MICROSCAN

HS-21/HS-41X Handheld Reader User Manual

P/N 84-100052 Rev B

Microscan Systems, Inc.
Tel: +1.425.226.5700 / 800.762.1149
Fax: +1.425.226.8250

Disclaimer

The information and specifications described in this manual are subject to change without notice.

Latest Manual Version

For the latest version of this manual, see the Download Center on our web site at: **www.microscan.com**.

Technical Support

For technical support, e-mail: **helpdesk@microscan.com**.

Warranty

For current warranty information, see: **www.microscan.com/warranty**.

Microscan Systems, Inc.

United States Corporate Headquarters
+1.425.226.5700 / 800.762.1149

United States Northeast Technology Center
+1.603.598.8400 / 800.468.9503

European Headquarters
+31.172.423360

Asia Pacific Headquarters
+65.6846.1214

Table of Contents

About the HS-21 and HS-41X Handheld Readers

The HS-21 is a general-purpose 2D reader. Its many features include dual field optics for both high density and wide angle performance, a ruggedized design, IP54 sealing against dust and water, and compact size.

The HS-41X is a special-purpose 2D reader for decoding direct part marks (DPM). Microscan's X-Mode decode algorithms make the HS-41X an ideal solution for reading difficult marks on many surfaces, including PCBs, electrical components, castings, and sheet metal. Its tough design makes it a good choice for manufacturing and light industrial applications.

Both readers can be configured and tested easily using the intuitive tree controls and user interface of Microscan's **ESP Software**.

Note: The HS-21 and HS-41X have unique algorithm licenses, and the HS-21 cannot be field-upgraded to an HS-41X.

About This Manual

This manual provides complete information on setting up, installing, and configuring the HS-21 and HS-41X. The chapters are presented in the order in which the reader would be assembled, configured, and optimized.

Highlighting

Cross-references and web addresses are highlighted in **blue bold**.

References to **ESP**, its toolbar headings, and menu headings are highlighted in **Bold Initial Caps**.

Statement of Agency Compliance

The HS-21 and HS-41X have been tested for compliance with FCC regulations and were found to be compliant with all applicable FCC Rules and Regulations.

IMPORTANT NOTE: To comply with FCC RF exposure compliance requirements, this device must not be co-located or operate in conjunction with any other antenna or transmitter.

CAUTION: Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

The HS-21 and HS-41X have been tested for compliance to CE (Conformité Européenne) standards and guidelines and were found to conform to applicable CE standards, specifically the EMC requirements EN 55024, ESD EN 61000-4-2, Radiated RF Immunity EN 61000-4-3, ENV 50204, EFT EN 61000-4-4, Conducted RF Immunity EN 61000-4-6, EN 55022, Class B Radiated Emissions, and Class B Conducted Emissions.

Statement of RoHS Compliance

All Microscan readers with a ‘G’ suffix in the FIS number are RoHS-Compliant. All compliant readers were converted prior to March 1, 2007. All standard accessories in the Microscan Product Pricing Catalog are RoHS-Compliant except 20-500013-01 and 98-000039-02. These products meet all the requirements of “Directive 2002/95/EC” European Parliament and the Council of the European Union for RoHS compliance. In accordance with the latest requirements, our RoHS-Compliant products and packaging do not contain intentionally added Deca-BDE, Perfluorooctanes (PFOS) or Perfluorooctanic Acid (PFOA) compounds above the maximum trace levels. To view the document stating these requirements, please visit:

http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:32002L0095:EN:HTML

and

http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2006:372:0032:0034:EN:PDF

Please contact your sales manager for a complete list of Microscan’s RoHS-Compliant products.

This declaration is based upon information obtained from sources which Microscan believes to be reliable, and from random sample testing; however, the information is provided without any representation of warranty, expressed or implied, regarding accuracy or correctness. Microscan does not specifically run any analysis on our raw materials or end product to measure for these substances.

The information provided in this certification notice is correct to the best of Microscan’s knowledge at the date of publication. This notice is not to be considered a warranty or quality specification. Users are responsible for determining the applicability of any RoHS legislation or regulations based on their individual use of the product.

In regards to “RoHS Directive 2011_65_EU” Microscan produces Monitoring and Control Instruments as well as Industrial Monitoring & Control Instruments as defined within the directive. Microscan has developed and is implementing a RoHS2 compliance plan with the intention of bringing all active products listed in our current marketing literature within full compliance as per the directive deadlines.

Key milestones for the transition plan are as follows:

- Complete internal product audit by July 2014.
- Initial “Monitoring and Control Instruments” RoHS2 compliant products available by December 2014
- Initial “Industrial Monitoring & Control Instruments” RoHS2 compliant products available by July 2015
- All new products introduced in 2015 are expected to be WEEE & RoHS2 compliant.

Microscan will mark the products with the ‘CE’ marking that complies with the RoHS2 process to acquire ‘CE’ certification per the example given: Example >> Machinery directive + EMC directive + RoHS2 = Declaration of Conformity.

1 Quick Start

Contents

This section is designed to get your HS-21 or HS-41X Handheld Reader up and running quickly using Microscan's **ESP** Software so you can get a sense of its capabilities and test sample symbols. Detailed setup information for configuring the reader for your specific application can be obtained in the subsequent sections.

Your interface type will determine how data is received by the host. When sending data by USB, you must open a text editor in your host computer. When sending data by RS-232, you must use a terminal program such as **ESP**'s **Terminal** view.

Check Required Hardware

The HS-21/HS-41X is shipped with one of the following cables:

6' USB Straight Cable ***8' RS-232 Coiled Cable***

USB Hardware

- HS-21/HS-41X Handheld Reader
- USB Cable

RS-232 Hardware

- HS-21/HS-41X Handheld Reader
- RS-232 Cable
- RS-232 Power Supply

Changing Cable and Communications Interface

The reader can be converted from USB to RS-232 or from RS-232 to USB by changing the cable and scanning the appropriate communications programming symbol below. To detach the USB or RS-232 cable from the reader, press a paper clip into the hole on the side of the handle and gently pull the cable out of the connector.

USB Interface

RS-232 Interface

M0166_01

Save Settings

USB Interface

Note: The USB interface draws its power from the host computer.

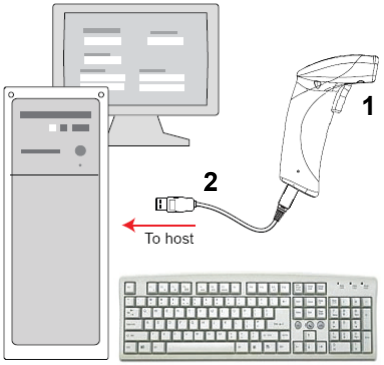
USB Configuration

Item	Description	Part Number
1	HS-21 Handheld Reader	FIS-HS21-000XG
	HS-41X Handheld Reader	FIS-HS41X-000XG
2	USB Cable, 6’ Straight	61-000224-01
	USB Cable, 8’ Coiled	61-000224-02

Installation Steps for USB

To power on the reader, plug the USB cable into the reader’s handle and into the host’s USB port. After several seconds, the reader will beep twice and the LED will turn off. The reader will now be ready to use.

Important: If you use a USB hub, be sure that it is a powered hub.



USB Configuration

Q0001_2

Default to USB (HID)

Read the **Save Settings** symbol.

M0166_01

Save Settings

Test Symbol
(ABCDEFGHIJKLMNOP)

RS-232 Interface

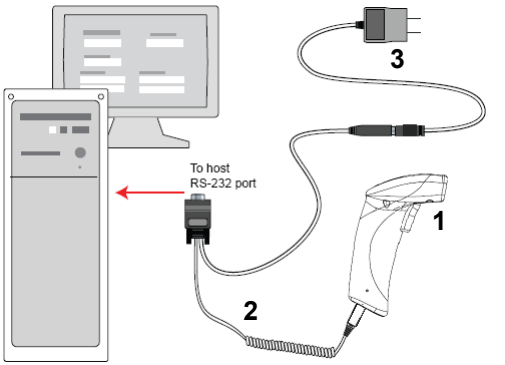
Note: Unlike USB, the RS-232 interface does not draw its power from the host computer, and requires a power supply.

RS-232 Configuration

Item	Description	Part Number
1	HS-21 Handheld Reader	FIS-HS21-000XG
	HS-41X Handheld Reader	FIS-HS41X-000XG
2	RS-232 Cable, 8' Coiled	61-000224-03
3	RS-232 Power Supply, U.S.	20-000335-02
	RS-232 Power Supply, Euro	20-000336-02
	RS-232 Power Supply, UK	20-000337-02

Installation Steps for RS-232

To power on the reader, plug the RS-232 cable into the reader's handle, plug the power supply into the AC outlet, plug the power supply cord into the barrel jack on the cable, and then connect the cable to the appropriate serial port on the host. After several seconds, the unit will beep twice and the LED will turn off. The reader will now be ready to use.



RS-232 Configuration

M0017_01

Reset to RS-232 Factory Defaults

Read the **Save Settings** symbol.

M0166_01

Save Settings

Test Symbol
(ABCDEFGHIJKLMNOP)

Install ESP

ESP Software is Microscan's configuration and testing software. Use ESP to set up your HS-21 or HS-41X Handheld Reader.

ESP can be found on the Microscan Tools Drive that is packaged with the reader.

1. Follow the prompts to install ESP from the Tools Drive.
2. Click on the ESP icon to run the program.

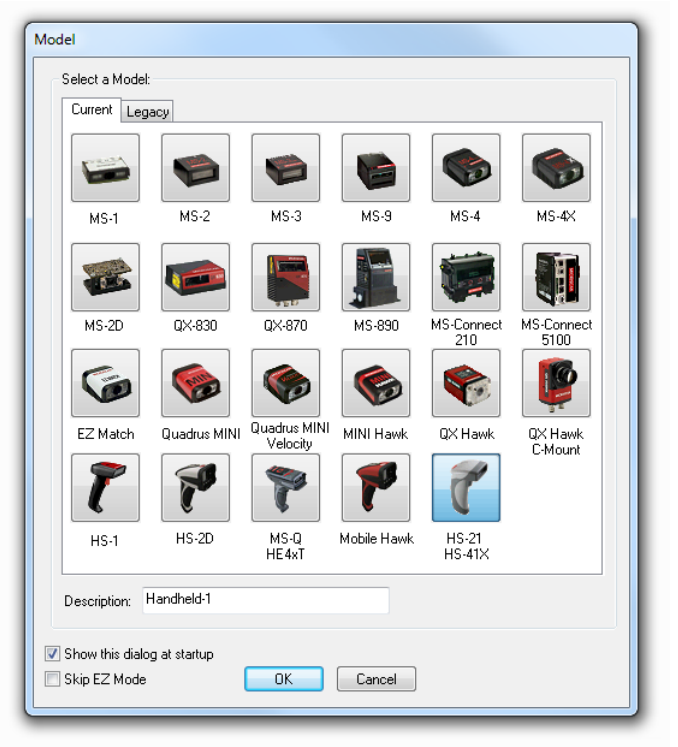
Note: ESP can also be installed from the **Download Center** at **www.microscan.com**.

Minimum System Requirements

- 233 MHz Pentium PC
- Windows 8, 7, Vista, or XP operating system (32-bit or 64-bit)
- Internet Explorer 6.0 or higher
- 128 MB RAM or greater
- 160 MB free disk space
- 800 x 600 256 color display (1024 x 768 32-bit color recommended)

Select Model

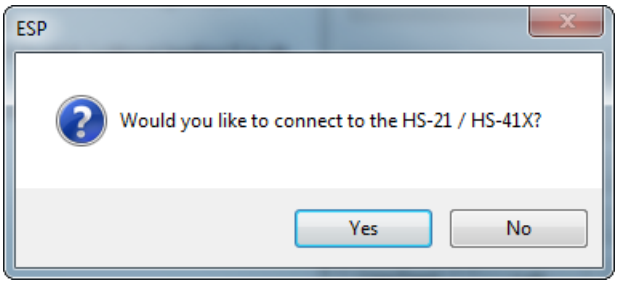
When you start **ESP**, the following menu will appear:



1. Click the HS-21/HS-41X button and then click **OK**. If you do not want to make this selection every time you start **ESP**, uncheck “Show this dialog at startup”. If you need to select another model later, click **Switch Model** at the top of the screen.

   Note: You can also type a name of your choice in the **Description** text field and click **OK**.

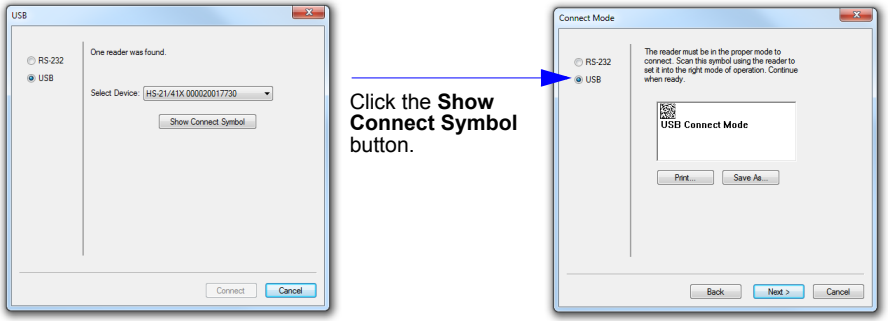
2. Click **Yes** when this dialog appears:



Select Interface and Connect to Reader

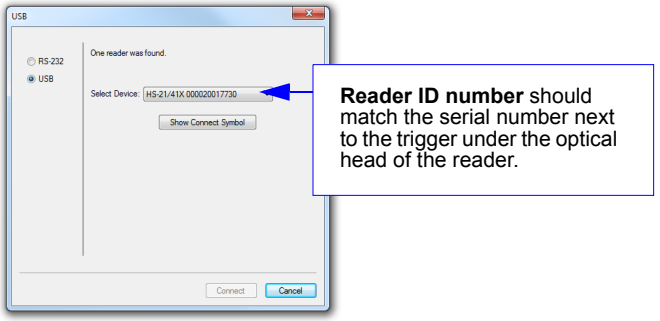
USB

- In the communications dialog box, select the communications interface you are using and click **Next**.



- Click the **Show Connect Symbol** button, print the **USB Connect Mode** symbol, and decode it with the reader to ensure that you are in the correct communications interface. Keep the printed symbol in a convenient place for future use.
- Click **Next** when you are finished.

The **Select Device** dialog will then reappear:

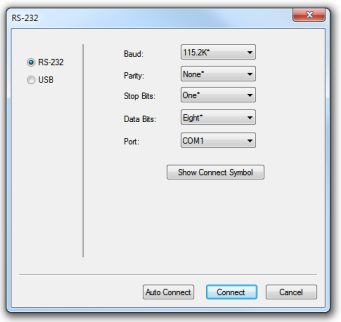


- You will see a " Reader" ID number in the **Select Device** field. Click **Connect**.
- When you are connected successfully, the **CONNECTED** message will appear in a green box in the status bar at the bottom right of the screen.

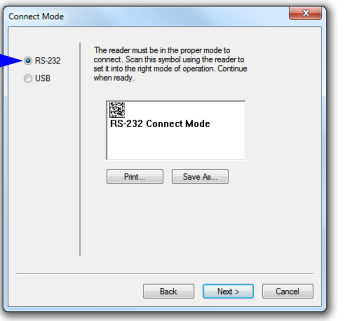
You are now ready to configure your reader using **ESP**. Subsequent sections provide more detailed information about **ESP**'s configuration options.

RS-232

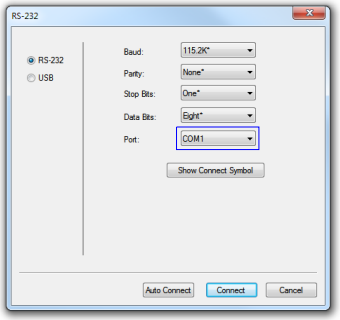
- In the **Select Protocol** dialog box, select the communications interface you are using and click **Next**.



Click the **Show Connect Symbol** button.



- Print the **RS-232 Connect Mode** symbol (also shown in the **Install ESP** step) and decode it with the reader to ensure that you are in the correct communications interface. Keep the printed symbol in a convenient place for future use.
- Click **Next** when you are finished.
- The **Com Port** dialog will then reappear. Select which communications port you are using. If you don't see your communications port listed on the dropdown menu, select **Other**.



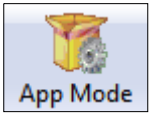
- Click **Connect**.
- When you are connected successfully, the **CONNECTED** message will appear in a green box in the status bar at the bottom right of the screen.

If the connection attempt fails, enable a different communications port, check your port connections, and try again.

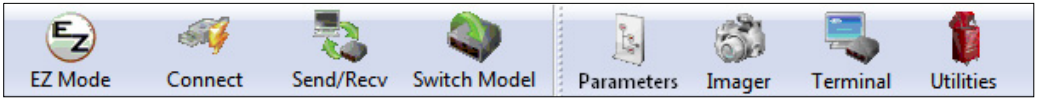
You are now ready to configure your reader using **ESP**. Subsequent sections provide more detailed information about **ESP**'s configuration options.

Configure the Reader

Click the **App Mode** button to make configuration changes.

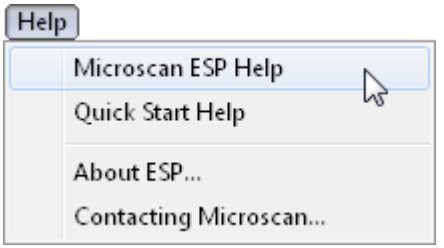


The following modes are accessible by clicking the buttons in the first row of **App Mode** icons:



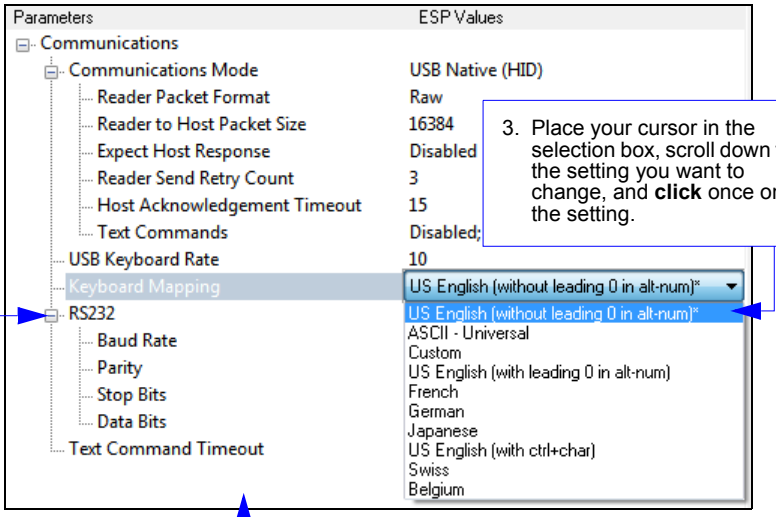
- Click the **EZ Mode** button to return to the EZ Mode view.
- Click the **Autoconnect** button to establish communication.
- Click the **Send/Recv** button to send or receive commands.
- Click the **Switch Model** button to open the model menu, or to return to a previous model.
- Click the **Parameters** button to show the tabbed tree controls for Communication, Read Cycle, Symbologies, and I/O Parameters.
- Click the **Imager** button to capture and decode symbols and to use Window of Interest for Near Field and Far Field.
- Click the **Terminal** button to display decoded symbol data and to send serial commands to the reader using text or macros.
- Click the **Utilities** button to show the tabbed interfaces for Device Control, Differences from Default, Firmware, and Advanced settings.

For further details, see **ESP Help** in the dropdown Help menu.



Save Changes in ESP

To make changes to a configuration setting:



1. **Left-click** on the **+** to expand the desired tree.
2. **Double-click** on the desired parameter and click once in the selection box to view options.
3. Place your cursor in the selection box, scroll down to the setting you want to change, and **click** once on the setting.
4. **Left-click** again on the open screen to complete your selection.
5. **Right-click** on the open screen and select **Save to Reader** to implement the command in the reader.

Saving Options

- **Send, No Save**. Changes will be lost when power is re-applied to the reader.
- **Send and Save**. This activates all changes in current memory *and* saves to the reader for power-on.

2 Using ESP

Contents

This section is designed to help you understand the structure and application of **ESP**.

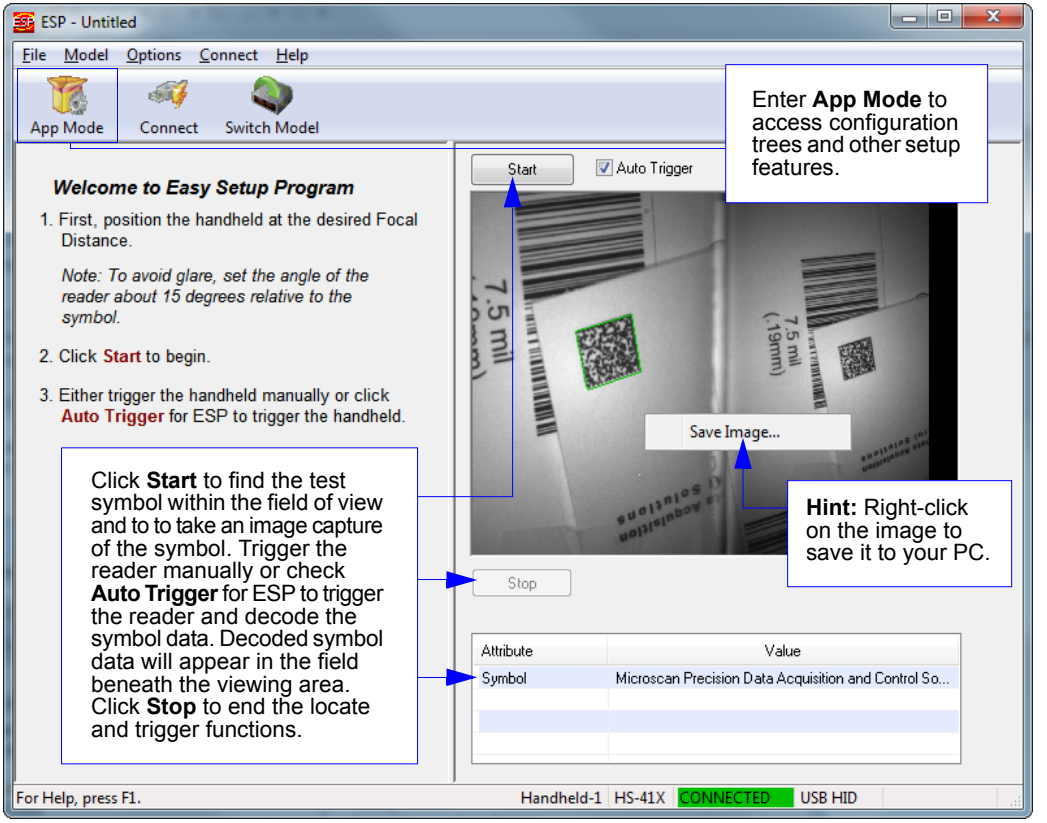
When you open **ESP**, unless otherwise specified in the **ESP Preferences** dialog accessible from the **Options** heading on the menu toolbar, you will enter **EZ Mode** for initial setup. From there, you can enter **Application Mode** (**App Mode**) and access several configuration menus (**Communications**, **Read Cycle**, **Symbologies**, **I/O Parameters**, an **Imager** interface, a **Terminal** interface, and a **Utilities** interface).

ESP can be used to configure the HS-21 and HS-41X Handheld Readers in the following ways:

- **Tree Controls:** Each configuration menu contains a list of all option settings that pertain to that specific element of reader operation. For example, the **Communications** menu shows a **Communications Mode** command, and then the options **RS-232 Serial**, **USB Keyboard**, and **USB Native (HID)**, all of which are accessible from a dropdown menu.
- **Graphic User Interfaces:** Settings can be configured using such point-and-click tools as radio buttons, tabs, spin boxes, check boxes, and drag-and-drop functions.
- **Terminal: ESP**'s **Terminal** interface allows you to send configuration and utility commands directly to the reader by typing them in the provided text field.

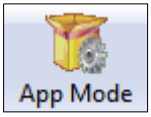
EZ Mode

EZ Mode offers instructions on positioning the reader in relation to a test symbol and triggering the reader to decode the symbol.

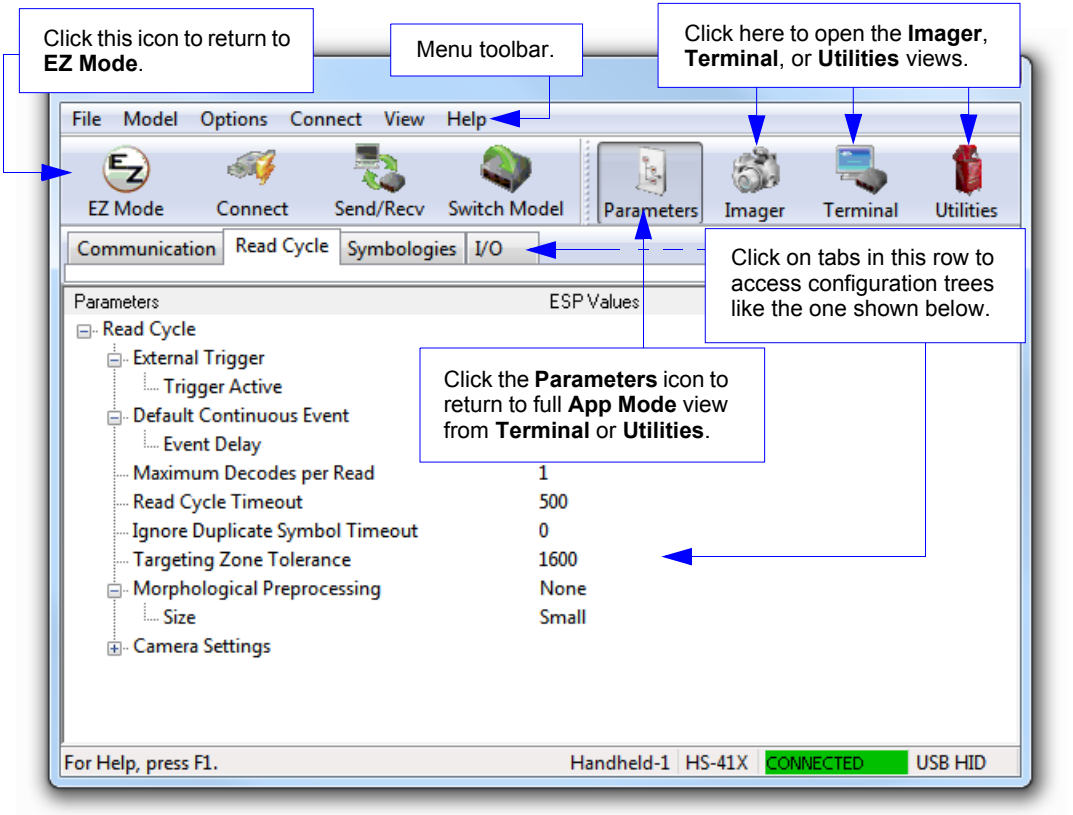


Application Mode

From **EZ Mode**, you can click on the **App Mode** button to access specific configuration menus, **Utilities** tools, and a **Terminal** window where serial commands can be entered.



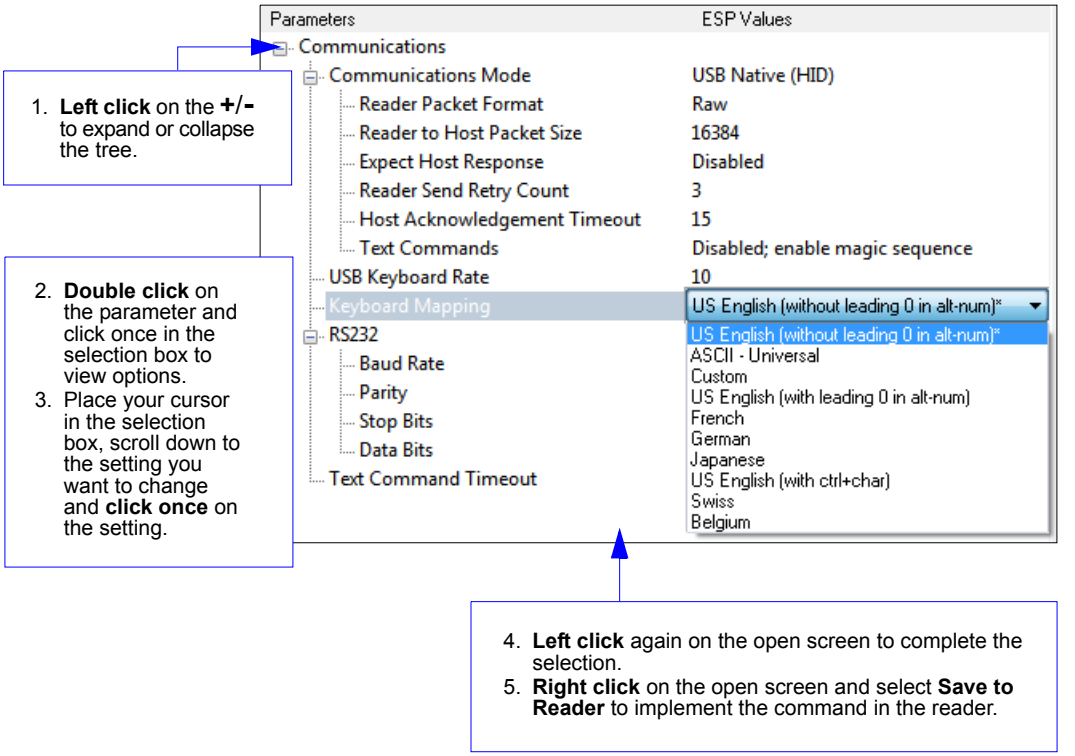
Note: The **App Mode** and **EZ Mode** buttons appear in the same position to allow easy switching between these primary modes.



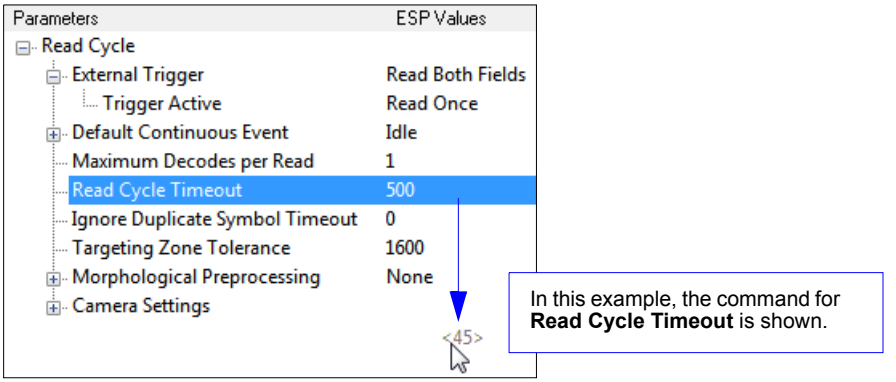
Note: See the corresponding sections of this documentation for specific information on any of the views or modes mentioned above.

Tree Controls

To make changes to configuration settings in the tree control menus:



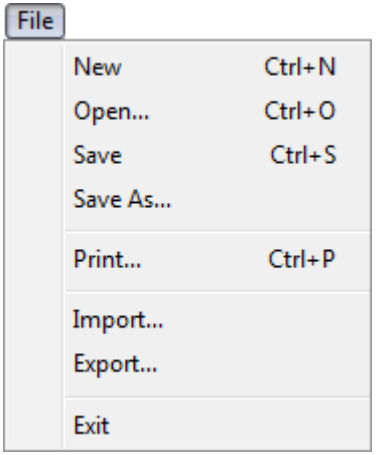
Hint: To see the underlying serial command that corresponds with each tree control item, click on the item in the tree control and drag the mouse to the open screen. The command will be displayed between angle brackets.



Menu Toolbar

File > New

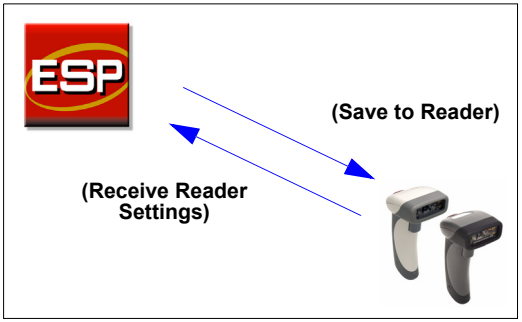
Whenever **New** is selected from the **File** menu, the default configuration of **ESP** is loaded.



Open / Save

When **Save** or **Save As** is selected, the **ESP** configuration is saved to the host computer's hard drive and available whenever the same file is selected under **Open**.

When you save menu changes to your hard drive, these changes are not saved to your reader. The diagram below shows how settings can be saved and received between **ESP** and the reader, and **ESP** and the host hard drive.



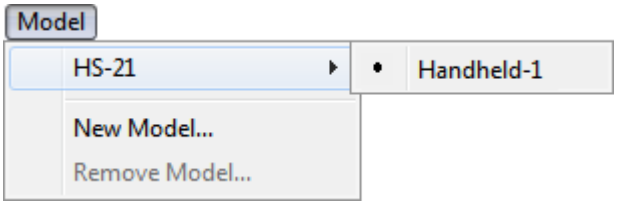
Import / Export

Import converts the ASCII settings from a text file to **ESP** configuration settings.

Export converts the active **ESP** configuration settings to an ASCII text file.

Model

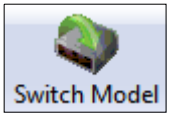
The **Model** menu allows you to select between reader models. When you choose another model, the current connection with your present model will be terminated.



New Model

To connect to another model, select **New Model**, choose the model you want, and click **OK**.

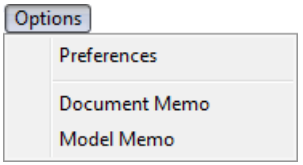
All models you have selected and enabled will continue to appear in the dropdown model menu. The **New Model** option is repeated when you click the **Switch Model** button on the top row of icons.



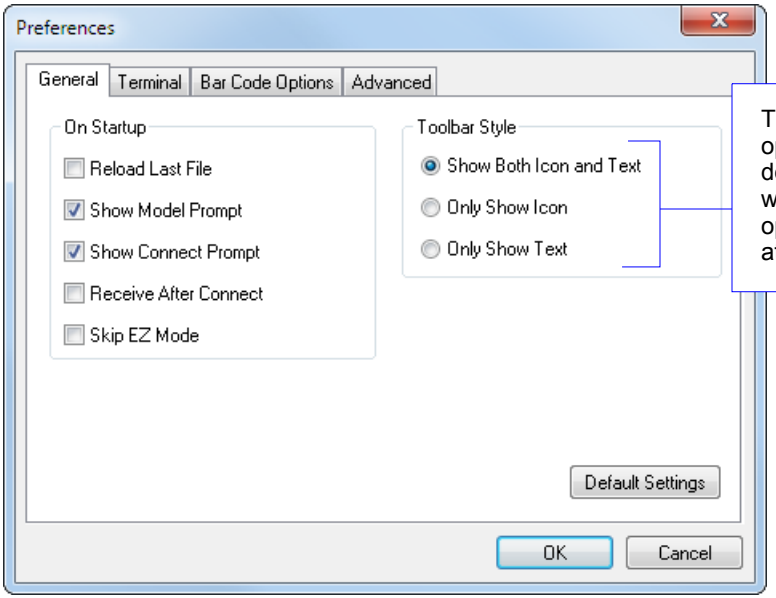
Options

You can use the **Options** menu to save memos and set up **ESP** preferences.

Preferences will be saved and loaded into **ESP** the next time **ESP** is opened, whether or not you save the **ESP** file to the host computer.



Preferences > General Tab



The **Toolbar Style** options allow you to determine how **ESP** will display the mode options in the two rows at the top of the screen.

Reload Last File

At startup, reloads the last file saved to the computer.

Show Model Prompt

At startup, remembers the last connected model and displays it in the **Connecting...** dialog whenever you attempt to connect.

Show Connect Prompt

At startup, displays the **Would you like to connect...** prompt.

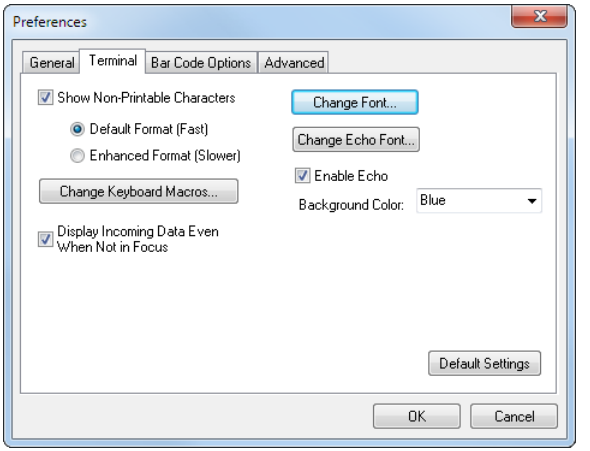
Receive After Connect

At startup, loads the reader's settings into **ESP**. (This is not recommended if you want to preserve your **ESP** settings for future use.)

Skip EZ Mode

At startup, skips **EZ Mode** and opens directly in **App Mode**.

Preferences > Terminal Tab

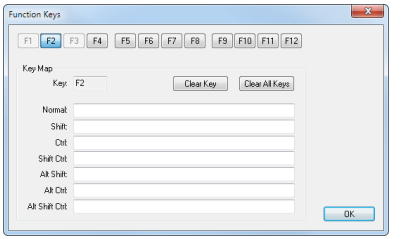


Show Non-Printable Characters

When **Show Non-Printable Characters** is enabled, characters such as “CRLF” will be displayed in the Terminal window. When **Enhanced Format** is checked, the characters are displayed with more detailed formatting.

Change Keyboard Macros

Clicking the **Change Keyboard Macros** button brings up the **Function Keys** dialog. In this dialog you can select the desired function key and then enter your macro keystrokes in the associated key map. For example, to make **Ctrl-F2** the keystroke to send a trigger character, select **F2**, then in the **Ctrl** row, enter **<trigger character>** and click **OK**. Then whenever the **Ctrl-F2** keystroke is pressed, the trigger character will start the read cycle.



Note: The **F1** key is reserved for opening **ESP** Help and the **F3** key is reserved for the **Find Next** function.

Change Font

Allows you to modify the font used for decode data received from the reader on the Terminal screen.

Change Echo Font

Allows you to modify the font used for command characters typed into the Terminal view.

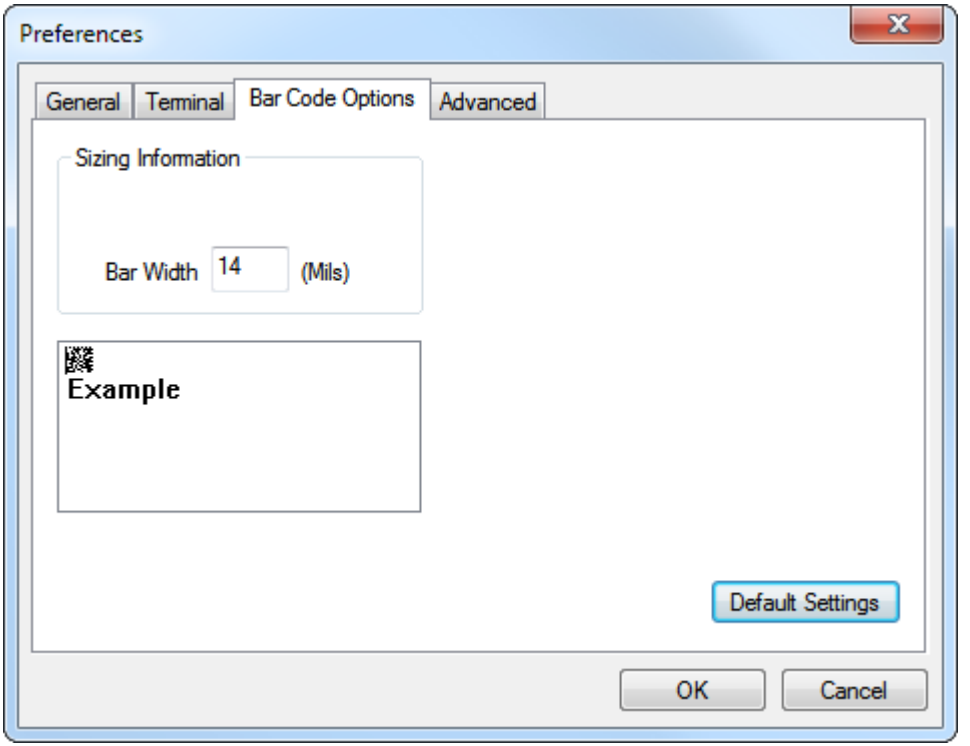
Enable Echo

Allows you to enter command characters in Terminal.

Display Incoming Data Even When Not in Focus

When **Display Incoming Data Even When Not in Focus** is enabled, data from the reader will continue to appear in the Terminal even when **ESP** is not the top window.

Preferences > Bar Code Options Tab



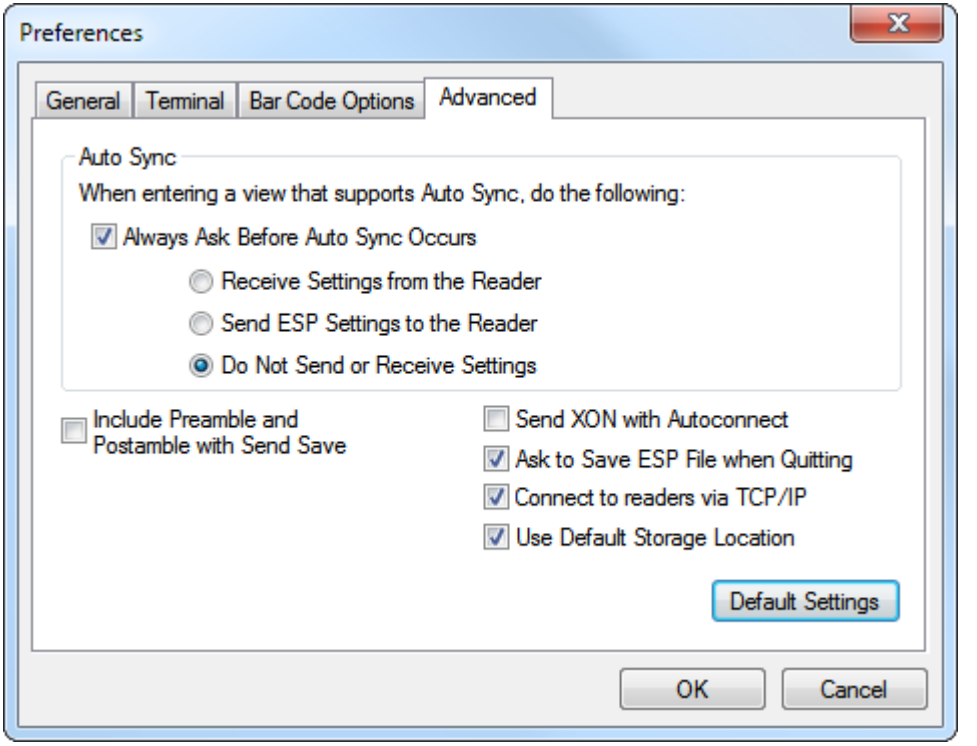
The **Bar Code Options** dialog allows you to set the size of user-created symbols.

Sizing Information

Sets the bar width or module width (in **mils**, or thousandths of an inch) of user-created symbols.

Example: A bar width of 14 is 0.014 inches.

Preferences > Advanced Tab



The **Auto Sync** options at the top of the **Advanced** tab allow the user to determine whether Auto Sync will be enabled automatically in sections of **ESP** where it is used, or if it will ask before it enables Auto Sync functions.

Always Ask Before Auto Sync Occurs

If this option box is checked, specific Auto Sync functions can be enabled. **Receive Settings from the Reader** will automatically send the reader's settings to **ESP** when Auto Sync is enabled. **Send ESP Settings to the Reader** will automatically send all reader configuration settings chosen in **ESP** to the reader. **Do Not Send or Receive Settings** creates a condition in which Auto Sync will not automatically send reader settings to **ESP**, or send **ESP** settings to the reader.

Include Preamble and Postamble with Send Save

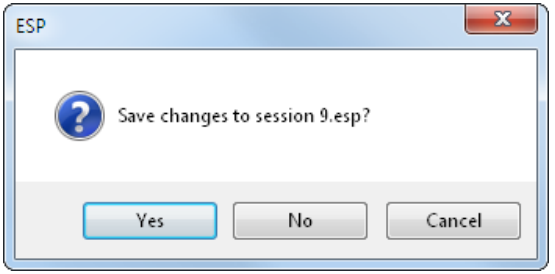
When this option box is checked, the user-configured Preamble and Postamble characters will be sent along with other parameters.

Send XON with Autoconnect

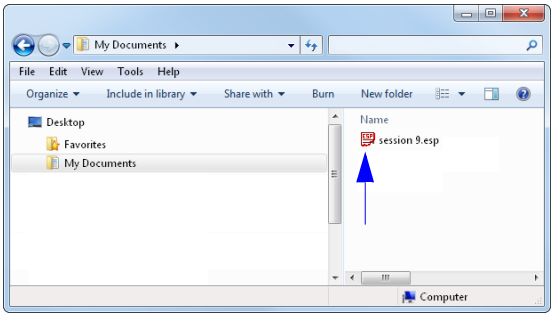
Sends an **XON** (**Begin Transmission**) command to the reader before starting the **Autoconnect** routine.

Ask to Save ESP File when Quitting

When enabled, prompts the user to save a **.esp** file when ending a session.



The **.esp** file will be saved in the location of your choice.

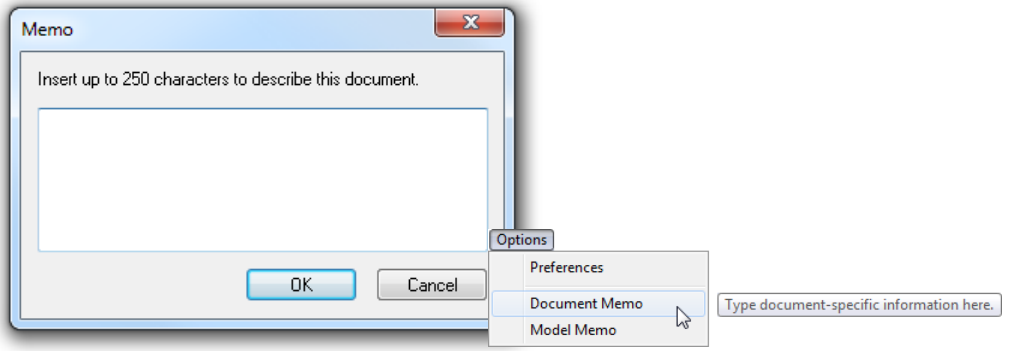


Use Default Storage Location

When enabled, automatically stores data in **ESP**'s Application Data folder.

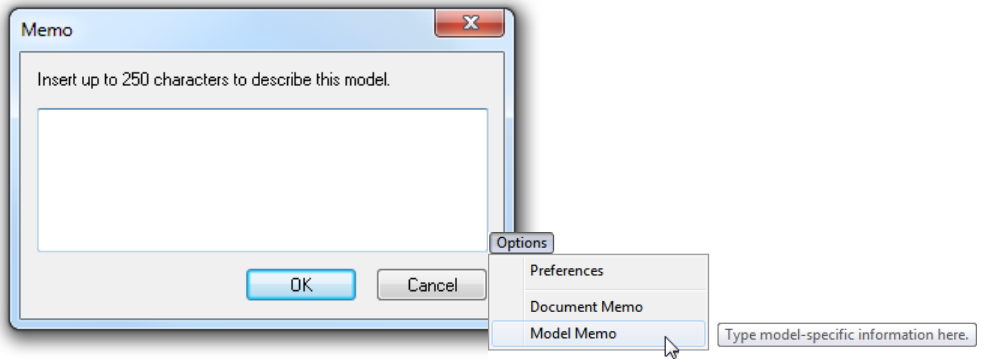
Document Memo

The information you type in the **Document Memo** field will appear in a context-sensitive text box whenever your cursor hovers over the **Document Memo** item on the **Options** menu.



Model Memo

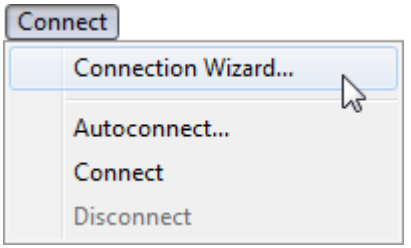
Similar to **Document Memo**, the information you type in the **Model Memo** field will appear in a context-sensitive text box whenever your cursor hovers over the **Model Memo** item on the **Options** menu. Memos created in **Model Memo** are specific to the model enabled when the message was created.



Note: Memos must be saved in a **.esp** file if you want them to available in your next session. If you do not save your current session, any memos that you have entered during the session will be discarded, and will be unavailable in your next session.

Connect

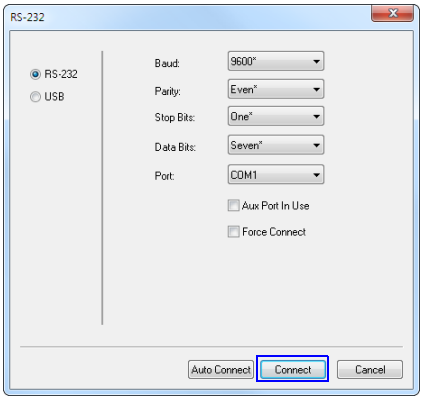
The **Connect** dropdown menu allows the user to access the **Connection Wizard**, as well as the **Autoconnect** and **Configure Multidrop** dialogs. **Connect** and **Disconnect** can also be performed directly from the dropdown menu without opening a dialog.



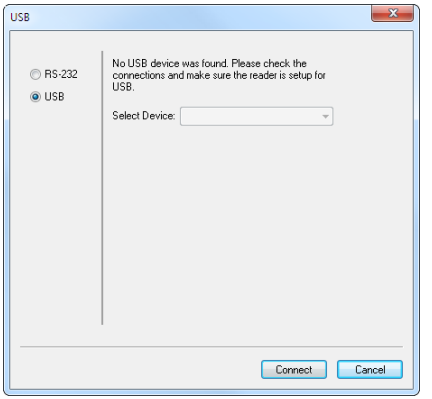
Connection Wizard

To connect using the Connection Wizard:

- Click **Connect** on **ESP**'s menu toolbar, and then select **Connection Wizard**.
- Select **RS-232** or **USB** to activate the appropriate display.
- Configure RS-232 or USB settings as required by the application, and click **Connect**.



RS-232 Connection Wizard



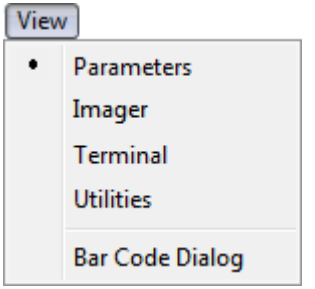
USB Connection Wizard

- When a connection is established, the green indicator in the status bar at the bottom right of the screen will be visible.

- If your RS-232 connection attempt fails, click the **Auto Connect** button to establish a connection between the reader and the host.

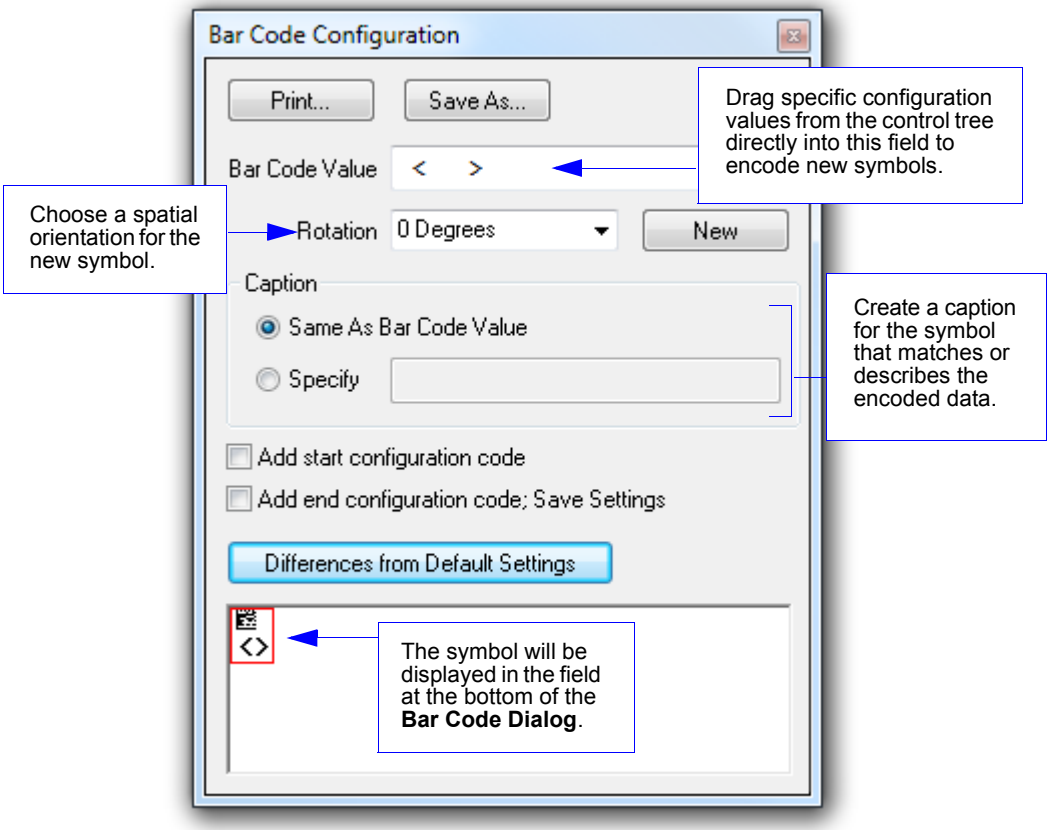
View

The **View** menu allows the user to move quickly between the **Parameters**, **Imager**, **Terminal**, and **Utilities** interfaces without using the icon buttons on the **App Mode** toolbar. It also allows the user to access the **Bar Code Dialog**, shown below.



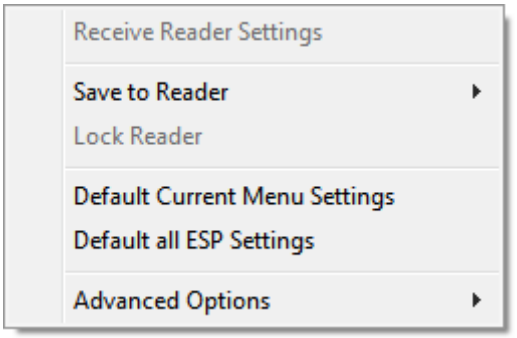
Bar Code Dialog

Symbols can be created in the **Bar Code Dialog** by typing the text to be encoded. This is a useful tool for creating configuration symbols, allowing the user to configure the reader by reading the user-created symbols.



Send/Receive

To access **Receive**, **Save**, **Lock**, **Default**, and **Advanced** options, click the **Send/Recv** button or right-click in the tree control areas..



You can also access these options by right-clicking in any of the configuration views.

Receive Reader Settings

From the **Send/Recv** menu, select **Receive Reader Settings.**

This option is useful if you want to receive the reader's settings and save them as a file for later retrieval. For example, if your reader has settings that you do not want to change, choosing **Receive Reader Settings** will allow you to load those settings to **ESP** and save them as an **ESP** file.

Receiving the reader's settings also assures that you will not subsequently save any unwanted configuration changes previously made in **ESP**.

Select this option if you want to upload the reader's settings to **ESP**. For example, if your **ESP** file has a number of custom settings that you want to maintain and download to the reader, you will lose those **ESP** settings if you choose to receive settings from the reader.

Save to Reader

Send, No Save

This saves **ESP** settings to current memory.

Send and Save

This activates all changes in current memory *and* saves to the reader.

Lock Reader

This locks in the most recently sent and saved configuration to the reader.

Default Current Menu Settings

This option returns the settings in the current tree control to their defaults.

Important: When you select **Default Current Menu Settings** you are *only* defaulting settings in **ESP**. The reader is not affected unless you download new settings.

Default all ESP Settings

This option returns all settings in ESP to their defaults.

Important: When you select **Default all ESP Settings** you are *only* defaulting settings in **ESP**. The reader is not affected unless you download new settings.

Advanced Options

Send Current View

This is the same as **Save to Reader > Send, No Save** except that only the commands in the current tree control are sent.

Send Current Command

This is the same as **Send Current View** except that it only saves the command that is currently selected.

3 Basic Operations

Contents

This section explains how to practice targeting and triggering, and how to begin configuring the reader.

Practice Targeting

When first connecting, allow approximately 3 seconds for the reader to initialize.

1. Hold the reader steady and point it at a test symbol.
2. Squeeze and hold the trigger.
3. Move the reader toward or away from the symbol in a fluid motion until the two side-by-side blue bars converge in the middle of the symbol. When the reader is at the optimal distance (about **4 inches** or **10 cm**), it will decode the symbol and will beep and vibrate while emitting a green LED flash to indicate a Good Read. At this optimal distance, the two blue bars should just be touching. Note that the bars overlap as you continue to draw the reader away from the symbol.
4. If no decode occurs, slowly draw away from or move closer to the symbol while holding the blue bars centered steadily on the symbol.

When the reader is closer to the symbol, you will see two separate bars.

As you draw the reader away from the symbol, the two bars converge. At the optimal distance, the two bars should just be touching, as shown above.

Test Symbol

ABCDEFGHIJKLMNOP

Targeting Suggestions

- Typically, you should not hold the reader exactly perpendicular to the symbol. Position the reader at an angle to avoid specular reflection.
- Use smooth, fluid motion when targeting the symbol. Do not wave the reader side-to-side or up-and-down, or attempt to sweep across a symbol, as sudden movements will create blurred images.
- The reader is omnidirectional and can decode symbols in any orientation. When decoding 1D symbols, be sure that the entire symbol falls well within the field of view.

Motion Detect Mode

The HS-21 or HS-41X can be placed in a presentation stand and used in Motion Detect Mode.

Important: **Motion Detection** must also be enabled in ESP for Motion Detect Mode to function.

To decode a symbol, simply place it within the reader's field of view. The reader will beep, vibrate, and emit a green LED flash upon Good Read.

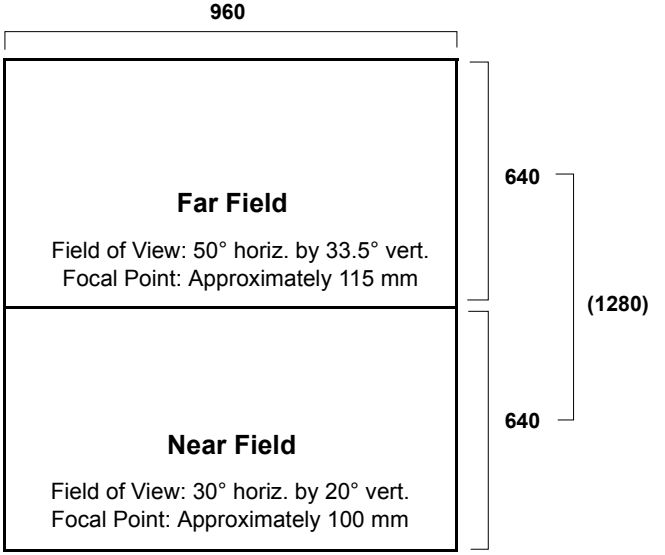
HS-21 with Presentation Stand

Dual Optics

The reader's dual field optical system can read small 2D symbols as well as larger 1D symbols. An image is captured from each field. The decoder first operates on the image (Near or Far) which was successfully decoded on the last cycle. If unsuccessful, the next image is decoded.

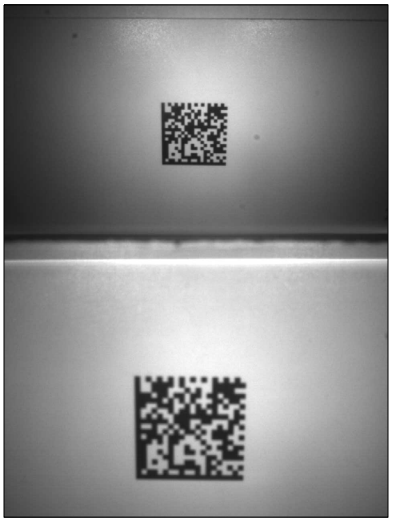
Move the reader closer to decode smaller symbols and farther away to decode larger symbols.

Imaging Area

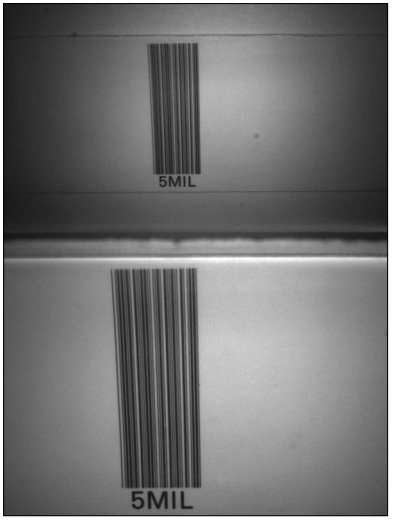


The reader's optics are divided into Near Field and Far Field decode zones. Each decode zone is 960 x 640 pixels.

Dual Optics Examples



20 mil Data Matrix



5 mil Code 39

Operational Feedback

Condition	Top LED Light	Sound	Vibration
Reader Successfully Powers Up	Green LED flashes	1 Beep	Handle Vibrates
Reader Successfully Enumerates with Host (via Cable)	Once enumerated, the green LED turns Off	1 Beep	Handle Vibrates
Attempting to Decode	Green LED is Off	None	No Vibration
Successful Decode and Data Transfer via Cable	Green LED flashes	1 Beep	Handle Vibrates
Configuration Symbol Successfully Decoded and Processed	Green LED flashes	2 Beep	Handle Vibrates
Configuration Symbol Successfully Decoded but Not Successfully Processed	Green LED flashes	4 Beeps	Handle Vibrates

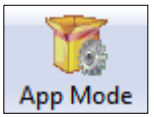
4 Communications

Contents

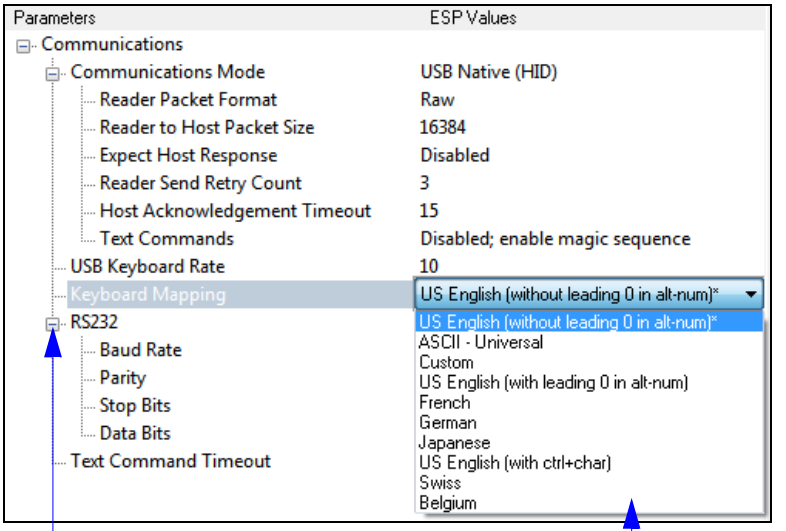
This section explains how to set up communications between the reader and a host.

With Microscan's **ESP** (Easy Setup Program), configuration changes can be made in the **ESP** tree controls and then sent and saved to the reader. The Data Matrix symbols in this section can also be decoded to configure the reader's Communications parameters.

Communications by ESP



Click this button to bring up the **App Mode** view, then click the **Communication** tab.

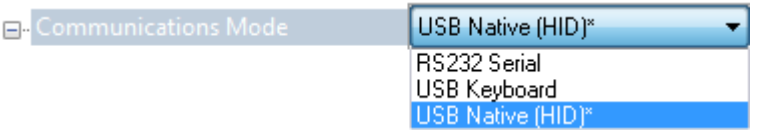


To open nested options, **single-click** the **+**.

To change a setting, **double-click** the setting and use your cursor to scroll through the options.

Communications Overview

Whenever you default the reader, it will return to the default settings of whichever interface you are using. Defaulting the reader does not remove preamble and postamble formatting. The reader is in **USB Native (HID)** by default.



USB

With USB communications, the reader connects directly to the host's USB port from which it draws its power. Data is displayed by any open Windows-based program that can capture text in USB Keyboard Mode.

RS-232

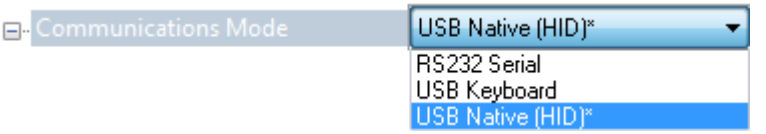
With RS-232 communications the reader communicates with the host through a communications program such as ESP's Terminal.

Default settings for establishing RS-232 communications are:

Baud Rate: **115.2K**
Parity: **None**
Stop Bits: **1**
Data Bits: **8**

USB Interface

The reader is in **USB Native (HID)** by default.



USB Native (HID)

This mode is the standard way of transferring unformatted, unpacketized data to the reader through the USB port.


Q0003_1

USB Keyboard Mode (Windows)

Data is output as keyboard sequences.


M0002_01

USB Virtual COM One-Way Mode

This mode allows a reader in a USB configuration to function as a virtual serial COM port. This mode requires installation of a USB Virtual COM driver, which is available from Microscan by request.


M0005_01

RS-232 Interface

Enabling **RS-232 Interface** will disable USB communications and require you to default the reader or read the USB Keyboard Mode symbol to return to USB.

Baud Rate (RS-232)

Baud Rate is the rate at which the reader and host transfer data. It only needs to be changed if necessary to match the host setting.


M0020_01

1200


M0024_01

19.2K


M0021_01

2400


M0025_01

38.4K


M0022_01

4800


M0026_01

57.6K


M0023_01

9600


M0027_01

115.2K (Default)

Parity (RS-232)

Parity is an error detection routine in which one data bit in each character is set to **1** or **0** so that the total number of 1 bits in the data field is even or odd. It only needs to be changed if necessary to match the host setting.


M0030_01

None (Default)


M0029_01

Odd


M0028_01

Even

Data Bits (RS-232)

Data Bits are the total number of bits in each character. This setting only needs to be changed if necessary to match the host setting.


M0019_01

8 Data Bits (Default)


M0018_01

7 Data Bits

Preamble

A **preamble** is a character or series of characters that is added to the beginning of a decoded data string. Preamble characters will appear in the order that they are enabled (left to right). For example, if you enable a comma and then a space, and then decode a symbol containing the data 'ABC', your output will look like this:

```
, ABC
```

The only limit to the number of preambles enabled is the total memory size available. Set the desired preamble by reading the appropriate symbol below.


M0130_02

Comma


M0131_02

Space


M0133_02

Tab (RS-232 Only)


M0132_02

Tab (USB Keyboard Only)


M0135_02

Carriage Return Line Feed (RS-232 Only)


M0134_02

Erase (None)


M0145_02

Erase Preamble and Postamble Data

Postamble

A **postamble** is a character or series of characters that is added to the end of a decoded data string. Postamble characters will appear in the order that they are enabled (left to right). For example, if you enable a space and then a comma, and then decode a symbol containing the data 'ABC', your output will look like this:

ABC ,

The only limit to the number of postambles enabled is the total memory size available.

Set the desired postamble by reading the appropriate symbol below.


M0137_02

Comma


M0140_02

Space


M0143_02

Tab (RS-232 Only)


M0142_02

Tab (USB Keyboard Only)


M0136_02

Carriage Return (RS-232 Only)


M0138_02

Line Feed (RS-232 Only)


M0139_02

Carriage Return Line Feed (RS-232 Only)


M0141_02

Enter (USB Keboard Only)


M0144_02

Erase (None)

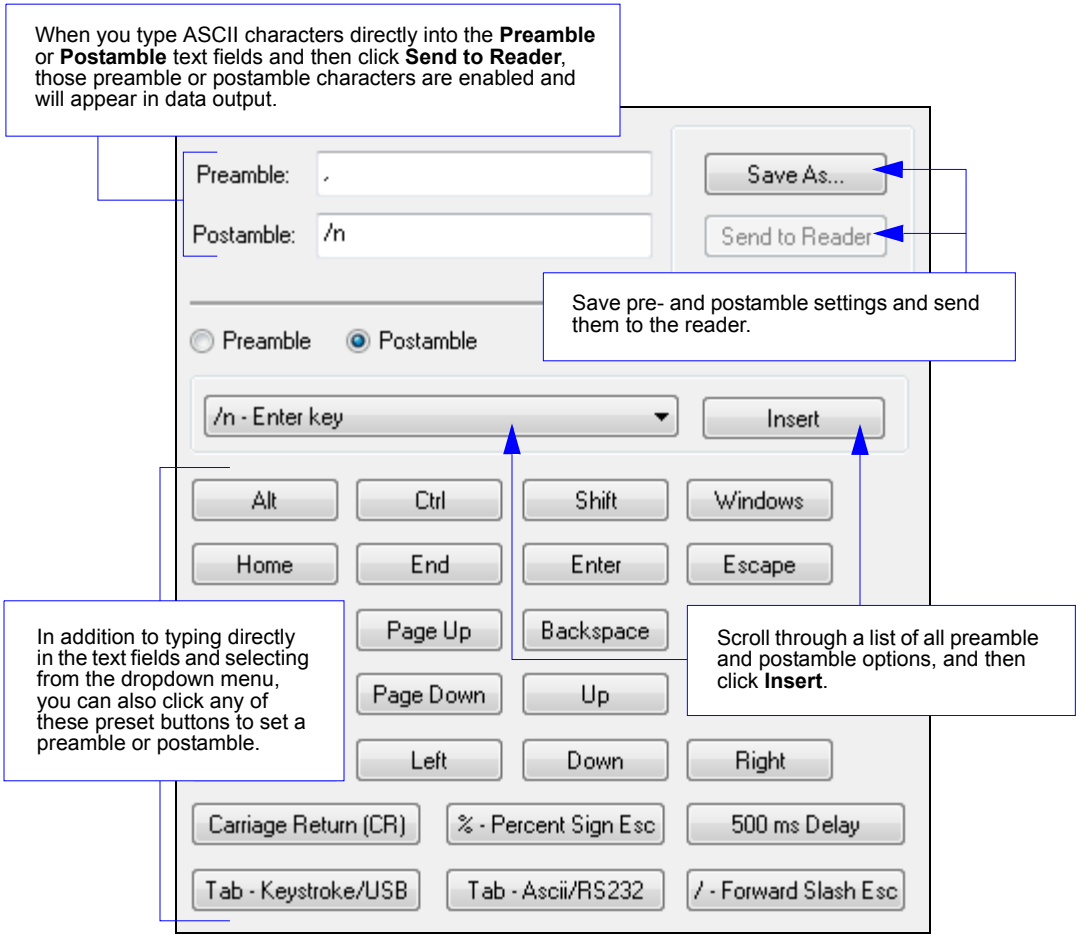

M0145_02

Erase Preamble and Postamble Data

Preamble and Postamble by ESP

Characters can also be added to the beginning and end of data strings using **ESP**. There are a few different ways to do this, using the interface shown below.

You will see the Communications tree control on the left, and the Preamble/Postamble interface on the right.



Keyboard Mapping

The **Keyboard Mapping** feature provides alternatives for keyboards that do not conform to U.S. English mapping.

Note: Universal Keyboard mapping is slightly slower than the other language-specific options, because it maps data by reference to the full set of ASCII characters. The advantage of Universal Keyboard mapping is that it allows any language and keyboard layout to be mapped.

Important: Keyboard Mapping is not to be confused with USB Keyboard Mode, which has an entirely different function—namely to enable USB cabled communications.


M0008_02
U.S., No Leading 0 (Default)


M0007_02
U.S. with Leading 0


M0009_02
U.S., Ctrl + Char.


M0010_01
French


M0011_01
German


M0012_01
Japanese

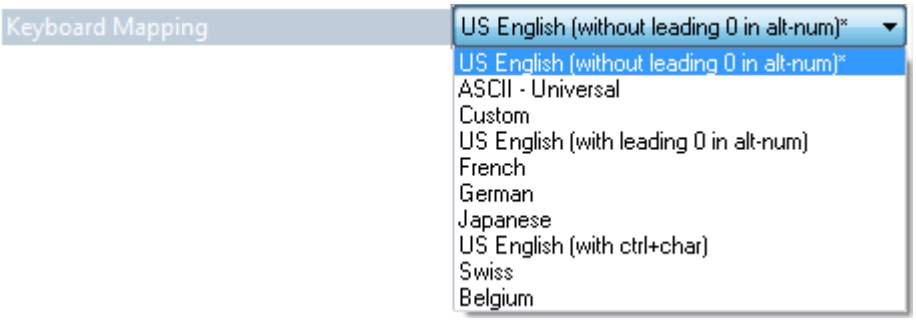

M0013_01
Universal


M0014_01
Custom


M0189_01
Belgian


M0190_01
Swiss

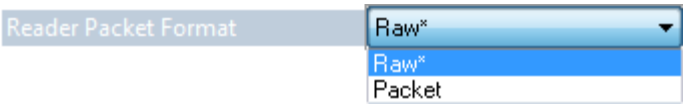
Keyboard Mapping by ESP



Communications Mode

Some **ESP** Communications options are unique to the software, and do not have corresponding programming symbols. These options are explained below.

Reader Packet Format



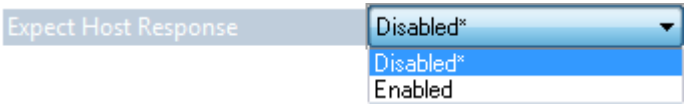
Data that is sent from the reader to the host in **Raw** format is sent without packet framing or check characters. **One-Way** communication is in a raw format, no response is expected from the host, and data is not resent.

Packetized data is sent with framing (a preamble communicating the amount of data to be transmitted, and a postamble containing error detection) and check characters, and a response is expected from the host. **Two-Way** communication is in packet format.

Reader to Host Packet Size

The **Reader to Host Packet Size** is the amount of data (in bytes) that is sent to the host in packet format. This feature allows you to set the maximum allowable packet size.

Expect Host Response



When **Expect Host Response** is enabled, the reader will re-transmit data if it doesn't receive acknowledgement from the host.

Reader Send Retry Count

Reader Send Retry Count sets the number of times the reader will re-transmit data before abandoning further send attempts. The minimum retry count is **1**, which represents the initial transmission.

Host Acknowledgement Timeout

The **Host Acknowledgement Timeout** is the amount of time (in seconds) that the reader will wait for an acknowledgement from the host before re-sending data.

Text Commands

When the **Text Commands** feature is enabled, the reader can accept text commands via RS-232 connections and USB Virtual COM modes.

Note: Text Commands are not supported in USB HID Mode.

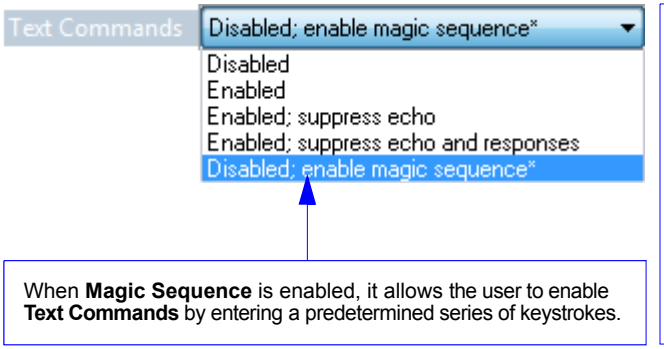
M0146_01

Enable Text Commands

M0147_01

Disable Text Commands (Default)

Text Commands by ESP



When **Magic Sequence** is enabled, it allows the user to enable **Text Commands** by entering a predetermined series of keystrokes.

When **Text Commands** are set to **Enabled; Suppress Echo**, text that a user enters in the Terminal will not be shown. When **Text Commands** are set to **Enabled; Suppress Echo and Responses**, neither user-entered data or reader responses will be shown, and only decoded symbol data will appear in the Terminal.

See **Terminal Right-Click Menu** for a way to change Echo settings directly in the Terminal view.

Entering Magic Sequence

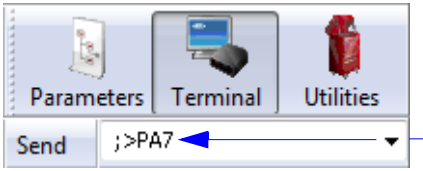
The magic sequence is **;>PA** followed by a numeric value of **1**, **3**, or **7**.

1 = Enable Text Commands

3 = Enabled; Suppress Echo

7 = Enabled; Suppress Echo and Responses

In the example below, the magic sequence entered will Enable Text Commands and Suppress Echo and Responses.



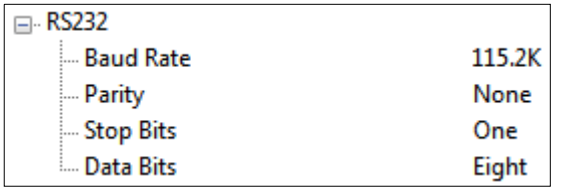
Enter the magic sequence in this text field and click **Send**.

Once the magic sequence has been sent, you can send text commands from the same text field.

USB Keyboard Rate

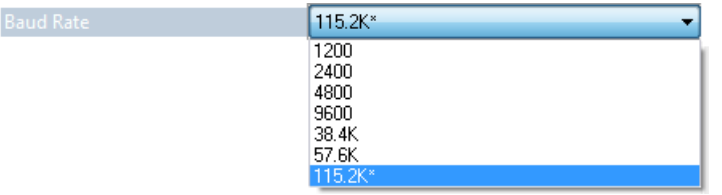
Requests that the host polls the USB reader at the rate specified (**1** to **255** ms).

RS-232



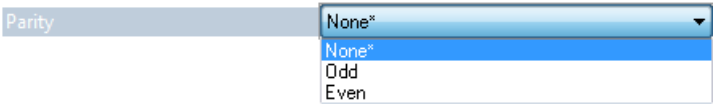
Baud Rate

Baud Rate is the rate at which the reader and host transfer data. It only needs to be changed if necessary to match the host setting.



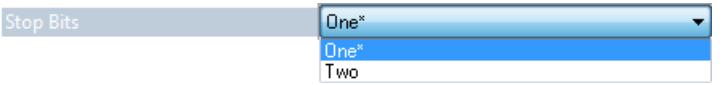
Parity

Parity is an error detection routine in which one data bit in each character is set to **1** or **0** so that the total number of 1 bits in the data field is even or odd. It only needs to be changed if necessary to match the host setting.



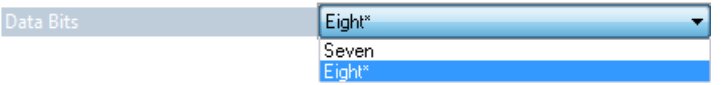
Stop Bits

Stop Bits are added to indicate the end of each character. This setting should only be changed if necessary to match the host setting.



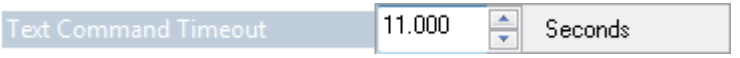
Data Bits

Data Bits are the total number of bits in each character. This setting only needs to be changed if necessary to match the host setting.



Text Command Timeout

Text Command Timeout allows you to set the maximum time during which a complete text command from the host must be received. Pending text command data is discarded when the timeout is exceeded.

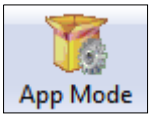


5 Read Cycle

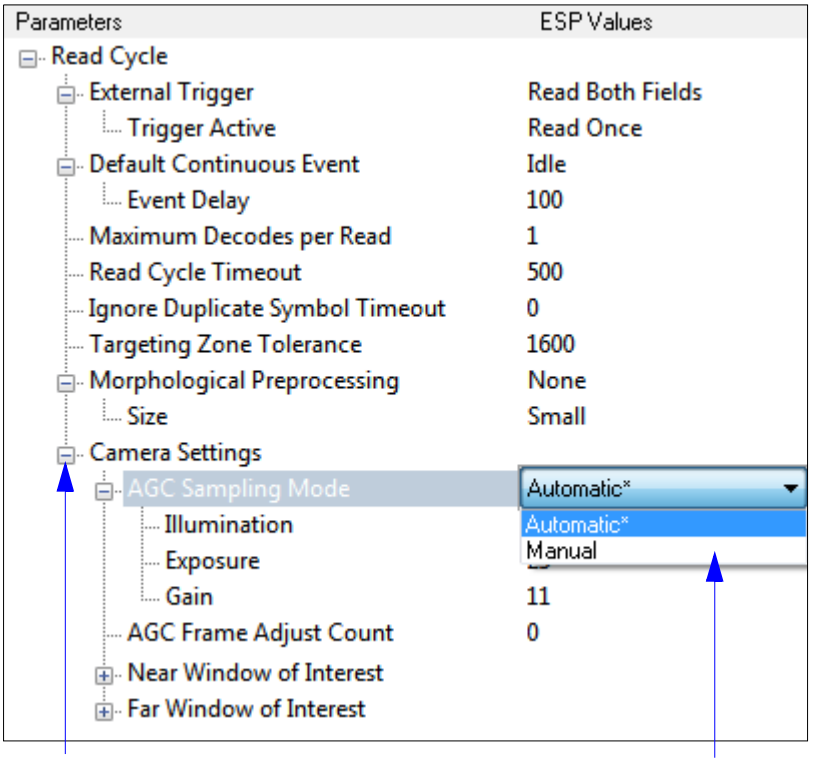
Contents

After you've established communications you will need to address the spatial and timing parameters associated with your application. This section explains those parameters. The Data Matrix symbols in this section can also be decoded to configure Read Cycle parameters.

Read Cycle by ESP



Click this button to bring up the **App Mode** view, and then click the **Read Cycle** tab.

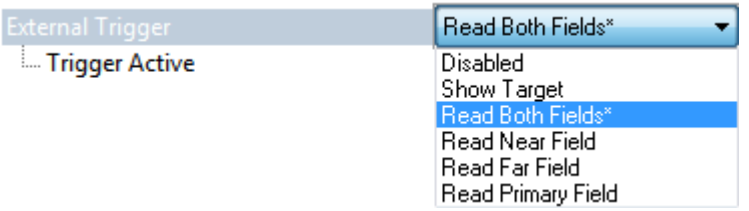


To open nested options, **single-click** the **+**.

To change a setting, **double-click** the setting and use your cursor to scroll through the options.

External Trigger

The **External Trigger** parameter allows you to determine reader behavior when triggered externally.



Show Target

The target LEDs will illuminate when the reader is triggered externally.

Read Both Fields (Default)

Both Near Field and Far Field will be activated to capture an image when the reader is triggered externally.

Read Near Field

Near Field will be activated to capture an image when the reader is triggered externally.

Read Far Field

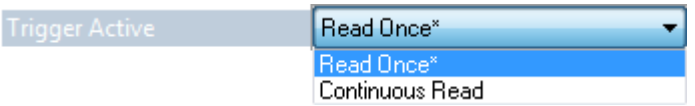
Far Field will be activated to capture an image when the reader is triggered externally.

Read Primary Field

When **Read Primary Field** is selected, the most recent field to have produced a Good Read (Near Field or Far Field) will be activated to capture an image when the reader is triggered externally.

Trigger Active

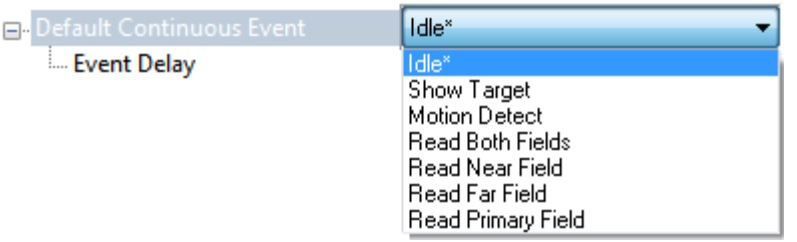
When an external trigger is active, the reader will either decode once and stop or decode continuously, depending on how this parameter is set. **Trigger Active** is set to Read Once by default.



Important: **Ignore Duplicate Symbol Timeout** should be set to a value greater than 0 when Trigger Active is set to Continuous Read.

Default Continuous Event

This parameter allows you to determine the default state of the reader.



Idle (Default)

When Default Continuous Event is set to **Idle**, the reader will remain inactive until triggered.

Show Target

When Default Continuous Event is set to **Show Target**, the reader will display the target LEDs but remain inactive until triggered externally.

Motion Detect

When Default Continuous Event is set to **Motion Detect**, the reader will remain inactive until motion occurs in the field of view (if a symbol is hand-presented, for example).

Read Both Fields

Both Near Field and Far Field will be continuously activated to capture an image.

Read Near Field

Near Field will be continuously activated to capture an image.

Read Far Field

Far Field will be continuously activated to capture an image.

Read Primary Field

When **Read Primary Field** is selected, the most recent field to have produced a Good Read (Near Field or Far Field) will be continuously activated to capture an image.

Event Delay

The default Event Delay is 0.100 seconds.

Maximum Decodes per Read

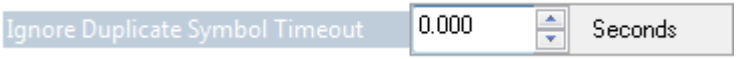
Maximum Decodes per Read allows you to set how many decodes can be performed in a single read cycle.

Read Cycle Timeout

Read Cycle Timeout determines the duration of the read cycle. The default Read Cycle Timeout is 0.500 seconds.

Ignore Duplicate Symbol Timeout

Ignore Duplicate Symbol Timeout sets the reader not to output the same symbol data multiple times within the time period designated.



Targeting Zone Tolerance

Targeting Zone Tolerance is particularly useful in environments where closely spaced symbols of various sizes need to be targeted. It allows the reader to narrow the field of view relative to the size of a symbol, and to determine the distance the target must be from the symbol for a decode event to occur.

See **Window of Interest** for more precise control of the active pixel area.

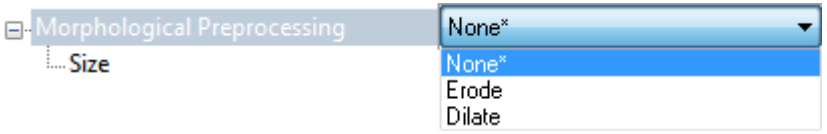
The default Targeting Zone Tolerance is 1600%.

Formula for calculating Targeting Zone Tolerance:

2 x distance from target to symbol (in pixels) / symbol width or height (in pixels) x 100

Morphological Preprocessing

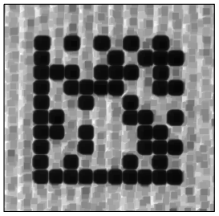
Morphological Preprocessing allows you to select the method for processing captured images, and to choose the operator size for that method. It is set to None by default.



Note: This feature is only available in the HS-41X Handheld Reader.

Erode

Erode increases the dark cell size of a symbol. Useful for increasing the dark cell size of a dark-on-light Data Matrix symbol.

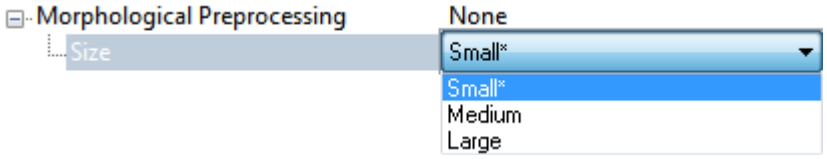


Dilate

Dilate increases the light cell size of a symbol. Useful for increasing the light cell size of a light-on-dark Data Matrix symbol.

Size

Size determines the size of the area or “pixel neighborhood” (measured in pixels) in which the morphological operation is being performed.

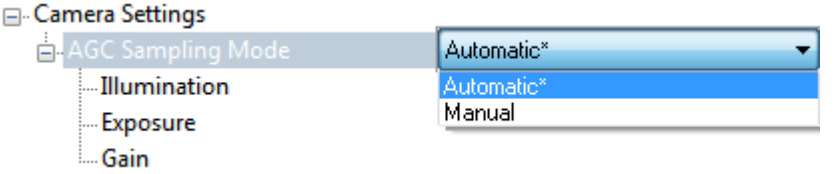


Camera Settings

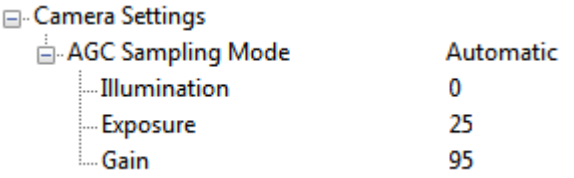
Camera Settings allow you to set AGC Sampling Mode, to set the percentage values for Illumination, Exposure, and Gain, to set the AGC Frame Adjust Count, and also to define Window of Interest dimensions.

AGC Sampling Mode

When **AGC Sampling Mode** is set to Automatic (default), each time a No Read occurs, the reader adjusts the gain and exposure for the next capture to optimize symbol contrast.



The values for **Illumination**, **Exposure**, and **Gain** can be set to any value between 0% and 100%. The default values are shown below.



AGC Frame Adjust Count

Automatic Gain Control (AGC) is a system that controls gain in order to maintain high performance over a range of input levels. Gain is essentially the ratio of output to input. Gain settings affect how the reader decodes symbols and captures images.

AGC Frame Adjust Count sets the number of image frames captured and discarded before the main image capture. This feature gives the gain control time to adjust.

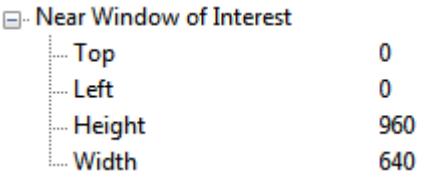
Window of Interest

The active pixel area of the image sensor is called the **Window of Interest** (WOI). The WOI allows the user to select an area of the field of view in which the desired symbol is located.

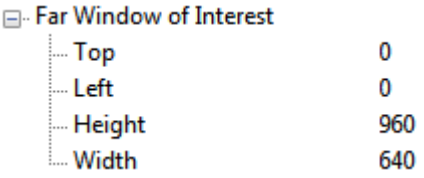
The programmable window of interest increases decode speed, improves threshold, and makes it easy to select specific symbols from among several in the field of view. The user provides the upper-left pixel location and the size of the window to define the Window of Interest.

Window of Interest can also be controlled using a graphic interface in ESP.

High Density



Wide

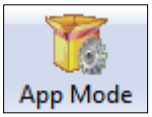


6 Symbologies

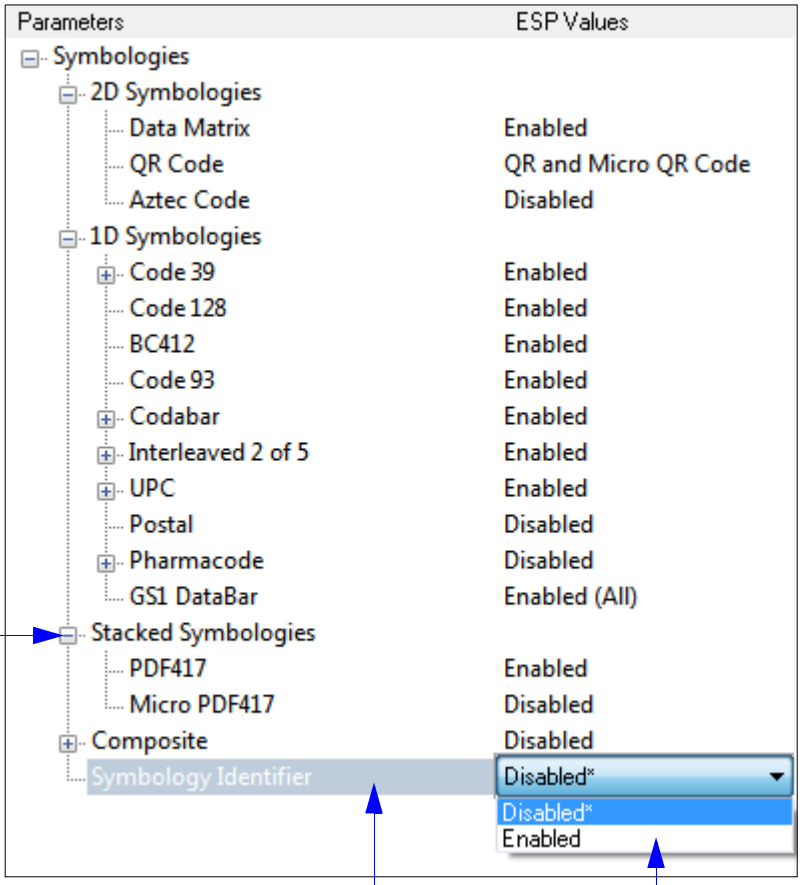
Contents

This section describes the various symbol types that can be decoded by the HS-21 and HS-41X. The Data Matrix symbols in this section can also be decoded to configure Symbologies parameters.

Symbologies by ESP



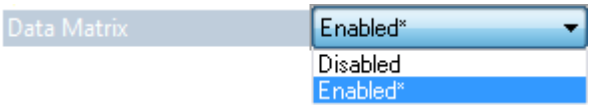
Click this button to bring up the **App Mode** view, and then click the **Symbologies** tab.



To open nested options, **single-click** the **+**.

To change a setting, **double-click** the setting and use your cursor to scroll through the options.

Data Matrix



Data Matrix Enabled (Default)

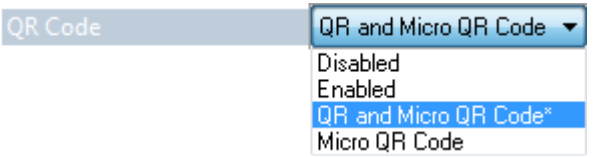
Data Matrix Disabled

Sample Data Matrix Symbol

If you disable the Data Matrix symbology, programming symbols will not be decodable by the reader and Data Matrix will need to be re-enabled using ESP.

Use the **Data Matrix Disabled** programming symbol with caution.

QR Code



QR Code Inverse and Standard Enabled

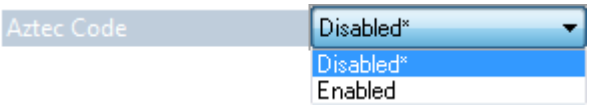
QR Code Disabled

QR Code and Micro QR Code Enabled

Sample QR Code Symbol

Sample Micro QR Code Symbol

Aztec

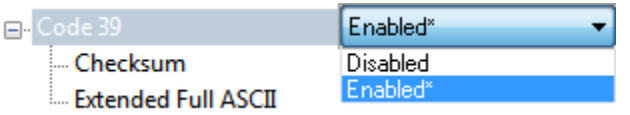


Aztec Enabled

Aztec Disabled (Default)

Sample Aztec Symbol

Code 39



Code 39 Disabled


M0047_01

Code 39 Enabled (Default)


M0046_01

Code 39 Enable Checksum


M0048_01

Code 39 Disable Checksum (Default)


M0049_01

Code 39 Enable Checksum and Strip from Result


M0050_01

Code 39 Extended Full ASCII Enabled

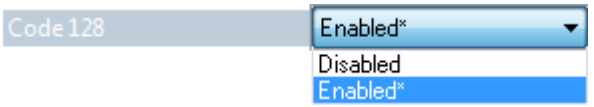

M0051_01

Code 39 Extended Full ASCII Disabled (Default)


M0052_01

Sample Code 39 Symbol

Code 128

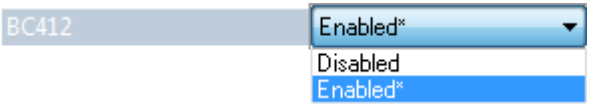


Code 128 Enabled (Default)

Code 128 Disabled

Sample Code 128 Symbol

BC412

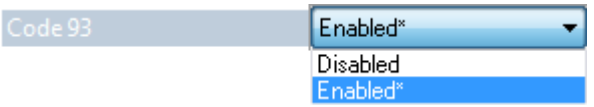


BC412 Enabled (Default)

BC412 Disabled

Sample BC412 Symbol

Code 93



Code 93 Enabled (Default)

M0059_01

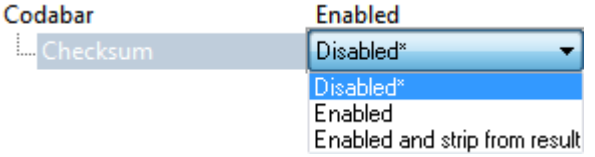
Code 93 Disabled

M0060_01

Sample Code 93 Symbol

123456789A

Codabar



Codabar Enabled (Default)

Codabar Disabled

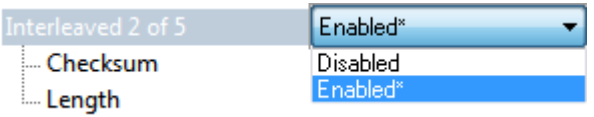
Codabar Checksum Enabled

Codabar Checksum Disabled (Default)

Codabar Checksum Enabled and Strip from Result

Sample Codabar Symbol

Interleaved 2 of 5



Interleaved 2 of 5 Enabled (Default)

Interleaved 2 of 5 Disabled

Interleaved 2 of 5 Checksum Enabled

Interleaved 2 of 5 Checksum Disabled (Default)

Interleaved 2 of 5 Checksum Enabled and Strip from Result

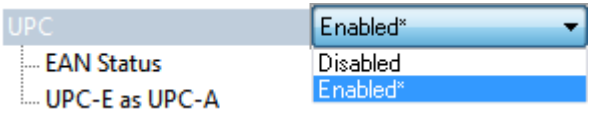
Interleaved 2 of 5 Two Digit Minimum

Interleaved 2 of 5 Four Digit Minimum

Interleaved 2 of 5 Six Digit Minimum (Default)

Sample Interleaved 2 of 5 Symbol

UPC



UPC Enabled (Default)

M0120_01

UPC Disabled

M0121_01

EAN Status Enabled (Default)

Q0016_01

EAN Status Disabled

Q0017_01

UPC-E as UPC-A Enabled

Q0018_01

UPC-E as UPC-A Disabled (Default)

Q0019_01

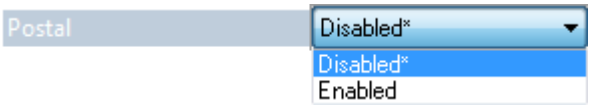
Sample UPC-E Symbol

0 012345

Sample UPC-A Symbol

0 12345 67891

Postal



Postal Enabled


Q0031_01

Postal Disabled (Default)


M0102_01

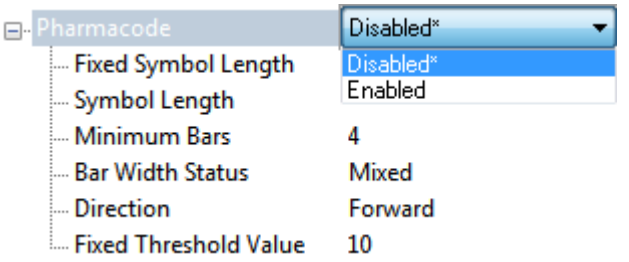
Supported Postal Symbologies

- USPS OneCode (4CB)
- POSTNET
- PLANET
- Japanese Post
- Australian Post
- Royal Mail
- KIX Code

Sample Postnet Symbol

Sample Royal Mail Symbol

Pharmacode



Pharmacode Enabled


Q0020_01

Pharmacode Disabled (Default)


Q0021_01

Fixed Symbol Length Enabled


Q0022_01

Fixed Symbol Length Disabled (Default)


Q0023_01

Bar Width Status: Mixed (Default)


Q0024_01

Bar Width Status: All Narrow


Q0025_01

Bar Width Status: All Wide


Q0026_01

Bar Width Status: Fixed Threshold


Q0027_01

Direction: Forward (Default)


Q0028_01

Direction: Reverse


Q0029_01

Fixed Symbol Length Status

When enabled, the reader will check the symbol length against the symbol length field. If disabled, any length will be considered valid.

Symbol Length

Specifies the exact number of bars that must be present for the reader to recognize and decode the Pharmacode symbol.

Minimum Bars

Sets the minimum number of bars that a Pharmacode symbol must have to be considered valid.

Bar Width Status

If set to **Mixed**, the reader will autodiscriminate between narrow bars and wide bars. If set to **All Narrow**, all bars will be considered as narrow bars. If set to **All Wide**, all bars will be considered as wide bars. If set to **Fixed Threshold**, it will use the fixed threshold value to determine whether the bars are narrow or wide. The **Bar Width Status** setting will be ignored when the reader is able to tell the difference between the narrow and the wide bars.

Direction

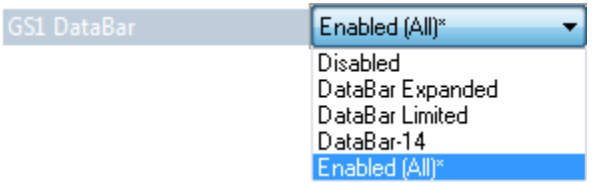
Specifies the direction in which a symbol can be read.

Fixed Threshold Value

Used when **Bar Width Status** is set to **Fixed Threshold**. Defines the minimum difference in pixels that will distinguish a narrow bar from a wide bar.

Sample Pharmacode Symbol

GS1 DataBar



All GS1 DataBar Enabled (Default)

All GS1 DataBar Disabled

GS1 DataBar Expanded Enabled

GS1 DataBar Limited Enabled

GS1 DataBar-14 Enabled

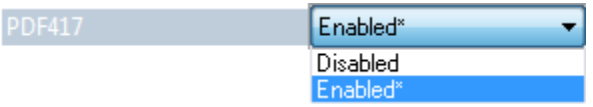
Sample DataBar-14 Limited Symbol

Sample DataBar-14 Stacked Symbol

Sample DataBar Expanded Symbol

Sample DataBar-14 Symbol

PDF417

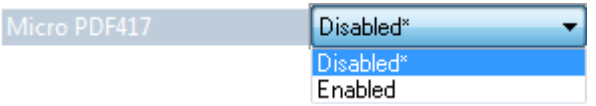


PDF417 Enabled (Default)

PDF417 Disabled

Sample PDF417 Symbol

MicroPDF417



MicroPDF417 Disabled (Default)

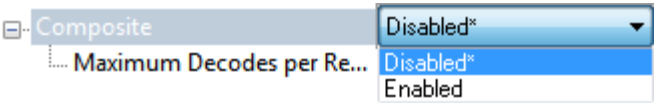
MicroPDF417 Enabled

Sample MicroPDF417 Symbol

Composite

Composite consists of a 1D component associated with an adjacent 2D component. A successful decode is required for both the 1D and 2D components before the reader outputs a result. When Composite is enabled, the unit decodes the 1D component first.

Important: EAN-8, EAN-13, UPC-A, and UPC-E cannot be decoded individually when Composite is enabled.



Maximum Decodes per Read

Maximum Decodes per Read represents the maximum number of candidate symbols in the field of view (1 - 100) that can be decoded during a read cycle. Note that decode speed will decrease as the Maximum Decodes per Read value is increased.

Composite Disabled (Default)

Composite Enabled

Sample Composite Symbol

Symbology Identifier

When **Symbology Identifier** is enabled, an AIM (Association for Automatic Identification and Mobility) preamble is added to decoded data output (see the **AIM Symbology Identifiers** list). This preamble identifies what kind of symbology has been decoded.

AIM Symbology Identifiers

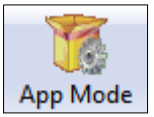
A	Code 39
C	Code 128
d	Data Matrix
e	GS1 DataBar / Composite
E	UPC/EAN
F	Codabar
G	Code 93
I	Interleaved 2 of 5
L	PDF417 / MicroPDF417
Q	QR Code / Micro QR Code
X	Other (Pharmacode)
z	Aztec

7 I/O Parameters

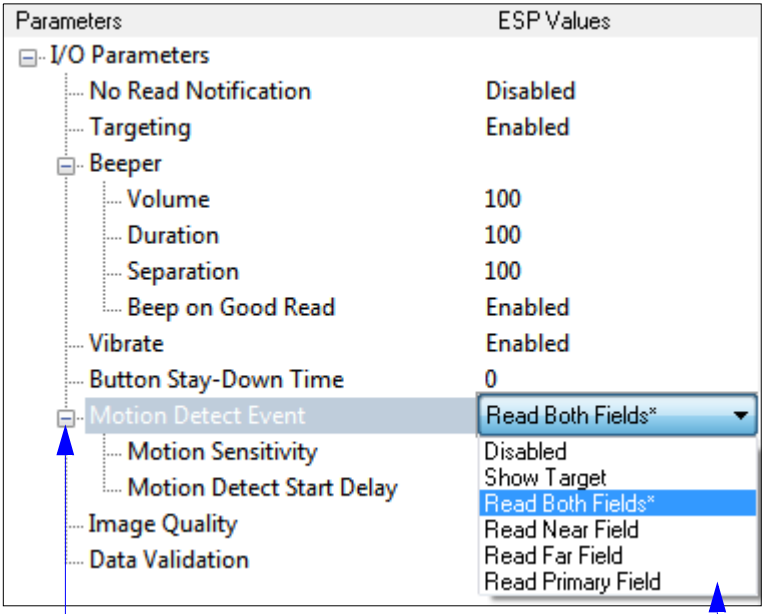
Contents

This section includes instructions on setting up conditions for changing input/output electrical transitions for control of the reader's internal and external devices. A discrete I/O (in/out) signal is an electrical transition from one voltage level to another so that digital switching can occur. The Data Matrix symbols in this section can also be decoded to configure I/O parameters.

I/O Parameters by ESP



Click this button to bring up the **App Mode** view, and then click the **Read Cycle** tab.



To open nested options, **single-click** the **+**.

To change a setting, **double-click** the setting and use your cursor to scroll through the options.

No Read Notification

No Read Notification allows you to enable or disable user feedback alerting you when a symbol is not decoded successfully.

The No Read message output is **ap/r**, indicating that the reader did not decode the symbol.

Targeting

The **Targeting** parameter allows you to turn the targeting LEDs on or off. They are on by default.

Read the configuration symbols below to enable or disable **Targeting**.


M0196_01

Targeting Off


M0197_01

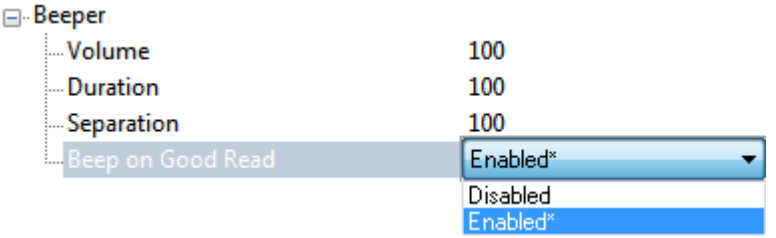
Targeting On

Beeper

The **Beeper** parameters allow you to set the Volume, Duration, and Separation of the beep, and whether or not it will beep on a Good Read.

Beeper volume is 100% by default, 0.100 seconds Duration by default, and 0.100 seconds Separation by default.

Beep on Good Read is enabled by default.



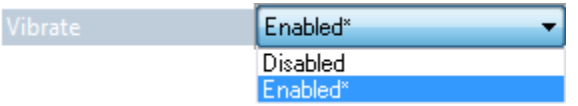
Read the configuration symbols below to enable or disable **Beeper**.

Beeper Off

Beeper On

Vibrate

The **Vibrate** parameter allows you to turn Vibrate on or off. It is on by default.

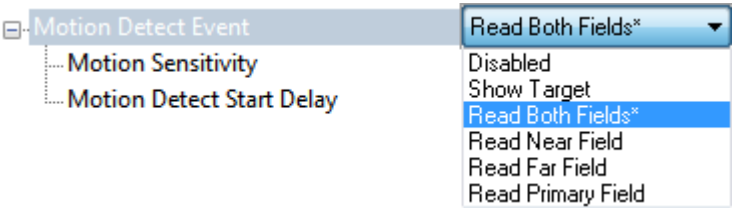


Button Stay-Down Time

Button Stay-Down Time sets the amount of time (in seconds) that the reader will continue to process the current “decode symbol” event. The reader will behave as if the trigger is being activated for this specified amount of time.

Motion Detect Event

Motion Detect Event allows you to determine the reader's behavior when motion is detected in the field of view.



Motion Sensitivity allows you to determine the sensitivity of motion detection (lower is more sensitive - 5 is default).

Motion Detect Start Delay allows you to set the amount of delay before a motion detect event occurs. (0 seconds is default.)

Show Target

The target LEDs will illuminate when a motion detect event occurs.

Read Both Fields (Default)

Both Near Field and Far Field will be activated to capture an image when a motion detect event occurs.

Read Near Field

Near Field will be activated to capture an image when a motion detect event occurs.

Read Far Field

Far Field will be activated to capture an image when a motion detect event occurs.

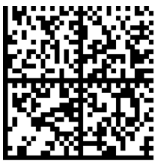
Read Primary Field

When Read Primary Field is selected, the most recent field to have produced a Good Read (Near Field or Far Field) will be activated to capture an image when a motion detect event occurs.

Read the configuration symbols below to enable or disable **Motion Detection**.


M0129_07

Motion Detection Off


M0162_01

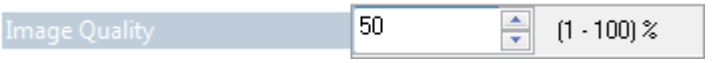
Motion Detection On, Start Delay 500 ms


M0161_01

Motion Detection On Start Delay 0

Image Quality

Image Quality allows you to determine the quality of images that are output from image captures. Image Quality is set to 50% by default.



Data Validation

Data Validation is used to confirm that a decoded string from the imager has complied with a particular company, industry, or ISO standard. HS-21/HS-41X Data Validation is compliant with Department of Defense Unique Identification and ISO/IEC 15434 (Information Technology – Transfer Syntax for High-Capacity ADC Media) requirements.

Unique Identification

Unique Identification is a mandatory Department of Defense (DoD) requirement on all solicitations issued January 1, 2004 or later. This policy mandates the use of Unique Item Identifiers (UIIs) encoded within Data Matrix symbols on equipment and parts procured by DoD. The HS-21/HS-41X complies with Department of Defense Standard Practice Identification (MIL-STD-130).

Once the imager decodes the Data Matrix symbol, and if Unique Item Identifier (UII), Current Part Number (CPN), and Lot/Batch Number (LBN) are turned on, the HS-21/HS-41X checks the ISO/IEC 15434 syntax with ISO/IEC 15418 (ANSI MH10.8.2 – AI and DI) and ISO/IEC 21849 (ATA – TEI) semantics to construct the UII, CPN, and LBN.

Unique Identification Features

The following data output options are applicable to Data Matrix ECC 200 symbols only and have no effect on other symbologies.

UII Enabled	Allows the imager to read only message streams encoded in Data Matrix ECC 200 symbols, then to construct and output a UII string. The message streams include validation of Unique Item Identifier (UII), Current Part Number (CPN), and Lot/Batch Number (LBN) strings. When the imager decodes a symbol but the symbol data does not compy with UII format, it will stop capturing images and the green LED will illuminate without beeping, vibrating, or outputing the string.
UII Enabled with Pass Through	Allows the imager to read UII messages in Data Matrix ECC 200 symbols and non-UII messages in any type of symbols. The imager's behavior is the same as with UII Enabled.
UII Enabled with Error Messages	Allows the imager to read UII messages in Data Matrix ECC 200 symbols and output detailed information such as construction type, data components, or error messages. The imager's behavior is the same as with UII Enabled.
Data Validation Disabled	Disables both UII and ISO/IEC 15434 data validation.

Unique Identification Output Examples

UII Enabled

UII:UN123456789ABCDEFG
CPN:87654321
LBN:87654321
UII:12345678 CPN:87654321
UII:12345678 LBN:87654321

UII Enabled with Pass Through

UII:UN123456789ABCDEFG
CPN:87654321
LBN:87654321
UII:12345678 CPN:87654321
UII:12345678 LBN:87654321
DATA:Microscan Precision Data Acquisition and Control Solutions

UII Enabled with Error Messages

UII:UN123456789ABCDEFG;Construct_1;25SUN123456789ABCDEFG;;;;;;
CPN:87654321;PNR;PNR 87654321;;;;;;
LBN:87654321;30T;30T87654321;;;;;;
UII:12345678 CPN:87654321;Construct_1_2/PNR;UID 12345678;PNR 87654321;;;;;
UII:12345678 LBN:87654321;Construct_1/30T;25S12345678;30T87654321;;;;;
(15434 ERROR: HEADER - 1ST POSITION);Microscan Precision Data Acquisition and Control Solutions;;;;;;;

Data Validation Disabled

The imager will return to normal output behavior without performing data validation.

ISO/IEC 15434

ISO/IEC 15434 specifies a transfer structure, syntax, and coding of messages and data formats when using high capacity automatic data capture (ADC) technologies.

The following ISO/IEC 15434 data output options are applicable to Data Matrix ECC 200 symbols only and have no effect on other symbologies.

ISO/IEC 15434 Enabled	Allows the imager to read only ISO/IEC 15434-compliant message streams in Data Matrix ECC 200 symbols then output the ISO/IEC 15434 string. This implementation only checks the header/trailer format and proper format indicator (00-99 and DD). The output string has a prefix, a format indicator, and data components.
ISO/IEC 15434 Enabled with Error Messages	Allows the imager to read only ISO/IEC 15434-compliant messages in Data Matrix ECC 200 symbols and output detailed information such as prefix, format indicator, data components, or error messages.
Data Validation Disabled	Disables both UII and ISO/IEC 15434 data validation.

ISO/IEC 15434 Output Examples

ISO/IEC 15434 Enabled

(15434);05;0100061414199999;211A0B9C3D6;;;;;

(15434);06;7L0A1B3C;1P4202435;S10936;;;;

(15434);06;17V0A1B2;1P4202435;S10936;;;;

ISO/IEC 15434 Enabled with Error Messages

(15434);05;0100061414199999;211A0B9C3D6;;;;;

(15434);06;7L0A1B3C;1P4202435;S10936;;;;

(15434);06;17V0A1B2;1P4202435;S10936;;;;

(15434 ERROR: HEADER - 3RD POSITION);[)<▲DD↔CAG 12345↔SER 67890123▲♦;;;;;;;

(15434 ERROR: TRAILER - END OF TRANSMISSION);[)>▲12↔CAG 12345↔SER 67890123▲♣;;;;;;;

(15434 ERROR: HEADER - GROUP SEPARATOR);[)>▲12▲CAG 12345♦029SER 67890123▲♦;;;;;;;

Data Validation Disabled

The imager will return to normal output behavior without performing data validation.

Data Validation Settings

The following symbols control Data Validation functions:


Q048_01

UII Enabled


Q049_01

UII Enabled with Pass Through


Q050_01

UII Enabled with Error Messages


Q051_01

ISO/IEC 15434 Enabled


Q052_01

ISO/IEC 15434 Enabled with Error Messages


Q053_01

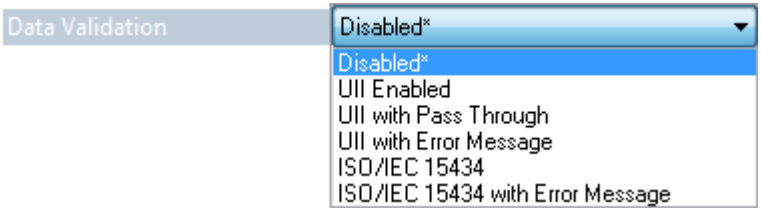
Data Validation Disabled (Default)


M10159_01

Save Settings

Data Validation by ESP

Each of the **Data Validation Settings** can also be enabled in **ESP**'s **I/O Parameters** tree control.



Detailed Output Format

The table below describes data validation output in detail.

Note: UII Enabled with Pass Through will add the prefix **DATA** to non-UII output for all symbologies.

Note: UII Enabled with Error Messages will output the following format: **UII/CPN/LBN; DF0; DF1; DF2; DF3; DF4; DF5; DF6; DF7**.

Note: When **ISO/IEC 15434** output is in compliance with the standard, the format is **(15434); DF0; DF1; DF2; DF3; DF4; DF5; DF6; DF7**. When it is not in compliance with the standard, the output is **(15434 ERROR: xxxx);;;;;;;;** where DF0 is the format indicator showing which type of data qualifier is in use.

UII Enabled		
Content of Decoded Data Matrix Symbol	**UII/CPN/LBN**	
Valid UII	*UII:*UII_data *Example:* UII:12345678	
Valid CPN Valid LBN	*CPN:*CPN_data *Example:* CPN:87654321 *LBN:*LBN_data *Example:* LBN:87654321	
Valid UII and CPN Valid UII and LBN	*UII:*UII_data CPN:CPN_data *Example:* UII:12345678 CPN:87654321 *UII:*UII_data LBN:LBN_data *Example:* UII:12345678 LBN:87654321	
Valid UII and Invalid CPN Valid UII and Invalid LBN	*UII:*UII_data (CPN ERROR) *Example:* UII:12345678 (CPN ERROR) *UII:*UII_data (LBN ERROR) *Example:* UII:12345678 (LBN ERROR)	
Invalid UII and Valid CPN Invalid UII and Valid LBN	*(UII ERROR) CPN:*CPN_data *Example:* (UII ERROR) CPN:87654321 *(UII ERROR) LBN:*LBN_data *Example:* (UII ERROR) LBN:87654321	
None of the above (Invalid UII; Invalid CPN; Invalid LBN; Invalid UII and Invalid CPN; Invalid UII and Invalid LBN)	No output data	
UII Enabled with Error Messages		
Content of Decoded Data Matrix Symbol	**UII/CPN/LBN**	**DFO**
Valid UII	*UII:*UII_data *Example:* UII:12345678	Constructed UII type *Example:* Contruct_1

Valid CPN Valid LBN	*CPN:*CPN_data *Example:* CPN:87654321 *LBN:*LBN_data *Example:* LBN:87654321	Constructed CPN type *Example:* PNR Constructed LBN type *Example:* 30T
Valid UII and CPN Valid UII and LBN	*UII:*UII_data CPN:CPN_data *Example:* UII:12345678 *CPN:*87654321 *UII:*UII_data LBN:LBN_data *Example:* UII:12345678 *LBN:*87654321	Constructed UII/CPN type *Example:* Construct_1/PNR Constructed UII/LBN type *Example:* Construct_1/30T
Valid UII and Invalid CPN Valid UII and Invalid LBN	*UII:*UII_data (30P ERROR: xxxx) *UII:*UII_data (PNR ERROR: xxxx) *UII:*UII_data (240 ERROR: xxxx) *UII:*UII_data (30T ERROR: xxxx)	Constructed UII type *Example:* Construct_1
Invalid UII and Valid CPN Invalid UII and Valid LBN	(UII ERROR: xxxx) CPN:CPN_data (UII ERROR: xxxx) LBN:LBN_data	Constructed CPN type: 30P, PNR, 240 Constructed LBN type: 30T
Invalid UII	(UII ERROR: xxxx) (15434 ERROR: xxxx) *Example:* (UII ERROR: DATA ELEMENT CHARACTER)	Original decoded data
Invalid CPN Invalid LBN	(30P ERROR:xxxx) (PNR ERROR:xxxx) (240 ERROR:xxxx) (30T ERROR:xxxx) (15434 ERROR: xxxx)	Original decoded data
Invalid UII and Invalid CPN Invalid UII and Invalid LBN	(UII ERROR: xxxx) (30P ERROR: xxxx) (UII ERROR: xxxx) (PNR ERROR: xxxx) (UII ERROR: xxxx) (240 ERROR: xxxx) (UII ERROR: xxxx) (30T ERROR: xxxx)	Original decoded data

Error Messages

The following is a list of potential error messages.

15434 ERROR: DATA ELEMENT SEPARATOR
15434 ERROR: DOUBLE TRAILER
15434 ERROR: FORMAT INDICATOR
15434 ERROR: HEADER - 1ST POSITION
15434 ERROR: HEADER - 2ND POSITION
15434 ERROR: HEADER - 3RD POSITION
15434 ERROR: HEADER - 4TH POSITION
15434 ERROR: HEADER - GROUP SEPARATOR
15434 ERROR: TRAILER - END OF TRANSMISSION
15434 ERROR: TRAILER - RECORD SEPARATOR
PNR ERROR: TOO LONG
PNR ERROR: TOO SHORT
PNR ERROR: CHARACTER
30P ERROR: TOO LONG
30P ERROR: TOO SHORT
30P ERROR: CHARACTER
240 ERROR: TOO LONG
240 ERROR: TOO SHORT
240 ERROR: CHARACTER
UII ERROR: DATA ELEMENT CHARACTER
UII ERROR: DATA ELEMENT TOO LONG
UII ERROR: DATA ELEMENT TOO SHORT
UII ERROR: LOWER CASE CHARACTER
UII ERROR: NEED UII ELEMENT FIRST
UII ERROR: SPACE AFTER TEI DATA QUALIFIER
UII ERROR: TEI DATA QUALIFIER
UII ERROR: UII ELEMENT INCOMPLETE
UII ERROR: WRONG FORMAT INDICATOR
UII ERROR: UII STRING TOO LONG

Additional Notes

- DF1 – DF7: If the UII/CPN field is “(15434 ERROR: xxxx)”, DF1 – DF7 are filled in with an empty string. Otherwise, the fields are used to display data elements. If there are fewer than seven data elements, an empty string is filled in at the end. If there are more than seven elements, only the first seven elements are displayed.
- There is a space between UII and CPN in both tables (UII:12345678 CPN:87654321).
- The constructed UII type can be Contruct_1, Contruct_2, Construct_1_2, or IUID_EQUIVALENT.
- The constructed CPN type can be PNR, 30P, or 240. The constructed LBN type can be 30T.

8 Advanced Operations

Contents

This section introduces several settings that can be applied to speed up processing or improve readablility in various circumstances.

Continuous Read

Read the following symbols to enable or disable Continuous Read.

M0127_02

Continuous Read On

M0126_03

Continuous Read Off

Mirroring

Mirroring allows the reader to decode symbols that are reversed. When Mirroring is enabled, all other decode functionality is disabled.

Note: Once the reader has been set to **Mirroring On**, it can only return to its default mode by reading the **Mirroring Off** symbol below.

Mirroring On

M0152_01

Mirroring Off (Default)

M0153_01

Motion Detection

Motion Detection causes the reader to attempt a decode whenever it senses motion in its field of view.

Motion Detection On, Start Delay 0 ms

M0161_01

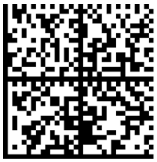
Motion Detection On, Start Delay 500 ms

M0162_01

Motion Detection Off (Default)

M0129_07

Motion Detection On, Start Delay 0 ms, Dark Environment



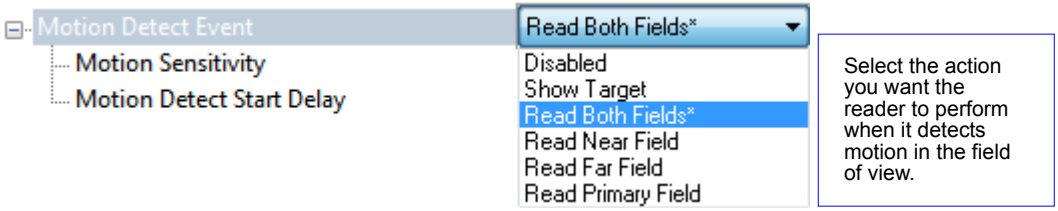
M0163_01

Motion Detection On, Start Delay 500 ms, Dark Environment

M0164_01

Motion Detection by ESP

Motion Detection settings can be refined further using the options in **ESP**.



Motion Sensitivity

The reader's sensitivity to motion in the field of view can be configured using the **Motion Sensitivity** parameter. (The lower the number, the greater the sensitivity.)

Important: For **Motion Sensitivity** to function correctly, **Button Stay-Down Time** should be increased.

Motion Detect Start Delay

Motion Detect Start Delay allows you to set the amount of delay before a **Motion Detect Event** occurs. (0 seconds is default.)

Window of Interest

Window of Interest allows you to shrink the processing area of the reader's field of view. Because the reader has far less processing to do in a smaller window, read rates typically increase dramatically.

- Click the **Camera** icon in App Mode to bring up **Window of Interest**.

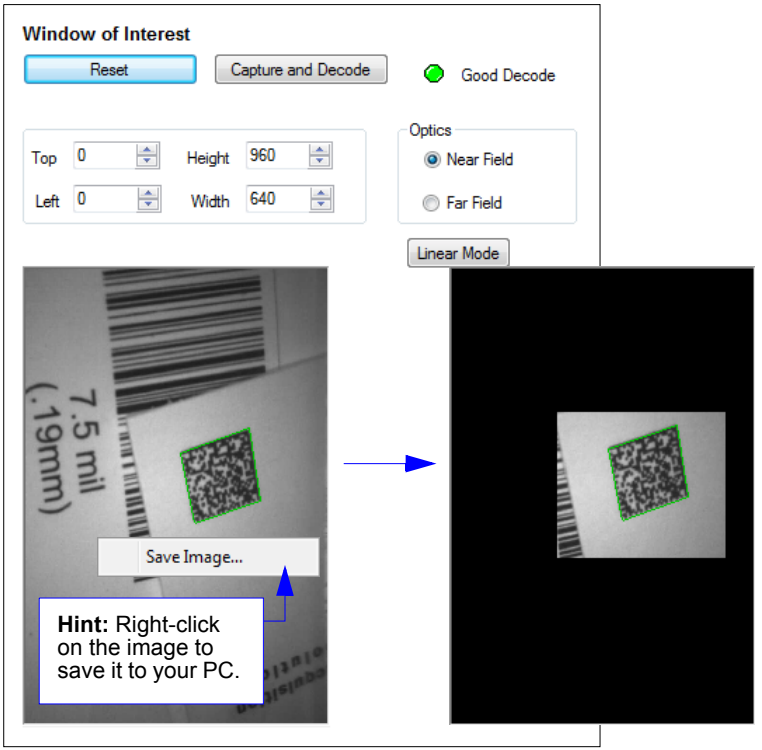
Click the **Capture and Decode** button in the Window of Interest view to decode the symbol in the field of view. If successful, the **Good Decode** indicator will be green and the symbol will be outlined in green.

Note: You can resize the image by clicking and dragging the **ESP** window from the lower right corner. This is useful where very small symbols are being read.

- Click and drag the cursor to define a rectangle over the symbol that you want to isolate. Notice that the surrounding pixels become black.

  You can use the anchor points on the image area that you have just drawn to resize the Window of Interest. You can also click on the center of the window to move it.

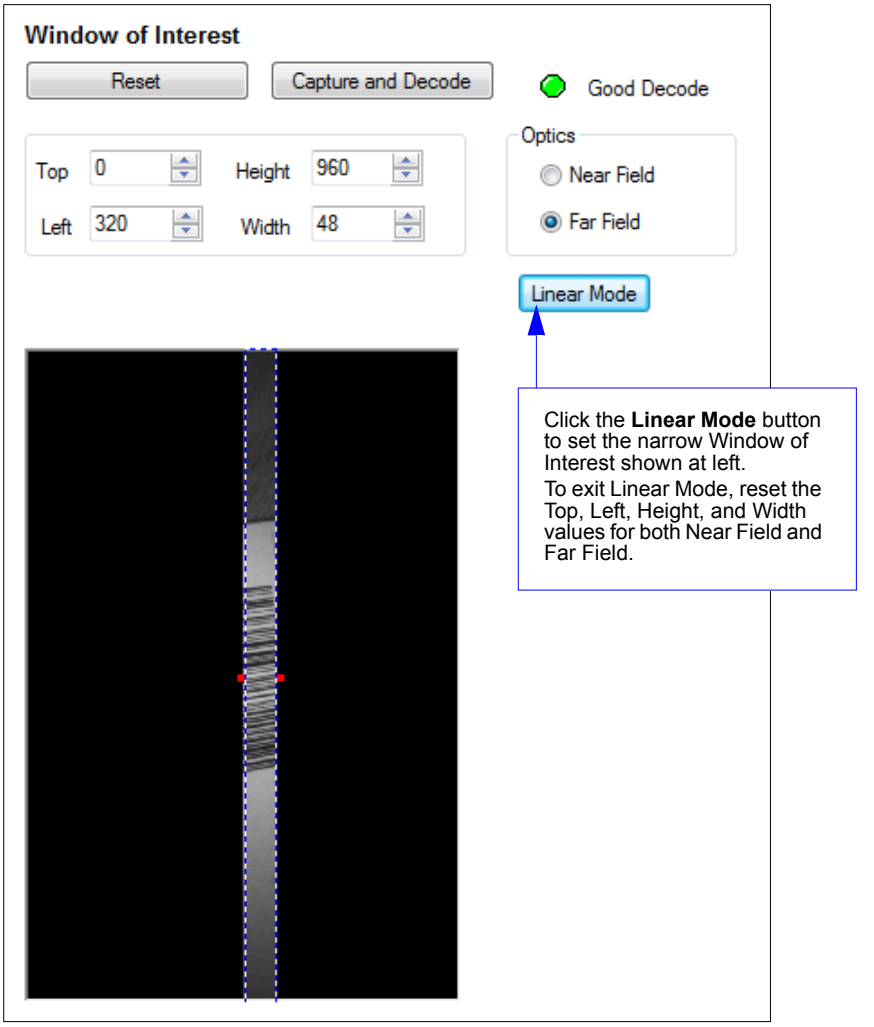
- Click the **Reset** button to remove the Window of Interest.



Linear Mode

Linear Mode is intended for use with 1D (linear) symbologies. The combination of Far Field optics and narrow field of view is ideal for decoding a series of closely-spaced 1D symbols, such as a warehouse “pick list”.

Clicking the Linear Mode button automatically defines the narrow Window of Interest in Far Field as shown below. The Near Field Window of Interest is also reduced to maintain fast decode times of 1D symbols while still allowing Data Matrix decoding (including Data Matrix programming symbols).



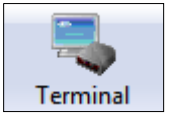
9 Terminal

Contents

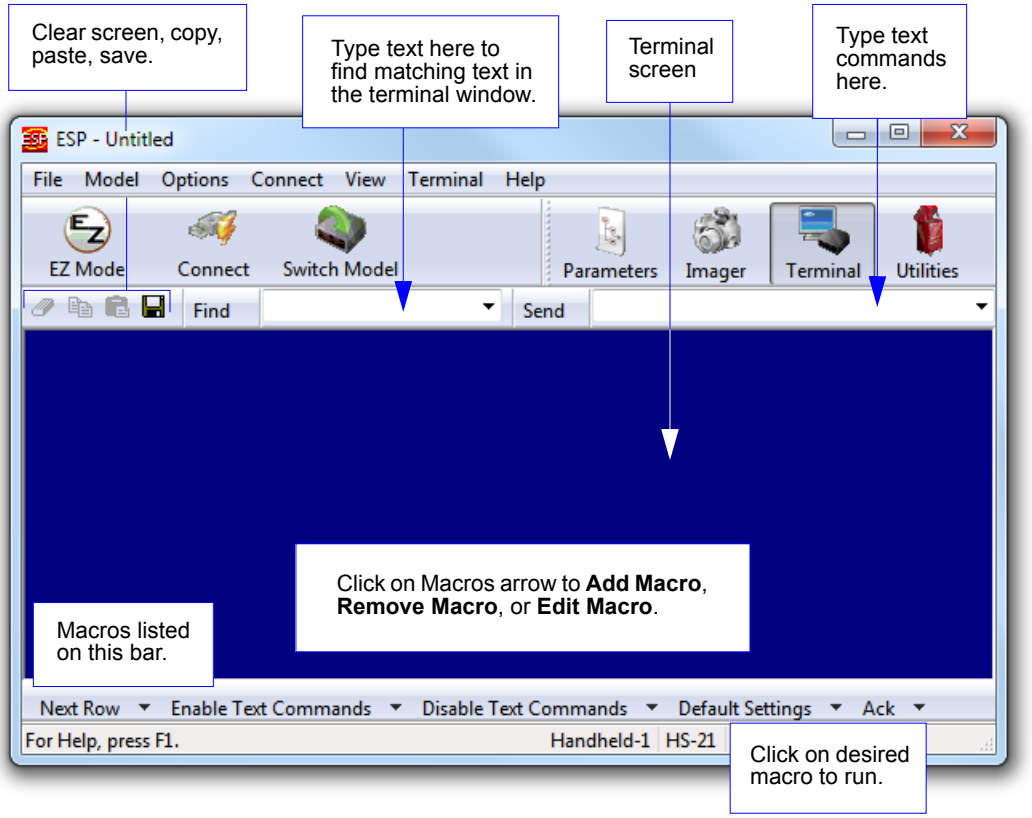
This section describes the **Terminal** interface and macro functions in **ESP**.

Terminal View

Click the **Terminal** button.



You will see the following view:



The Terminal interface allows you to send commands to the reader by using macros, by copying and pasting, or by typing commands in the **Send** text field.

The Terminal view also displays symbol data or information from the reader.

You can also right click on the Terminal screen to bring up a menu of further options.

Find

The **Find** function allows you to enter text strings to be searched for in the terminal window. For example, suppose a series of symbols have been scanned into the terminal view and you want to determine if a particular symbol whose data begins with “ABC” has been read.

1. Type “ABC” into the **Find** box.

2. Press **Enter**.

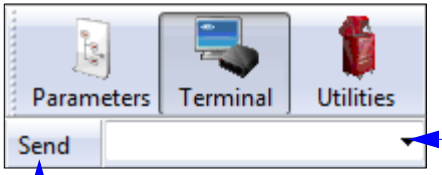
   The first instance of “ABC” will be highlighted in the terminal window.

3. Click the **Find** button to the left of the text field to locate additional instances of “ABC”.

Send

The **Send** function allows you to enter text commands and then send them to the reader. (See **Text Commands**.)

For example, suppose you want to disable the vibrate function in the reader. To disable vibrate using a text command, you would enter “P%A10” (the command that disables vibrate) in the text field and click **Send**.



Click here to access the dropdown menu of stored text commands.

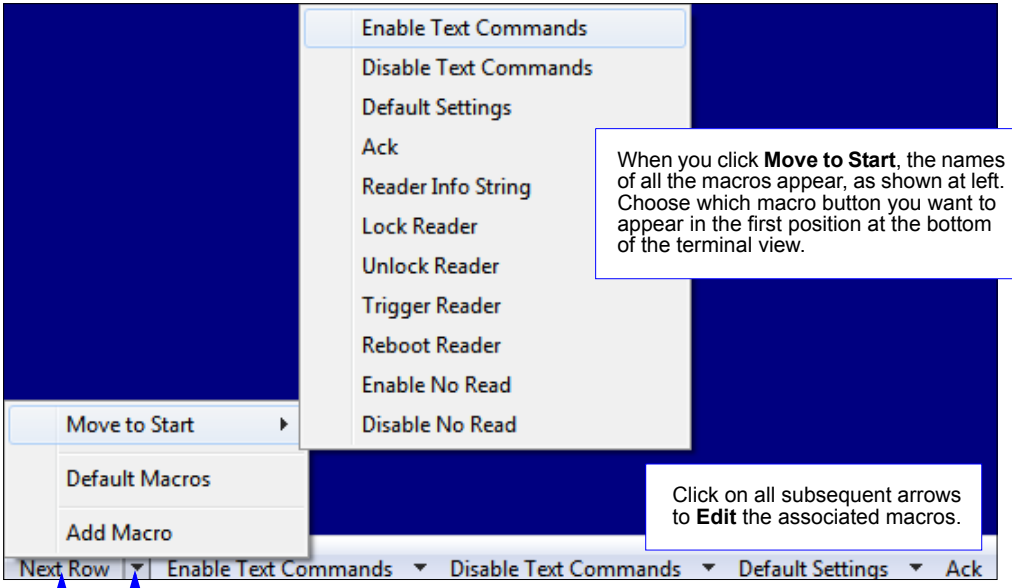
Type text commands in the text field and then click the **Send** button to initiate. Clicking **Send** repeatedly will initiate the command repeatedly.

Once text commands are initiated, they are saved in a dropdown menu that can be accessed by clicking the arrow to the right of the text field.

You can also send the current command repeatedly by clicking the **Send** button repeatedly.

Macros

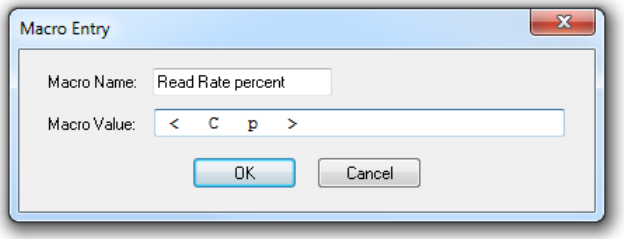
Macros can be stored in a macro selection bar, edited in a separate window, and executed by clicking on the macro name.



Clicking on a macro button executes the related command. The command is also sent to the reader at the same time it is displayed.

Editing a Macro

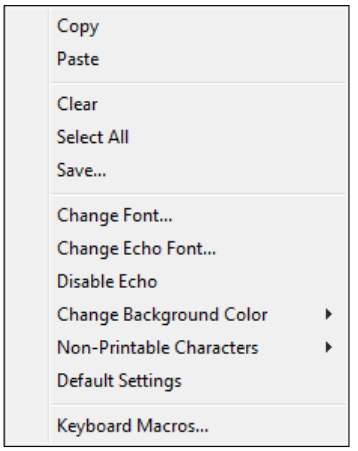
When you click the arrow next to a any macro and select **Edit**, the following dialog appears:



You can edit an existing macro or type in the **Macro Name** text field and define it in the **Macro Value** text field.

Terminal Right-Click Menu

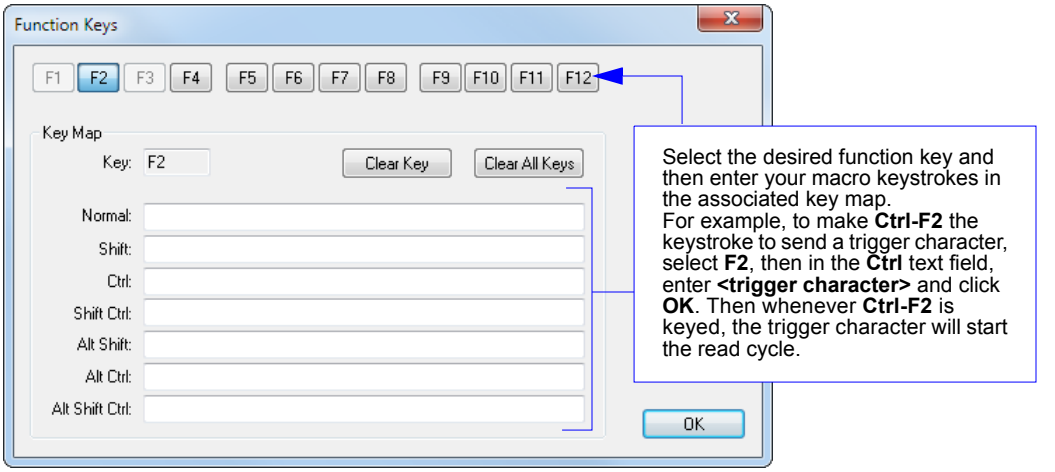
Right click in the terminal window to display the following menu:



- **Copy** selected text to clipboard.
- **Paste** from terminal or other text.
- **Clear** all text in terminal window.
- **Select All** text in the terminal window.
- **Save...** incoming and outgoing data into a text file.
- **Change Font...** of data received from the reader.
- **Change Echo Font...** to change the appearance of user-entered data.
- **Disable Echo** to hide user-entered data.
- **Change Background Color** of the terminal window.
- **Non-Printable Characters** can be shown or hidden in the terminal view in **Standard** or **Enhanced** format.
- **Default Settings** to return all of the above to original settings.
- **Keyboard Macros** brings up the **Function Keys** dialog, which allows you to create customized macro functions.

Function Keys

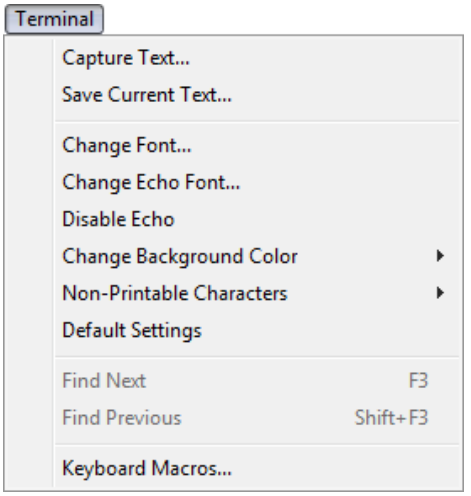
The **Function Keys** dialog allows you to assign commands to specific function keys on a standard keyboard. Note that the **F1** key is reserved for opening **ESP** Help, and the **F3** key is reserved for the **Find Next** function.



Note: This feature is also available from the **Terminal Dropdown Menu** and the **Terminal** tab of the **Preferences** dialog.

Terminal Dropdown Menu

The terminal dropdown menu allows you to capture and save current text, and it also includes the functions defined for the **Terminal Right-Click Menu**.



- **Capture Text...** lets you append data in real time to a text file of your choice. While in operation, the text file cannot be opened. You can select **Pause** to interrupt the capture flow or **Stop** to end the flow and open the file.
- **Save Current Text...** saves all text in the terminal window to a text file of your choice.
- **Find Next** locates the next instance of the specified data string in the terminal. This function can also be activated by pressing **F3**.
- **Find Previous** locates the most recently occurring instance of the specified data string in the terminal.
- **Keyboard Macros** brings up the **Function Keys** dialog, which allows you to create customized macro functions.

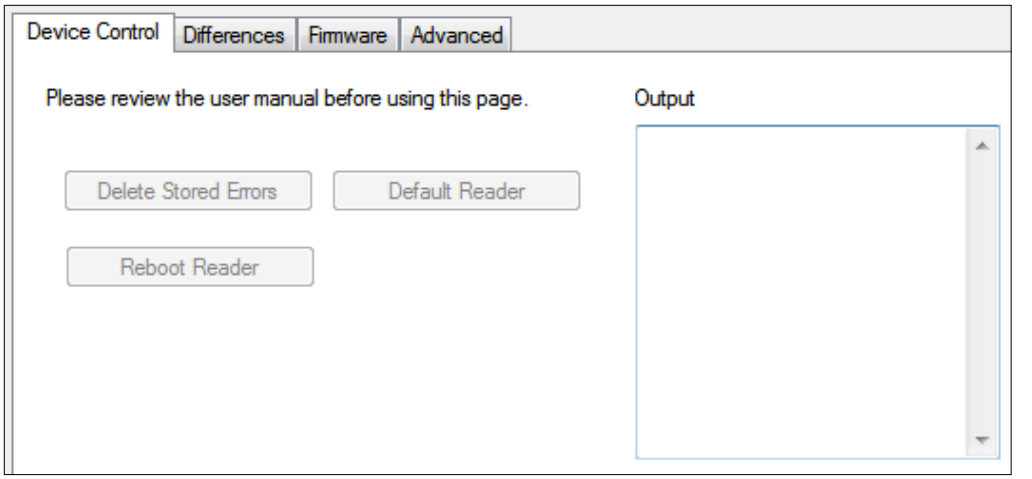
10 Utilities

Contents

This section explains **ESP**'s **Utilities** features. These include **Device Control**, an interface that lets you perform major operations with one click; **Differences from Default**, which shows all currently enabled reader settings that are not default settings; **Firmware**, where you can update your reader's firmware; and **Advanced**, which allows you to collect batch files for customized reader configuration and optimization.

Device Control

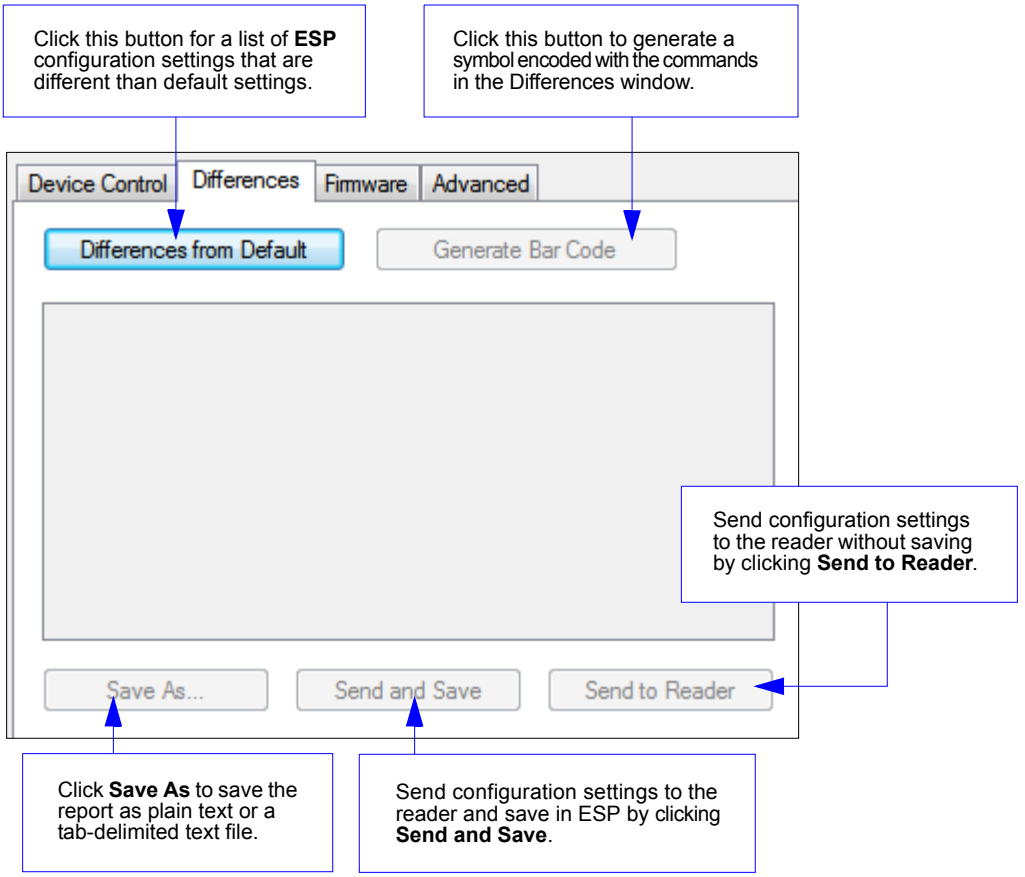
This feature allows you to delete stored errors, to reboot the reader, and to default the reader.



- **Delete Stored Errors** erases all logged errors whether you have looked at them or not.
- **Default Reader** returns the reader to its default state, without any optimization or configuration.
- **Reboot Reader** refreshes the reader's memory and functionality, returning it to the most recent configuration you have saved.

Differences from Default

Clicking the **Differences from Default** button will cause **ESP** to check all stored configuration settings and compare them to default settings. All settings that are different than default will appear in the left column (shown below), and descriptions of those settings will appear in the right column.

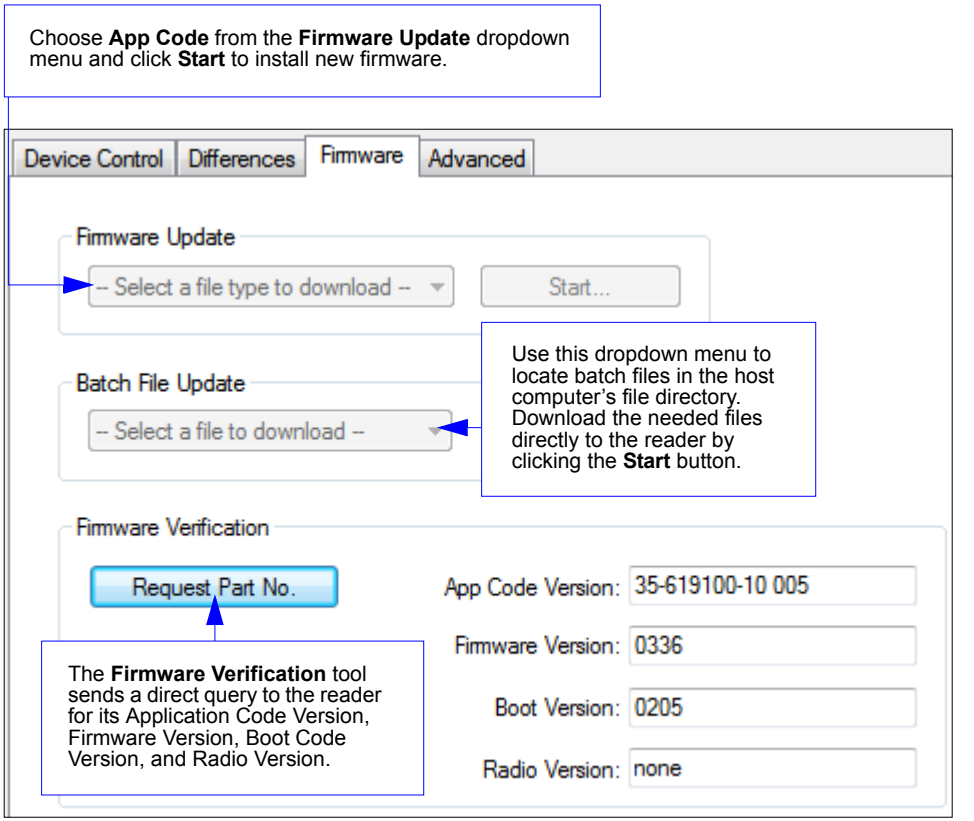


- To save the **Differences from Default** report, either as plain text or as a tab-delimited text file, click **Save As**.
- Click **Send and Save** to send the settings to the reader and save them, or **Send to Reader** to send the settings without saving them.

Important: The use the **Differences from Default** feature, you must connect to the reader and **Receive Reader Settings** via the **Send/Recv** button on the toolbar.

Firmware

The Firmware view in ESP Utilities is a simple way to update and verify your reader's firmware and to update batch files.



ID and Firmware Version

Another way to query the 2D Engine for its identifying information is by reading the following symbol:


M0148_01

ID and Firmware Version

The host's text program will output a data string containing the device's identifying information in the format shown below.

Example:

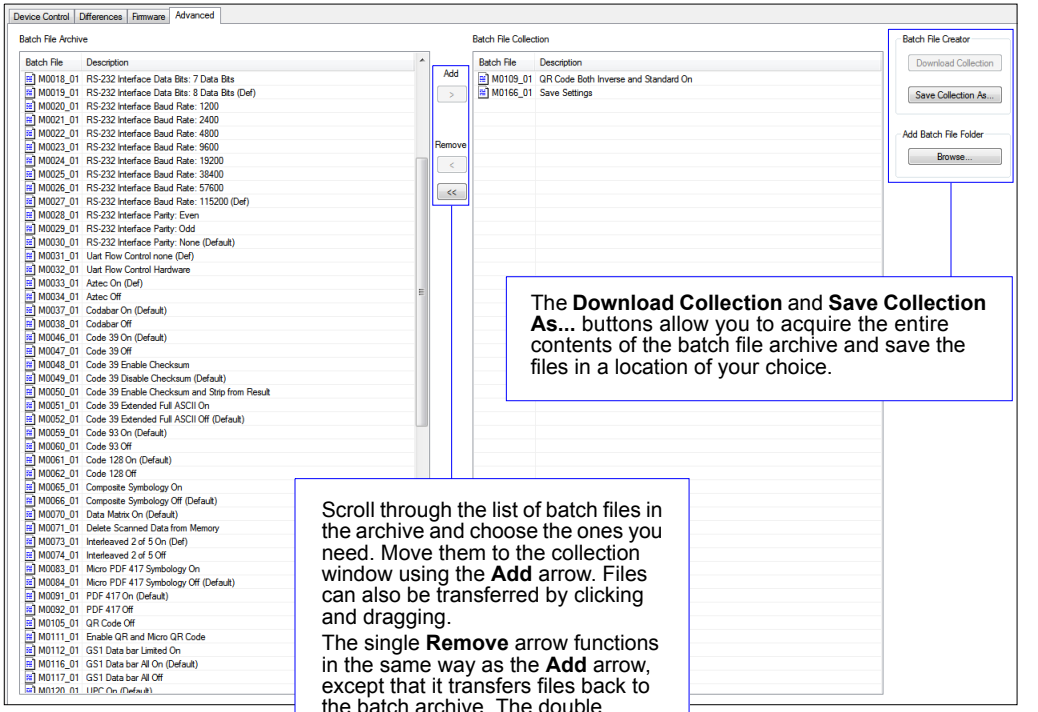
i03360205none0020019795A06000000600080014004900002<TAB>35-619100-10 005

0336	Application Version Number
0205	Bootloader Firmware Version
None	Radio Firmware Version
0020019795	Serial Number
A	A – Running Application
06	N/A
0	N/A
0000	N/A
06	Hardware Identifier
0008	Hardware Type Identifier
0014	Boot Application Version
0049	Operating System Kernel Version
0002	Root File System Versions
<TAB>	ASCII TAB Character
35-619100-10 005	Decoder Version PN and BN

Advanced

The **Advanced** tab in **Utilities** features an archive of all batch files containing reader configuration commands. Each batch file's extension is .crb, and each file contains the fundamental code for programming the reader. Notice that the names of the batch files correspond with the numbers beneath all the Data Matrix configuration symbols.

This tool allows you to use the batch file data to create your own symbols, or to collect only the files that you use frequently to configure the reader for your application.



Appendices

Contents

Appendix A — General Specifications

Mechanical

Height:	5.2” (131.6 mm)
Width:	2.0” (52 mm)
Depth:	3.6” (91.1 mm)
Weight:	3.9 oz. (110 g)

Environmental

Operating temperature: –20° to 55° C (–4° to 131°F)
Storage temperature: –30° to 65° C (–22° to 150°F)
Humidity: 5 to 95% (non-condensing)
Shock: Withstands multiple drops of 6’ (1.8 meters)

CE Standards

Immunity: EN 55024
ESD: EN 61000-4-2
Radiated RF: EN 61000-4-3
Keyed Carrier: ENV50204
EFT: EN 61000-4-4
Conducted RF: EN 61000-4-6
Emissions: EN 55022, Class B Radiated, Class B Conducted
CB Test Certificate: IEC 60950-1:2005, 2nd Edition

Symbologies

2D Symbologies: Data Matrix, QR Code, Micro QR Code, Aztec Code
Stacked Symbologies: PDF417, MicroPDF417, Composite
Linear Symbologies: UPC, Code 39, Code 128, Interleaved 2 of 5, Codabar, GS1 DataBar, Code 93
Postal Symbologies: USPS OneCode (4CB), POSTNET, PLANET, Japanese Post, Australian Post, Royal Mail, KIX Code

Light Collection Options

Sensor: CMOS 1.2 Megapixel grayscale
Sensor Array: 1280 by 960
Field Selection: Near Field or Far Field
Field of View: Near Field: 30° horizontal by 20° vertical; Far Field: 50° horizontal by 33.5° vertical
Focal Point: Approximately 100 mm
Optical Resolution: Near Field: 960 x 640; Far Field: 960 x 640

Communication Protocols

Standard Interface: RS-232, USB 2.0 (Generic HID, HID Keyboard, Virtual COM Port)

Read Parameters

Pitch: ±60° (front to back)
Skew: ±60° (from plane parallel to symbol, side-to-side)
Rotational Tolerance: ±180°
Print contrast Resolution: 25% (1D symbologies); 35% (2D symbologies); absolute dark/light reflectance differential, measured at 650 nm
Ambient Light Immunity: Sunlight: Up to 9,000 ft.-candles / 96,890 lux
Target Beam: 2 blue bars

Indicators

Status Indicators: Beep, vibrate, LED flash

Image Output Options

Format: JPEG

Memory Capacity

128MB Flash ROM, 32MB RAM

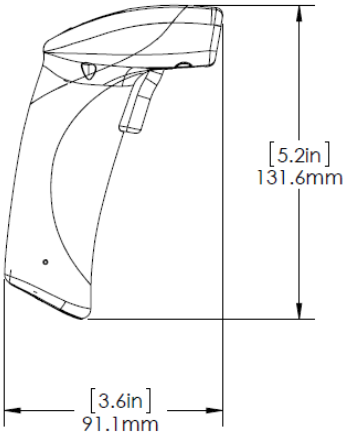
Data Editing

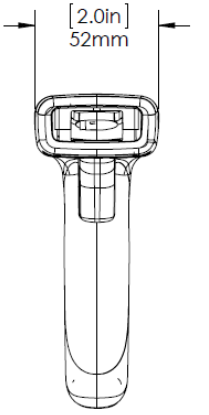
JavaScript (Additional License Required)

Electrical

Power Requirements: Reader @ 5VDC (mA): Typical: Less than 450 mA; Idle: Less than 80 mA; Sleep: Less than 31mA

Dimensions





Note: Nominal dimensions shown. Typical tolerances apply.

Read Ranges

Narrow Bar	Read Range
STANDARD DENSITY	
1D	
.0050” (.127 mm)	3.7 to 5.0” (94 to 127 mm)
.0075” (.191 mm)	2.2 to 6.5” (56 to 165 mm)
.010” (.254 mm)	1.5 to 8.0” (38 to 203 mm)
.020 (.508 mm)	2.3 to 15.5” (58 to 394 mm)
2D	
.0050” (.127 mm)	3.7 to 4.6” (94 to 117 mm)
.0075” (.191 mm)	1.5 to 6.0” (38 to 152 mm)
.010” (.254 mm)	1.6 to 7.7” (41 to 196 mm)
.020 (.508 mm)	1.6 to 9.4” (41 to 239 mm)

FIS and Demo Kit Options; Accessories

HS-21/HS-41X Handheld Readers	
HS-41X, Handheld, X-Mode, Dark Gray, USB, 6' Straight Cable	FIS-HS41X-0001G
HS-41X, Handheld, X-Mode, Dark Gray, RS-232, 8' Coiled Cable, U.S. P/S	FIS-HS41X-0002G
HS-41X, Handheld, X-Mode, Dark Gray, RS-232, 8' Coiled Cable, Euro P/S	FIS-HS41X-0003G
HS-41X, Handheld, X-Mode, Dark Gray, RS-232, 8' Coiled Cable, UK P/S	FIS-HS41X-0004G
HS-21, Handheld, Std, Light Gray, USB, 6' Straight Cable	FIS-HS21-0001G
HS-21, Handheld, Std, Light Gray, RS-232, 8' Coiled Cable, U.S. P/S	FIS-HS21-0002G
HS-21, Handheld, Std, Light Gray, RS-232, 8' Coiled Cable, Euro P/S	FIS-HS21-0003G
HS-21, Handheld, Std, Light Gray, RS-232, 8' Coiled Cable, UK P/S	FIS-HS21-0004G
Kit, Demo, X-Mode, Dark Gray, USB, 6' St Cable	98-000263-01
Kit, Demo, Std., Light Gray, USB, 8' Coiled Cable	98-000264-01
Presentation Stand for fixed-mount Motion Detect operation	98-000270-03
Microscan Tools Drive: Software, User Manuals, Quick Start Guides, Configuration Guides, links to Microscan website	37-000001

Safety Certifications

FCC, CE, RoHS/WEEE

Product specifications are given for typical performance at 25°C (77°F) using grade A symbols. Performance characteristics may vary at high temperatures or other environmental extremes. Five Year Limited Warranty on parts and labor.

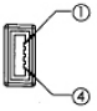
Appendix B — Electrical Specifications

Power Requirements

Reader @ 5VDC (mA): *Typical:* Less than 450 mA; *Idle:* Less than 80 mA; *Sleep:* Less than 31mA

USB Cable Pinouts

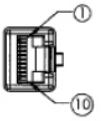
Connector A



WIRING TABLE:

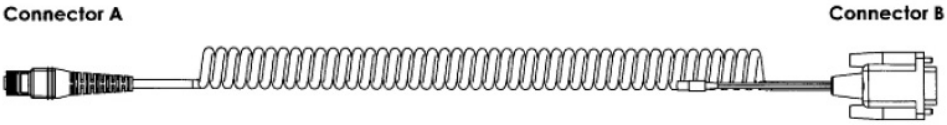
CONNECTOR A	NAME	WIRE	COLOR	CONNECTOR B
1	VIN	24AWG	RED	1
2	DM	28AWG	WHITE	2
3	DP	28AWG	GREEN	3
4	GND	24AWG	BLACK	10
SHELL	-	SHIELD	BARE	SHELL

Connector B

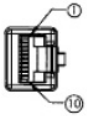


Maximum Voltage Tolerance = 5V +/- 10%.
Caution: Exceeding the maximum voltage will void manufacturer warranty.

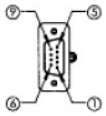
RS-232 Cable Pinouts



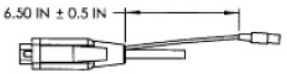
Connector A



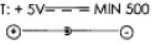
Connector B



Connector C



- MATING PLUG OD: 5.5 mm
- CENTER PIN OD 2.0 mm, LENGTH 8.5 mm.
- INPUT: + 5V⎓ MIN 500 mA

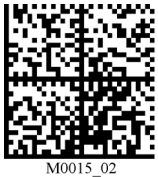
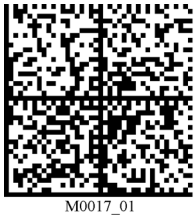


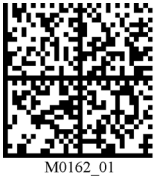
WIRING TABLE:

CONN A	NAME	WIRE	COLOR	CONN B	WIRE	COLOR	CONN C
1	VIN	24AWG	RED	9	24AWG	RED	TIP
4	TX	28AWG	BROWN	2			
5	RTS	28AWG	ORANGE	8			
6	RX	28AWG	YELLOW	3			
7	CTS	28AWG	GREEN	7			
10	GND	24AWG	BLACK	5	24AWG	BLACK	RING
SHIELD	-	SHIELD		SHIELD			

Maximum Voltage Tolerance = 5V +/- 10%.
Caution: Exceeding the maximum voltage will void manufacturer warranty.

Appendix C — Configuration Symbols

A1  Q0001_2 Default to USB (HID)	**A2**  M0002_01 USB Keyboard (Windows)	**A3**  Q0003_1 USB Native (HID) Mode	**A4**  M0005_01 USB Virtual COM Mode
B1  M0015_02 USB Enable Alternate OS	**B2**  M0016_02 USB Disable Alternate OS	**B3**  M0017_01 Reset to RS-232 Defaults	**B4**  M0018_01 RS-232 7 Data Bits
C1  M0019_01 RS-232 8 Data Bits (Default)	**C2**  M0020_01 RS-232 1200 Baud	**C3**  M0021_01 RS-232 2400 Baud	**C4**  M0022_01 RS-232 4800 Baud
D1  M0023_01 RS-232 9600 Baud	**D2**  M0024_01 RS-232 19200 Baud	**D3**  M0025_01 RS-232 38400 Baud	**D4**  M0026_01 RS-232 57600 Baud
E1  M0027_01 RS-232 115200 Baud (Default)	**E2**  M0028_01 RS-232 Even Parity	**E3**  M0029_01 RS-232 Odd Parity	**E4**  M0030_01 RS-232 No Parity (Default)

A1  M0031_01 UART Flow Control None (Default)	**A2**  M0032_01 UART Flow Control Hardware	**A3**  M0149_01 RS-232 Raw Mode (Default)	**A4**  M0150_01 RS-232 Packet Mode
B1  M0196_01 Targeting Off	**B2**  M0197_01 Targeting On	**B3**  M0167_01 Beep Off	**B4**  M0168_01 Beep On
C1  M0126_03 Continuous Trigger Off	**C2**  M0129_07 Motion Detect Off	**C3**  M0161_01 Motion Detect On, Start Delay 0	**C4**  M0162_01 Motion Detect On, Start Delay 500 ms
D1  M0146_01 Reader Text Commands On	**D2**  M0147_01 Reader Text Commands Off	**D3**  M0130_02 Preamble - Comma	**D4**  M0131_02 Preamble - Space
E1  M0132_02 Preamble - Tab (Keyboard Only)	**E2**  M0133_02 Preamble - Tab (Serial Only)	**E3**  M0134_02 Preamble - Erase (None)	**E4**  M0135_02 Preamble - CR LF (Serial Only)

A1  M0136_02 Postamble - CR (Serial Only)	**A2**  M0137_02 Postamble - Comma	**A3**  M0138_02 Postamble - LF (Serial Only)	**A4**  M0139_02 Postamble - CR LF (Serial Only)
B1  M0140_02 Postamble - Space	**B2**  M0141_02 Postamble - Enter (Keyboard Only)	**B3**  M0142_02 Postamble - Tab (Keyboard Only)	**B4**  M0143_02 Postamble - Tab (Serial Only)
C1  M0144_02 Postamble - Erase (None)	**C2**  M0145_02 Preamble and Postamble - Erase	**C3**  M0007_02 U.S. Keyboard Mapping (Default)	**C4**  M0008_02 U.S. Keyboard without Leading 0
D1  M0009_02 U.S. Keyboard with Ctr+Char	**D2**  M0010_01 French Keyboard Mapping	**D3**  M0011_01 German Keyboard Mapping	**D4**  M0012_01 Japanese Keyboard Mapping
E1  M0013_01 Universal Keyboard Mapping	**E2**  M0014_01 Custom Keyboard	**E3**  M0189_01 Belgian Keyboard Mapping	**E4**  M0190_01 Swiss Keyboard Mapping

A1  Data Matrix On	**A2**  Data Matrix Off	**Important:** If you disable the Data Matrix symbology, programming symbols will not be decodable by the reader, and you will need to re-enable Data Matrix using ESP. Use the **Data Matrix Off** programming symbol with caution.	
B1  M0033_01 Aztec On	**B2**  M0034_01 Aztec Off (Default)	**B3**  Q0009_01 BC412 On (Default)	**B4**  Q0010_01 BC412 Off
C1  M0037_01 Codabar On (Default)	**C2**  M0038_01 Codabar Off	**C3**  Q0011_01 Codabar Checksum Enabled	**C4**  Q0012_01 Codabar Checksum Disabled (Default)
D1  Q0030_01 Codabar Checksum Enabled, Strip from Result	**D2**  M0046_01 Code 39 On (Default)	**D3**  M0047_01 Code 39 Off	**D4**  M0048_01 Code 39 Enable Checksum
E1  M0049_01 Code 39 Disable Checksum (Default)	**E2**  M0050_01 Code 39 Enable Checksum and Strip from Result	**E3**  M0051_01 Code 39 Extended Full ASCII On	**E4**  M0052_01 Code 39 Extended Full ASCII Off (Default)

A1  M0059_01 Code 93 On (Default)	**A2**  M0060_01 Code 93 Off	**A3**  M0061_01 Code 128 On (Default)	**A4**  M0062_01 Code 128 Off
B1  M0065_01 Composite On	**B2**  M0066_01 Composite Off (Default)	**B3**  M0073_01 Interleaved 2 of 5 On (Default)	**B4**  M0074_01 Interleaved 2 of 5 Off
C1  Interleaved 2 of 5 Two Digit Minimum	**C2**  Interleaved 2 of 5 Four Digit Minimum	**C3**  Interleaved 2 of 5 Six Digit Minimum	**C4**  M0151_01 Interleaved 2 of 5 with Control Character Stripped
D1  Q0013_01 Interleaved 2 of 5 Checksum Enabled	**D2**  Q0014_01 Interleaved 2 of 5 Checksum Disabled (Default)	**D3**  Q0015_01 Interleaved 2 of 5 Checksum Enabled, Strip from Result	**D4**  M0083_01 MicroPDF417 On
E1  M0084_01 MicroPDF417 Off (Default)	**E2**  M0091_01 PDF417 On (Default)	**E3**  M0092_01 PDF417 Off	**E4**  Q0031_01 Postal On

A1  M0102_01 Postal Off (Default)	**A2**  M0105_01 QR Code Off	**A3**  M0109_01 QR Code Inverse and Standard On	**A4**  M0111_01 QR Code and Micro QR Code On
B1  M0112_01 GS1 DataBar Limited On	**B2**  M0116_01 All GS1 DataBar On (Default)	**B3**  M0117_01 All GS1 DataBar Off	**B4**  M0120_01 UPC On (Default)
C1  M0121_01 UPC Off	**C2**  Q0016_01 EAN Status Enabled (Default)	**C3**  Q0017_01 EAN Status Disabled	**C4**  Q0018_01 UPC-E as UPC-A Enabled
D1  Q0019_01 UPC-E as UPC-A Disabled (Default)	**D2**  Q0020_01 Pharmacode On	**D3**  Q0021_01 Pharmacode Off (Default)	**D4**  Q0022_01 Pharmacode Fixed Symbol Length Enabled
E1  Q0023_01 Pharmacode Fixed Symbol Length Disabled (Default)	**E2**  Q0024_01 Pharmacode Bar Width Status Mixed (Default)	**E3**  Q0025_01 Pharmacode Bar Width Status All Narrow	**E4**  Q0026_01 Pharmacode Bar Width Status All Wide

A1  Q0027_01 Pharmacode Bar Width Status Fixed Threshold	**A2**  Q0028_01 Pharmacode Direction Forward (Default)	**A3**  Q0029_01 Pharmacode Direction Reverse	**A4**  M0071_01 Delete Scanned Data from Memory
B1  M0148_01 ID and Firmware Version	**B2**  M0165_01 Clear All JavaScript Rules	**B3**  M0166_01 Save Settings	**B4**  M0191_01 Disable Duplicate Symbol Timeout
C1  M0192_01 1 Second Duplicate Scan Delay	**C2**  M0193_01 2 Second Duplicate Scan Delay	**C3**  M0194_01 3 Second Duplicate Scan Delay	**C4**  M0195_01 Default Trigger Delay

Configuration Symbol Reference

RS-232 Settings

Symbologies

USB Settings

Other Commands

Appendix D — Serial Commands

Text Commands

Text commands may be sent to the reader in RS-232 or USB Virtual COM mode using any serial communications software, e.g., ESP Terminal.

Encoded-data is decoded by the reader by replacing %xx by a single byte with the value specified by the two hex-digits xx, e.g., **%25** would be replaced by character number 0x25, which is ASCII '%'.

text-command: *command-type encoded-dataopt carriage-return*

command-type: Single ASCII character in the set defined in **Command Types**.

encoded-data: *encoded-datum I encoded-data encoded-datum*

encoded-datum: *printable-character | % hex-digit hex-digit*

printable-character: any byte value in the range [0x20,0x7e]

hex-digit: **'0' | '1' | '2' | '3' | '4' | '5' | '6' | '7' | '8' | '9' |'A' | 'B' | 'C' | 'D' | 'E' | 'F' | 'a' | 'b' | 'c' | 'd' | 'e' | 'f'**

carriage-return: **0x0d**

In order to eliminate inadvertent commanding of the reader, Text Commands are disabled by default. To enable Text Commands requires an initial sequence: ;>PAx where x is as defined in the Reader Settings Table, register setting 41. (Note: 'A' is the ASCII character that corresponds to 41 HEX.)

For example, to send the reader commands by typing commands in ESP Terminal:

;>PA1

P(xx)yy

P(xx)yy

~

PA8

Where **;>PA1** enables text commands with echo and command responses; **P%xxyy** can be any desired commands; **~** saves the settings just sent (the ~ command saves all but communication-related settings); and **PA8** turns text commands back off (except for the initial sequence). (**Note:** *'A'* is the ASCII character that corresponds to 41 hex, thus P%418 would be equivalent.)

Note: ;>PA1 is used for interactive text commands. If the commands are to be saved in a file and sent non-interactively, use **;>PA7** instead; this enables text commands but disables echo and command responses. (See **Command Types**, **Reader Settings**, and **CRB System** for additional information.)

The following two examples can be sent to a reader in RS-232 mode from ESP Terminal by just typing the example text.

Example 1 (make the reader beep 3 times):

#%03 *Expected output: should make reader beep 3 times*

Example 2 (set reader to continuous read, High Density field (FOI0) only):

P(C4)5 *Expected output: should set reader to continuous read, High Density field (FOI0) only*

Example 3 (set reader to trigger read mode):

P(C4)255 *Expected output: should set reader to trigger read*

Packetized Commands

Packetized commands consist of packetized data sent from Host-to-Reader to configure and cause the reader to perform certain functionalities (e.g. settings). Packetized commands are always enabled, unlike text commands. In addition, they include error detection data, making them more robust than text commands.

normal-command structure:

[preamble] [command-type] [data-size] [dataopt] [reserved field] [crc14]

preamble:	**0xEE 0xEE 0xEE 0xEE**
command-type:	Single ASCII character in the set defined in **Command Types**.
data-size:	byte value in range [0,240], which indicates size of data (in bytes) following this before *[reserved field]*
dataopt:	*datum* or *data datum* *data* or *datum* is any byte value in the range [0,255]
reserved field:	**0x00**
crc14:	Two consecutive bytes, each in range [0,127], representing the crc16 value and with the value 0x7F7F, most significant byte first. The packet crc16 is calculated over the entire packet, excluding the preamble and the crc14 itself. (See source files *crc16.[hc]* (Appendix) for details on the crc16 algorithm and polynomials to be used.)

Note: Most terminal programs will omit or not transmit a NULL (0x00) character entered in the terminal. So, if one tries to compose the command structure manually and send it through a terminal program, one might have to send everything before the reserved field, then send the 0x00 (key Ctrl+@ in ESP terminal view) by itself, and then the CRC14 checksum bytes.

In the first of the examples shown below, for instance, “0xEE 0xEE 0xEE 0xEE 0x23 0x01 0x03” can be sent using ESP’s Terminal, then the NULL character can be sent by typing “Ctrl + Shift + 2”, and then “0x4E 0x71” can be sent to complete the command sequence.

Examples:

0xEE 0xEE 0xEE 0xEE 0x23 0x01 0x03 **0x00** 0x4E 0x71

This executes the beep command, causing the reader to beep 3 times. “#%03” in text command format. The first 4 0xEE are the preamble, followed by the ‘#’ character or 23h the beep command, and followed by data size of 1 with a data datum of 3 following it. 0x00 is the reserved field followed by two bytes CRC14 of the command + data size + data datum.

0xEE 0xEE 0xEE 0xEE 0x49 0x00 **0x00** 0x03 0x3C

This executes the info command, “I” in text command format. Since it does not have any data datum associated with it, 0x00 following 0x49 (ASCII “I”) but before the reserved field and CRC14 (0x00 0x03 0x3C) indicating it has no *dataopt*.

0xEE 0xEE 0xEE 0xEE 0x50 0x04 0xC4 0x32 0x35 0x35 **0x00** 0x43 0x3C

This will change the reader to trigger-read mode, “P(C4)255” in text command format. 0x50 is the command-type, 0x04 indicates that following 4 bytes are the data option with 0xC4 being the register that needs to be updated and 0x32 0x32 0x35 being the new value.

0xEE 0xEE 0xEE 0xEE 0x50 0x03 0xA9 0x35 0x63 **0x00** 0x77 0x6B

This will change the reader Bypass Gain to 92 (0x5c) percent, “P(A9)5c” in text command format. 0x50 is the command-type, 0x03 indicates that the following 3 bytes are the data option with 0xA9 being the register that needs to be updated and 0x35 0x63 (5c) being the new value.

0xEE 0xEE 0xEE 0xEE 0x50 0x07 0x28 0x31 0x45 0x41 0x29 0x46 0x41 **0x00** 0x14 0x18

This will change Good Read Duration to 250ms, “P(1EA)FA” in text command format. In this case the register 1EA is more than one byte can hold; the register is converted to its individual ASCII hex value bounded by 0x28 and 0x29, underlined above.

Command Types

#	Causes the reader to beep the specified number of times; *data* contains the number as a single character in the range [0,127]. (The reader will respond with *d* or *e*.) Example – beep three times: #%03
$	Posts an event to the reader; *data* contains the event number as a single character. See setting 39 in **Reader Settings** for a list of the event numbers. (The reader will respond with *d* or *e*.)
(	Causes the reader to upload any logged error messages (no *data*) (The reader will respond with a g packet, zero or more z packets, and a final *d* or *e*. Each *z* packet contains a portion of the requested data in its *data* field. **Note:** This is very similar to the response to the X command; however, *p* packets are not applicable and the *g* and *d/e* packets are not suppressed even in raw mode.)
)	Causes the reader to erase its log of error messages (no *data*) (The reader will respond with *d* or *e*.)
,	Causes the reader to send a list of current reader settings (no *data*) (The reader will respond with d containing a space-separated list of all setting values (in order, expressed as hexadecimal ASCII characters) or with *e*.)
/	Toggle a bit (or bits) in a reader setting; *data* contains a printable ASCII string in the following format: hexadecimal setting number in parentheses followed by a 32-bit signed integer value, expressed in ASCII hexadecimal characters (with optional minus sign) or ASCII decimal characters preceded by the '#' character, e.g., /(2e)1000 or /(2e)#4096; the specified integer is XOR'ed with the existing setting value. (The reader will respond with *d* or *e*.) **Note:** See **Reader Settings** for possible reader settings.
1	Indicates the start of a file download; *data* is empty. This command is followed by a sequence of *2* commands containing the file data and a download-end command (e.g., *5*). (The reader will respond with *d* or *e*.)
2	Indicates a continuation of a file download; *data* contains the next portion of the file data. (The reader will not send any response.)
5	Indicates the end of a regular file download; *data* contains the name of the file, which is from 1 to 200 letters, digits, periods, hyphens, and underscores, terminated with ASCII NUL. (The reader will respond with *d*, *e*, or *f*.)

9	Requests the reader to delete a file from its storage; *data* contains the file name, terminated with ASCII NUL.
;	Reserved (nop – treated as a comment)
<	Causes the reader to send a list of saved reader settings (no *data*) (see ',' command)
=	Puts setting directly to reader's non-volatile memory so that it will take effect upon next reboot; *data* is as defined in the / command; the specified integer replaces the existing setting value. **Note:** This command can be used to set communication modes without losing communication during the process. (The reader will respond with *d* or *e*.)
>	Causes the reader to send a string of text to the host as a *z* packet; *data* contains the text to send. (The reader will respond with a z packet containing the text.)
@	Causes the reader to reset its internal date/timestamp to the specified time; *data* contains the date and/or time in one of the following formats. yyyy-mm-dd hh:mm:ss yyyy-mm-dd hh:mm hh:mm:ss hh:mm **Note:** The separators are optional; only digits are significant. (The reader will respond with *d* or *e*.) Examples: Set to midnight: @00:00 Set to Sept 1, 2005 11:52:02 PM: @2005-09-01 23:52:02 **Note:** On units without a battery-backed real-time clock, the date and time will reset to 2000-01-01 00:00:00 upon power-up.)
A	Notifies the reader that the previously sent data were rejected for one of the following reasons: • The packet was encrypted and the decryption failed. • The host is locked to a different reader. The reader should indicate to the user that the packet has been rejected; e.g., it may sound error beeps. See related setting 0x12f, notify-of-packet-rejection. (The reader will not respond to the host.)

G	Get setting from reader; *data* contains a single character (0-255), which is the setting number. (The reader will respond with *d* and the setting value as a sequence of 8 ASCII hexadecimal digits or with *e*.) **Note:** See **Reader Settings** for possible reader settings.
I	Requests the reader to send its information string (no *data*). (The reader will respond with *i* or *e*.)
J	Requests the reader to restore settings to defaults (no *data*). (The reader will respond with *d* or *e*.)
J1	Complete restore of factory setup. Will overwrite the apps and settings.
L	Requests the reader to send a list of its stored files. *data* is: • (no *data*) or "0"; all non-hidden files. • "1"; hidden files (The reader will respond in the same manner as with the '(' command, each *z* packet containing a file name as a NUL-terminated string of printable ASCII characters.)
O	Set a bit (or bits) in a reader setting; *data* is as defined in the / command; the specified integer is ORed with the existing setting value. (The reader will respond with *d* or *e*.) **Note:** See **Reader Settings** for possible reader settings.
P	Put setting to reader; *data* is as defined in the / command; the specified integer replaces the existing setting value. (The reader will respond with *d* or *e*.) **Note:** See **Reader Settings** for possible reader settings.
Q	Clear a bit (or bits) in a reader setting; *data* is as defined in the / command; the ones-complement of the specified integer is AND'ed with the existing setting value. (The reader will respond with *d* or *e*.) **Note:** See **Reader Settings** for possible reader settings.
R	Requests that the previously sent packet be re-sent by the reader; *data* may specify a maximum packet size the receiver will accept: *data* is either empty or specifies a 16-bit big-endian unsigned integer (2 bytes). If *data* is empty or specifies a size less than 32 (the minimum packet size), the reader will use its preferred maximum packet size. Otherwise, it will use the specified max packet size (or less) and will fragment data across multiple smaller packets when necessary. (The reader will respond by re-sending its previous packet or with *e* if there was no previous packet. If the max data size has changed, it may resend the previous data in a sequence of more than one packet.)

T	Requests the current date and time (no *data*). (The reader will respond with *d* with *data* containing the date and time formatted as yyyy-mm-dd hh:mm:ss.) **Note:** On units without a battery-backed real-time clock, the date and time will reset to 2000-01-01 00:00:00 upon power-up.
U	Reserved for script engine.
W	Requests the reader to write its current settings from RAM to its non-volatile memory. (The reader will respond with *d* or *e*.)
Y	Acknowledge the receipt of a packet; *data* specifies the received packet number (one byte). (The reader will not respond.)
Z	Request the reader to reboot. *data* is: • *empty* or '0'; reboot the reader. • '1'; restart application. (The reader will respond with *d* or *e* before it reboots.)
^	Requests the reader to upload the specified stored file; *data* contains the file name, terminated with ASCII NUL. The reader will respond with: • 'g' packet containing *"filename<tab>(size)"* • 'z' packet(s) • 'd' packet containing "EOF<*tab*>(CRC16)" **Note:** *filename* "help" is reserved to send command information.
–	Causes the reader to wait for all buttons to be released and clear its event queue. (The reader will respond with *d* or *e*.)
|	Process *data* as a decoded string. (The reader will respond with *d* or *e*.)

Simple Protocol

The file is split into blocks of 236 or less bytes each and downloaded to the reader via *1, 2,* and *5* commands using the following sequence:

1. Send a *1* command to initialize the download.
2. Wait for a *d* or *e* response from the reader or a timeout.
   a. If timeout or *e* response, restart the sequence at step 1.
   b. If *d* response, continue to step 3.
3. Send a series of *2* commands, each with a portion of the file. (The reader will not send any response.)
4. Send a *5* command to end the download and install the file.
5. Wait for a *d, e,* or *f* response from the reader or a timeout.
   a. If *f* response or timeout, restart the sequence at step 1.
   b. If *e* response, repeat step 5.
   c. If *d* response, file download has completed successfully.

Note: The timeout will need to be increased from the normal response timeout to allow the firmware time to write the file to the flash memory.

Reader Settings

The host sets the reader settings using the */, O, P, Q,* and = commands and reads them using the *G*, **,**, and < commands.

For example, the following P command sets register 2C to the value C8.

P(2C)C8

Note: For two-digit setting numbers (i.e., settings 00 through fd), an alternative format may be used: in place of the parentheses and hexadecimal setting number, substitute a single character, which represents the setting number. The equivalent to the example above is P,C8 (the ASCII '**,**' character has the hexadecimal value 2c). (In certain circumstances, such as with text commands, "percent-encoding" may be used for encoding a character as a sequence consisting of the percent character followed by two hexadecimal digits. With percent-encoding, the example may be expressed as P%2CC8.)

In the **Reader Settings Table**, the **Reg** column is the setting number, in **hexadecimal**, to be used with the commands identified above. In the **Default** column, all values are in **hexadecimal** unless otherwise specified. To use decimal values in commands you must precede the data with a pound sign '**#**'. The following P command sets register 2C to the same value as the example above:

P(2C)#200

Since the single digit values of 0 through 9 are identical in decimal and hexadecimal, no indicator is needed.

Binary Dip Switch

Some registers are what Microscan terms a 'Binary Dip Switch' where the value of each bit of the data string switches on or off some part of the behavior of that register. The bits are numbered from least significant to most (right to left). Each bit can be on or off (1 or 0).

An example of this is register 0B, 'Codabar Checksum'. The following settings are possible:

Bit (R to L)	Controls	Value
0	Codabar Checksum Checking	0: Disabled
		1: Enabled
1	Strip Checksum from Output	0: Disabled
		1: Enabled

Given the settings above, the binary string turns Codabar Checksum ON and strips it from output.

Thus, the command to implement the settings above would be:

P(48)3

or

P(48)#03

Field of Interest

The reader optics are typically split into two separate fields - Field Of Interest 0 (FOI0) and Field Of Interest 1 (FOI1). In certain circumstances, these fields can be customized to the requirements of the user. In the default configuration of these fields FOI0 is the High Density (HD) field and FOI1 is the Wide (W) field.

At a given focus distance, the HD field is designed to read small, low-mil symbols while the Wide field is designed to pick up large, wide symbols.

This document will refer to FOI0 as HD and FOI1 as Wide.

Reader Settings Table

<table>
<tr><th>Reg</th><th>Setting Name</th><th>Default (Hex)</th><th>Comment</th></tr>
<tr><td>04</td><td>Continuous Illumination During Read</td><td>0</td><td>0: Minimal Illumination
1: Leave Illumination On Until End Read Cycle
Leave illumination on during read.</td></tr>
<tr><td>08</td><td>Reader Packet Format</td><td>1</td><td>1: Raw
2: Packet Mode Version 1
For example, USB “two-way” native:
1B: 5 (USB Native)
08: 2 (packet mode)
42: 1 (expect response)
Also see registers: 1B, 42</td></tr>
<tr><td>0A</td><td>NEC 2 of 5 Symbology</td><td>1</td><td>Binary Dip Switch
<table>
<tr><th>Bit</th><th>Controls</th><th>Value</th></tr>
<tr><td rowspan="2">0</td><td rowspan="2">NEC 2 of 5 Decoding</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Checksum checking</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Strip checksum from result</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">3</td><td rowspan="2">1 Digit Symbol Allowed</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">4</td><td rowspan="2">2 Digit Symbol Allowed</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
</table>
Note: All symbol lengths greater than 2 are always enabled when NEC 2 of 5 Decoding is enabled.</td></tr>
</table>

<table>
<tr><td>0B</td><td>Matrix 2 of 5 Symbology</td><td>1</td><td>Binary Dip Switch<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="2">0</td><td rowspan="2">Matrix 2 of 5 Decoding</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">1</td><td rowspan="2">Checksum checking</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">2</td><td rowspan="2">Strip checksum from result</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">3</td><td rowspan="2">1 Digit Symbol Allowed</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">4</td><td rowspan="2">2 Digit Symbol Allowed</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr></table>Note: All symbol lengths greater than 2 are always enabled when Matrix 2 of 5 Decoding is enabled.</td></tr>
<tr><td>0C</td><td>Telepen Symbology</td><td>1</td><td>0: Disabled
1: Enabled</td></tr>
<tr><td>0D</td><td>Enable Non-Square Data Matrix Symbology</td><td>0</td><td>0: Disabled
1: Enabled</td></tr>
<tr><td>0F</td><td>Targeting Control</td><td>1</td><td>0: Targeting Disabled
1: Targeting Enabled</td></tr>
<tr><td>16</td><td>Data Matrix Rectangular Symbology</td><td>0</td><td>0: Disabled
1: Enabled</td></tr>
<tr><td>19</td><td>Data Matrix Symbology</td><td>1</td><td>Binary Dip Switch<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="2">0</td><td rowspan="2">Data Matrix Decoding</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">1</td><td rowspan="2">Inverse Data Matrix Decoding</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr></table></td></tr>
</table>

1B	Communications Mode	8	1: RS232 serial 2: USB keyboard 5: USB Native (HID) 6: USB VComm 8: Dynamic (1 if decode is RS232; 2 if decode is USB) This setting is used in conjunction with settings 08 and 42 to configure the communication mode between standard “one-way” and “two-way” modes. For example, USB “two-way” native: 1b: 5 (USB Native) 08: 2 (packet mode) 42: 1 (expect response) **Note:** The following must be completed within 1 second. first output report with numlock set and capslock clear second output report with numlock set and capslock clear third output report with capslock set numlock clear fourth output report with capslock set numlock clear fifth output report with numlock set and capslock clear sixth output report with numlock set and capslock clear On the last output report comm protocol is set to raw mode, comm expect response is false and comm mode is USB Downloader mode. Also see registers: 08, 42
1C	Serial Baud Rate	1C200 (#115200)	All standard baud rates up #115200 • #9600 (2580) • #19200 (4B00) • #38400 (9600) • #57600 (E100) • #115200 (1C200)
1D	Serial Stop Bits	1	1: One 2: Two
1E	Serial Data Bits	8	7: Seven 8: Eight

<table>
<tr><td>22</td><td>Serial Parity</td><td>0</td><td>0: None
1: Odd
2: Even</td></tr>
<tr><td>26</td><td>Beep Volume (percent)</td><td>64 (#100)</td><td>Valid Range: 0 to 64 (#100) Percent
This is the current percentage of full volume potential.
Also see registers: 59, A7</td></tr>
<tr><td>29</td><td>PDF417 Symbology</td><td>1</td><td>0: Disabled
1: Enabled
Also see registers: 2A, CF</td></tr>
<tr><td>2A</td><td>MicroPDF417 Symbology</td><td>0</td><td>0: Disabled
1: Enabled
Also see registers: 29, CF</td></tr>
<tr><td>2B</td><td>QR Code Symbology</td><td>1</td><td>Binary Dip Switch
<table>
<tr><th>Bit</th><th>Controls</th><th>Value</th></tr>
<tr><td rowspan="2">0</td><td rowspan="2">QR Code Decoding</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Inverse QR Code Decoding</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Micro QR Code Decoding</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">3</td><td rowspan="2">Inverse Micro QR Code Decoding</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
</table></td></tr>
</table>

<table>
<tr>
<td>2C</td>
<td>Idle Mode Countdown Timer (ms)</td>
<td>64 (#100)</td>
<td>Valid Range: 0 to 7FFFFFFF Milliseconds
Counts down to the change to Idle Mode.
The most significant bit (MSB) of the 32-bit register indicates whether this timer is enabled. Enable or Disable the timer by setting the MSB.
You can change the big directly by setting the register value (such as setting to #100) or you can change the value of just the MSB using the O (set), Q (clear) or / (toggle) bit commands. See Command Types for more information on these commands.
<table>
<tr><th>Action</th><th>Command</th></tr>
<tr><td>Enable</td><td>O(32)#-2147483648</td></tr>
<tr><td>Disable</td><td>Q(32)#-2147483648</td></tr>
<tr><td>Toggle</td><td>/(32)#-2147483648</td></tr>
</table>
Idle Mode is the time between the last user interaction with the reader (button press, etc.) or firmware interaction (communications, etc.) and Standby Mode. There are many user and firmware events that will reset the timer. Therefore, it may seem that the timer is longer than the value set.
This state: Active (highest power usage)
Next state: Idle</td>
</tr>
<tr>
<td>2D</td>
<td>Keyboard Maps</td>
<td>0</td>
<td>0: US English (without leading 0 in the ALT _ Number)
1: ASCII (ALT+number) - universal
2: Custom (requires user to download keyboard map)
3: US English (with leading 0 in the ALT + number for non-printable ASCII)
4: French Keyboard
5: German Keyboard
6: Japanese Keyboard
7: US English (with CTRL + char for non-printable ASCII)</td>
</tr>
</table>

34	Maximum Candidate Decodes Per Read	1	The Reader will process up to this number of codes per “read code” event. If there are more than this many codes in the field of view and within target tolerance, only the first ones will be decoded. For fastest performance with single codes, set to 1.
35	Button Stay-Down Time (ms)	0	Valid Range: 0 to 7FFFFFFF Milliseconds Keep processing the “read code” events for this amount of time (act as if the button stays down for this time)
36	Number of Control Frames Before Picture Capture	0	Valid Range: 0 to 7FFFFFFF Frames Number of frames captured and discarded before live picture to give the gain control time to adjust. Also see registers: 43, AC, AD, AE, AF
39	Trigger 1	3	The specified event is posted upon press of this button. The events are defined below: 0: No Action 1: Keep Awake 2: Show Target 3: Read In Both Fields (Default) 4: Default Event Selected By Hardware 5: Read In High Density field (FOI0). 6: Read In Wide field (FOI1). 7: Take Picture 8: Read In Most Recently Successful Field 255: Idle
40	Text Command Timeout (ms)	2AF8 (#11000)	Valid Range: 0 to 7FFFFFFF Milliseconds The maximum time during which a complete text command from Host must be received. (Pending text command data is discarded when the timeout is exceeded.)

<table>
<tr>
<td>41</td>
<td>Text Commands</td>
<td>8</td>
<td>
Binary Dip Switch
<table>
<tr><th>Bit</th><th>Controls</th><th>Value</th></tr>
<tr><td rowspan="2">0</td><td rowspan="2">Text Commands</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Suppress Echo</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Suppress Responses</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">3</td><td rowspan="2">Disable Text Commands but Enable Magic Sequence</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">4</td><td rowspan="2">Suppress URL Decode; See Below</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">5</td><td rowspan="2">Accept On Time-out</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
</table>
Magic Sequence:
The Magic Sequence is the string ";>PAx" where x is 1, 3, or 7 as defined above. This would normally be used in command text files, which would begin with the text-command-on sequence and end with the command to return to this special mode. For example:
;>PA7
;any desired commands here
PA8
Suppress URL Decode:
For example, if enabled, P%418 will not equal PA8. The % is not recognized as an escape character
Accept On Timeout:
</td>
</tr>
</table>

<table>
<tr><td>42</td><td>Expect Acknowledgement From Host</td><td>0</td><td>0: Reader doesn’t wait for acknowledge
1: Reader will retransmit data when Host doesn’t acknowledge receipt
This setting is used in conjunction with settings 1B and 42 to configure the communication mode between standard “one-way” and “two-way” modes.
For example, USB “two-way” native:
1B: 5 (USB Native)
08: 2 (packet mode)
42: 1 (expect response)
Also see registers: 08, 1B</td></tr>
<tr><td>43</td><td>JPEG Picture Quality (percent)</td><td>32 (#50)</td><td>Valid Range: 0 to 64 (#100) Percent
0: Raw Image (No JPEG Compression)
1 To 100: JPEG Compression Quality Percent
Also see registers: 36, AC, AD, AE, AF</td></tr>
<tr><td>45</td><td>Read Cycle Timeout</td><td>1F4 (#500)</td><td>Valid Range: 0 to FFFF (#65535) ms</td></tr>
<tr><td>47</td><td>Maxicode Symbology</td><td>0</td><td>Binary Dip Switch<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="2">0</td><td rowspan="2">Maxicode Decoding, Mode 0</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">1</td><td rowspan="2">Maxicode Decoding, Mode 1</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">2</td><td rowspan="2">Maxicode Decoding, Mode 2</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">3</td><td rowspan="2">Maxicode Decoding, Mode 3</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">4</td><td rowspan="2">Maxicode Decoding, Mode 4</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">5</td><td rowspan="2">Maxicode Decoding, Mode 5</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">6</td><td rowspan="2">Maxicode Decoding, Mode 6</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr></table></td></tr>
<tr><td>48</td><td>Codabar Checksum</td><td>0</td><td>Binary Dip Switch<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="2">0</td><td rowspan="2">Codabar Checksum Checking</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">1</td><td rowspan="2">Strip Checksum from Output</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr></table></td></tr>
</table>

<table>
<tr><td>49</td><td>Code 39 Symbology</td><td>0</td><td>0: Disabled
1: Enabled
Code 39 Full ASCII Decoding</td></tr>
<tr><td>4A</td><td>Composite Codes</td><td>0</td><td>0: Disabled
1: Enabled
Composite Code Decoding
Also see register: D8</td></tr>
<tr><td>4B</td><td>Postal Code Symbology</td><td>0</td><td>0: Disabled
1: Enabled</td></tr>
<tr><td>4C</td><td>GS1 Symbology</td><td>1F (#31)</td><td>Binary Dip Switch
<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="2">0</td><td rowspan="2">GS1 Expanded decoding</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">1</td><td rowspan="2">GS1 Expanded Stacked decoding</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">2</td><td rowspan="2">GS1 Limited decoding</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">3</td><td rowspan="2">GS1-14 and GS1-14 Truncated decoding</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">4</td><td rowspan="2">GS1-14 Stacked and GS1-14 Stacked Omnidirectional decoding</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr></table></td></tr>
<tr><td>4D</td><td>UPC Expansion</td><td>0</td><td>0: Disabled
1: Enabled
Also see registers: 4E, 6A, 74</td></tr>
<tr><td>4E</td><td>UPC Supplemental</td><td>0</td><td>0: Disabled
1: Enabled
Also see registers: 4D, 6A, 74</td></tr>
</table>

<table>
<tr>
<td>4F</td>
<td>MSI Plessey Symbology</td>
<td>1</td>
<td>Binary Dip Switch
<table>
<tr><th>Bit</th><th>Controls</th><th>Value</th></tr>
<tr><td rowspan="2">0</td><td rowspan="2">MSI Plessey</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="8">[3:1]</td><td rowspan="8">Checksum</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td>1: 1 mod 10</td></tr>
<tr><td>2: mod 10 and mod 11</td></tr>
<tr><td>3: 2 mod 10</td></tr>
<tr><td>5: 1 mod 10 strip cs</td></tr>
<tr><td>6: mod 10 and mod 11 strip cs</td></tr>
<tr><td>7: 2 mod 10 strip cs</td></tr>
<tr><td rowspan="2">4</td><td rowspan="2">Improved Bounds</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
</table></td>
</tr>
<tr>
<td>50</td>
<td>Aztec Symbology</td>
<td>1</td>
<td>Binary Dip Switch
<table>
<tr><th>Bit</th><th>Controls</th><th>Value</th></tr>
<tr><td rowspan="2">0</td><td rowspan="2">Aztec decoding</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Inverse Aztec decoding</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
</table></td>
</tr>
<tr>
<td>53</td>
<td>Decoder HD field (FOI0) Width</td>
<td>280 (#640)</td>
<td>Valid Range: 1 to 640 pixels
Decoder uses only the specified pixel width in the HD field (FOI0).
Also see registers: 54, 98, 99</td>
</tr>
<tr>
<td>54</td>
<td>Decoder HD field (FOI0) Height</td>
<td>3C0 (#960)</td>
<td>Valid Range: 1 to 960 pixels
Decoder uses only the specified pixel height in the HD field (FOI0).
Also see registers: 53, 98, 99</td>
</tr>
<tr>
<td>55</td>
<td>Notify Of Read Failure</td>
<td>0</td>
<td>0: Disabled
1: Send “r” packet on no-read (See “r” packet in Packet Data.)
0x100xx: post event on no-read, where the lower 8 bits specify the event number. For example, 0x10009 to post Event 0x09.</td>
</tr>
</table>

<table>
<tr><td>59</td><td>Beep Duration</td><td>64 (#100)</td><td>Valid Range: 0 to 7FFFFFFF Milliseconds
Also see registers: 26, A7</td></tr>
<tr><td>66</td><td>Bypass Illumination</td><td>0</td><td>Valid Range: 0 to 64 (#100) percent</td></tr>
<tr><td>6A</td><td>UPC Symbology</td><td>1</td><td>0: Disabled
1: Enabled
Also see registers: 4D, 4E, 74</td></tr>
<tr><td>6B</td><td>Code 39 Symbology</td><td>1</td><td>0: Disabled
1: Enabled
Also see register: 70</td></tr>
<tr><td>6C</td><td>Code 93 Symbology</td><td>1</td><td>0: Disabled
1: Enabled</td></tr>
<tr><td>6D</td><td>Code 128 Symbology</td><td>1</td><td>0: Disabled
1: Enabled</td></tr>
<tr><td>6E</td><td>Interleaved 2 Of 5 Symbology</td><td>1</td><td>0: Disabled
1: Enabled
Also see registers: 71, C9</td></tr>
<tr><td>6F</td><td>Codabar Symbology</td><td>1</td><td>0: Disabled
1: Enabled</td></tr>
<tr><td>70</td><td>Code 39 Checksum</td><td>0</td><td>Binary Dip Switch
<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="2">0</td><td rowspan="2">Code 39 Checksum Checking</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">1</td><td rowspan="2">Strip Checksum from Output</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr></table>Also see register: 6B</td></tr>
<tr><td>71</td><td>Interleaved 2 Of 5 Checksum</td><td>0</td><td>Binary Dip Switch
<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="2">0</td><td rowspan="2">Interleaved 2 of 5 Checksum Checking</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">1</td><td rowspan="2">Strip Checksum from Output</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr></table>Also see register: 6E, C9</td></tr>
<tr><td>74</td><td>UPC Short Margin</td><td>1</td><td>0: Disabled
1: Enabled
Also see registers: 4D, 4E, 6A</td></tr>
<tr><td>78</td><td>Settings Lock</td><td>1</td><td>1: Settings unlocked
3: Settings locked (except settings Lock)</td></tr>
</table>

<table>
<tr><td>85</td><td>Trioptic Options</td><td>0</td><td>Binary Dip Switch
<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="2">0</td><td rowspan="2">Trioptic Decoding, Normal Quiet Zones</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">1</td><td rowspan="2">Allow Short Quiet Zones</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">2</td><td rowspan="2">No Quiet Zones (requires firmware version 3280+)</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr></table></td></tr>
<tr><td>86</td><td>Motion Detection: Event</td><td>3</td><td>Valid Range: (see register 39)
Motion detection is enabled by setting register C4 to 0xF0.
This register is reset to 0 (disabled) when register C4 is changed away from 0xF0.
When motion is detected, this event is posted.
See register 39 for list of events.
Also see registers: 20E</td></tr>
<tr><td>87</td><td>Motion Sensitivity</td><td>5</td><td>Valid Range: 0 to FFFF (#65535) ms
Also see registers: 86, 20E</td></tr>
<tr><td>93</td><td>Suppress Beep On Decode</td><td>0</td><td>0: Beep indicating decode before JavaScript processing
1: Call JavaScript without beeping to indicate decode
Normally, the Reader beeps as soon as decodes are read and processes them via JavaScript if necessary after the beep. To enable JavaScript to control the beep feedback, change this setting to 1; this will suppress the beep; the JavaScript would typically beep if the decode is valid or start another read cycle if it isn’t.
This setting does not suppress beeps for anything but a successful decode event.</td></tr>
<tr><td>98</td><td>Decoder HD field (FOI0) X Offset</td><td>0</td><td>Valid Range: 0 to 639 pixels
Decoder uses the pixels after the specified pixel offset in the HD field (FOI0).
Also see registers: 53, 54, 99</td></tr>
</table>

99	Decoder HD field (FOI0) Y Offset	0	Valid Range: 0 to 959 pixels Decoder uses the pixels after the specified pixel offset in the HD field (FOI0). Also see registers: 53, 54, 98
9A	Decoder Wide field (FOI1) X Offset	0	Valid Range: 0 to 639 pixels Decoder uses the pixels after the specified pixel offset in the Wide field (FOI1). Also see registers: 9B, C7, C8
9B	Decoder Wide field (FOI1) Y Offset	0	Valid Range: 0 to 959 pixels Decoder uses the pixels after the specified pixel offset in the Wide field (FOI1). Also see registers: 9A, C7, C8
9D	Target Tolerance (percent)	640 (#1600)	Valid Range: 0 to 7FFFFFFF Percent For the Reader to accept a code, the target dot must be within the code rectangle or in proximity to the symbol. The nearness is defined as this percentage of the code's smaller dimension. For example, with a 10 x 20 mm code and a setting of 150 (%), the target dot must be within 15 mm of the code. Any value over 1000 is considered infinite tolerance, and no target checking is performed.
A2	Default Event Delay (ms)	64 (#100)	Valid Range: 0 to 7FFFFFFF Milliseconds The Reader will pause for this amount of time between each posting of the default event (used with "continuous read" mode). Also see register C4
A7	Beep Pulse Separation (ms)	64 (#100)	Valid Range: 0 to 7FFFFFFF Milliseconds The spacing in milliseconds between beeps. Also see registers: 26, 59
AC	Wide field (FOI1) Picture Window Left Position	0	Specify left edge of window used with "take picture." The position and size are relative to the virtual image (i.e., not the rotated physical image). **Note:** Overall image is 960 pixels by 1280 pixels. Upper half is Wide field (FIO1); lower half is High Density (FOI0). Also see registers: 36, 43, AD, AE, AF

AD	Wide field (FOI1) Picture Window Upper Position	0	Specify upper edge of window used with “take picture.” The position and size are relative to the virtual image (i.e., not the rotated physical image). **Note:** Overall image is 960 pixels by 1280 pixels. Upper half is Wide field (FIO1); lower half is High Density (FOI0). Also see registers: 36, 43, AC, AE, AF
AE	Wide field (FOI1) Picture Window Width	500 (#1280)	Specify width of window used with “take picture.” The position and size are relative to the virtual image (i.e., not the rotated physical image). **Note:** Overall image is 960 pixels by 1280 pixels. Upper half is Wide field (FIO1); lower half is High Density (FOI0). Also see registers: 36, 43, AC, AD, AF
AF	Wide field (FOI1) Picture Window Height	3C0 (#960)	Specify height of window used with “take picture.” The position and size are relative to the virtual image (i.e., not the rotated physical image). **Note:** Overall image is 960 pixels by 1280 pixels. Upper half is Wide field (FIO1); lower half is High Density (FOI0). Also see registers: 36, 43, AC, AD, AE
B0	Target On Before Picture (ms)	3E8 (#1000)	Valid Range: 0 to 7FFFFFFF Milliseconds 0: Target off before picture capture Also see registers: 36, 43, AC, AD, AE
B3	Number Of Retries Before Reader Gives Up Sending Packet	3	**Note:** The value 1 is defined as the original send attempt but no resends. Also see register: 42
BF	USB Keyboard Poll Rate	A (#10)	Valid Range: 1 to FF (#255) Milliseconds The Host is requested to poll the USB device at the specified period.

C4	Default (Continuous) Event	FF (#255)	The default value of FF (idle event) disables “continuous scanning”. Use one of the read events to enable “continuous scanning.” See setting 39 for the list of events. When no button is pressed but the Reader is still in active mode (i.e., not power-saving idle or sleep modes), this event will be posted. Also see register: 39
C7	Decoder Wide field (FOI1) Width	280 (#640)	Valid Range: 1 to 280 (#640) pixels Decoder uses only the specified pixel height in the Wide field (FOI1). Also see registers: 9A, 9B, C8
C8	Decoder Wide field (FOI1) Height	3C0 (#960)	Valid Range: 1 to 3C0 (#960) pixels Decoder uses only the specified pixel height in the Wide field (FOI1). Also see registers: 9A, 9B, C7
C9	Interleaved 2 Of 5 Lengths	0	FFFFFFFC: 2 and 4 digit disabled FFFFFFFD: 2 digit enabled FFFFFFFE: 4 digit enabled Also see registers: 6E, 71
CF	Macro PDF417 Symbology	0	0: Disabled 1: Enabled Also see registers: 29, 2A
D8	Composite Codes Require Both Elements	1	0: Accept any composite element 1: Only accept composite codes if both elements could be decoded. Also see register 4A
EB	Maximum Reader To Host Packet Data Size	4000 (#16384)	Valid Range: 1 to 4000 (#16384)
EC	Host Acknowledgement Time Limit Multiplier (ms)	F (#15)	Valid Range: 0 to 7FFFFFFF Milliseconds When Expect Acknowledgement From Host (register 42) is nonzero, the Reader will wait up to Host Acknowledgement Time Limit (register 37) + dataSize * Host Acknowledgement Time Limit Multiplier (register EC) milliseconds to receive an acknowledgement from the Host.

ED	Prefix Decode Result With AIM Symbology Identifiers	0	0: Don't prefix with AIM identifier 1: Prefix decode result with ISO/IEC standard 15424/AIM symbology identifier
F0	Allow Code 128 Short Margin	1	0: Disabled 1: Enabled
F6	Code 39 Short Margin	1	0: Disallow short margin Code 39 symbol decoding 1: Allow short margin Code 39 symbol decoding
F8	PharmaCode Symbology	0	0: Disabled 1: Enabled
F9	PharmaCode Bar Count	1004 (#4100)	Valid Range: Each 8 bits can be 04 to 10 (#16) Bit 0 – Bit 7: min bar count, 04 to 10 (#16) Bit 9 – Bit 15: max bar count, 04 to 10 (#16)
FA	PharmaCode Min Value	F (#15)	Valid Range: F (#15) to 1FFFE (#131070)
FB	PharmaCode Max Value	1FFFE (#131070)	Valid Range: F (#15) to 1FFFE (#131070)
10B	Enable JavaScript	1	0: Disabled 1: Enabled When set to 0 installed scripts are disabled. This can be useful from boot mode for recovering the unit if a non-responsive script is installed.
10D	Data Matrix Symbol Identification Effort	2	0: Normal effort 1: Increase effort 2: Max effort Increases the decoder's effort to find a Data Matrix symbol in an image.
12C	Data Matrix Improvement	1	0: Disabled 1: Enabled Improves the decoding capability of the Reader on low contrast or pixelated Data Matrix bar codes

<table>
<tr><td>12D</td><td>Hong Kong 2 Of 5 Symbology</td><td>0</td><td>Binary Dip Switch<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="2">0</td><td rowspan="2">Hong Kong 2 of 5 decoding</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">1</td><td rowspan="2">1 Digit Symbol Allowed</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr><tr><td rowspan="2">2</td><td rowspan="2">2 Digit Symbol Allowed</td><td>0: Disabled</td></tr><tr><td>1: Enabled</td></tr></table></td></tr>
<tr><td>137</td><td>PDF417 Handle Invalid Shift</td><td>0</td><td>0: Disabled
1: Enabled
Allows the decoding of PDF417 bar codes that were improperly encoded</td></tr>
<tr><td>159</td><td>Ignore Duplicate Code (ms)</td><td>0</td><td>Valid Range: 0 to 7FFFFFF Milliseconds
Consecutive duplicate codes (i.e., codes that contain the same data) are blocked for this amount of time (in milliseconds). 0 turns off blocking of duplicate codes.</td></tr>
<tr><td>1D7</td><td>Morphology</td><td>0</td><td>Binary Dip Switch<table><tr><th>Bit</th><th>Controls</th><th>Value</th></tr><tr><td rowspan="3">1-0</td><td rowspan="3">Technique</td><td>0: None</td></tr><tr><td>1: Erode</td></tr><tr><td>2: Dilate</td></tr><tr><td rowspan="3">3-2</td><td rowspan="3">Size</td><td>0: Small 3x3</td></tr><tr><td>1: Med. 5x5</td></tr><tr><td>2: Large 7x7</td></tr></table></td></tr>
<tr><td>1D8</td><td>BC412 Status</td><td>1</td><td>0: Disabled
1: Enabled</td></tr>
<tr><td>1D9</td><td>UPC/EAN Status</td><td>1</td><td>0: Disabled
1: Enabled</td></tr>
</table>

<table>
<tr><td>1DC</td><td>Pharmacode Settings</td><td>28A40</td><td>Binary Dip Switch<table>
<tr><th>Bit</th><th>Controls</th><th>Value</th></tr>
<tr><td rowspan="2">0</td><td rowspan="2">Direction</td><td>0: Forward</td></tr>
<tr><td>1: Reverse</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">Fixed Symbol Length Status</td><td>0: Disabled</td></tr>
<tr><td>1: Enabled</td></tr>
<tr><td rowspan="4">3-2</td><td rowspan="4">Bar Width Status</td><td>0 = Mixed</td></tr>
<tr><td>1 = All Narrow</td></tr>
<tr><td>2 = All Wide</td></tr>
<tr><td>3 = Use Fixed Threshold</td></tr>
<tr><td>8-4</td><td>Minimum Number of Bars Value</td><td>4-10 (#16) (Default 4)</td></tr>
<tr><td>13-9</td><td>Fixed Symbol Length Value</td><td>1-10 (16) (Default 5)</td></tr>
<tr><td>29-14</td><td>Fixed Bar Width Threshold</td><td>0-FFFF (#65535) (Default 10)</td></tr>
</table></td></tr>
<tr><td>1EA</td><td>Good Read Duration (ms)</td><td>C8 (#200)</td><td>Valid Range: 0 to 7FFFFFF Milliseconds</td></tr>
<tr><td>1EB</td><td>Decoder Data Matrix Module Size</td><td>32 (#50)</td><td></td></tr>
<tr><td>20E</td><td>Motion Detection: Start Delay (ms)</td><td>0</td><td>Valid Range: 0 to 7FFFFFFF Milliseconds
A built-in delay of 200 ms prevents motion detect to detect motion right after a successful decode. This allows the bar code to be removed without triggering a new decode. Use this to add an additional delay amount.
Also see register: 86</td></tr>
</table>

CRB System

The CRB system is a convenient method for creating and maintaining a set of commands that can be easily sent to the reader. These CRB files can be created in any text editor with the file extension of .crb. The CRB system accepts all of the valid *text commands*. The most commonly used commands are *J*, *N*, *P*, and ~. There should be one command per line. The CRB file may contain empty lines and comments as well.

The .crb files can be sent directly to the reader using the normal file transfer. As CRB files are just a list of *text commands*, they can also be sent by a serial terminal program. **Note: if using a serial terminal program the reader will first need to be set to "text command mode"; see Text Commands.**

You can request a copy of all reader configuration settings in .crb format.

Example CRC16 C Code

CRC16.h:

```
// crc16.h
#ifndef crc16_h
#define crc16_h
#include <stdint.h>
#include <stddef.h>
#ifdef __cplusplus
extern "C" {
#endif
typedef uint16_t crc_t;
crc_t crc
(       crc_t initialCrc
        , const unsigned char* bufPtr
        , size_t length
);
#ifdef __cplusplus
} // extern "C"
#endif
#endif
/*eof*/
```

CRC16.c:

```
// crc16.c
#include <crc16.h>
crc_t crc
(         crc_t initialCrc
          , const unsigned char* p
          , size_t n
)
{
          enum
          {
                crcBits = 16,
                charBits = 8,
                diffBits = crcBits - charBits
          };
          crc_t c = initialCrc;
          #include "crc16tab.h"
          while( n-- )
                c = (c << charBits) ^ crcTab[( c >> diffBits ) ^ *p++];
          return c;
}
/*eof*/
```

CRC16tab.h:

```
/* crc16tab.h
 * crc16 table of partial remainders generated by
 * mkcrctab.c with polynomial 1021.
 * included only from within crc() function in file crc16.c
 */
static const crc_t crcTab[] =
{
    0x0000, 0x1021, 0x2042, 0x3063, 0x4084, 0x50a5, 0x60c6, 0x70e7,
    0x8108, 0x9129, 0xa14a, 0xb16b, 0xc18c, 0xd1ad, 0xe1ce, 0xf1ef,
    0x1231, 0x0210, 0x3273, 0x2252, 0x52b5, 0x4294, 0x72f7, 0x62d6,
    0x9339, 0x8318, 0xb37b, 0xa35a, 0xd3bd, 0xc39c, 0xf3ff, 0xe3de,
    0x2462, 0x3443, 0x0420, 0x1401, 0x64e6, 0x74c7, 0x44a4, 0x5485,
    0xa56a, 0xb54b, 0x8528, 0x9509, 0xe5ee, 0xf5cf, 0xc5ac, 0xd58d,
    0x3653, 0x2672, 0x1611, 0x0630, 0x76d7, 0x66f6, 0x5695, 0x46b4,
    0xb75b, 0xa77a, 0x9719, 0x8738, 0xf7df, 0xe7fe, 0xd79d, 0xc7bc,
    0x48c4, 0x58e5, 0x6886, 0x78a7, 0x0840, 0x1861, 0x2802, 0x3823,
    0xc9cc, 0xd9ed, 0xe98e, 0xf9af, 0x8948, 0x9969, 0xa90a, 0xb92b,
    0x5af5, 0x4ad4, 0x7ab7, 0x6a96, 0x1a71, 0x0a50, 0x3a33, 0x2a12,
    0xdbfd, 0xcbdc, 0xfbbf, 0xeb9e, 0x9b79, 0x8b58, 0xbb3b, 0xab1a,
    0x6ca6, 0x7c87, 0x4ce4, 0x5cc5, 0x2c22, 0x3c03, 0x0c60, 0x1c41,
    0xedae, 0xfd8f, 0xcdec, 0xddcd, 0xad2a, 0xbd0b, 0x8d68, 0x9d49,
    0x7e97, 0x6eb6, 0x5ed5, 0x4ef4, 0x3e13, 0x2e32, 0x1e51, 0x0e70,
    0xff9f, 0xefbe, 0xdfdd, 0xcffc, 0xbf1b, 0xaf3a, 0x9f59, 0x8f78,
    0x9188, 0x81a9, 0xb1ca, 0xa1eb, 0xd10c, 0xc12d, 0xf14e, 0xe16f,
    0x1080, 0x00a1, 0x30c2, 0x20e3, 0x5004, 0x4025, 0x7046, 0x6067,
    0x83b9, 0x9398, 0xa3fb, 0xb3da, 0xc33d, 0xd31c, 0xe37f, 0xf35e,
    0x02b1, 0x1290, 0x22f3, 0x32d2, 0x4235, 0x5214, 0x6277, 0x7256,
    0xb5ea, 0xa5cb, 0x95a8, 0x8589, 0xf56e, 0xe54f, 0xd52c, 0xc50d,
    0x34e2, 0x24c3, 0x14a0, 0x0481, 0x7466, 0x6447, 0x5424, 0x4405,
    0xa7db, 0xb7fa, 0x8799, 0x97b8, 0xe75f, 0xf77e, 0xc71d, 0xd73c,
    0x26d3, 0x36f2, 0x0691, 0x16b0, 0x6657, 0x7676, 0x4615, 0x5634,
    0xd94c, 0xc96d, 0xf90e, 0xe92f, 0x99c8, 0x89e9, 0xb98a, 0xa9ab,
    0x5844, 0x4865, 0x7806, 0x6827, 0x18c0, 0x08e1, 0x3882, 0x28a3,
    0xcb7d, 0xdb5c, 0xeb3f, 0xfb1e, 0x8bf9, 0x9bd8, 0xabbb, 0xbb9a,
    0x4a75, 0x5a54, 0x6a37, 0x7a16, 0x0af1, 0x1ad0, 0x2ab3, 0x3a92,
    0xfd2e, 0xed0f, 0xdd6c, 0xcd4d, 0xbdaa, 0xad8b, 0x9de8, 0x8dc9,
    0x7c26, 0x6c07, 0x5c64, 0x4c45, 0x3ca2, 0x2c83, 0x1ce0, 0x0cc1,
    0xef1f, 0xff3e, 0xcf5d, 0xdf7c, 0xaf9b, 0xbfba, 0x8fd9, 0x9ff8,
    0x6e17, 0x7e36, 0x4e55, 0x5e74, 0x2e93, 0x3eb2, 0x0ed1, 0x1ef0,
};

/*eof*/
```

Appendix E — Communications Protocol

Communications Protocol Command Table

Protocol Command (Mnemonic displayed on menu)	Control Characters (Entered in menu or serial command)	Hex Value	Effect of Command
RES	^D	04	Reset
REQ	^E	05	Request
EOT	^D	04	Reset
STX	^B	02	Start of Text
ETX	^C	03	End of Text
ACK	^F	06	Acknowledge
NAK	^U	15	Negative Acknowledge
XON	^Q	11	Begin Transmission
XOFF	^S	13	Stop Transmission

Appendix F — ASCII Table

Dec	Hex	Mne	Ctrl
00	00	NUL	^@
01	01	SOH	^A
02	02	STX	^B
03	03	ETX	^C
04	04	EOT	^D
05	05	ENQ	^E
06	06	ACK	^F
07	07	BEL	^G
08	08	BS	^H
09	09	HT	^I
10	0A	LF	^J
11	0B	VT	^K
12	0C	FF	^L
13	0D	CR	^M
14	0E	SO	^N
15	0F	SI	^O
16	10	DLE	^P
17	11	DC1	^Q
18	12	DC2	^R
19	13	DC3	^S
20	14	DC4	^T
21	15	NAK	^U
22	16	SYN	^V
23	17	ETB	^W
24	18	CAN	^X
25	19	EM	^Y
26	1A	SUB	^Z
27	1B	ESC	^[
28	1C	FS	^\
29	1D	GS	^]
30	1E	RS	^^
31	1F	US	^_

Dec	Hex	Ch
32	20	SP
33	21	!
34	22	“
35	23	#
36	24	$
37	25	%
38	26	&
39	27	'
40	28	(
41	29	)
42	2A	*
43	2B	+
44	2C	,
45	2D	-
46	2E	.
47	2F	/
48	30	0
49	31	1
50	32	2
51	33	3
52	34	4
53	35	5
54	36	6
55	37	7
56	38	8
57	39	9
58	3A	:
59	3B	;
60	3C	<
61	3D	=
62	3E	>
63	3F	?

Dec	Hex	Ch
64	40	@
65	41	A
66	42	B
67	43	C
68	44	D
69	45	E
70	46	F
71	47	G
72	48	H
73	49	I
74	4A	J
75	4B	K
76	4C	L
77	4D	M
78	4E	N
79	4F	O
80	50	P
81	51	Q
82	52	R
83	53	S
84	54	T
85	55	U
86	56	V
87	57	W
88	58	X
89	59	Y
90	5A	Z
91	5B	[
92	5C	\
93	5D	]
94	5E	^
95	5F	_

Dec	Hex	Ch
96	60	`
97	61	a
98	62	b
99	63	c
100	64	d
101	65	e
102	66	f
103	67	g
104	68	h
105	69	i
106	6A	j
107	6B	k
108	6C	l
109	6D	m
110	6E	n
111	6F	o
112	70	p
113	71	q
114	72	r
115	73	s
116	74	t
117	75	u
118	76	v
119	77	w
120	78	x
121	79	y
122	7A	z
123	7B	{
124	7C	|
125	7D	}
126	7E	~
127	7F	D

Appendix G — Maintenance

The HS-21 and HS-41X Handheld Readers require only a minimum of maintenance to operate.

Cleaning the HS-21 and HS-41X Handheld Readers

The following substances are approved for cleaning of the HS-21 and HS-41X.

Product	**Chemical Content**
Alcohol Wipes	Isopropyl Alcohol
CaviWipes® Disinfecting Towelettes and CaviCide®	Isopropyl Alcohol, Ethylene Glycol Monobutyl Ether
Clorox® Disinfecting Wipes	Isopropyl Alcohol, n-Alkyl Dimethyl-benzyl Ammonium Chloride, n-Alkyl Dimethyl Ethylbenzyl Ammonium Chloride, Alkyl Polyglucoside, Propylene Glycol Propyl Ether
Clorox® Bleach Solution (10% Clorox bleach, 90% tap water)	Sodium Hypochlorite, Sodium Chloride, Sodium Carbonate, Sodium Hydroxide, Sodium Polyacrylate
Formula 409® Glass and Surface Cleaner	n-Alkyl Dimethyl Benzyl Ammonium Chloride, n-Propoxypropanol
Sani-Cloth® HB, Super Sani-Cloth® Germicidal, Sani-Cloth® Plus Germicidal Disposable Wipes	Quaternary Ammonium Compounds/Chlorides
Virex® II Disinfectant Cleaner	n-Alkyl Dimethyl Benzyl Ammonium Chloride, Didecyl Dimethyl Ammonium Chloride
Gentle dish soap and water	

Appendix H — Glossary of Terms

Aberration — The failure of an optical lens to produce an exact point-to-point correspondence between the object and its resulting image. Various types are chromatic, spherical, coma, astigmatism and distortion.

Absorption — The loss of light of certain wavelengths as it passes through a material and is converted to heat or other forms of energy. (–)

Active Illumination — Lighting an area with a light source coordinated with the acquisition of an image. Strobed flash tubes and pulsed lasers are examples.

ADC — See **Analog-to-Digital Converter**.

A/D Converter — See **Analog-to-Digital Converter**.

AGC — See **Automatic Gain Control**.

Ambient Light — Light which is present in the environment of the front end of a reader and generated from outside sources. This light, unless used for actual illumination, will be treated as background noise by the reader.

Analog — A smooth, continuous voltage or current signal or function whose magnitude (value) is the information.

Analog-to-Digital Converter (A/D Converter or **ADC)** — A device that converts an analog voltage or current signal to a discrete series of digitally encoded numbers (signal) for computer processing.

Application-Specific Integrated Circuit (ASIC) — An integrated circuit that is customized for a particular kind of use, rather than general use. All vision system elements including firmware can be integrated into one ASIC.

Automatic Gain Control (AGC) — Adjustment to signal strength that seeks to maintain a constant level regardless of the distance between a reader and symbol.

Auxiliary Port — RS-232 connection to an auxiliary terminal or device for remote viewing.

Baud Rate — The number of discrete signal events per second; bits per second.

CCD — See **Charge-Coupled Device**.

Charge-Coupled Device (CCD) — A semiconductor device with an array of light-sensitive elements that converts light images into electrical signals.

Check Character — A Modulus 43 or Modulus 10 character that is added to encoded symbol data for additional data integrity.

Connector — A plug or socket on a device or cable providing in/out connectivity for various circuits and pins.

Concentrator — Intermediary device that relays data from readers to a host and commands from the host to the readers or other devices.

DAC — See **Digital-to-Analog Converter**.

Daisy Chain — Linkage of primary and secondary readers allowing data to be relayed up to the host via auxiliary port connections.

Decode — A **Good Read**. The successful interpretation and output of the information encoded in a symbol.

Default — Restores **ROM** or flash settings and initializes serial commands.

Delimited — A delimited command or field is bracketed by predefined characters.

Decode Rate — The number of good reads per second ahieved by a reader.

Darkfield Illumination — Lighting of objects, surfaces, or particles at very shallow or low angles, so that light does not directly enter a reader's optical hardware.

Depth-of-Field — The in-focus range of a reader. Measured from the distance behind an object to the distance in front of the object with all objects appearing in focus.

Diffused Lighting — Scattered soft lighting from a wide variety of angles used to eliminate shadows and specular glints from profiled, highly reflective surfaces.

Digital-to-Analog Converter (DAC) — A **VLSI** circuit used to convert digitally processed images to analog for display on a monitor.

Digital Signal Processor (DSP) — A **VLSI** chip designed for ultra-high-speed arithmetic processing.

Discrete I/O — Inputs and outputs characterized by discrete signal transitions from one voltage level to another so that digital switching can occur.

Direct Memory Access (DMA) — A capability provided by some computer bus architectures that allows data to be sent directly to memory from an attached device.

DSP — See **Digital Signal Processor**.

EPROM — See **Erasable Programmable Read-Only Memory**.

Embedded Memory — Onboard memory device such as **EPROM** or flash.

End of Read Cycle — The time or condition at which the reader stops expecting symbol information to decode.

Erasable Programmable Read-Only Memory (EPROM) — A memory chip that retains data when its power supply is turned off; "non-volatile memory".

External Edge — Allows a read cycle to be initiated by a trigger signal from an object detector when it detects the appearance of an object (rising edge). The read cycle ends with a good read, a timeout, or a new trigger.

External Level — Allows a read cycle to be initiated by a trigger signal from an object detector. The read cycle ends when the object moves out of the detector's range.

Falling Edge — A change of state (to inactive) associated with a level trigger.

Field-Programmable Gate Array (FPGA) — A semiconductor device containing programmable interconnects and logic components.

Firmware — Software hard-coded in non-volatile memory (**ROM**), and closely tied to specific pieces of hardware.

Fixed Symbol Length — Increases data integrity by ensuring that only a symbol length will be accepted.

Focal Distance — In optics, the distance from the lens to the focal plane.

Focal Plane — Usually found at the image sensor, it is a plane perpendicular to the lens axis at the point of focus (–).

Focus — Any given point in an image at which light converges; the focal point.

FPGA — See **Field-Programmable Gate Array**.

Full Duplex — A communications system in which signals can travel simultaneously between devices.

Good Read — A decode. The successful scanning and decoding of the information encoded in a bar code symbol.

Half Duplex — A communications system in which signals can travel between devices in both directions, but not simultaneously.

Host — A computer, **PLC**, or other device that is used to execute commands and process data and discrete signals.

Image Sensor — A device that converts a visual image to an electrical signal; a **CCD**, for example.

Initialize — Implement serial configuration commands into the reader's active memory.

Input — A channel or communications line. Decoded data or a discrete signal that is received by a device.

Ladder Orientation — A linear symbol orientation in which the bars are parallel to the symbol's direction of travel.

Light-Emitting Diode (LED) — A semiconductor device that emits light when conducting current.

Lens — A transparent piece of material with curved surfaces which either converge or diverge light rays.

Multidrop — A communications protocol for networking two or more readers or other devices with a concentrator (or controller) and characterized by the use of individual device addresses and the RS-485 standard.

Normally Closed — A discrete output state that is only active when open.

Normally Open — A discrete output state that is only active when closed.

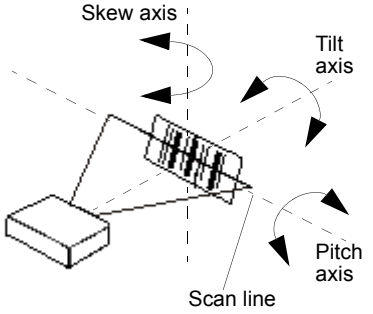
Object Plane — An imaginary plane in the field of view, focused by a reader's optical system at the corresponding image plane on the sensor.

Output — A channel or communications line. Data or discrete signals that are transmitted or displayed by a device.

Parity — An error detection routine in which one data bit in each character is set to **1** or **0** so that the total number of **1** bits in the data field is even or odd.

Picket Fence Orientation — A linear symbol orientation in which the bars are perpendicular to the symbol's direction of travel.

Pitch — Rotation of a linear or 2D symbol around an axis parallel to the symbol length on the substrate. See the illustration below.



PLC — See **Programmable Logic Controller**.

Port — Logical circuit for data entry and exit. (One or more ports may be included within a single connector.)

Programmable Logic Controller (PLC) — An electronic device used in industrial automation environments such as factory assembly lines and automotive manufacturing facilities.

Protocol — The rules for communication between devices, providing a means to control the orderly flow of information between linked devices.

Random Access Memory (RAM) — A data storage system used in computers, composed of integrated circuits that allow access to stored data in any sequence without movement of physical parts.

Read Cycle — A programmed period of time or condition during which a reader will accept symbol input.

Read-Only Memory (ROM) — A data storage medium used in computers and other electronics, primarily used to distribute firmware.

Skew — Rotation of a linear or 2D symbol around an axis parallel to the symbol height on the substrate. See the illustration under the definition of **Pitch**.

Substrate — The surface upon which a symbol is printed, stamped, or etched.

Symbol Transitions — The transition of bars and spaces on a symbol, used to detect the presence of a symbol on an object.

Symbology — A symbol type, such as Code 39 or Code 128, with special rules to define the widths and positions of bars and spaces to represent specific numeric or alphanumeric information.

Tilt — Rotation of a linear or 2D symbol around an axis perpendicular to the substrate. See the illustration under the definition of **Pitch**.

Trigger — A signal, transition, or character string that initiates a read cycle.

Very Large-Scale Integration (VLSI) — The creation of integrated circuits by combining thousands of transistor-based circuits on a single chip.

VLSI — See **Very Large-Scale Integration**.