



ZBA Inc.

# **ZB-3062**

## **Barcode Scanner with 2D imager**



**ZBA, Inc**  
**94 Old Camplain Road,**  
**Hillsborough, NJ 08844**  
**[www.zbausa.com](http://www.zbausa.com)**



# Table of Contents

- 1 DOCUMENT STATUS..... 6**
- 2 REGULATORY ITEMS ..... 7**
- 3 INTRODUCTION..... 8**
- 3.1 Installation ..... 8**
  - 3.1.1 RS232 ..... 8
  - 3.1.2 USB (HID) ..... 8
  - 3.1.3 USB (Virtual-COM) ..... 8
- 4 DEFAULT SETTINGS..... 9**
- 5 ZB3062 SPECIFICATION ..... 10**
- 5.1 Reading Skills..... 13**
- 6 PROGRAMMING THE ZB-3062..... 13**
- 6.1 Interface Selection..... 14**
  - 6.1.1 RS232 ..... 14
  - 6.1.2 TTL Level 232 Interface ..... 17
  - 6.1.3 USB HID ..... 17
  - 6.1.4 USB Virtual COM ..... 17
- 6.2 Pin Assignments ..... 18**
- 7 SYSTEM CONTROL ..... 19**
- 7.1 Scan..... 19**
- 7.2 Scan ..... 20**
- 7.3 Indication ..... 21**
- 7.4 Thermal and Centering ..... 26**
- 7.5 Decode Search Mode..... 28**
- 7.6 Output Sequence..... 29**
- 7.7 Print Weight ..... 32**
- 7.8 Video Reverse..... 32**
- 7.9 Working Orientation..... 32**
- 8 CODE OPTION..... 33**
- 8.1 Codabar ..... 34**
- 8.2 Code 39..... 36**
- 8.3 Code 32..... 37**
- 8.4 Interleaved 2 of 5..... 38**
- 8.5 Code 93..... 38**
- 8.6 Straight 2 of 5 Industrial ..... 39**
- 8.7 Straight 2 of 5 IATA..... 40**
- 8.8 Matrix 2 of 5 ..... 40**
- 8.9 Code 11..... 41**
- 8.10 Code 128..... 42**



**8.11 Telepen..... 43**

**8.12 UPC-A ..... 44**

**8.13 UPC-E0..... 45**

**8.14 EAN/JAN-13 ..... 47**

**8.15 EAN/JAN-8..... 49**

**8.16 MSI ..... 50**

**8.17 Plessey Code ..... 50**

**8.18 GS1 DataBar Omnidirectional ..... 51**

**8.19 GS1 DataBar Limited ..... 51**

**8.20 GS1 DataBar Expanded ..... 51**

**8.21 PosiCode ..... 52**

**8.22 Codablock F ..... 52**

**8.23 Code 16K..... 53**

**8.24 Code 49..... 53**

**8.25 PDF417 ..... 54**

**8.26 MicroPDF417 ..... 54**

**8.27 EAN•UCC Composite Codes ..... 55**

**8.28 Postal Codes ..... 56**

    8.28.1 Postnet..... 56

    8.28.2 Planet Code..... 56

    8.28.3 Postal Codes..... 57

    8.28.4 China Post ..... 57

    8.28.5 Korea Post..... 58

**8.29 QR Code..... 58**

**8.30 Data Matrix ..... 59**

**8.31 MaxiCode ..... 59**

**8.32 Aztec Code..... 60**

**9 STRING FORMAT ..... 61**

**9.1 Prefix/Suffix..... 61**

**9.2 Data Formatting ..... 64**

**9.3 Data Format Editor Commands..... 65**

**9.4 OCR Programming ..... 67**

**9.5 OCR Fonts ..... 67**

    9.5.1 U.S. Currency Font ..... 68

    9.5.2 MICR E-13B Font ..... 68

    9.5.3 SEMI Font..... 68

**9.6 OCR Templates ..... 69**

**9.7 Utilities ..... 75**

**9.8 Serial Prog. Commands ..... 77**



<b>10</b>	<b><i>TEST CHART</i></b> .....	<b>79</b>
<b>11</b>	<b>INTERFACE ID</b> .....	<b>82</b>
<b>12</b>	<b>PRODUCT CODE ID</b> .....	<b>83</b>
<b>13</b>	<b>ASCII CODE TABLE</b> .....	<b>85</b>
<b>14</b>	<b>OCR PROG. CHART</b> .....	<b>86</b>
<b>15</b>	<b>PROGRAM CHART</b> .....	<b>87</b>



**Disclaimer:**

No part of this document may be copied or reproduced in any form or by any means, or transferred to any third party, without the prior written consent of an authorized representative of ZBA, Inc. (“ZBA”). The information in this document is subject to change without notice. ZBA assumes no responsibility for any errors or omissions that may appear in this document, and disclaims responsibility for any consequences resulting from the use of the information set forth herein. ZBA makes no commitments to update or to keep current information contained in this document. The products listed in this document are not suitable for use in applications such as, but not limited to, aircraft control systems, aerospace equipment, submarine cables, nuclear reactor control systems and life support systems. Moreover, ZBA does not recommend or approve the use of any of its products in life support devices or systems or in any application where failure could result in injury or death. If a customer wishes to use ZBA products in applications not intended by ZBA, said customer must contact an authorized ZBA representative to determine ZBA’s willingness to support a given application. The information set forth in this document does not convey any license under the copyrights, patent rights, trademarks or other intellectual property rights claimed and owned by owned by ZBA. The information set forth in this document is considered to be “Proprietary” and “Confidential” property

ALL PRODUCTS SOLD BY ZBA ARE COVERED BY THE PROVISIONS APPEARING IN ZBA’S TERMS AND CONDITIONS OF SALE ONLY, INCLUDING THE LIMITATIONS OF LIABILITY, WARRANTY AND INFRINGEMENT PROVISIONS. ZBA MAKES NO WARRANTIES OF ANY KIND, EXPRESS, STATUTORY, IMPLIED OR OTHERWISE, REGARDING INFORMATION SET FORTH HEREIN OR REGARDING THE FREEDOM OF THE DESCRIBED PRODUCTS FROM INTELLECTUAL PROPERTY INFRINGEMENT, AND EXPRESSLY DISCLAIMS ANY SUCH WARRANTIES INCLUDING WITHOUT LIMITATION ANY EXPRESS, STATUTORY OR IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.



# 1 Document Status

Date	Comments
October 30 2009	Initial Draft
November 21 2013	Correct Grammar in Data editing
June 3 2015	Increase size of BC on Programming Chart for ease of printing.

To make a request for change, correction, additions or information on references, please contact:

ZBA, Inc.  
94 Old Camplain Road  
Hillsborough, NJ 08844  
Phone: 908-359-2070  
Fax 908-595-0909  
E-mail: [techsupport@zbausa.com](mailto:techsupport@zbausa.com)  
Website : [www.zbausa.com](http://www.zbausa.com)



## 2 Regulatory items

### NOTICE:

<p><b>FCC:</b> This device complies with Part 15 of the FCC Rules, (1) This device may not cause harmful interface, and (2) This device must accept any interface received, including interface that may cause undesirable operation.</p> <p><b>CE:</b> This product conforms to the following standards EMC:EN55022:2006+A1:2007, class B EN55024:1998+A1:2001+A2:2003</p> <p><b>BSMI :</b> This device is compliant with requirement of BSMI and granted ID No. R3A078</p>
--

These limits are designed to provide a reasonable protection against harmful interface when the equipment is operated under a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interface to radio communications. Operation of this equipment in a residential area is likely to cause harmful interface in which case the user will be required to correct the interface at his or her own expenses.

*Note: All brands and trademarks shall belong to their respective owner.*

*Note: Specification is subject to changes without notice*

### Using the ZB3062

The ZB3062 automatically scan barcode at a distance.

Simply aim and pull the trigger. Code scanning is performed along the center of the light bar emitted from the reading window. This bar must cover the entire code.

Successful scanning shall be obtained by tilting the scanner with respect to the barcode to avoid direct reflections that impair the reading performance, especially for 2D barcode.

### Recommended Steps

When the required settings have been configured, all settings are stored in non- volatile memory of scanner after reading configuration barcode labels. Recommended steps are as follows.

- 1) Set right host interface for your scanner.
- 2) Set interface to optimize protocol of scanner with your host in interface section.
- 3) Set system control of scanner, such as specific adjustments double confirm, power saving, indicator and scanning mode which you prefer usage in system control section.
- 4) Set code options of scanner for your usage in code option section. You must make sure to enable the symbology first, then Min./Max. code length, code ID checksum and truncate digits are also converted.
- 5) Set string format of the scanner, such as preamble, postamble Prefix, suffix, code ID and code name transmission for your application in string format section.

*Note: If the scanner is still not working properly. Please contact your dealer for further information.*



## 3 Introduction

### 3.1 Installation

#### 3.1.1 **RS232**

- 1) Disconnect power to the terminal/computer.
- 2) Connect the external power supply (DC adapter) to the serial interface cable of the scanner.
- 3) Plug the serial connector into the serial port on the back of your computer/terminal. Tighten the two screws to secure the connector to the port.
- 4) Plug the power pack into power source.
- 5) Once the scanner has been fully connected, turn the terminal / computer power back on.

#### 3.1.2 **USB (HID)**

- 1) Connect the USB cable between scanner and PC.
- 2) Windows will automatically detect the USB device.

#### 3.1.3 **USB (Virtual-COM)**

- 1) Install driver, USB Serial Emulation Driver, from CD-ROM, The driver will use the next available COM port number
- 2) Connect the USB cable between scanner and PC.
- 3) Windows will automatically detect the USB device.

**Note:** If any of the above operation is incorrect, turn off the power immediately and check any improper connections. Go through all above steps again.



## 4 Default settings

For each barcode shown as below:

V = Enabled as default setting

- = Not supported

Empty space = Not enabled at default setting

Code Type	Read Enable	Checksum Verification Enable	Checksum Transmission Enable	Code ID
UPC-A	V	V	V	c
UPC-E0	V	V	V	E
UPC-E1				E
EAN-13	V	V	V	d
EAN-8	V	V	V	D
Code-39	V			b
Interleaved 2 of 5	V			e
Industrial 2 of 5		-	-	e
IATA 2 of 5				f
Matrix 2 of 5				m
Codabar	V			a
Code-128	V			j
Code-93	V			i
Code-11		V (2 digits)		h
MSI				g
Plessey				n
GS1 DataBar Omnidirectional	V	-	-	y
GS1 DataBar Limited	V	-	-	y
GS1 DataBar Expanded	V	-	-	y
PosiCode A&B	V			W
Codablock F				q
Code 16K				o
Code 49	V			I
PDF 417	V			r
MicroPDF 417	V			R
EAN UCC Composite				y
Postnet				P
Planet Code				L
British Post				B



Canada Post		C
Kix Post (Netherlands)		K
Australian Post		A
Japanese Post		J
China Post		Q
Korea Post		?
QR Code	V	O
Data Matrix	V	w
MaxiCode	V	x
Aztec Code	V	z
Aztec Runes		z

## 5 ZB3062 Specification

Specification	Model ZB3062
<b>Operational</b>	
Light Source	626 nm ± 30 nm Visible Red LED
Aiming Source	526 nm ± 30 nm Visible Green LED
Optical System	752 x 480 CMOS sensor
Motion Tolerance	4 inches / sec
Depth of Scan Field (PCS=90%,10 mils)	1D : 37 ~ 175 mm ( CODE 39,) 2D : 62 ~ 137 mm ( QR code )
Resolution (min)	1D : 0.127 mm (5 mils) 2D : 0.21 mm ( 8 mils)
Print Contrast	30% or more
Focal Point	114 mm ( 4.5 inches ) from lens plate
Scanning Angle	Pitch: ± 40° Skew: ±40°
Rotational Sensitivity	360°
Decode Capability	Auto-discriminates all standard barcodes; Other symbologies can be ordered optionally
Beeper Operation	Volume x 3 and Frequency x 3 or no beep
Indicator	Blue led and adjustable beeper
<b>Mechanical</b>	
Length	165.1 mm
Width-handle	32 mm
Width-head	72.8 mm
Depth-handle	54.7 mm
Depth-head	82.5 mm
Weight	146



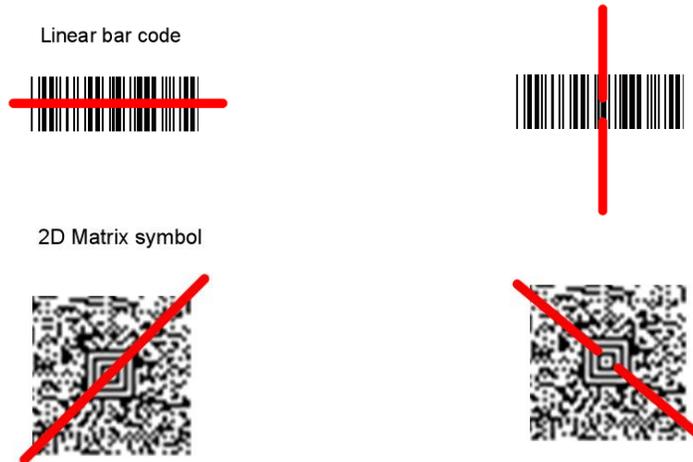
Cable	Straight 2.0 m	
Connector type	RJ-45 phone jack connector	
Case material	ABS	
Cushion material	Rubber	
<b>Electrical</b>		
Input Voltage	3.5 ~ 5.5 VDC	
Power	RS232	USB HID
Operating	Max 300mA @ 5V	
Standby	Max 55mA @ 5V	Max 50mA @ 5V
Low Power Mode	Max 30mA @ 5V	*
Agency listing	EMI: FCC, CE, BSMI Safety: UL, BSMI, CB	
<b>Environmental</b>		
Operating Temperature	0°C to 50°C (32°F to 122°F)	
Storage Temperature	-20°C to 60°C (-4°F to 140°F)	
Humidity	Up to 95% relative humidity, non-condensing @ 50°C	
Light Level	Dark to 100,000 Lux.	
Shock	24 drops from 1.5 m to concrete	
Resistance	IP 42	
Contaminants	Seals to resist airborne particulate contaminants	
Ventilation	None required	
<b>Programming</b>		
Programming method	1.Executing DOS Command by RS-232 or USB Virtual COM. 2. A scanner reads the programming codes.	
Program upgrade	Enabled built-in flash memory	



Programmable characteristics	Code type selection, check digit selection Decoding option Decoding option Transmitted character delay, Header selection, trailer selection, message suffix, good read beep tone and volume, scanner trigger selection Keyboard emulation type (intermessage delay, keyboard type and keyboard language) Serial interface type (ACK/NAK, Xon/Xoff, RTS/CTS, good read LED control, start/stop bits)
------------------------------	--

## 5.1 Reading Skills

The engine has a view finder that projects a bright red or green aiming beam that corresponds to the engine's horizontal field of view. The aiming beam should be centered over the bar code, but it can be positioned in any direction for a good read.



The aiming beam is smaller when the engine is closer to the code and larger when it is farther from the code. Symbologies with smaller bars or elements (mil size) should be read closer to the unit. Symbologies with larger bars or elements (mil size) should be read farther from the unit. To read single or multiple symbols Linear bar code 2D Matrix symbol (on a page or on an object), hold the engine at an appropriate distance from the target, send a trigger command, and center the aiming beam on the symbol. If the code being scanned is highly reflective (e.g., laminated), it may be necessary to tilt the code +5° to prevent unwanted reflection.

## 6 Programming the ZB-3062

### ZB-3062 Series Scanner

To program the ZB-3062, you must scan a series of programming barcodes in the correct order. On the last page of this manual you will see a table of alphanumeric barcodes, which are used to program the various options as per your application needs.

#### To program each option, you must:

1. Scan the **Program** barcode on the parameter setting part.
2. Enter the option mode by scanning the **option barcodes** (also on the Parameter setting part).
3. To the right of the option barcode, the necessary alphanumeric inputs are listed. Scan these alphanumeric entries from the **Program Chart**. To confirm above steps, you must scan the **Save** barcode.

## 6.1 Interface Selection

This decoder built-in scanner comes in one model and supports interfaces such as RS232 serial, USB virtual COM and USB HID. In most of the cases, simply selecting an appropriate cable and configure the proper interface by following interface selection.

**Interface selection:** You can change factory interface default (USB Virtual COM ) for other type interface. By plugging different cables, setting right interface, then scan the interface barcode, power cycle the scanner will be changed to another interface. However, you must make sure which cable you need.

### 6.1.1 RS232





**RS-232 (continued)**

Default Setting: Baud Rate: 115200 bps; Data Format 8 data bits, no parity bit, 1 stop bit

**Baud Rate**



300



9600



600



19200



1200



38400



2400



57600



4800



\*115200

**Data Bits** sets the word length at 7 or 8 bits of data per character. If an application requires only ASCII Hex characters 0 through 7F decimal (text, digits, and punctuation), select 7 data bits. For applications which require use of the full ASCII set, select 8 data bits per character. Default = 8.

**Stop Bits** sets the stop bits at 1 or 2. Default = 1.

**Parity** provides a means of checking character bit patterns for validity. Default = None.

**Data Format**



7 Data, 1 Stop, Parity Even



7 Data, 2 Stop, Parity Odd



7 Data, 1 Stop, Parity None



8 Data, 1 Stop, Parity Even



7 Data, 1 Stop, Parity Odd



\*8 Data, 1 Stop, Parity None



7 Data, 2 Stop, Parity Even



8 Data, 1 Stop, Parity Odd



7 Data, 2 Stop, Parity None



**RS-232 continued**

**RTS/CTS**-If the scanner wants to send the barcode data to host computer, it will issue the RTS signal first, wait for the CTS signal from the host computer, and then perform the normal data communication. If there is no replied CTS signal from the host computer after the timeout (Response Delay) duration, the scanner halts transmission until it detects another active CTS signal.

**Xon/Xoff**- When the host computer is unable to accept data, it sends a Xoff code to inform the scanner to suspend data transmission, and Xon to continue.

**ACK/NAK**- When the ACK/NAK protocol is used, the scanner waits for an ACK (acknowledge) or (not acknowledge) from the host computer after data transmission, and will resend in response to a NAK.

**Response Delay**

The unit stays awake to receive data until the RS-232 Receiver Time-Out expires. A trigger command resets the time-out. When an RS-232 receiver is sleeping, a character may be sent to wake up the receiver and reset the time-out. A transaction on the CTS line will also wake up the receiver. The receiver takes 300 milliseconds to completely wake up. Change the RS-232 receiver time-out by scanning the bar code below, then scanning digits from the inside back cover of this manual, then scanning **Save**. The range is 0 to 300 seconds. Default = 0 seconds (no time-out - always on).

**Handshaking**



RTS/CTS On



\*RTS/CTS Off



XON/XOFF On



\*XON/OFF Off



ACK/NAK On



\*ACK/NAK Off



Response Delay

### 6.1.2 **TTL Level 232 Interface**

The ZB-3062 provides a TTL level serial 232 communication link. TTL Level 232 Inverted is the “normal” mode that allows direct communication from the ZB-3062 to a standard PC RS-232 serial port with the use of cable. Non-Inverted serial TTL communication is provided to allow direct connection to a TTL UART port, or to an RS-232 driver integrated circuit.

Note that if the TTL Level 232 Non-Inverted bar code is scanned, the ZB-3062 will no longer communicate with a standard PC RS-232 port. TTL Level 232 Inverted with Polarity Override allows direct communication from the ZB-3062 to a standard PC RS-232 serial port with the use of cable.

This setting also allows you to override the polarity externally.

Default = TTL Level 232 Inverted with Polarity Override.



TTL Level 232 Inverted



TTL Level 232 Non-Inverted



\*TTL Level 232 Inverted  
with Polarity Override

### 6.1.3 **USB HID**

Scan the following code to program the ZB-3062 for USB HID bar code imagers.



USB HID (PC)



USB HID (MAC)

### 6.1.4 **USB Virtual COM**

Scan the following code to program the ZB-3062 to emulate a regular RS-232-based COM port. If you are using a PC, you will need to download a driver from CD-ROM, The driver will use the next available COM port number



USB Virtual COM

#### **CTS/RTS Emulation & ACK/NAK Mode**



CTS/RTS On



\*CTS/RTS Off

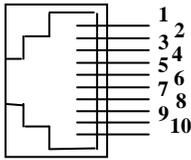


ACK/NAK On



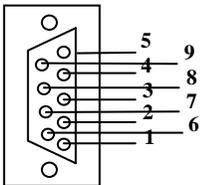
\*ACK/NAK Off

## 6.2 Pin Assignments



10-pin RJ-45 Connector to Scanner Side

Pin	RS-232	USB
1	NC	NC
2	VCC	VCC
3	TXD	TXD
4	NC	NC
5	NC	NC
6	CTS	D+
7	RXD	D-
8	RTS	RTS
9	GND	GND
10	GND	GND



4

Connector to Host Side:

Pin	Definition	USB
1	NC	+
2	TXD	D+
3	RXD	D-
4	NC	-
5	GND	*
6	NC	*
7	CTS	*
8	RTS	*
9	VCC	*



## 7 System Control

### 7.1 Scan

#### **Scanning output:**

You can set the image size to a VGA resolution to accommodate older applications that require a smaller image size. When Image VGA is set to On, the resultant image is 640x480 pixels. When Image VGA is Off, the image is 752x480 pixels.

#### **Scanning mode:**

##### ***Manual/Serial Trigger Mode***

You can activate the imager either by providing an external hardware trigger, or using a serial trigger command. When in manual trigger mode, the imager scans until a bar code is read, or until the hardware trigger is released.

When in serial mode, the imager scans until a bar code has been read or until the deactivate command is sent. In serial mode, the imager can also be set to turn itself off after a specified time has elapsed (see [Read Time-Out](#), which follows).

##### ***Read Time-Out***

Use this selection to set a time-out (in milliseconds) of the imager's trigger when using serial commands to trigger the imager, or if the imager is in manual trigger mode. Once the imager has timed out, you can activate the imager either by pressing the trigger or using a serial trigger command. After scanning the **Read Time-Out** bar code, set the time-out duration (from 0-300,000 milliseconds) by scanning digits from the inside back cover, then scanning **Save**. Default = 0 (infinite, or no time-out).

##### ***Manual Trigger, Low Power*** (For RS-232 only)

The imager powers down until the trigger is pulled. When the trigger is pulled, the imager powers up and operates until there is no triggering for the time set with the **Low Power Time-Out** bar code below. There is a delay of up to one second in operation when the imager is first triggered, but there is no delay when operating in low power time-out mode.

##### ***Low Power Time-Out Timer***

Scan the Low Power Time-Out bar code to change the time-out duration (in seconds). Then scan the time-out duration (from 0-300 seconds) from the inside back cover, and **Save**. Default = 120 seconds.

If the unit remains idle during the low power time-out interval, the unit goes into low power mode. Whenever the trigger is enabled, the low power time-out timer is reset.

##### ***Presentation Mode***

This programs the imager to work in Presentation mode. The LEDs are either off or at the lowest power for ambient conditions until a bar code is presented to the imager. Then the LEDs turn on automatically to read the code. Presentation Mode uses ambient light to detect the bar codes. If the light level in the room is not high enough, Presentation Mode may not work properly.

## 7.2 Scan

### Scanning Output



VGA Off



\*VGA On

### Scanning Mode



\*Manual/Serial Trigger



Read Time-Out



Manual Trigger, Low Power



Presentation Mode



Low Power Time-Out

### *Presentation LED Behavior after Decode*

When an imager is in presentation mode, the LEDs remain on and continue scanning for a short time after a bar code is decoded. If you wish to turn the LEDs off immediately after a bar code is decoded, scan the **LEDs Off** bar code, below. Default = LEDs On.

### *Presentation LED Time-Out*

When using [Presentation LED Behavior after Decode](#) (above), you may want to set the time the LEDs remain off after a decode. To set the duration of this delay, scan the bar code below, then set the time-out by scanning digits (0 - 9,999 ms) , then scanning **Save**. Once the unit has completed this time-out, it will immediately resume scanning.

### *Presentation Sensitivity*

Presentation Sensitivity is a numeric range that increases or decreases the imager's reaction time to bar code presentation. To set the sensitivity, scan the **Sensitivity** bar code, then scan the degree of sensitivity (from 0-20) from the inside back cover, and **Save**. 0 is the most sensitive setting, and 20 is the least sensitive. Default = 1.

### *Hands Free Time-Out*

The Presentation Modes is referred to as “hands free” modes. If a trigger command is sent, or the hardware trigger is pulled when using a hands free mode, the imager changes to manual trigger mode. You can set the time the imager should remain in manual trigger mode by setting the Hands Free Time-Out. Once the time-out value is reached, (if there have been no further trigger pulls) the imager reverts to the original hands free mode. Scan the **Hands Free Time-Out** bar code, then scan the time-out duration (from 0-300,000 milliseconds) from the inside back cover, and **Save**. Default = 5,000 ms.

### *Presentation*



\*LEDs On



LEDs Off

### *LED Time-Out*



LED Time-Out Duration



Sensitivity



Hands Free Time-Out

## 7.3 Indication

### *Beeper*

The beeper may be programmed **On** or **Off** in response to a good read. Turning this option off, only turns off the beeper response to a good read indication. All error and menu beeps are still audible.

### *Volume*

The beeper volume codes modify the volume of the beep the imager emits on a good read.

### *Pitch*

The beeper pitch codes modify the pitch of the beep the imager emits on a good read.

### *Duration*

The beeper duration codes modify the length of the beep the imager emits on a good read.

### *Beeper*



\*On



Off

### *Volume*



Low



Medium



High



Off

### *Pitch*



Low(1600 Hz)



\*Medium (3250 Hz)



High (4200 Hz)

### *Duration*



\*Normal Beep



Short Beep



### Indicator LED

The LED indicator can be programmed **On** or **Off** in response to a good read.

#### Polarity

LED Good Read Polarity sets the idle and active states of the LED signal. When set to **Active High**, the LEDs turn on when the signal shifts from low to high. When set to **Active Low**, the LEDs turn on when the signal shifts from high to low. If the LED is enabled for a good read, the polarity change takes effect after the next successful decode or a power cycle. If the LED is disabled, a polarity change only takes effect after a power cycle.

**Note:** LED Good Read Polarity sets the LED signal voltage level to either high (Active Low) or low (Active High). This will affect LED-Good Read setting. If the LED has been disabled, make sure it is not turned on accidentally by a polarity change.

#### LED



\*On



Off

#### Polarity



Active High



\*Active Low

### Number of Beeps

The number of beeps of a good read can be programmed from 1 - 9. The same number of beeps will be applied to the beeper and LED in response to a good read. For example, if you program this option to have five beeps, there will be five beeps and five LED flashes in response to a good read. The beeps and LED flashes are in sync with one another. To change the number of beeps, scan the bar code below and then scan a digit (1-9) bar code and the **Save** bar code.

#### Beep Polarity

Beep Polarity sets the idle and active states of the beeper signal. When set to **Active High**, the beeper sounds when the signal shifts from low to high. When set to **Active Low**, the beeper sounds when the signal shifts from high to low.

If the beeper is enabled for a good read, the polarity change takes effect after the next successful decode or a power cycle. If the beeper is disabled, a polarity change only takes effect after a power cycle.

**Note:** Beep Polarity sets the beeper signal voltage level to either high (Active Low) or low (Active High). This will affect Beeper-Good Read settings. If the beeper has been disabled, make sure it is not turned on accidentally by a polarity change.

#### Good Read Delay

This sets the minimum amount of time before the imager can read another bar code. Default = No Delay.

#### User-Specified Good Read Delay

If you want to set your own length for the good read delay, scan the bar code below, then set the delay (from 0-30,000 milliseconds) by scanning digits from the inside back cover, then scanning **Save**.

#### Number of Beep



Number of Pulses

#### Beep Polarity



Active High



\*Active Low

#### Good Read Delay



\*No Delay



Short Delay (500 ms)



Medium Delay (1,000 ms)



Long Delay (1,500 ms)

**User-Specified Good Read Delay**



User-Specified Good Read Delay

**Reread Delay**

This sets the time period before the imager can read the same bar code a second time. Setting a reread delay protects against accidental rereads of the same barcode. Longer delays are effective in minimizing accidental rereads at POS (point of sale). Use shorter delays in applications where repetitive bar code scanning is required. Reread Delay only works when in Presentation Mode

**User-Specified Reread Delay**

If you want to set your own length for the reread delay, scan the bar code below, then set the delay (from 0-30,000 milliseconds) by scanning digits from the inside back cover, then scanning **Save**.

**Reread Delay**



Short (500 ms)



\*Medium (750ms)



Long (1000 ms)



Extra Long (2000 ms)

**User-Specified Reread Delay**



User-Specified Reread Delay

## Indication (continued)

### LED Power Level

This selection allows you to adjust LED and aimer brightness. **Off** is used when no illumination is needed. **Low** is used if low illumination is sufficient. **High** is the brightest setting. If you have an aimer delay programmed, the aimer will be at 100% power during the delay, regardless of the LED Power Level.

**Note:** If you scan the **Off** bar code, both the aimer and illumination lights turn off, making it difficult to scan bar codes in low light. To turn the LED Power Level back on, move to a brightly lit area and scan either the **Low** or the **High** bar code below.

### Illumination Lights

If you want the illumination lights on while reading a bar code, scan the **Lights On** bar code, below. However, if you want to turn just the lights off, scan the **Lights Off** bar code.

**Note:** This setting does not affect the aimer light. The aiming light can be set using [Aimer Modes](#).

### Imager Time-Out

Imager Time-Out powers down the imager after the unit has been idle for the specified time. To prevent the imager from powering down, set this time-out to 0. Scan the bar code below, then set the time-out by scanning digits (from 0 -999,999 ms) from the inside back cover, then scanning **Save**. Default = 1 ms.

### LED Power Level



Off



Low (50%)



\*High (100%)

### Illumination Light



\*Lights On



Lights Off

### Imager Time-Out



Imager Time-Out

## Indication

### Aimer Delay

The aimer delay allows a delay time for the operator to aim the imager before the picture is taken. Use these codes to set the time between when the trigger is activated and when the picture is taken. During the delay time, the aiming light will appear, but the LEDs won't turn on until the delay time is over.

### User-Specified Aimer Delay

If you want to set your own length for the duration of the delay, scan the bar code below, then set the time-out by scanning digits (0 - 4,000 ms) from the back cover of this manual, then scan **Save**.

### Aimer Modes

**Interlaced**, the illumination and aiming timing is automatically synchronized to the imager exposure period by the Image Engine. The engine turns illumination on while the image is being exposed, and it turns the aiming off at all other times. The interlaced mode provides the lowest overall current draw and is recommended for most applications. It also provides the brightest aimer in most applications. The Image Engine software automatically maintains an approximate 25% aimer duty cycle, even when the imager exposure time is at its maximum in dark operating environments.

**Concurrent** is provided for backwards compatibility with the 4X00 Image Engine series, and is not recommended for most applications. In concurrent mode, the illumination LEDs are on continuously, while the aimer LEDs turn off during the imager exposure period, and on while the imager is not exposing. Concurrent mode is used to eliminate any flicker of the illumination LEDs that may be objectionable to the user, especially when running the



engine at 12 MHz. The illumination LED current is reduced compared to interlaced mode to limit engine peak current. The image engine software automatically maintains an approximate 25% aimer duty cycle, even when the imager exposure time is at its maximum in dark operating environments. Concurrent mode provides the brightest appearance of the illumination LEDs of any of the imager operating modes. This mode may be useful for applications when an operator is using the illumination LEDs for aiming, such as in fixed mount, kiosk, or auto trigger applications. Select **Off** if you don't want to use either aimer mode.

*Aimer Delay*



200Milliseconds



400Milliseconds



\*Off ( no delay)



Delay Duration

*Aimer Modes*



Concurrent



\*Interlaced



Off

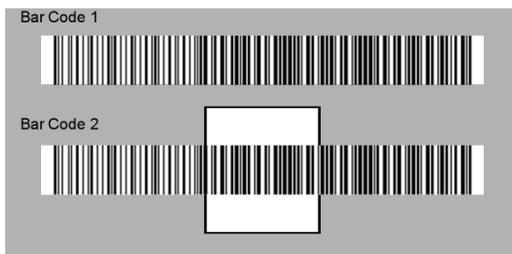
## 7.4 Thermal and Centering

### Thermal Considerations

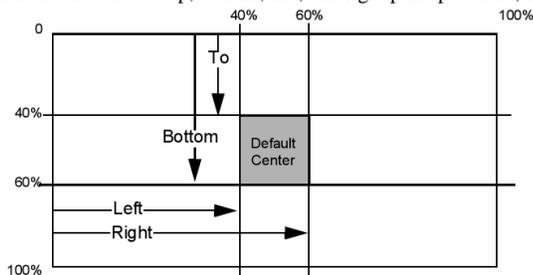
Care must be taken when designing the Image Engine into any system. Internal heating of the Image Engine can occur in high duty cycle scanning applications in several ways. The high visibility aimer dissipates a significant amount of power as heat. The illumination and aiming LEDs also release heat, and are a major contributor to thermal increases in high use or in presentation mode. An increase in temperature around an Image Engine can cause noise levels on the imager, degrading image quality. The thermal rise can also affect the laser diode. In a continuous scanning or high use environment, the Image Engine temperature can rise 15° to 20°C. Under high ambient temperature conditions, the laser diode is at risk of thermal breakdown and possible failure. The image quality and decode performance will also degrade. The Power Control PWM can be used to reduce the effect of the Illumination LEDs on thermal rise, however, this also reduces the intensity of the illumination. Reducing the intensity of the illumination reduces total power used but can also reduce the depth of field in low light environments.

### Centering

Use Centering to narrow the imager's field of view to make sure the imager reads only those bar codes intended by the user. For instance, if multiple codes are placed closely together, centering will insure that only the desired codes are read. (Centering can be used in conjunction with [Aimer Delay](#), for the most error-free operation in applications where multiple codes are spaced closely together. Using the Aimer Delay and Centering features, the imager can emulate the operation of older systems, such as linear laser bar code imagers. In the example below, the gray area is the full imager field of view and the white area is the centering window. Bar Code 1 will not be read, while Bar Code 2 will be.



The default centering window is a 128x96 pixel area (640x480 default image size) in the center of the imager's field of view. The following diagram illustrates the default top, bottom, left, and right pixel positions, measured from the top and the left side of the imager's field of view.



## Thermal and Centering

If a bar code is not within the predefined window, it will not be decoded or output by the imager. If centering is turned on by scanning **Centering On**, the imager only reads codes that intersect the centering window you specify using the **Top, Bottom, Left, or Right** bar codes. Scan **Centering On**, then scan one of the following bar codes to change the top, bottom, left, or right of the centering window. Then scan the percent you want to shift the centering window using digits on the inside back cover of this manual. Scan **Save**. Default Centering = 40% for Top and Left, 60% for Bottom and Right.



Centering On



\*Centering Off



Top of Centering Window



Left of Centering Window



Bottom of Centering Window



Right of Centering Window

## 7.5 Decode Search Mode

There are three selectable decode (scanning) modes:

**Full Omnidirectional** - Searches for bar code features beginning at the center of an image, and searches to the image's limits. This mode reads all symbologies, in any orientation. The Full Omni-directional search is very thorough which may slow performance time.

**Quick Omnidirectional** - This is an abbreviated search for bar code features around the center region of an image. This mode quickly reads all symbologies in any orientation. The Quick Omnidirectional mode may miss some off-center symbols, as well as larger Data Matrix and QR Code.

**Advanced Linear Decoding** - Performs quick horizontal linear scans in a center band of the image. This mode does quickly read linear and stacked bar codes. Advanced Linear Decoding cannot read 2D, OCR, or Postal symbols.



Full Omni-directional

**Note:** This search mode is the default setting for the 2D ZB-3062 Engines.



Quick Omni-directional



Advanced Linear Decoding

**Note:** This search mode is the default setting for the point-and-shoot ZB-3062

## 7.6 Output Sequence

### Require Output Sequence

When turned off, the bar code data will be output to the host as the Imager decodes it. When turned on, all output data must conform to an edited sequence or the Imager will not transmit the output data to the host device.

**Note:** This selection is unavailable when the Multiple Symbols Selection is turned on.

### Output Sequence Editor

This programming selection allows you to program the Imager to output data (when scanning more than one symbol) in whatever order your application requires, regardless of the order in which the bar codes are scanned. Reading the **Default Sequence** symbol programs the Imager to the Universal values, shown below. These are the defaults. Be **certain** you want to delete or clear all formats before you read the **Default Sequence** symbol.

**Note:** To make Output Sequence Editor selections, you'll need to know the code I.D., code length, and character matches your application requires. Use the Alphanumeric symbols (inside back cover) to read these options.

**Note:** You must hold the trigger while reading each bar code in the sequence.

To Add an Output Sequence

1. Scan the **Enter Sequence** symbol

2. **Code I.D.**

On the **Product Code ID**, find the symbology to which you want to apply the output sequence format. Locate the Hex value for that symbology and scan the 2 digit hex value from the Programming Chart (inside back cover).

3. **Length**

Specify what length (up to 9999 characters) of data output will be acceptable for this symbology. Scan the four digit data length from the Programming Chart. (Note: 50 characters is entered as 0050. 9999 is a universal number, indicating all lengths.) When calculating the length, you must count any programmed prefixes, suffixes, or formatted characters as part of the length (unless using 9999).

4. **Character Match Sequences**

On the ASCII Chart, find the Hex value that represents the character(s) you want to match. Use the Programming Chart to read the alphanumeric combination that represents the ASCII characters. (99 is the Universal number, indicating all characters.)

5. **End Output Sequence Editor**

Scan **F F** to enter an Output Sequence for an additional symbology, or **Save** to save your entries. Other Programming Selections

•**Discard**

This exits without saving any Output Sequence changes.

### Output Sequence

#### Output Sequence Example

In this example, you are scanning Code 93, Code 128, and Code 39 bar codes, but you want the imager to output Code 39 1st, Code 128 2nd, and Code 93 3rd, as shown below.

**Note:** Code 93 must be enabled to use this example.



A - Code 39



B - Code 128



C - Code 93

You would set up the sequence editor with the following command line:



**SEQBLK62999941FF6A999942FF69999943FF**

The breakdown of the command line is shown below:

SEQBLK sequence editor start command 62 code identifier for **Code 39** 9999 code length that must match for Code 39, 9999 = all lengths 41 start character match for Code 39,

41h = "A" FF termination string for first code

6A code identifier for **Code 128** 9999 code length that must match for Code 128,

9999 = all lengths 42 start character match for Code 128,

42h = "B" FF termination string for second code

69 code identifier for **Code 93**

9999 code length that must match for Code 93,

9999 = all lengths 43 start character match for Code 93,

43h = "C"

FF termination string for third code To program the previous example using specific lengths, you would have to count any programmed prefixes, suffixes, or formatted characters as part of the length. **SEQBLK62001241FF6A001342FF69001243FF**

The breakdown of the command line is shown below:

SEQBLK sequence editor start command

62 code identifier for **Code 39**

0012 A - Code 39 sample length (11) plus CR suffix (1) = 12

41 start character match for Code 39, 41h = "A"

FF termination string for first code

6A code identifier for **Code 128**

0013 B - Code 128 sample length (12) plus CR suffix (1) = 13

42 start character match for Code 128, 42h = "B"

FF termination string for second code

69 code identifier for **Code 93**

0012 C - Code 93 sample length (11) plus CR suffix (1) = 12

43 start character match for Code 93, 43h = "C"

FF termination string for third code

## Output Sequence

### Require Output Sequence

When an output sequence is **Required**, all output data must conform to an edited sequence or the imager will not transmit the output data to the host device. When it's **On/Not Required**, the imager will attempt to get the output data to conform to an edited sequence, but if it cannot, the imager transmits all output data to the host device as is. When the output sequence is **Off**, the bar code data is output to the host as the imager decodes it.

**Note:** This selection is unavailable when the Multiple Symbols Selection is turned on.

### Output Sequence Editor



Enter Sequence



Default Sequence

### Require Output Sequence



Required



On/Not Required



\*Off

### Multiple Symbols

When this programming selection is turned **On**, it allows you to read multiple symbols when the trigger is activated. If you press and hold the trigger, aiming the Imager at a series of symbols, it reads unique symbols once, beeping (if turned on) for each read. The imager attempts to find and decode new symbols as long as the trigger is activated. When this programming selection is turned **Off**, the Imager will only read the symbol closest to the aiming beam.

### No Read

With No Read turned **On**, the Imager notifies you if a code cannot be read. If using a Quick\*View Scan Data Window, an "NR" appears when a code cannot be read. If No Read is turned **Off**, the "NR" will not appear.

### Multiple Symbols

**Note:** This feature does not work when the Imager is in Low Power mode.



On



\*Off

### No Read



On



\*Off

## 7.7 Print Weight

Print Weight is used to adjust the way the imager reads Matrix symbols. If a imager will be seeing consistently heavily printed matrix symbols, then a print weight of 6 may improve the reading performance. For consistently light printing, a print weight of 2 may help. After scanning the **Set Print Weight** bar code, set the print weight (from 1-7) by scanning digits from the inside back cover, then scanning **Save**. Default = 4.

### Print Weight



Set Print Weight



\*Default

## 7.8 Video Reverse

Video Reverse is used to allow the imager to read bar codes that are inverted. The “Off” bar code below is an example of this type of bar code. If additional menuing is required, Video

Reverse must be disabled to read the menu bar codes and then re-enabled after menuing is completed.

Note: Images downloaded from the unit will not be reversed. This is a setting for decoding only.

### Video Reverse



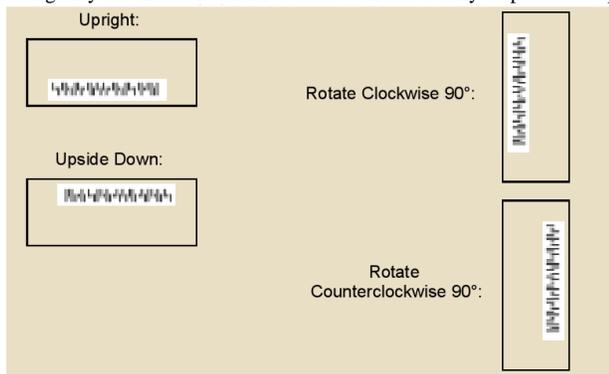
On



\* Off

## 7.9 Working Orientation

Some bar codes are direction-sensitive. For example, Kix codes and OCR can misread when scanned sideways or upside down. Use the working orientation settings if your direction-sensitive codes will not usually be presented upright to the scanner. Default = Upright.





**Working Orientation**



## 8 Code Option

if you want to decode all the symbologies allowable for your imager, scan the **All Symbologies On** code. On the other hand, you want to decode only a particular symbology, scan All Symbologies Off followed by the On symbol for that particular symbology.

**All Symbologies**



**Message Length Description**

You may set the valid reading length of some of the bar code symbologies. If the data length of the scanned bar code doesn't match the valid reading length, the imager will issue an error beep. You may set the same value for minimum and maximum length to force the imager to read fixed length bar code data. This helps reduce the chances of a misread.

**EXAMPLE:** Decode only those bar codes with a count of 9-20 characters. Min. length = 09 Max. length = 20

**EXAMPLE:** Decode only those bar codes with a count of 15 characters. Min. length = 15 Max. length = 15. For a value other than the minimum and maximum message length defaults, scan the bar codes included in the explanation of the symbology, then scan the digit value of the message length and **Save** bar codes. The minimum and maximum lengths and the defaults are included with the respective symbologies

**Start/Stop Characters**

Start/Stop characters identify the leading and trailing ends of the bar code.

## 8.1 Codabar

### Check Character

Codabar check characters are created using different “modulos.” You can program the imager to read only Codabar bar codes with Modulo 16 check characters. **No Check Character** indicates that the imager reads and transmits bar code data with or without a check character. When Check Character is set to **Validate and Transmit**, the imager will only read Codabar bar codes printed with a check character, and will transmit this character at the end of the scanned data. When Check Character is set to **Validate, but Don’t Transmit**, the unit will only read Codabar bar codes printed with a check character, but will not transmit the check character with the scanned data.



<Default All Codabar Settings>



\*On



Off

### Start /Stop Character



Transmit



\*Don't Transmit

### Check Character



\*No Check Character

### Validate and Transmit



Validate Modulo 16 and  
Transmit



Validate Modulo 16 ,but Don't  
Transmit

### Concatenation

Codabar supports symbol concatenation. When you enable concatenation, the imager looks for a Codabar symbol having a “D” start character, adjacent to a symbol having a “D” stop character. In this case the two messages are concatenated into one with the “D” characters omitted. Default = On.



Select Require to prevent the imager from decoding a single “D” Codabar symbol without its companion. This selection has no effect on Codabar symbols without Stop/Start D characters.



**Message Length**

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 2-60. Minimum Default = 4, Maximum Default = 60.

**Message Length**



\*On



Off



Require



Minimum Message Length



Maximum Message Length

## 8.2 Code 39

### *Start/Stop Characters*

Start/Stop characters identify the leading and trailing ends of the bar code. You may either transmit, or not transmit Start/Stop characters. Default = Don't Transmit.

### *Check Character*

**No Check Character** indicates that the imager reads and transmits bar code data with or without a check character. When Check Character is set to **Validate, but Don't Transmit**, the unit only reads Code 39 bar codes printed with a check character, but will not transmit the check character with the scanned data. When Check Character is set to **Validate and Transmit**, the imager only reads Code 39 bar codes printed with a check character, and will transmit this character at the end of the scanned data. Default = No Check Character.



<Default All Code 39 Settings>



\*On



Off



Transmit



\*Don't Transmit



\*No Check Character



Validate and Transmit



Validate, but Don't Transmit

### *Message Length*

Scan the bar codes below to change the message length.. Minimum and Maximum lengths = 0-48. Minimum Default = 0, Maximum Default = 48.

### *Append*

This function allows the imager to append the data from several Code 39 bar codes together before transmitting them to the host computer. When this function is enabled, the imager stores those Code 39 bar codes that start with a space (excluding the start and stop symbols), and does not immediately transmit the data. The imager stores the data in the order in which the bar codes are read, deleting the first space from each. The imager transmits the appended data when it reads a Code 39 bar code that starts with a character other than a space.



Code 39



Minimum Message Length



Maximum Message Length



Append On



\*Append Off

Code 39 Full ASCII

If Full ASCII Code 39 decoding is enabled, certain character pairs within the bar code symbol will be interpreted as a single character. For example: \$V will be decoded as the ASCII character SYN, and /C will be decoded as the ASCII character #.

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

Character pairs /M and /N decode as a minus sign and period respectively. Character pairs /P through /Y decode as 0 through 9.

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, scan the bar code below, select the code page with which the bar codes were created and scan the value and the Save bar code. The data characters should then appear properly.



Full ASCII On



\*Full ASCII Off



Code 39 Code Page

### 8.3 Code 32

Code 32 Pharmaceutical is a form of the Code 39 symbology used by Italian pharmacies. This symbology is also known as PARAF.



**Note:** Trioptic Code must be turned off while scanning Code 32 Pharmaceutical codes.

**Code 32**



On



\*Off

## 8.4 Interleaved 2 of 5

**Check Digit**

**No Check Digit** indicates that the imager reads and transmits bar code data with or without a check digit.

When Check Digit is set to **Validate, but Don't Transmit**, the unit only reads Interleaved 2 of 5 bar codes printed with a check digit, but will not transmit the check digit with the scanned data.

When Check Digit is set to **Validate and Transmit**, the imager only reads Interleaved 2 of 5 bar codes printed with a check digit, and will transmit this digit at the end of the scanned data.

**Message Length**

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 2-80. Minimum Default = 4, Maximum Default = 80.



<Default All Interleaved 2 of 5 Settings>



\*On



Off



\*No Check Digit



Validate, but Don't Transmit



Validate and Transmit



Minimum Message Length



Maximum Message Length

## 8.5 Code 93

### Code 93 Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 0-80. Minimum Default = 0, Maximum Default = 80.

### 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, scan the bar code below, select the code page with which the bar codes were created and scan the value and the **Save** bar code from the Programming Chart on the inside the back cover of this manual. The data characters should then appear properly.



<Default All Code 93 Settings>



\*On



Off



Minimum Message Length



Maximum Message Length



Code 93 Code Page

## 8.6 Straight 2 of 5 Industrial

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-48. Minimum Default = 4, Maximum Default = 48.



<Default All Straight 2 of 5 Industrial Settings>



On



\*Off



Minimum Message Length



Maximum Message Length

## 8.7 Straight 2 of 5 IATA

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-48. Minimum Default = 4, Maximum Default = 48



<Default All Straight 2 of 5 IATA Settings>



On



\*Off



Maximum Message Length



Minimum Message Length

## 8.8 Matrix 2 of 5

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-80. Minimum Default = 4, Maximum Default = 80.



<Default All Matrix 2 of 5 Settings>



On



\*Off



Minimum Message Length



Maximum Message Length

## 8.9 Code 11

### *Check Digits Required*

This option sets whether 1 or 2 check digits are required with Code 11 bar codes. Default = Two Check Digits.

### *Message Length*

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-80. Minimum Default = 4, Maximum Default = 80.



<Default All Code 11 Settings>



On



\*Off



One Check Digit



\*Two Check Digits



Minimum Message Length



Maximum Message Length



## 8.10 Code 128

### *ISBT 128 Concatenation*

The use of ISBT formats requires a paid license. The ISBT 128 Application Specification describes 1) the critical data elements for labeling blood products, 2) the current recommendation to use Code 128 due to its high degree of security and its space-efficient design, 3) a variation of Code 128 that supports concatenation of neighboring symbols, and 4) the standard layout for bar codes on a blood product label. Use the bar codes below to turn concatenation on or off. Default = Off.

### *Message Length*

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 0-80. Minimum Default = 0, Maximum Default = 80.

### *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 bar code being scanned was created using a code page that is different from the one the host program is expecting. If this is the case, scan the bar code below, select the code page with which the bar codes were created, and scan the value and the **Save**. The data characters should then appear properly. Default = 2.



<Default All Code 128 Settings>



\*On



Off



ISBT 128 On



\* ISBT 128 Off



Minimum Message Length



Maximum Message Length

### *Code Page*



Code 128 Code Page

## 8.11 Telepen

### Telepen Output

Using AIM Telepen Output, the imager reads symbols with start/stop pattern 1 and decodes them as standard full ASCII (start/stop pattern 1). When Original Telepen Output is selected, the imager reads symbols with start/stop pattern 1 and decodes them as compressed numeric with optional full ASCII (start/stop pattern 2). Default = AIM Telepen Output.

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-60. Minimum Default = 1, Maximum Default = 60.



<Default All Telepen Settings>



On



\*Off



\*AIM Telepen Output



Original Telepen Output



Minimum Message Length



Maximum Message Length



## 8.12 UPC-A

### *Check Digit*

This selection allows you to specify whether the check digit should be transmitted at the end of the scanned data or not. Default = On.

### *Number System*

The numeric system digit of a U.P.C. symbol is normally transmitted at the beginning of the scanned data, but the unit can be programmed so it will not transmit it. Default = On.



<Default All UPC-A Settings>



\*On



Off

### *Check Digit*



\*On



Off

### *Number System*



\*On



Off

### *Addenda*

This selection adds 2 or 5 digits to the end of all scanned UPC-A data. Default = Off for both 2 Digit and 5 Digit Addenda.

### *Addenda Required*

When **Required** is scanned, the imager will only read UPC-A bar codes that have addenda. You must then turn on a 2 or 5 digit addenda. Default = Not Required.

### *Addenda Separator*

When this feature is on, there is a space between the data from the bar code and the data from the addenda. When turned off, there is no space. Default = On.

### *UPC-A/EAN-13 with Extended Coupon Code*

Use the following codes to enable or disable UPC-A **and** EAN-13 with Extended Coupon Code. Default = On.



**Addenda**



2 Digit Addenda On



\*2 Digit Addenda Off



5 Digit Addenda On



\*5 Digit Addenda Off

**Addenda Required**



Required



\*Not Required

**Addenda Separator**



\*On



Off

**UPC-A/EAN-13 with Extended**

**Coupon Code**



\*On



Off

## 8.13 UPC-E0

Most U.P.C. bar codes lead with the 0 number system. For these codes, use the UPC-E0 selection. If you need to read codes that lead with the 1 number system. Default = On.

**UPC-E0 Expand**

UPC-E Expand expands the UPC-E code to the 12 digit, UPC-A format. Default = Off.

**Addenda Required**

When Addenda Required is set to on, the imager will only read UPC-E bar codes that have addenda.

**Addenda Separator**

When this feature is on, there is a space between the data from the bar code and the data from the addenda. When turned off, there is no space.

**Check Digit**

Check Digit specifies whether the check digit should be transmitted at the end of the scanned data or not. Default = On.



**Number System**

The numeric system digit of a U.P.C. symbol is normally transmitted at the beginning of the scanned data, but the unit can be programmed so it will not transmit it. Default = On.



<Default All UPC-E Settings>



\*UPC-E0 On



UPC-E0 Off

**UPC-E0 Expand**



On



\*Off

**Addenda Required**



Required



\*Not Required

**Addenda Separator**



\*On



Off

**Check Digit**



\*On



Off

**Number System**



\*On



Off

**Addenda**

This selection adds 2 or 5 digits to the end of all scanned UPC-E data. Default = Off for both 2 Digit and 5 Digit Addenda

**UPC-E1**

Most U.P.C. bar codes lead with the 0 number system. For these codes. If you need to read codes that lead with the 1 number system, use the UPC-E1 selection. Default = Off

**Addenda**



2 Digit Addenda On



\*2 Digit Addenda Off



5 Digit Addenda On



\*5 Digit Addenda Off

**UPC-E1**



UPC-E1 On



\*UPC-E1 Off

## 8.14 EAN/JAN-13

**Check Digit**

This selection allows you to specify whether the check digit should be transmitted at the end of the scanned data or not.

**ISBN Translate**

This selection causes EAN-13 Bookland symbols to be translated into their equivalent ISBN number format.



<Default All EAN/JAN Settings>



\*On



Off

**Check Digit**



\*On



Off

**ISBN Translate**



On



\*Off



### EAN/JAN-13

#### *Addenda*

This selection adds 2 or 5 digits to the end of all scanned EAN/JAN-13 data.

#### *Addenda Required*

When Addenda required is set to on, the imager will only read EAN/JAN-13 bar codes that have addenda.

#### *Addenda Separator*

When this feature is on, there is a space between the data from the bar code and the data from the addenda. When turned off, there is no space.

#### *Addenda*



2 Digit Addenda On



\*2 Digit Addenda Off



5 Digit Addenda On



\*5 Digit Addenda Off

#### *Addenda Required*



Required



\*No Required

#### *Addenda Separator*



\*On



Off



## 8.15 EAN/JAN-8

### Check Digit

This selection allows you to specify whether the check digit should be transmitted at the end of the scanned data or not. Default = On.

### Addenda

This selection adds 2 or 5 digits to the end of all scanned EAN/JAN-8 data. Default = Off for both 2 Digit and 5 Digit Addenda.



<Default All EAN/JAN-8 Settings>



\*On



Off

### Check Digit



\*On



Off

### Addenda



2 Digit Addenda On



\*2 Digit Addenda Off



5 Digit Addenda On



\*5 Digit Addenda Off

### Addenda Required

When Addenda required is set to on, the imager will only read EAN/JAN-8 bar codes that have addenda.

### EAN/JAN-8 Addenda Separator

When this feature is on, there is a space between the data from the bar code and the data from the addenda. When turned off, there is no space.

### Addenda Required



Required



\*Not Required

### Addenda Separator



\*On



Off

## 8.16 MSI

### MSI Check Character

Different types of check characters are used with MSI bar codes. You can program the imager to read MSI bar codes with Type 10 check characters.

When Check Character is set to **Validate and Transmit**, the imager will only read MSI bar codes printed with the specified type check character, and will transmit this character at the end of the scanned data. When Check Character is set to **Validate, but Don't Transmit**, the unit will only read MSI bar codes printed with the specified type check character, but will not transmit the check character with the scanned data.

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 4-48. Minimum Default = 4, Maximum Default = 48.



<Default All MSI Settings>



On



\*Off



\*Validate Type 10, but Don't  
Transmit



Validate Type 10 and Transmit



Minimum Message Length



Maximum Message Length

## 8.17 Plessey Code

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 4-48. Minimum Default = 4, Maximum Default = 48.



<Default All Plessey Code Settings>



On



\*Off



Minimum Message Length



Maximum Message Length

### 8.18 GS1 DataBar Omnidirectional



<Default All GS1 DataBar Omnidirectional Settings>



\*On



Off

### 8.19 GS1 DataBar Limited



<Default All GS1 DataBar Limited Settings>



\*On



Off

### 8.20 GS1 DataBar Expanded

#### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 4-74. Minimum Default = 4, Maximum Default = 74.



<Default All GS1 DataBar Expanded Settings>



\*On



Off



Minimum Message Length



Maximum Message Length

## 8.21 PosiCode

You have to have **PosiCode A and B** on to read any of the PosiCode symbologies.

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 2-80. Minimum Default = 4, Maximum Default = 48.

### Trioptic Code

**Note:** If you are going to scan Code 32 Pharmaceutical codes, Trioptic Code must be off. Trioptic Code is used for labeling magnetic storage media



<Default All PosiCode Settings>



\* On



Off



A and B On ( No Limited )



A and B and Limited On



A and B and Limited B On

( Limited A Off )



( Limited B Off )

### Message Length



Minimum Message Length



Maximum Message Length

### Trioptic Code



On



\*Off

## 8.22 Codablock F

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-2048. Minimum Default = 1, Maximum Default = 2048



<Default All Codablock F Settings>



On



\*Off



Minimum Message Length



Maximum Message Length

## 8.23 Code 16K

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 0-160. Minimum Default = 1, Maximum Default = 160.



<Default All Code 16K Settings>



On



\*Off



Minimum Message Length



Maximum Message Length

## 8.24 Code 49

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-81. Minimum Default = 1, Maximum Default = 81.



<Default All Code 49 Settings>



\*On



Off



Minimum Message Length



Maximum Message Length

## 8.25 PDF417

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-2750. Minimum Default = 1, Maximum Default = 2750.



<Default All PDF417 Settings>



\*On



Off



Minimum Message Length



Maximum Message Length

## 8.26 MicroPDF417

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-366. Minimum Default = 1, Maximum Default = 366.



<Default All Micro PDF417 Settings>



\* On



Off



Minimum Message Length



Maximum Message Length



## 8.27 EAN•UCC Composite Codes

Linear codes are combined with a unique 2D composite component to form a new class called EAN•UCC Composite symbology. EAN•UCC Composite symbologies allow for the co-existence of symbologies already in use.

### UPC/EAN Version

Scan the **UPC/EAN Version On** bar code to decode EAN•UCC Composite symbols that have a UPC or EAN linear component. (This does not affect EAN•UCC Composite symbols with a UCC/EAN-128 or GS1 DataBar linear component.)

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-2435. Minimum Default = 1, Maximum Default = 2435

### Emulation

The imager can automatically format the output from any EAN•UCC data carrier to emulate what would be encoded in an equivalent UCC/EAN-128 or GS1 DataBar and Composite symbol. EAN•UCC data carriers include UPC-A and UPC-E, EAN-13 and EAN-8, ITF-14, UCC/EAN-128, and EAN•UCC GS1 DataBar and Composites. Data from 2D symbols such as Aztec Code, Data Matrix, or QR Code, which encode a leading FNC1, also invoke EAN•UCC emulation. If UCC/EAN-128 Emulation is selected, the AIM Symbology Identifier is reported as “jC1”. If GS1 DataBar Emulation is selected, the AIM Symbology Identifier is reported as “je0.” Any application that accepts EAN•UCC data can be simplified since it only needs to recognize one data carrier type.

### TCIF Linked Code 39 (TLC39)

This code is a composite code since it has a Code 39 linear component and a MicroPDF417 stacked code component. All bar code readers are capable of reading the Code 39 linear component. The MicroPDF417 component can only be decoded if **TLC39 On** is selected. The linear component may be decoded as Code 39 even if TLC39 is off.

### EAN•UCC Composite Codes



On



\*Off

### UPC/EAN Version



UPC / EAN Version On



\*UPC / EAN Version Off

### Message Length



Minimum Message Length



Maximum Message Length

### Emulation

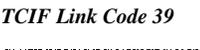


GS1 DataBar Emulation



128 Emulation

### \*EAN.UCC Emulation Off



### TCIF Link Code 39



TLC39 On



\*TLC39 Off

## 8.28 **Postal Codes**

Note: For best performance when reading a postal symbology, all other postal symbologies should be turned off. The following postal codes can only be read by the ZB-3062.

### 8.28.1 **Postnet**

#### **Check Digit**

This selection allows you to specify whether the check digit should be transmitted at the end of the scanned data.



On



\*Off



Transmit Check Digit



\*Don't Transmit Check Digit

### 8.28.2 **Planet Code**

#### **Check Digit**

This selection allows you to specify whether the check digit should be transmitted at the end of the scanned data.



On



\*Off



Transmit Check Digit



\*Don't Transmit Check Digit



### 8.28.3 Postal Codes

#### *British Post*



On



\*Off

#### *Canadian Post*



On



\*Off

#### *Kix (Netherlands) Post*

Note: Kix code can misread when scanned sideways or upside down.



On



\*Off

#### *Australian Post*



On



\*Off

#### *Japanese Post*



On



\*Off

### 8.28.4 China Post

#### *Message Length*

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 2-80. Minimum Default = 4, Maximum Default = 80.



<Default All China Post Settings>



On



\*Off



Minimum Message Length



Maximum Message Length

### 8.28.5 Korea Post

#### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 2-80. Minimum Default = 4, Maximum Default = 48.



<Default All Korea Post Settings>



On



\*Off



Minimum Message Length



Maximum Message Length

### 8.29 QR Code

This selection applies to both QR Code and Micro QR Code.

#### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-3500. Minimum Default = 1, Maximum Default = 3500.



<Default All QR Code Settings>



\*On



Off



Minimum Message Length



Maximum Message Length

## 8.30 Data Matrix

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-1500. Minimum Default = 1, Maximum Default = 1500.



<Default All Data Matrix Settings>



\*On



Off



Minimum Message Length



Maximum Message Length

## 8.31 MaxiCode

### Message Length

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-150. Minimum Default = 1, Maximum Default = 150.



<Default All MaxiCode Settings>



\*On



Off



Minimum Message Length



Maximum Message Length

### 8.32 **Aztec Code.**

#### **Message Length**

Scan the bar codes below to change the message length. Minimum and Maximum lengths = 1-3750.  
Minimum Default = 1, Maximum Default = 3750.



<Default All Aztec Code Settings>



\*On



Off



Minimum Message Length



Maximum Message Length

#### **Aztec Runes**

Select **Enable Runes** if you are scanning Aztec runes, which are the smallest type of Aztec Code symbol with the ability to encode a very short license plate message.



Enable Runes



\*Disable Runes



## 9 String Format

### 9.1 Prefix/Suffix

When a bar code is scanned, additional information is sent to the host computer along with the bar code data. This group of bar code data and additional, user-defined data is called a “message string.” The selections in this section are used to build the user-defined data into the message string. Prefix and Suffix characters are data characters that can be sent before and after scanned data. You can specify if they should be sent with all symbologies, or only with specific symbologies. The following illustration shows the breakdown of a message string:

**Prefix / Suffix characters setting:** Characters defined as prefix or suffix characters will be transmitted immediately with the scanned data for all symbologies. Up to 11 ASCII characters can be defined for both Prefix and Suffix.

#### *Tips*

- A prefix or suffix may be added or cleared from one symbology or all symbologies.
- You can add any prefix or suffix f, plus Code I.D. and AIM I.D.
- You can string together several entries for several symbologies at one time.
- Enter prefixes and suffixes in the order in which you want them to appear on the output.
- When Setting up for specific symbologies, instead of All Symbologies, the symbology ID value counts as an added prefix or suffix character. Prefix Scanned Data Suffix

#### *To Add a Prefix or Suffix:*

**Step 1.** Scan the **Add Prefix** or **Add Suffix** symbol

**Step 2.** Determine the 2 digit Hex value from the Symbology Chart or the symbology to which you want to apply the prefix or suffix. For example, for Code 128, Code ID is “j” and Hex ID is “6A”.

**Step 3.** Scan the 2 hex digits or scan **9, 9** for all symbologies.

**Step 4.** Determine the hex value from the for the prefix or suffix you wish to enter.

**Step 5.** Scan the 2 digit hex value.

**Step 6.** Repeat Steps 4 and 5 for every prefix or suffix character.

**Step 7.** To add the Code I.D., scan **5, C, 8, 0**.

To add AIM I.D., scan **5, C, 8, 1**.

To add a backslash (\), scan **5, C, 5, C**.

**Note:** To add a backslash (\) as in Step 7, you must scan 5C twice – once to create the leading backslash and then to create the backslash itself.

**Step 8.** Scan **Save** to exit and save, or scan **Discard** to exit without saving. Repeat Steps 1-6 to add a prefix or suffix for another symbology.

**Example: Add a Suffix to a specific symbology**

#### *To send a CR (carriage return) Suffix for UPC only:*

**Step 1.** Scan **Add Suffix**.

**Step 2.** Determine the 2 digit hex value for UPC.

**Step 3.** Scan **6, 3**.

**Step 4.** Scan **0, D** (carriage return).from the

**Step 5.** Scan **Save**, or scan **Discard** to exit without saving.



**To Clear One or All Prefixes or Suffixes:**

You can clear a single prefix or suffix, or clear all prefixes/suffixes for a symbology. When you Clear One Prefix (Suffix), the specific character you select is deleted from the symbology you want. When you Clear All Prefixes (Suffixes), all the prefixes or suffixes for a symbology are deleted.

**Step 1.** Scan the **Clear One Prefix** or **Clear One Suffix** symbol.

**Step 2.** Determine the 2 digit Hex value for the symbology from which you want to clear the prefix or suffix.

**Step 3.** Scan the 2 digit hex value or scan **9, 9** for all symbologies. Your change is automatically saved.

**To Add a Carriage Return Suffix to all Symbologies**

Scan the following bar code if you wish to add a carriage return suffix to all symbologies at once. This action first clears all current suffixes, then programs a carriage return suffix for all symbologies.

**Prefix Selections**



Add Prefix



Clear All Prefix



Clear One Prefix

**Suffix Selections**



Add Suffix



Add CR Suffix All Symbologies



Clear One Suffix



Clear All Suffixes

**Function Code Transmit**

When this selection is enabled and function codes are contained within the scanned data, the imager transmits the function code to the terminal.

**Function Code Transmit**



\*Enable



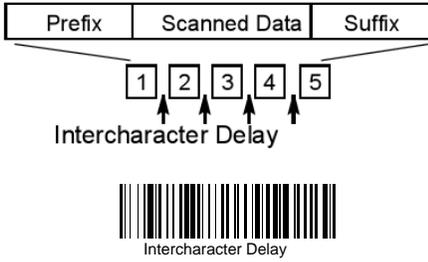
Disable

**Intercharacter, Interfunction, and Intermessage Delays**

Some terminals drop information (characters) if data comes through too quickly. Intercharacter, interfunction, and intermessage delays slow the transmission of data, increasing data integrity. Each delay is composed of a 5 millisecond step. You can program up to 99 steps (of 5 ms each) for a range of 0-495 ms.

**Intercharacter Delay**

An intercharacter delay of up to 495 milliseconds may be placed between the transmissions of each character of scanned data. Scan the **Intercharacter Delay** bar code below, then scan the number of milliseconds and the **SAVE** bar code



To remove this delay, scan the **Inter-character Delay** bar code, then set the number of steps to 0. Scan the **SAVE** bar code.  
**Note:** Intercharacter delays are not supported in USB serial emulation.

**User Specified Inter-character Delay**

An inter-character delay of up to 495 milliseconds may be placed after the transmission of a particular character of scanned data. Scan the **Delay Length** bar code below, then scan the number of milliseconds and the **SAVE** bar code. Next, scan the **Character to Trigger Delay** bar code, then the 2-digit hex value for the ASCII character that will trigger the delay

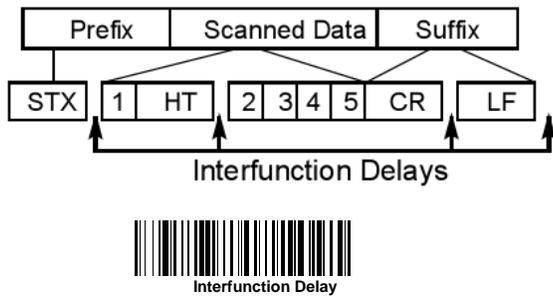
**User Specified Inter-character Delay**



To remove this delay, scan the **Delay Length** bar code, and set the number of steps to 0. Scan the **SAVE** bar code

**Interfunction Delay**

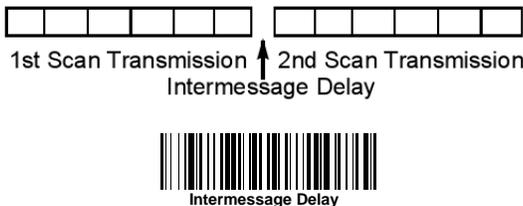
An interfunction delay of up to 495 milliseconds may be placed between the transmission of each segment of the message string. Scan the **Interfunction Delay** bar code below, then scan the number of milliseconds and the **SAVE**



To remove this delay, scan the **Interfunction Delay** bar code, then set the number of steps to 0. Scan the **SAVE** bar code.

**Intermessage Delay**

An intermessage delay of up to 495 milliseconds may be placed between each scan transmission. Scan the **Intermessage Delay** bar code below, then scan the number of milliseconds and the **SAVE** bar code



To remove this delay, scan the **Intermessage Delay** bar code, then set the number of steps to 0. Scan the **SAVE** bar code.



## 9.2 Data Formatting

### **Data Format Editor Introduction**

You may use the Data Format Editor to change the imager's output. For example, you can use the Data Format Editor to insert characters at certain points in bar code data as it is scanned. The selections in the following pages are used only if you wish to alter the output. Default Data Format setting = None. Normally, when you scan a bar code, it gets outputted automatically; however when you do a format, you must use a "send" command within the format program to output data. Multiple formats may be programmed into the imager. They are stacked in the order in which they are entered. However, the following list presents the order in which formats are applied:

1. Specific Term ID, Actual Code ID, Actual Length
2. Specific Term ID, Actual Code ID, Universal Length
3. Specific Term ID, Universal Code ID, Actual Length
4. Specific Term ID, Universal Code ID, Universal Length
5. Universal Term ID, Actual Code ID, Actual Length
6. Universal Term ID, Actual Code ID, Universal Length
7. Universal Term ID, Universal Code ID, Actual Length
8. Universal Term ID, Universal Code ID, Universal Length

If you have changed data format settings, and wish to clear all formats and return to the factory defaults, scan the **Default Data Format**

### **To Add a Data Format**

**Step 1.** Scan the **Enter Data Format** symbol

**Step 2. Primary/Alternate Format**

Determine if this will be your primary data format, or one of 3 alternate formats. (Alternate formats allow you "single shot" capability to scan one bar code using a different data format. After the one bar code has been read, the imager reverts to the primary data format. If you are programming the primary format, scan **0** using the inside the back cover of this manual. If you are programming an alternate format, scan **1, 2, or 3**, depending on the alternate format you are programming.

**Step 3. Interface Type**

Select the Interface ID number for your PC. Scan three numeric bar codes on the inside back cover to program the imager for your Interface ID (you must enter 3 digits). For example, scan **0 0 0** for RS-232.

**Note:** The wildcard for all terminal types is 099.

**Step 4. Code ID.**

Find the symbology to which you want to apply the data format. Locate the Hex value for that symbology and scan the 2 digit hex value.

**Step 5. Length**

Specify what length (up to 9999 characters) of data will be acceptable for this symbology. Scan the four digit data length from the back cover of this manual. (Note: 50 characters is entered as 0050. 9999 is a universal number, indicating all lengths.)

**Step 6. Editor Commands**

Refer to [Data Format Editor Commands](#) . Scan the symbols that represent the command you want to enter 94 alphanumeric characters may be entered for each symbology data format.

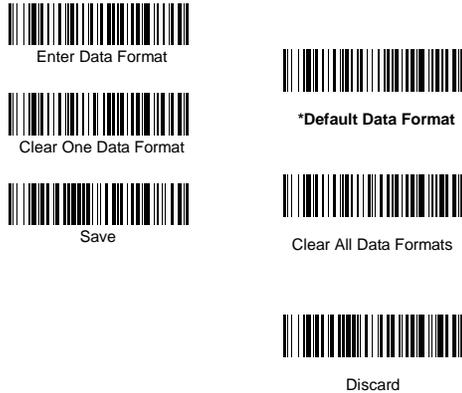
**Step 7.** Scan **Save** to save your entries.

### • Clear One Data Format

This deletes one data format for one symbology. If you are clearing the primary format, scan **0**. If you are clearing an alternate format, scan **1, 2, or 3**, depending on the alternate format you are clearing. Scan the Interface Type and Code I.D., and the bar code data length for the specific data format that you want to delete. All other formats remain unaffected.

• **Save:** This exits, saving any Data Format changes.

• **Discard :** This exits without saving any Data Format changes.



## 9.3 Data Format Editor Commands

### Send Commands

F1 Send all characters followed by “xx” key or function code, starting from current cursor position.

**Syntax = F1xx** (xx stands for the hex value for an ASCII code.)

F2 Send “nn” characters followed by “xx” key or function code, starting from current cursor position. **Syntax = F2nnxx** (nn stands for the numeric value (00-99) for the number of characters and xx stands for the hex value for an ASCII code.)

F3 Send up to but not including “ss” character (Search and Send) starting from current cursor position, leaving cursor pointing to “ss” character followed by “xx” key or function code. **Syntax = F3ssxx** (ss and xx both stand for the hex values for ASCII codes.)

F4 Send “xx” character “nn” times (Insert) leaving cursor in current cursor position. **Syntax = F4xxnn** (xx stands for the hex value for an ASCII code, and nn is the numeric value (00-99) for the number of times it should be sent.)

E9 Send all but the last “nn” characters, starting from the current cursor position. **Syntax = E9nn** (nn is the numeric value (00-99) for the number of characters that will not be sent at the end of the message.)

### Move Commands

F5 Move the cursor a head “nn” characters from current cursor position. **Syntax = F5nn** (nn stands for the numeric value (00-99) for the number of characters the cursor should be moved ahead.)

F6 Move the cursor back “nn” characters from current cursor position. **Syntax = F6nn** (nn stands for the numeric value (00-99) for the number of characters the cursor should be moved back.)

F7 Move the cursor to the beginning of the data string.

**Syntax = F7.** EA Move the cursor to the end of the data string. **Syntax = EA**

### Search Commands

F8 Search ahead for “xx” character from current cursor position, leaving cursor pointing to “xx” character.

**Syntax = F8xx** (xx stands for the hex value for an ASCII code.)

F9 Search back for “xx” character from current cursor position, leaving cursor pointing to “xx” character. **Syntax = F9xx** (xx stands for the hex value for an ASCII code.)

**E6** Search ahead for the first non “xx” character from the current cursor position, leaving cursor pointing to non “xx” character. **Syntax = E6xx** (xx stands for the hex value for an ASCII code.

**E7** Search back for the first non “xx” character from the current cursor position, leaving cursor pointing to non “xx” character. **Syntax = E7xx** (xx stands for the hex value for an ASCII code,

### Miscellaneous Commands

**FB** Suppress all occurrences of up to 15 different characters, starting at the current cursor position, as the cursor is advanced by other commands. When the FC command is encountered, the suppress function is terminated. The cursor is not moved by the FB command. **Syntax = FBnnxxyy . .zz** where nn is a count of the number of suppressed characters in the list and xxyy .. zz is the list of characters to be suppressed. (xx stands for the hex value for an ASCII code,

**FC** Disables suppress filter and clear all suppressed characters. **Syntax = FC.**

**E4** Replaces up to 15 characters in the data string with user specified characters. Replacement continues until the E5 command is encountered. **Syntax = E4nnxx1xx2yy1yy2...zz1zz2** where nn is the total count of both characters to be replaced plus replacement characters; xx1 defines characters to be replaced and xx2 defines replacement characters, continuing through zz1 and zz2.

**E5** Terminates character replacement. **Syntax = E5.**

**FE** Compare character in current cursor position to the character “xx.” , If characters are equal, increment cursor. If characters are not equal, no format match. **Syntax = FExx** (xx stands for the hex value for an ASCII code.

**EC** Check to make sure there is an ASCII number at the current cursor position. If character is not numeric, format is aborted. **Syntax = EC.**

**ED** Check to make sure there is a non-numeric ASCII character at the current cursor position. If character is numeric, format is aborted. **Syntax = ED.**

### Data Formatter

When Data Formatter is turned off, the bar code data is output to the host as read (including prefixes and suffixes). Choose one of the following options. Default = Data Formatter On, but Not Required.



\*Data Formatter On, but Not Required



Data Formatter Off

When Data Formatter is required, all input data must conform to an edited format or the imager does not transmit the input data to the host device.



Data Formatter On, Format Required

### Alternate Data Formats

Alternate formats allow you “single shot” capability to scan one bar code using a different data format than your primary format. When data formats are programmed, you must input whether you are programming the primary format, or an alternate format numbered 1, 2, or 3. An alternate format is initiated by scanning one of the 3 alternate format bar codes below. The imager will scan the next bar code, formatting the data with the selected alternate format, then revert immediately to the primary format.



Alternate Data Format 1



Alternate Data Format 2



Alternate Data Format 3

## 9.4 OCR Programming

Use this section to program the Imager for optical character recognition (OCR). The 2D ZB-3062 Engine reads 6 to 60 point OCR typeface.

**Note:** OCR is not as secure as bar codes. To enhance security in OCR applications, create an OCR template to match the data, and print an OCR check character.

The 2D ZB-3062 Engine will read OCR-A, OCR-B, U.S. Currency Serial Number (Money), MICR E-13B, and SEMI fonts. You can either select an OCR default, or create your own custom template for the type of OCR format you intend to read. See "OCR" on page 9-1 for programming codes that will enable your imager to read OCR-A, OCR-B, U.S.

Currency, MICR E-13B, or SEMI fonts. See "OCR Templates" on page 9-3 if you want to create a custom "template," or character string that defines the length and content of OCR strings that will be read with your imager.

**Note:** Setting the template and check character options are essential for OCR reading.

## 9.5 OCR Fonts

**Default All OCR Settings** turns off all OCR capability in the imager, so the imager will be able to scan linear, stacked, matrix, and composite bar codes, but not OCR fonts. In addition, any OCR templates you have created are erased.

The eight digit default templates are reinstated for any future use of the **OCR On** codes listed below.

### OCR

**Note:** OCR symbols can misread when scanned sideways or upside down. Use [Working Orientation](#), if your OCR symbols will not usually be presented upright to the scanner. Only one OCR symbology can be read at a time. **OCR-A On** allows you to scan characters in the OCR-A font. The default setting allows you to scan any eight digit combination. If you have created an OCR template, character combinations that fit the template can be scanned. **OCR-B On** allows you to scan characters in the OCR-B font. The default setting allows you to scan any eight digit combination. If you have created an OCR template, character combinations that fit the template can be scanned.



<Default All OCR Settings>



OCR-A On



OCR-B On

### 9.5.1 **U.S. Currency Font**

**U.S. Currency On** allows you to scan characters in the font used on U.S. currency. The default setting allows you to scan any eight digit combination. If you have created an OCR template, character combinations that fit the template can be scanned



### 9.5.2 **MICR E-13B Font**

**MICR E-13B On** allows you to scan MICR characters on a bank check. The default setting allows you to scan any eight digit combination. If you have created an OCR template, character combinations that fit the template can be scanned



**Note:** TOAD characters (Transit, On Us, Amount and Dash) are output in the following manner:

-  the transit character is output as **T**
-  the amount character is output as **A**
-  the on us character is output as **O**
-  the dash character is output as **D**

### 9.5.3 **SEMI Font**

**SEMI Font On** allows you to scan the SEMI font used in the semiconductor industry.



**All OCR Off** turns off all OCR capability in the imager, so the imager will be able to scan linear, stacked, matrix, and composite bar codes, but not OCR fonts. However, any OCR templates you have created will be retained in memory.



## 9.6 OCR Templates

You can create a custom “template,” or character string that defines the length and content of OCR strings that will be read with your imager. There are several choices when creating a custom template for your application. You can create a template for a single format, you can string together several formats, and you can create a template for a user-defined variable. These choices are described in detail below.

### Creating an OCR Template

A single template allows you to program the imager to read any combination of characters in the order you specify. Refer to examples that follow the Template Characters table below.

#### Template Characters

a	represents any alphanumeric character (digit or letter)
c	represents a check character position
d	represents any digit
e	represents any available OCR character
g	represents character from user-defined variable “g”
h	represents character from user-defined variable “h”
l	represents any uppercase letter
t	marks the start of a new template
r	multi row indicator
All other characters represent themselves. Spaces can be used. Note: In MICR E-13B templates, TOAD characters (capital letters T, O, A, and D), represent Transit, On Us, Amount, and Dash.	

**Note:** OCR templates default to eight digits, no check character.

To Add an OCR Template

1. Turn on the OCR font you want to read .
2. Begin building the template.

Scan the **Enter OCR Template** symbol

3. Scan the characters for the string.

Use the Template Characters chart above to determine what characters you need to create your format. Use the [OCR Programming Chart](#) to scan the characters for your template.

**Example:** You need to read any combination of eight digits. The template would be: dddddddd

To create this template, you would enable the OCR-A font. Scan the **Enter OCR Template** symbol , then scan the **d** from the [OCR Programming Chart](#) in the back of this manual eight times. Scan **Save OCR Template** This would let you read any string of eight digits, for example:

37680981

#### Character Match Sequences

On the [ASCII Chart](#), find the Hex value that represents the character(s) you want to match.

**Example:** You need to read three digits, three specific characters (ABC), three digits. The template would be:

ddd414243ddd  
  
 hex codes for  
 letters A, B, and C



To create this template, you would enable the OCR-A font. Scan the **Enter OCR Template** symbol. Scan the **d** from the [OCR Programming Chart](#) in the back of this manual three times. Then scan **414243** from the inside back cover (the hex characters for “A,” “B,” and “C”), and scan the **d** three more times. Scan **Save OCR Template**. This would let you read any string of three digits, “ABC,” then any string of three digits, for example:

551ABC983

**Adding Spaces**

You may also need to put spaces in your template.

**Example:** You need to read three digits, space, three specific characters (ABC), space, three digits. The template would be:

ddd2041424320ddd



hex code for a space

To create this template, you would enable the OCR-A font. Scan the **Enter OCR Template** symbol. Scan the **d** from the [OCR Programming Chart](#) in the back of this manual three times, then scan **2041424320** from the inside back cover (the hex characters for “space,” “A,” “B,” “C,” “space”). Scan the **d** three more times, then scan **Save OCR Template**. This would let you read any string of three digits space, “ABC,” space, then any string of three digits. For example:

551 ABC 983

**Note:** If using Quick\*View to program, use the space bar to designate a space and not the hex value of 20.4. Exit OCR Template Editor Scan **Save OCR Template** to save your entries. **Discard OCR Template** exits without saving any OCR Template changes.

**Stringing Together Multiple Formats  
(Creating “Or” Statements)**

You may want to program the imager to accept many OCR formats. To do this, you would string together each format with a “t.” This tells the imager to read optical characters that match any one of the formats in the template.

**Example:** You need to read any combination of eight digits, or a combination of four digits, two uppercase letters, and two digits. The template would be: dddddddtdddlldd

To create this template, you would enable the OCR-A font. Scan the **Enter OCR Template** symbol. Scan the **d** from the [OCR Programming Chart](#) in the back of this manual eight times, then scan the **t** to create the “or” statement. Then you would scan the characters for the second template. Scan **d** four times, scan **l** two times, then scan **d** two more times. Scan **Save OCR Template**. This would let you read either type of format, for example:

99028650

or

9902XZ50

You can string together as many templates as you need.



### OCR User-Defined Variables

You can create up to two of your own user variables for an OCR template. These variables will represent any OCR readable characters. The user-defined variables are stored under the letters "g" and "h." Creating a user variable follows the same steps as creating a template, but instead of scanning the **Enter OCR Template** symbol, you scan the **Enter User-Defined Variable** symbol. The letters g and h can then be used in an OCR template to define the variable you specified.

**Example:** You need a variable to represent the letters "A," "B," or "C." The template for this variable would be:

414243 To create this template, you would enable the OCR-A font. Scan the **Enter User-Defined Variable g** symbol. Scan **414243** (the hex characters for "A," "B," and "C"). Scan **Save OCR Template**. This will let you read either

A or B or C in any position where you place the g. For example, you could create the following template: ddddddggg This template would then let you read data that began with six digits, and had an A, B, or C trailing. So you would be able to read:

654321ABC

or

654321BAC

or

654321CCC

### Reading Multi-Row OCR

The ZB-3062 is capable of decoding multi-row OCR text.

**Note:** Reading rows longer than sixteen characters is not recommended. Consider the following example. This example shows serial commands as they would be entered using Quick\*View.

**Example:** You need to read multiple rows of OCR-A data as shown below:

12345678

ABCDEFGH

First, enable the OCR-A font. To read the first row of OCR data, you would program the following template:

OCRTMP"ddddddd". This template is the default OCR template. If you wanted to read the second line of data, you would use the following template:

OCRTMP"llllllll". To read both lines of OCR at one time, use the variable **r** to indicate the start of a new row. All the other templating variables for the individual rows work the same as previously described. For instance, in the above example, you would use the following template to read both rows:

OCRTMP"ddddddrlllllll".

To read the three rows below, you would use the template command

"OCRTMP"ddddddrllllllrlllddd".

12345678

ABCDEFGH

ABCD1234.

## OCR Check Character

You may want to print and verify a check character in order to enhance the security of your OCR application. The ZB-3062 can be programmed for almost any type of check character. A number of presets are provided for common check character uses (e.g., modulo 10 and modulo 36).

Scan the **OCR Modulo 10** or **OCR Modulo 36 Check Character** bar code to specify the type of check character used in the OCR strings you're scanning. The imager will then only read OCR character strings with a valid check character. The ZB-3062 transmits the OCR data without the

check character data. You must specify the location of the check character in the template with a **c**.

**Example:** You need to read any combination of seven digits, with a modulo 10 check character in the eighth position. The template would be: ddddddc. To create this template, you would enable the OCR-A font. Scan the **Modulo 10 Check Character** symbol. Then scan the **Enter OCR Template** symbol, and scan the **d** from the [OCR Programming Chart](#) seven times, and scan the **c** once. Scan **Save OCR Template**. This template will let you read any combination of six digits with a correct check character after. (If the check character is invalid, the imager will issue an error beep.) For example, the following string could be scanned:

01234569

and the output would be: 0123456

### *OCR Modulo 10 Check Character*

Scan this symbol to program the OCR template for a simple modulo 10 checksum of the digits 0 through 9.



OCR Modulo 10 Check Character

### *OCR Modulo 36 Check Character*

Scan this symbol to program the OCR template for a simple modulo 36 checksum of the digits 0 through 9 and the letters A through Z.



OCR Modulo 36 Check Character



### OCR User-Defined Check Character

You can customize the check character calculation to suit your application. Each character of the check character alphabet can be programmed in its proper order. The number of characters you enter determines the modulo value for the calculation. By default, the check character computation is unweighted, but the imager also supports two weighted modulo 10 checking schemes.

**Example:** To program a modulo 11 check character, you would enter the following 11 characters in order:

**0123456789X**

Also enter the OCR template: **ddddddc**

Enable the OCR-A font, then scan the following string:

**6512351X**

The imager performs the following check character computation:

$(6 + 5 + 1 + 2 + 3 + 5 + 1 + X) \text{ modulo } 11 = 0$

Since the result is zero, the message is considered to be valid, so the reader outputs the message: 6512351

### Programming a User-Defined Check Character

1. Scan the **Enter OCR Check Character** bar code, below:



**Enter OCR Check Character**

OCR Modulo 36 Check Character Enter OCR Check Character

2. Enter the characters in order. For each character, look up the corresponding hex value from the [ASCII Chart](#).

3. Scan the **Save** bar code on the inside back cover.

**Example:** To program the modulo 11 check character from example #8 , enable the OCR-A font. Scan the **Enter OCR Check Character** bar code, then scan the following hex values in order:

**3031323334353637383958**

After you enter all the desired hex values, scan the **Save** bar code on the inside back cover of this manual.

### Weighting Options

By default, the check character computation is unweighted. It is possible to use one of two weighted modulo 10 schemes. Weighting is often used to detect if two neighboring characters are transposed, a common error when an operator keys in data.

### Weighted Modulo 10 Check Character

Starting with the check character and working backward through the message, the imager applies a multiplier of 1, then 3, then 1, then 3, and so on. This is the checking scheme used in many EAN•UCC symbologies, including U.P.C. and Interleaved 2 of 5 (when a check digit is invoked). To apply this weighting scheme, set the OCR check character to “**0123456789x3x1**” or scan the following symbol:



**a Weighted Modulo 10  
Check Character**

**Example:** Scan the 3-1-3-1 Weighted Modulo 10 Check Character symbol. Also enter the OCR template: **ddddddc**

Then scan the string below:

**01234565**

The reader performs the check character computation below:

$$(0 \times 3 + 1 \times 1 + 2 \times 3 + 3 \times 1 + 4 \times 3 + 5 \times 1 + 6 \times 3 + 5 \times 1) \text{ modulo } 10 = 0$$

**a. Weighted Modulo 10 Check Character**

Since the result is zero, the message is considered to be valid, so the reader outputs the message: 0123456

**b. 2-1-2-1 Weighted Modulo 10 Check Character**

Starting with the check character and working backward through the message, the imager applies a multiplier of 1, then 2, then 1, then 2, and so on. When the result of the multiplication is greater than 9, add both digits to the running sum. This is often referred to as the LUHN formula. One common application of the LUHN formula is validate credit card numbers. This is the modulo 10 checking scheme used in MSI Code and in Code 32 Pharmaceutical (PARAF). To apply this weighting scheme, set the OCR check character to "0123456789x2x1" or scan the following symbol:



**b. Weighted Modulo 10 Check Character**

**Example:** Scan the 2-1-2-1 Weighted Modulo 10 Check Character symbol. Also enter the OCR template: dddddd

Then scan the string below:

0128454

The reader performs the check character computation below:

$$(0 \times 1 + 1 \times 2 + 2 \times 1 + 8 \times 2 + 4 \times 1 + 5 \times 2 + 4 \times 1) \text{ modulo } 10$$

$$= (0 + 2 + 2 + (1 + 6) + 4 + (1 + 0) + 4) \text{ modulo } 10$$

$$= 0$$

Since the result is zero, the message is considered to be valid, so the reader outputs the message: 012845

**OCR ISBN Application Example**

One application of OCR is to read the ISBN characters typically encoded using the OCR-A or OCR-B font. This is especially useful when the ISBN number is not encoded in an EAN-13 bar code. The following example shows how to configure the imager to read the ISBN strings on books in Japan. After you have followed the steps below, you will be able to scan the following ISBN number or the additional data below it, depending on the line of text at which the imager is aimed.

ISBN4-594-03019-X

C0097 ¥838E

1. Scan the OCR-B On bar code
2. Program the user-defined variable "g" to comprise the ten digits plus the dash: 0123456789-
3. In Japan, it is common for two fields to follow the ISBN number, the three digit price field, and the four digit price field. The first field typically starts with a "C" (uppercase c), followed by four digits. The second field typically starts with a "P" or a yen symbol, followed by three or four digits, followed by an "E." Program the user-defined variable "h" to comprise the "P" and the yen symbol (represented by a backslash). P\
4. Scan the symbol below to set up three templates to handle the ISBN number, the three digit price field, and the four digit price field.



5. Finally, set up the ISBN check digit, which is a special position-weighted modulo 11 checksum. The imager automatically invokes the ISBN checksum for template rows that are:



- 1.) at least fourteen characters long,
- 2.) whose first four characters are the letters "ISBN,"
- 3.) whose last character is a check character, and
- 4.) when the modulo 11 check character "0123456789X" is programmed.

Please note that all these commands can be combined into a single serial programming command:

```
OCRENA2,TMP"ISBNggggggggggctCddd hdddEtCddd
hdddE",GPG"0123456789-",GPH"P",CHK"0123456789X".
```

These commands can be encoded into the following Aztec Code symbol:



### OCR Template Codes

Note: Reading more than three rows of OCR is not recommended. Contact the factory if you have an application that requires reading four or more rows of OCR.

#### OCR Template Codes



On



Enter User-Defined Variable "g"†



Enter User-Defined Variable "h"†

#### Exit Selections



Save OCR Template



Discard OCR Template

## 9.7 Utilities

### To Add a Test Code I.D. Prefix to All Symbologies

This selection allows you to turn on transmission of a Code I.D. before the decoded symbology. This action first clears all current prefixes, then programs a Code I.D. prefix for all symbologies. This is a temporary setting that will be removed when the unit is power cycled.

### Show Software Revision

Scan the bar code below to output the current software revision, unit serial number, and other product information.

### Show Data Format

Scan the bar code below to show current data format settings.

### Resetting the Standard Product Defaults

If you aren't sure what programming options are in your imager, or you've changed some options and want the standard product default settings restored, scan the **Standard Product Default Settings** bar code below.

The Menu Commands starting on page 11-5 lists the standard product default settings for each of the commands



**Test Menu**

When you scan the Test Menu **On** code, then scan a programming code in this manual, the imager displays the content of a programming code. The programming function will still occur, but in addition, the content of that programming code is output to the terminal.

**Note:** This feature should not be used during normal imager operation.

**To Add a Test Code I.D. Prefix to All Symbologies**



Add Code I.D. Prefix to All Symbologies (Temporary)

**Show Software Revision**

**Show Data Format**



Show Revision



Data Format Settings

**Standard Product Default Settings**



Standard Product Default Settings

**Test Menu**



On



\*Off

**2D PQA (Print Quality Assessment)**

Two-dimensional Print Quality Assessment (2D PQA) is a feature of Hand Held Products' image readers where the data from the successful read of a 2D bar code symbol is augmented with lines of text that both identify the symbol, and also report graded measurement parameters obtained from it.



## 9.8 Serial Programming Commands

The serial programming commands can be used in place of the programming bar codes. Both the serial commands and the programming bar codes will program the ZB-3062 Engine. For complete descriptions and examples of each serial programming command, refer to the corresponding programming bar code in this manual. The device must be set to an RS-232 interface. The following commands can be sent via a PC COM port using terminal emulation software.

### Conventions

The following conventions are used for menu and query command descriptions: parameter A label representing the actual value you should send as part of a command.

[option] An optional part of a command.

{Data} Alternatives in a command.

**bold** Names of menus, menu commands, buttons, dialog boxes, and windows that appear on the screen.

### Menu Command Syntax

Menu commands have the following syntax (spaces have been used for clarity only): Prefix Tag SubTag {Data} [, SubTag {Data}] [: Tag SubTag {Data}] [...]

Tag A 3 character case-insensitive field that identifies the desired menu command group. For example, all RS-232 configuration settings are identified with a Tag of **232**.

SubTag A 3 character case-insensitive field that identifies the desired menu command within the tag group. For example, the SubTag for the RS-232 baud rate is **BAD**. Data The new value for menu setting, identified by the Tag and Sub-Tag. Storage A single character that specifies the storage table to which the command is applied. An exclamation point (!) performs the command's operation on the device's volatile menu configuration table. A period (.) performs the command's operation on the device's non-volatile menu configuration table. Use the non-volatile table only for semi-permanent changes you want saved through a power cycle.

### Query Commands

Several special characters can be used to query the device about its settings.

^ What is the default value for the setting(s).

? What is the device's current value for the setting(s).

\* What is the range of possible values for the setting(s). (The device's response uses a dash (-) to indicate a continuous range of values. A pipe (|) separates items in a list of non-continuous values.)

### Tag Field Usage

When a query is used in place of a Tag field, the query applies to the entire set of commands available for the particular storage table indicated by the Storage field of the command. In this case, the SubTag and Data fields should not be used because they are ignored by the device.

### SubTag Field Usage

When a query is used in place of a SubTag field, the query applies only to the subset of commands available that match the Tag field. In this case, the Data field should not be used because it is ignored by the device.

### Data Field Usage

When a query is used in place of the Data field, the query applies only to the specific command identified by the Tag and SubTag fields.

### Concatenation of Multiple Commands

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

### Responses

The device responds to serial commands with one of 3 responses:

**ACK** Indicates a good command which has been processed.

**ENQ** Indicates an invalid Tag or SubTag command.



**NAK** Indicates the command was good, but the Data field entry was out of the allowable range for this Tag and SubTag combination, e.g., an entry for a minimum message length of 100 when the field will only accept 2 characters.

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

**Examples of Query Commands**

In the following examples, a bracketed notation [ ] depicts a non-displayable response.

**Example:** Example #1: What is the range of possible values for Codabar Coding Enable?

**Enter:** **cbrena\***.

**Response:** **CBRENA0-1[ACK]**

This response indicates that Codabar Coding Enable (CBRENA) has a range of values from 0 to 1 (off and on).

**Example:** Example #2: What is the default value for Codabar Coding Enable?

**Enter:** **cbrena^**.

**Response:** **CBRENA1[ACK]**

This response indicates that the default setting for Codabar Coding Enable (CBRENA) is 1, or on.

**Example:** Example #3: What is the device's current setting for Codabar Coding Enable?

**Enter:** **cbrena?**.

**Response:** **CBRENA1[ACK]**

This response indicates that the device's Codabar Coding Enable (CBRENA) is set to 1, or on.

**Example:** Example #4: What are the device's settings for all Codabar selections?

**Enter:** **cbr?**.

**Response:** **CBRENA1[ACK],**

**SSX0[ACK],**

**CK20[ACK],**

**CCT1[ACK],**

**MIN2[ACK],**

**MAX60[ACK],**

**DFT[ACK].**

This response indicates that the device's Codabar Coding Enable (CBRENA) is set to 1, or on; the Start/Stop Character (SSX) is set to 0, or Don't Transmit; the Check Character (CK2) is set to 0, or Not Required; concatenation (CCT) is set to 1, or Enabled; the Minimum Message Length (MIN) is set to 2 characters; the Maximum Message Length (MAX) is set to 60 characters; and the Default setting (DFT) has no value.

**Trigger Commands**

You can activate and deactivate the imager with serial trigger commands. First, the imager must be put in Manual/Serial Trigger Mode either by scanning the Manual/Serial Trigger Mode bar code, or by sending the Manual/ Serial Menu Command . Once the imager is in serial trigger mode, the trigger is activated and deactivated by sending the following commands:

Activate: **SYN T CR**

Deactivate: **SYN U CR**

The imager scans until a bar code has been read, until the deactivate command is sent, or until the serial time-out has been reached (see "Read Time-Out")

**Resetting the Standard Product Defaults**

If you aren't sure what programming options are in your imager, or you've changed some options and want the factory settings restored, scan the **Standard Product Default Settings** bar code below.



Standard Product Default Settings

The chart on the following pages lists the factory default settings for each of the menu commands (indicated by an asterisk (\*) on the programming pages).

### 10 Test Chart

(Bar code samples marked with symbol “\*” are not enabled initially.)

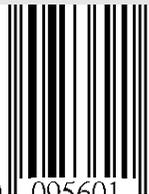
<b>CODABAR</b>  a89121121a
<b>CODE-11 *</b>  0289121121
<b>CODE-128</b>  258963
<b>CODE-39</b>  *741258*
<b>CODE-93</b>  951263
<b>EAN-13</b>  7 534539 789813
<b>PDF-417</b> 
<b>STANDARD-25 *</b>  65978
<b>CODE-16K *</b>  87549
<b>EAN-8</b>



9456 2156  
INDUSTRIAL-25 \*



04976  
UPC-E



0 095601 1

INTERLEAVED-25



46820  
MATRIX 25 \*



4563535663

MSI/PLESSEY \*



754268

UPCA



5 73648 64734 5

UK/PLESSEY PARA \*



64872

GS1 DataBar



Micro-PDF



QR Code



DataMatrix

**Data Matrix**



Test Symbol

Aztec



MaxiCode

MaxiCode



Test Message



## 11 Interface ID

If you want to change the pre-programmed interface of your image engine, below. For example, a USB Virtual COM device has a Interface ID of 130. You would scan the **Interface ID** bar code, then **1, 3, 0** , then **Save**. If you make an error while scanning the digits (before scanning Save), scan the **Discard** code.

Interface	Interface ID
RS-232	000
USB HID (PC)	124
USB HID (Mac)	125
USB Virtual COM	130



Terminal ID



Save



## 12 Product Code ID

Symbology	AIM ID	Code ID	Code ID (HEX )
All Symbologies			0x99
Australian Post	]x0	'A	0x41
Aztec Code	]zm	'z	0x7A
British Post	]X0	'B	0x42
Canadian Post	]X0	'C	0x43
China Post	]X0	'Q	0x51
Codabar	]Fm	'a	0x61
Codablock F	]Om	'q	0x71
Code 11	]H3	'h	0x68
Code 128	]Cm	'j	0x6A
Code 16K	]Km	'o	0x6F
Code 32	]X0	'<	0x3C
Code 39	]Am	'b	0x62
Code 49	]Tm	'I	0x6C
Code 93	]Gm	'i	0x69
Data Matrix	]dm	'w	0x77
EAN-13	]E0	'd	0x64
EAN-8	]E4	'D	0x44
EAN UCC Composite	]em	'y	0x79
EAN-13 w Extended Coupon Code	]E3	'd	0x64
Interleaved 2 of 5	]Lm	'e	0x65
Japanese Post	]X0	'J	0x4A
KIX Post	]X0	'K	0x4B
Korea Post	]X0	'?	0x3F
Matrix 2 of 5	]X0	'm	0x6D
MaxiCode	]Um	','x	0x78
MicriPDF417	]Lm	'R	0x52
MSI	]Mm	'g	0x67
No Read			0x9C
OCR-A	]o1	'O	0x4F

Symbology	AIM ID	Code ID	Code ID (HEX )
OCR-B	]o2	'O	0x4F
OCR MICR E-13B	]ZE	'O	0x4F
OCR US Money Font	]o3	'O	0x4F
SEMI Font	]o3	'O	0x4F
PDF 417	]Lm	'r	0x72
Planet Code	]X0	'L	0x4C
Plessey Code	]P0	'n	0x6E
PosiCode	]pm	'W	0x57
Postnet	]X0	'P	0x50
QR Code	]Qm	's	0x73
GS1 DataBar Family	]em	'y	0x79
Straight of 2 of IATA	]Rm	'f	0x66
TCIF Linked Code 39	]L2	'T	0x54



ZBA Inc.

<b>Telepen</b>	<b>JBm</b>	<b>'t</b>	<b>0x74</b>
<b>Trioptic Code</b>	<b>JX0</b>		<b>0x3D</b>
<b>UCC/EAN-128</b>	<b>JC1</b>	<b>'I</b>	<b>0x49</b>
<b>UPC-A</b>	<b>JE0</b>	<b>'c</b>	<b>0x63</b>
<b>UPC-A w Extended Coupon Code</b>	<b>JE3</b>	<b>'c</b>	<b>0x63</b>
<b>UPC-E</b>	<b>JE0</b>	<b>'E</b>	<b>0x45</b>
<b>VeriCode</b>	<b>JX0</b>	<b>'v</b>	<b>0x76</b>

# 13 ASCII Code Table

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

Dec	Hex	Char									
128	80	€	160	A0		192	C0	À	224	E0	à
129	81	☐	161	A1	ı	193	C1	Á	225	E1	á
130	82	,	162	A2	ª	194	C2	Â	226	E2	â
131	83	f	163	A3	£	195	C3	Ã	227	E3	ã
132	84	.	164	A4	«	196	C4	Ä	228	E4	ä
133	85	...	165	A5	¥	197	C5	Å	229	E5	å
134	86	†	166	A6	¦	198	C6	Æ	230	E6	æ
135	87	‡	167	A7	§	199	C7	Ç	231	E7	ç
136	88	ˆ	168	A8	¨	200	C8	È	232	E8	è
137	89	‰	169	A9	©	201	C9	É	233	E9	é
138	8A	Š	170	AA	ª	202	CA	Ê	234	EA	ê
139	8B	ı	171	AB	«	203	CB	Ë	235	EB	ë
140	8C	œ	172	AC	¬	204	CC	Ì	236	EC	ì
141	8D	☐	173	AD	-	205	CD	Í	237	ED	í
142	8E	Ž	174	AE	®	206	CE	Î	238	EE	î
143	8F	☐	175	AF	˜	207	CF	Ï	239	EF	ï
144	90	☐	176	B0	*	208	DO	Ð	240	FO	ð
145	91	'	177	B1	±	209	D1	Ñ	241	F1	ñ
146	92	ˆ	178	B2	²	210	D2	Ò	242	F2	ó
147	93	ˆ	179	B3	³	211	D3	Ó	243	F3	ó
148	94	ˆ	180	B4	´	212	D4	Ô	244	F4	ô
149	95	ˆ	181	B5	µ	213	D5	Õ	245	F5	ó
150	96	—	182	B6	¶	214	D6	Ö	246	F6	ö
151	97	—	183	B7	·	215	D7	×	247	F7	+
152	98	ˆ	184	B8	,	216	D8	Ø	248	F8	ø
153	99	™	185	B9	ˆ	217	D9	Ù	249	F9	ù
154	9A	š	186	BA	¸	218	DA	Ú	250	FA	ú
155	9B	ı	187	BB	»	219	DB	Û	251	FB	û
156	9C	œ	188	BC	¼	220	DC	Ü	252	FC	ü
157	9D	☐	189	BD	½	221	DD	Ý	253	FD	ý
158	9E	ž	190	BE	¾	222	DE	Þ	254	FE	þ
159	9F	ÿ	191	BF	¿	223	DF	ß	255	FF	ÿ

# 14 OCR Prog. Chart



a



d



g



l



t



c



e



h



r

# 15 Program Chart



0



1



2



3



4



5



6



7



8



9



A



B



C



D



E



F



Save



Discard