. The duration is from 10 to 10000ms.

Command for querying whether the engine supports this feature: LEDONS* or LEDONS&

Returning LEDONS<ACK> indicates the engine supports this feature.

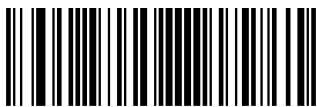
Command for querying the range of possible values for the setting: LEDONS^

Returning LEDONS-2C10-10000D <ACK> indicates the range for the length of time the LED stays lit is 10-10000ms.

Example: Turn on the Good Read LED for 1,000ms

Enter: `~<SOH>0000#LEDONS2C1000D;<ETX>`

Response: `<STX><SOH>0000#LEDONS2C1000D<ACK>;<ETX>`



@SETUPE0
** Exit Setup

Turn On Illumination LED

You may turn on the internal illumination LED on the engine for a certain period of time with a command sent from the host. Note that the engine **cannot** scan barcodes when it is executing this command. The duration is from 10 to 10000ms.

Command for querying whether the engine supports this feature: LEDONI* or LEDONI&
Returning LEDONI<ACK> indicates the engine supports this feature.

Command for querying the range of possible values for the setting: LEDONI^
Returning LEDONI-0C10-10000D <ACK> indicates the range for the length of time the LED stays lit is 10-10000ms.

Example: Turn on the illumination LED for 1,000ms

Enter: ~<SOH>0000#LEDONI0C1000D;<ETX>

Response: <STX><SOH>0000#LEDONI0C1000D<ACK>;<ETX>



@SETUPE1
Enter Setup

Chapter 14 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 Always On, Sense Mode, Decode Session Timeout = 2s**:

1. Input the commands:

```
@ILLSCN2;SCNMOD2;ORTSET2000;
```

2. Generate a batch barcode.

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



@BATCHS
Enable Batch Barcode



@SETUPE0
** Exit Setup

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 Always On, Sense Mode, Decode Session Timeout = 2s**:

1. Input the following commands:

```
@ILLSCN2;SCNMOD2;ORTSET2000;
```

2. Generate a PDF417 batch barcode.





@SETUPE1
Enter Setup

Use Batch Barcode

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



@SETUPE1
Enter Setup



@BATCHS
Enable Batch Barcode



Batch Barcode



@SETUPE0
Exit Setup



@SETUPE0
** Exit Setup

Appendix

Digit Barcodes

0~9



@DIGIT0

0



@DIGIT2

2



@DIGIT4

4



@DIGIT1

1



@DIGIT3

3



@DIGIT5

5



@DIGIT6

6



@DIGIT7

7



@DIGIT8

8



@DIGIT9

9

A~F



@DIGITA

A



@DIGITB

B



@DIGITC

C



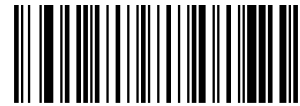
@DIGITD

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 engine is still in the setup mode.



Factory Defaults Table (ST.G01.1)

Parameter	Factory Default	Remark
System Settings		
Barcode Programming	Disabled (Exit Setup)	
Programming Barcode Data	Do not transmit	
Scan Mode	Level Mode	
Decode Session Timeout	3000ms	1-3,600,000ms; 0: Infinite
Image Stabilization Timeout (Sense Mode)	500ms	0-3,000ms
Image Change Trigger Sensitivity	6	
Scanning Interval	0	
Reread Timeout	Disabled, 50ms	1-3,600,000ms
Reread Timeout Reset	Off	
Good Read Delay	Disabled, 500ms	1-3,600,000ms
Scanning Preference	Normal Mode	
Image Flipping	Do Not Flip	
Bad Read Message	Off	
	NG	1-7 characters
Trigger Commands	Disabled	
Start Scanning Command	<SOH> T <EOT>	
Stop Scanning Command	<SOH> P <EOT>	
Illumination	Normal	
External Illumination	Off	
Aiming	On	
Good Read LED	On	
Good Read LED Duration	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	Loud	
Auto Sleep	Disabled	
Time Period from Idle to Sleep	3s	
Default Interface	USB CDC	
RS-232 Interface		

Baud Rate	9600	
Parity Check	None	
Data Bits	8	
Stop Bits	1	
USB Interface		
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
Emulate Keypad with Leading Zero	On	USB HID Keyboard
Function Key Mapping	Disable	USB HID Keyboard
Inter-Keystroke Delay	No Delay	USB HID Keyboard
Caps Lock	Caps Lock OFF, non-Japanese Keyboard	USB HID Keyboard
Convert Case	No Case Conversion	USB HID Keyboard
Emulate Numeric Keypad 1	Off	USB HID Keyboard
Emulate Numeric Keypad 2	Off	USB HID Keyboard
Fast Mode	Off	USB HID Keyboard
Polling Rate	4ms	USB HID Keyboard
Adaptive Wired Communication	On	
Symbologies		
Global Settings		
Surround GS1 AI's with Parentheses	Do Not Surround GS1 AI's with Parentheses	
Code 128		
Code 128	Enabled	
Maximum Length	48	
Minimum Length	1	
Check Character	Do Not Transmit	
FNC1 Output	On	
EAN-8		
EAN-8	Enabled	
Check Character	Transmit	
2-Digit Add-On Code	Disabled	
5-Digit Add-On Code	Disabled	
Add-On Code	Not Required	

Convert EAN-8 to EAN-13	Disabled	
EAN-13		
EAN-13	Enabled	
Check Character	Transmit	
2-Digit Add-On Code	Disabled	
5-Digit Add-On Code	Disabled	
Add-On Code	Not Required	
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	
UPC-E		
UPC-E	Enabled	
Check Character	Transmit	
2-Digit Add-On Code	Disabled	
5-Digit Add-On Code	Disabled	
Add-On Code	Not Required	
Transmit Preamble Character	System Character	
Convert UPC-E to UPC-A	Disabled	
UPC-A		
UPC-A	Enabled	
Check Character	Transmit	
2-Digit Add-On Code	Disabled	
5-Digit Add-On Code	Disabled	
Add-On Code	Not Required	
Transmit Preamble Character	System Character	
Interleaved 2 of 5		
Interleaved 2 of 5	Enabled	
Maximum Length	80	

Minimum Length	6	No less than 4
Check Character Verification	Disabled	
Febraban		
Febraban	Disabled	
Transmit Delay per Character	Disabled	
	70ms	
Transmit Delay per 12 Characters	Disabled	
	500ms	
ITF-14		
ITF-14	Disabled	
ITF-6		
ITF-6	Disabled	
Matrix 2 of 5		
Matrix 2 of 5	Enabled	
Maximum Length	80	
Minimum Length	4	No less than 4
Check Character Verification	Disabled	
Code 39		
Code 39	Enabled	
Maximum Length	48	
Minimum Length	1	
Check Character Verification	Disabled	
Start/Stop Character	Do not transmit	
Code 39 Full ASCII	Disabled	
Code 32 Pharmaceutical (PARAF)	Disabled	
Code 32 Prefix	Disabled	
Code 32 Start/Stop Character	Do not transmit	
Code 32 Check Character	Do not transmit	
Codabar		
Codabar	Enabled	
Maximum Length	60	
Minimum Length	2	
Check Character Verification	Disabled	
Start/Stop Character	Do not transmit	
	ABCD/ABCD	

Code 93		
Code 93	Disabled	
Maximum Length	48	
Minimum Length	1	No less than 1
Check Character Verification	Do Not Transmit Check Character After Verification	
GS1-128 (UCC/EAN-128)		
GS1-128	Enabled	
Maximum Length	48	
Minimum Length	1	
Check Character	Do Not Transmit	
FNC1 Output	On	
GS1 Databar		
GS1 Databar	Disabled	
Application Identifier "01"	Transmit	
Code 11		
Code 11	Disabled	
Maximum Length	48	
Minimum Length	4	No less than 4
Check Character Verification	Disabled	
Check Character	Transmit	
ISBN		
ISBN	Enabled	
Set ISBN Format	ISBN-10	
ISSN		
ISSN	Disabled	
Industrial 25		
Industrial 25	Disabled	
Maximum Length	48	
Minimum Length	6	No less than 4
Check Character Verification	Disabled	
Standard 25		
Standard 25	Disabled	
Maximum Length	48	
Minimum Length	6	No less than 4

Check Character Verification	Disabled	
Plessey		
Plessey	Disabled	
Maximum Length	48	
Minimum Length	4	No less than 4
Check Character Verification	Disabled	
MSI-Plessey		
MSI-Plessey	Disabled	
Maximum Length	48	
Minimum Length	4	No less than 4
Check Character Verification	One Check Character, MOD10	
Check Character	Transmit	
AIM 128		
AIM 128	Disabled	
Maximum Length	48	
Minimum Length	1	
Check Character	Do Not Transmit	
FNC1 Output	On	
PDF417		
PDF417	Enabled	
Maximum Length	6144	
Minimum Length	1	
PDF417 Twin Code	Single PDF417 Only	
Character Encoding	Default Character Encoding	
PDF417 ECI Output	Enabled	
QR Code		
QR Code	Enabled	
Maximum Length	6144	
Minimum Length	1	
QR Twin Code	Single QR Only	
Character Encoding	Default Character Encoding	
QR ECI Output	Enabled	
URL QR	Enabled	
Micro QR Code		
Micro QR	Enabled	

Maximum Length	6144	
Minimum Length	1	
Aztec		
Aztec Code	Disabled	
Maximum Length	6144	
Minimum Length	1	
Character Encoding	Default Character Encoding	
Aztec ECI Output	Enabled	
Data Matrix		
Data Matrix	Enabled	
Maximum Length	6144	
Minimum Length	1	
Data Matrix Twin Code	Single Data Matrix Only	
Rectangular Barcode	Enabled	
Character Encoding	Default Character Encoding	
Data Matrix ECI Output	Enabled	
Data Formatter		
Data Formatter	Disabled	
Non-Match Error Beep	Off	
Data Format Selection	Format_0	
Prefix & Suffix		
All Prefixes/Suffixes	Disabled	
Prefix Sequence	Code ID+ Custom+ AIM ID	
Custom Prefix	Disabled	
AIM ID Prefix	Disabled	
Code ID Prefix	Disabled	
Custom Suffix	Disabled	
Data Packing	Disable Data Packing	
Terminating Character Suffix	Disabled	

AIM ID Table(V2022.6)

Symbology	AIM ID	Possible AIM ID Modifiers (m)
Code 128]C0	
GS1-128 (UCC/EAN-128)]C1	
EAN-8]E4	
EAN-13]E0	
EAN-13 with Addon]E3	
UPC-E]E0	
UPC-E with Addon]E3	
UPC-A]E0	
UPC-A with Addon]E3	
Interleaved 2 of 5]Im	0, 1, 3
ITF-14]Im	1, 3
ITF-6]Im	1, 3
Matrix 2 of 5]X0	
Code 39]Am	0, 1, 3, 4, 5, 7
Codabar]Fm	0, 2, 4
Code 93]G0	
AIM 128]C2	
ISSN]X0	
ISBN]X0,]E0,]E3	
Industrial 25]S0	
Standard 25]Rm	0, 1, 3
Plessey]P0	
Code 11]Hm	0, 1, 3
MSI Plessey]Mm	0, 1
GS1 Databar(RSS)]e0	
PDF417]Lm	0-5
QR Code]Qm	0-6
Aztec]zm	0-9, A-C
Data Matrix]dm	0-6
Micro QR]Qm	0-6

Code ID Table(V1.00.0)

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

Symbology ID Number(V1.00.0)

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, Febraban	008
ITF-14	009
ITF-6	010
Matrix 2 of 5	011
Code 39	013
Codabar	015
Code 93	017
AIM 128	020
ISSN	023
ISBN	024
Industrial25	025
Standard25	026
Plessey	027
Code 11	028
MSI-Plessey	029
GS1 Databar (RSS)	031
PDF417	032
QR Code	033
Aztec	034
Data Matrix	035
Micro QR	043

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)

Keyboard Key References

6E	70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	•	•	•		
01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0F	4B	50	55	5A	5F	64	69
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	4C	51	56	5B	60	65	6A
1E	1F	20	21	22	23	24	25	26	27	28	29	2B				5C	61	66		
2C	2E	2F	30	31	32	33	34	35	36	37	39			53			5D	62	67	6C
3A	3B	3C	3D					3E	3F	38	40	4F	54	59	63	68				

104 Key U.S. Style Keyboard

6E	70	71	72	73	74	75	76	77	78	79	7A	7B	7C	7D	7E	•	•	•		
01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0F	4B	50	55	5A	5F	64	69
10	11	12	13	14	15	16	17	18	19	1A	1B	1C	2B	4C	51	56	5B	60	65	6A
1E	1F	20	21	22	23	24	25	26	27	28	29	1D				5C	61	66		
2C	2D	2E	2F	30	31	32	33	34	35	36	37	39			53		5D	62	67	6C
3A	3B	3C	3D					3E	3F	38	40	4F	54	59	63	68				

105 Key European Style Keyboard

SCANNING MADE SIMPLE

Newland EMEA
+31 (0) 345 87 00 33
info@newland-id.com

Rolweg 25
4104 AV Culemborg
The Netherlands



@NewlandEMEA