MICROSCAN®

HawkEye™ 1500 Series User Manual

v2.4.1, Nov 2008

EM-40391-1V241

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, email: helpdesk@microscan.com.

Microscan Systems, Inc.
1201 S.W. 7th Street
Renton, WA 98057
U.S.A.
Tel: 425 226 5700
Fax: 425 226 8250
helpdesk@microscan.com

Microscan Europe
Tel: 31 172 423360
Fax: 31 172 423366

Microscan Asia Pacific
R.O. Tel: 65 6846 1214
Fax: 65 6846 4641

Microscan Limited Warranty Statement and Exclusions

What Is Covered?
Microscan Systems Inc. warrants to the original purchaser that products manufactured by it will be free from defects in material and workmanship under normal use and service for a period of one year from the date of shipment. This warranty is specifically limited to, at Microscan’s sole option, repair or replacement with a functionally equivalent unit and return without charge for service or return freight.

What Is Excluded?
This limited warranty specifically excludes the following: (1) Any products or parts that have been subject to misuse, neglect, accident, unauthorized repair, improper installation, or abnormal conditions or operations; (2) Any products or parts that have been transferred by the original purchaser; (3) Customer mis-adjustment of settings contrary to the procedure described in the Microscan Systems Inc. owners manual; (4) Upgrading software versions at customer request unless required to meet specifications in effect at the time of purchase; (5) Units returned and found to have no failure will be excluded; (6) Claims for damage in transit are to be directed to the freight carrier upon receipt. Any use of the product is at purchaser’s own risk. This limited warranty is the only warranty provided by Microscan Systems Inc.

regarding the product. Except for the limited warranty above, the product is provided “as is.” To the maximum extent permitted by law, this express warranty excludes all other warranties, express or implied, including but not limited to, implied warranties of merchantability and. Technical support questions may be directed to: helpdesk@microscan.com Register your product with Microscan: www.microscan.com/register fitness for a particular purpose. Microscan Systems Inc. does not warrant that the functions contained in the product will meet any requirements or needs purchaser may have, or that the product will operate error free, or in an uninterrupted fashion, or that any defects or errors in the product will be corrected, or that the product is compatible with any particular machinery.

Limitation of Liability

In no event shall Microscan Systems Inc. be liable to you or any third party for any special, incidental, or consequential damages (including, without limitation, indirect, special, punitive, or exemplary damages for loss of business, loss of profits, business interruption, or loss of business information), whether in contract, tort, or otherwise, even if Microscan Systems Inc. has been advised of the possibility of such damages. Microscan Systems Inc.’s aggregate liability with respect to its obligations under this warranty or otherwise with respect to the product and documentation or otherwise shall not exceed the amount paid by you for the product and documentation. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages or limitations on an implied warranty, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which may vary from state to state.

Tel: 425.226.5700 | Fax: 425.226.8250 | helpdesk@microscan.com

Contents

PREFACE

Welcome!

Purpose of This Manual

This manual is designed to help you to understand how your HawkEye™ 1500 works, and how to use it quickly and efficiently.

Manual Conventions

The following typographical conventions are used throughout this manual.

- Items emphasizing important information is **bolded**.
- Menu selections, menu items and entries in screen images are indicated as: Run (triggered), Modify..., etc.

1

CHAPTER 1

Configurations

This chapter contains information about selecting the proper HawkEye™ 1500 Series Smart Camera-Based Reader for your application. It also contains information about selecting lenses and lighting.

Note: Throughout this manual, "HawkEye™ 1500 Series Camera" is used as a generic term for the HawkEye™ 1515, the HawkEye™ 1525, and the HawkEye™ 1510. When information is specific to a camera, that camera name is used.

Selecting the Correct HawkEye™ to Read Your Data Matrix

There are three main HawkEye™ models:

FIGURE 1–1. HawkEye™ 1515

FIGURE 1–2. HawkEye™ 1525

FIGURE 1–3. HawkEye™ 1510

Both the HawkEye™ 1515 and HawkEye™ 1525 have fixed optics and illumination built right into the units. The HawkEye™ 1510 allows you to select from a variety of off-the-shelf optics and illumination components. For complete information about the HawkEye™ 1510, see Appendix A, “HawkEye™ 1510,”.

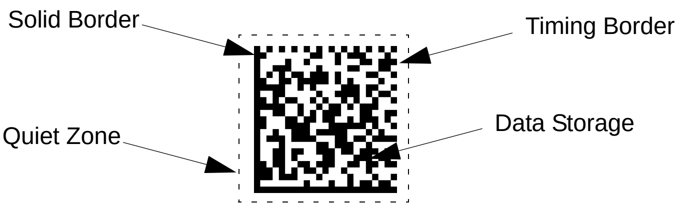
The HawkEye™ 1515 and HawkEye™ 1525 use the same lenses, which give the units a range of magnifications. The HawkEye™ 1515 and HawkEye™ 1525 use different illumination components.

Use the following paragraphs to help you pick the correct model and magnification for your Data Matrix or Barcode reading application.

Data Matrix Construction

A Data Matrix is made up of four major components, as shown in Figure 1–4.

FIGURE 1–4. Four Major Components of a Data Matrix



Selection Criteria

There are three main considerations for choosing the proper HawkEye™ type:

- Resolution
- Field of View
- Working Distance

The most important consideration is Resolution. All three of these features are governed by lens selection.

Resolution

A Data Matrix is comprised of a series of dark and light cells. To obtain optimal READ performance, each cell should be imaged by at least 4 to 5 camera pixels. To obtain optimal VERIFICATION performance, each cell should be imaged by at least 10 camera pixels.

A Barcode is comprised of a series of light and dark lines. To obtain optimal READ performance, each line should be seen as at least 2 pixels wide.

Field of View

A second and interrelated consideration is Field of View. The field of view should be small enough for the Data Matrix to have at least 4 pixels per cell resolution. At the same time, the field of view should be large enough to contain the Data Matrix or Barcode, as well as to leave enough space around the symbol to compensate for symbol positioning error and the required Quiet Zone.

Note: The Quiet Zone must be at least 1 cell in size. A Quiet Zone of 2 cells or more is allowed, and makes reading easier.

Working Distance

The Working Distance is the distance from the front of the light to the symbol. Typically, it is dictated by whatever clearance is required for part handling between the HawkEye™ 1500 camera and the part. These standoff distances vary from a low of 3 inches (76.2mm) to a high of 5 inches (127mm) with the HawkEye™ 1515 and the HawkEye™ 1525. If you have different standoff distance requirements, the HawkEye™ 1510 camera should be used.

Selecting A Lens

Standard Lens Selection Chart

Table 1–1 shows the different magnifications for the HawkEye™ 1515 and HawkEye™ 1525 with built in lenses. Each type has a specific Working Distance, Field of View, and Minimum Cell Size. To determine the correct type:

1. Determine the **Cell Size** for your Data Matrix. Do this by measuring the size of your Data Matrix in the horizontal direction, and by counting the number of cells in the horizontal direction.

   Cell Size = Matrix Size (H) / Number of Cells (H)

2. Determine the overall field of view required to contain the Data Matrix, Quiet Zone, and to allow for the positioning of the symbol.

3. Look in the chart to see which types have a Minimum Cell Sizes less than your cell size.

   Example: Assume a Data Matrix has 23 cells in the horizontal direction, is 0.75” (19.05mm) wide, and with a quiet zone, is over 1.125” (28.58mm) wide.

$$\text{Cell Size} = \frac{0.75\text{” } (19.05\text{mm})}{23} = 0.033\text{” } (0.84\text{mm})$$

   The only HawkEye™ Type that has a Minimum Cell Size less than 0.033” (0.84mm) and a field of view greater than 1.125” (28.58mm) is the Medium Density (MD).

TABLE 1–1. Lens Selection Chart

Type	Working Distance	Field of View at Focus	Minimum Cell Size (4 Pixels/Cell)
Medium Density (MD)	5.0” (127mm) ± 1.0” (25.4mm)	1.55”H (39.37mm) x 1.19”V (30.23mm)	DM: 0.010” (0.25mm) BC: 0.005” (0.127mm)
High Density (HD)	3.0” (76.2mm) ± 0.5” (12.7mm)	1.00”H (25.4mm) x 0.75”V (19.05mm)	DM: 0.006” (0.152mm) BC: 0.003” (0.076mm)
Long High Density (LHD)	5.0” (127mm) ± 0.5” (12.7mm)	1.00”H (25.4mm) x 0.75”V (19.05mm)	DM: 0.006” (0.152mm) BC: 0.003” (0.076mm)
Super High Density (SHD)	3.5” (88.9mm) ± 0.5” (12.7mm)	0.55”H (13.97mm) x 0.42”V (10.67mm)	DM: 0.003” (0.076mm) BC: 0.0015” (0.038mm)
Ultra High Density (UHD)	2.25” (57.15mm) ± 0.125” (3.175mm)	0.25”H (6.35mm) x 0.19”V (4.83mm)	DM: 0.0013” (0.033mm) BC: 0.0005” (0.0127mm)

Custom Lens Selection

In addition to the HawkEye™ 1515 and HawkEye™ 1525 cameras with built in lenses, the HawkEye™ 1510 is sold with a C or CS mount that allows the use of conventional lenses. The use of conventional lenses and extension tubes allow a much larger range of ‘fields of view’ and ‘working distances’ to be accomplished. Standard lens selection criteria should be employed to select the correct optics for your application.

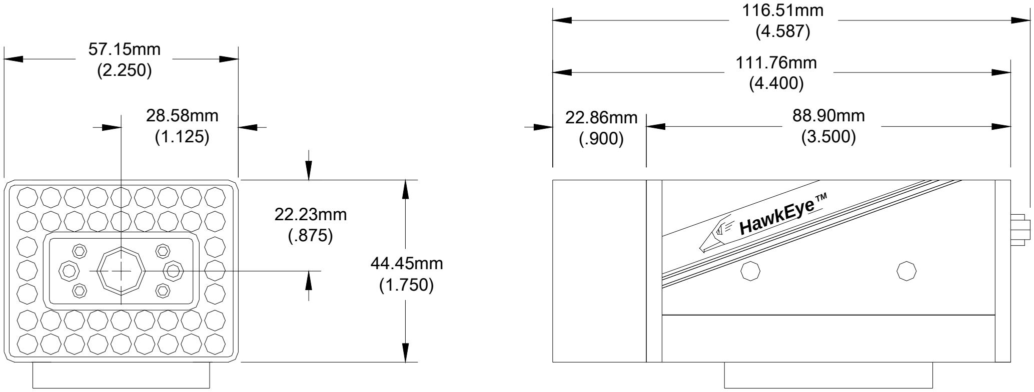
Selecting Lighting

Standard Light Selection

The HawkEye™ 1515 and HawkEye™ 1525 come with built in lights and built in lenses.

HawkEye™ 1515

FIGURE 1–5. HawkEye™ 1515

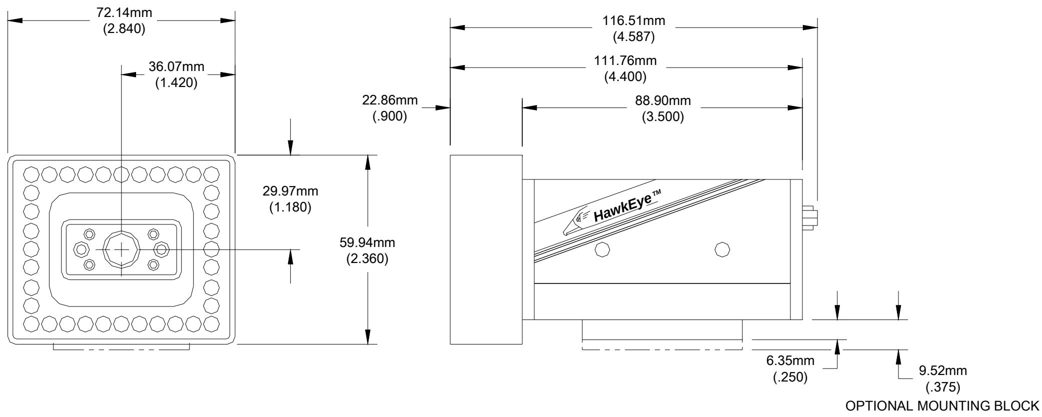


The HawkEye™ 1515 has a small light ring built on the front of the unit. The ring is 1.75"H (44.45mm) x 1.25"H (31.75mm) at the center of the emission zone. The ring is aimed in 15° to converge 3" (76.2mm) from the front on the unit. When the unit is used perpendicular to the part, the light acts as a bright field, high angle ring light. A second useful configuration is to angle the unit 20° off of vertical. In this configuration, the light acts as a directional high angle spotlight. The HawkEye™ 1515 configuration is the universal reader for the broadest range of Data Matrix and Barcode reading applications.

Note: The HawkEye™ 1515-XL does not have lasers.

HawkEye™ 1525

FIGURE 1–6. HawkEye™ 1525



The HawkEye™ 1525 has a medium size light ring built on the front of the unit. The ring is 2.375”H (60.33mm) x 1.875”H (47.63mm) at the center of the emission zone. The ring is aimed in 25° to converge 2.25” (57.15mm) from the front on the unit. When the unit is used perpendicular to the part, the light acts as a dark field, medium to low angle ring light. Typically, the HawkEye™ 1525 is used for highly reflective parts.

Note: The HawkEye™ 1525-XL does not have lasers.

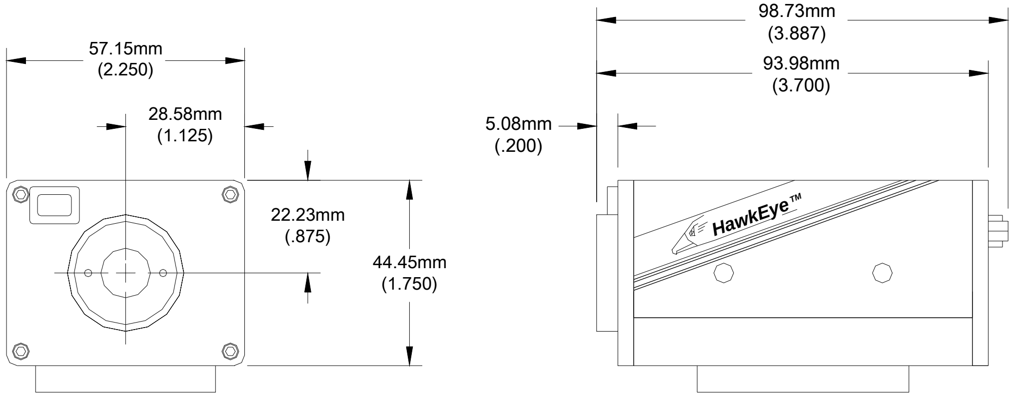
HawkEye™ XL details

The HawkEye™ XL is designed for applications requiring ease of integration utilizing fixed optics and integrated lighting but does not need features/benefits provided by laser targeting.

Custom Light Selection

HawkEye™ 1510

FIGURE 1–7. HawkEye™ 1510



The HawkEye™ 1510 is designed for applications requiring flexibility in the selection of lighting and optics. The unit is designed with a mounting block that allows the attachment of C and CS mount lenses. Additionally, it has brackets specially designed to mount a variety of the standard NER Lights. For complete information about the HawkEye™ 1510, see Appendix A, “HawkEye™ 1510,”.

Note: The HawkEye™ 1510 does not have a laser.

CHAPTER 2

Connecting to the HawkEye™ 1500

The chapter contains information to help you connect to the HawkEye™ 1500 camera. Specific information describes connectors, adapters, cables, pinouts, and signals.

Note: There are no user serviceable parts inside.

Connectivity

TCP/IP Port

When communicating over Ethernet, the camera uses the following predefined ports. The camera establishes connections as a Server and, therefore, listens for Host clients to initiate the connection on a particular port. Any number of clients can connect to the camera, each one with their private peer-to-peer connection.

TABLE 2–1. HawkEye™ 1500 TCP/IP Connectivity

Port Name	Protocol	Number	Note
Camera Query Port	UDP	49093	Discovers HawkEye™ 1500 cameras on the current subnet.
Camera Announce Port	UDP	49094	Broadcasts the HawkEye™ 1500 camera identity on the current subnet used by Network View in ReadRunner, provides general counters, camera name, IP, IP in control, camera status, and camera software version and capability.
COMMAND	TCP	49095	Used for all HawkEye™ 1500 ASCII Commands including accessing Buffered cycle read reports (PARTQ commands).
STANDARD	SERIAL	N/A	Programs the format of the data sent via the serial port.
STANDARD	TCP	49096	Used by ReadRunner and programming COM object library for reports with images. By default lossy up to 2 per second. Connection can be programmed to lossless, i.e., inline with the read cycle.
STANDARD	TCP	49097	Used by ReadRunner and programming COM object library for reports only. By default lossy but at maximum network rate. Connection can be programmed to lossless, i.e. inline with the read cycle.
TCP1 User Connection	TCP	49098	User format configurable TCP based connection. Uses ASCII and binary keywords (IDxx), (CDxx), etc…. Note that the connection is always lossless, i.e., inline with the read cycle.
TCP2 User Connection	TCP	49099	User format configurable TCP based connection. Uses ASCII and binary keywords (IDxx), (CDxx), etc… Note that the connection is always lossless, i.e. inline with the read cycle.
TCP3 User Connection	TCP	49100	User format configurable TCP based connection. Uses ASCII and binary keywords (IDxx), (CDxx), etc… Note that the connection is always lossless, i.e., inline with the read cycle.
TCP4 User Connection	TCP	49101	User format configurable TCP based connection. Uses ASCII and binary keywords (IDxx), (CDxx), etc…

Note: Ports COMMAND, TCP1…4 use a protocol that is either ASCII or defined by you at connection time. Therefore, these ports are fully supported on a non-Windows based Host, provided the Host supports TCP/IP and a socket level API. Ports 49096 and 49097 use a binary format that is parsed into easy to use COM events and objects on the Host and is, therefore, only applicable to Windows-based Hosts.

Note: You can enable/disable the UDP protocol through the command line. For more information, see the UDP_BROADCAST command in the HawkEye 1500 Series Reference & Programmers Manual.

Serial Port

When communicating over a serial line, you need to be aware of the limitations of this communication medium and how ReadRunner and the camera handle it.

There can be only one client connected to the camera. In addition, there can be only one channel to exchange data, i.e., commands and binary data.

The ReadRunner UI enables a special ACK scheme that may need to be disabled once the camera has been programmed and is connected serially to a PLC or the controlling host. You can accomplish this by sending special control characters (see Table 2–2):

TABLE 2–2. Serial Port Connectivity

Control Char	Function
Ctrl+O	Turn off report output.
Ctrl+P	Toggle the state of the prompt and echo.
Ctrl+Q	Turn off command output.
Ctrl+R	Release control of the unit and put it back online.
Ctrl+S	Take control of the unit and bring it offline (forcibly).
Ctrl+T	Dump the heartbeat string to the terminal.
Ctrl+U	Toggle the display of report strings on the terminal.
Ctrl+W	Turn on command output.

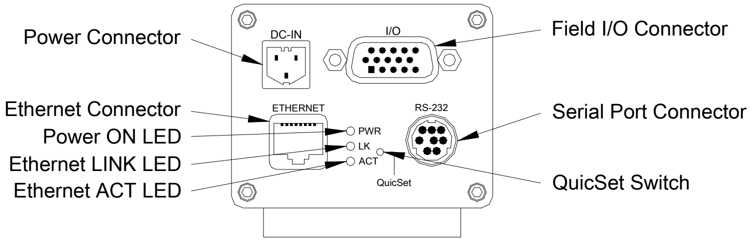
If the Serial port is programmed for 7 bits per character, certain features of the ReadRunner UI are not supported; in particular, “Hook Reports” and “Hook Images” are not available, as they transfer image and counters (binary data) and cannot be represented using a 7 bit per character ASCII format.

Note: Regardless of whether the 7 or 8 bits per character are programmed, “Hook Reports” and “Hook Images” must be disconnected before quitting the ReadRunner application before connecting the camera over serial to the controlling device. Output formatting settings for the serial port are available in ReadRunner under the STANDARD Tab in the Output Settings Form.

Rear Panel

Figure 2–1 details the layout of the rear panel.

FIGURE 2–1. Rear Panel Layout



- Power Connector – 24VDC in
- Field I/O Connector – DB15S – 1 Opto in, 3 Opto Out, 4 GPIO (strobe out optional on GPIO 1)
- Serial Port Connector – 8 pin mini-DIN
- Ethernet Connector – RJ45
- Ethernet LINK LED – Green
- Ethernet ACT LED – Yellow
- Power ON LED – Green
- QuicSet Switch – Recessed

Power Connector

Figure 2–2 shows the pinout of the Power connector.

FIGURE 2–2. Power Connector Pinout

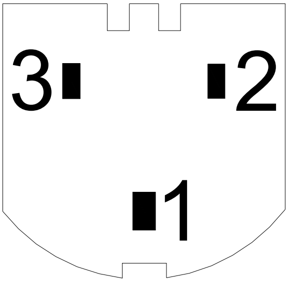


Table 2–3 lists the suppliers for the power connector mating connector.

TABLE 2–3. Power Connector Mating Connector Suppliers

Supplier	Part Number	Description
Microscan	HECONNPS	Housing and socket crimp Kit
HIROSE	RP34-8SP-3SC RP34-SC-112	Housing Socket crimp

Table 2–4 describes the power connector signals.

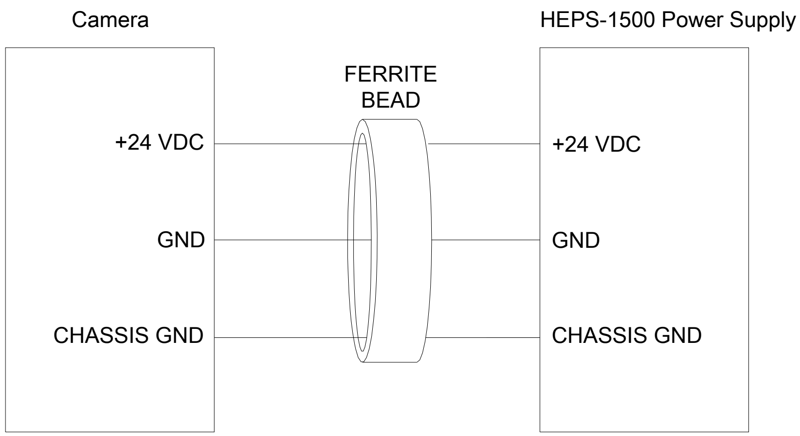
TABLE 2–4. Power Connector Signals

Pin	Signal Name
1	Chassis ground
2	+24 VDC
3	24V return

Power Supply Wiring

Figure 2–3 and Figure 2–4 show the wiring for the power supply.

FIGURE 2–3. Power Supplied Via the HEPS-1500 Wall Mount Power Supply

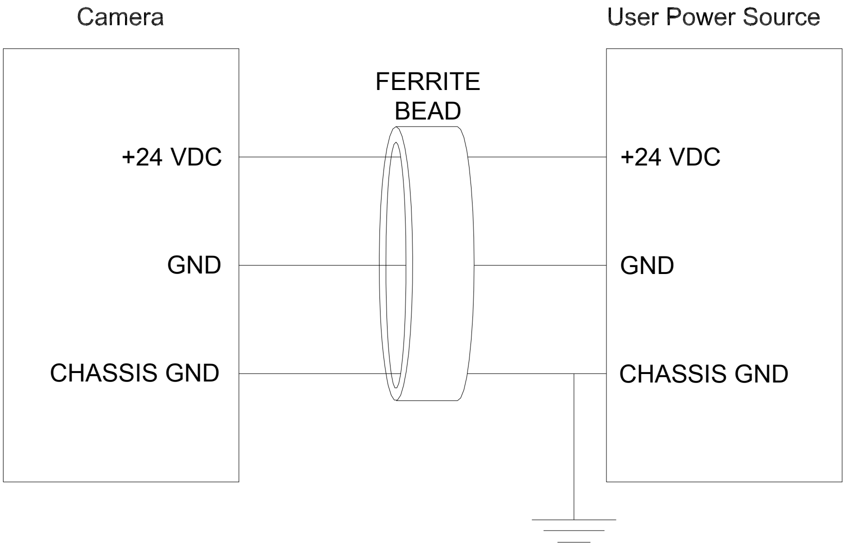


Note: Ferrite suppression bead is integrated on power supply cable.

TABLE 2–5. Power Supply Suppliers

Supplier	Part Number	Description
Microscan	HEPS-1500	Wall mount power supply

FIGURE 2–4. Power Supplied Via a User Supplied Source



Note: Ferrite suppression bead is required to meet CE radiated emission requirements.

TABLE 2–6. Power Cable and Ferrite Bead Suppliers

Suppliers	Part Number	Description
Microscan	HEPC-006	Single-ended power cable - 6 Ft. w/molded ferrite bead
Microscan	914-0012-1	EMI Round Cable Suppression Core
Fair-Rite	0443167251	

Field I/O Connector

Figure 2–5 shows the pinout for the HDB-15S connector.

FIGURE 2–5. Field I/O (HDB-15S) Connector

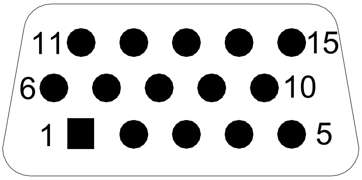


Table 2–7 lists the suppliers for the field I/O (HDB-15S) mating connector.

TABLE 2–7. Field I/O (HDB-15S) Mating Connector Suppliers

Supplier	Part Number	Description
Microscan	HECONNIO	Housing and Hood Kit
AMP	748676-1	Housing
HEIL	180-015-102-001	Hood

Table 2–8 describes the field I/O signals.

TABLE 2–8. Field I/O Signals

Pin	Signal	Direction	H/W Description	S/W Direction	S/W Name	Notes
6	sensor power	out	Fused +24V, 100 ma max.			1
10	sensor power return	common	DC ground			1
1	opto in a	in	Trigger input A	In	OPTO IN 1	2,6
11	opto in b	in	Trigger input B	In	OPTO IN 1	2,6
2	opto 1 out a	out	Pass output a	out	OPTO OUT PASS	3,7
12	opto 1 out b	out	Pass output b	out	OPTO OUT PASS	3,7
3	opto 2 out a	out	Fail output a	out	OPTO OUT FAIL	3,8
13	opto 2 out b	out	Fail output b	out	OPTO OUT FAIL	3,8
4	opto 3 out a	out	Data valid a	out	OPTO OUT DV	3,9

TABLE 2–8. Field I/O Signals (Continued)

14	opto 3 out b	out	Data valid b	out	OPTO OUT DV	3,9
5	TTL IO 1 / Ext. strobe	In/out	General purpose I/O 1/ external strobe 1	out	LIGHTING EXTERNAL	4,5, 10
7	TTL IO 2	In/out	General purpose I/O 2	out	NOT USED	5
8	TTL IO 3	In/out	General purpose I/O 3	out	RTE OUT	5,11
9	TTL IO 4	In/out	General purpose I/O 4	in	GPIO IN 4 DURATION	5,12
15	I/O return	common	DC ground			
shell	chassis ground					

Notes:

1. Non-isolated utility power for sensor and/or opto current loops.
2. Bipolar isolated current input 5-24V, 1-5 ma., 250VAC isolation.
3. Bipolar isolated output switch, Ron = 35 ohm max., Ion < 50 ma, Voff < 50VDC, 250VAC isolation.
4. 4V, 20 ma positive pulse in strobe mode.
5. OUT: 20 ma sink, 2.2K to +5V pullup. IN: TTL, 1V hysteresis.
6. Opto-isolated camera sensor trigger.

Note: Notes 7 through 12 refer to default I/O assignment. This may change according to user assignment.

7. Read Pass signal for I/O Pass/Fail modes. Read Pass/Fail for I/O Data Valid modes.
8. Read Fail signal; either decode, locate or match fail for I/O Pass/Fail modes. Not used for I/O Data Valid modes.
9. Data Valid signal used in all pipelined I/O modes. External equipment can read result of decode while this signal is asserted.
10. Controls the external strobe light. Set in Application mode -> Lighting.

11. Runtime Error signal: programmed to assert when Trigger, Read, Timeout Overruns and/or Network drops occur.

12. Used by StopAndScan to signal retry duration.

Figure 2–6 shows the Opto In equivalent circuit.

FIGURE 2–6. Opto In

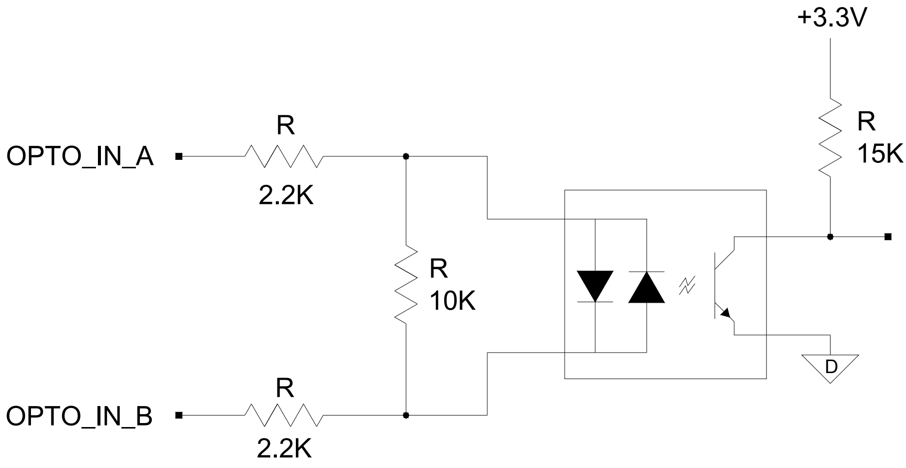


Figure 2–7 shows the Opto Out equivalent circuit.

FIGURE 2–7. Opto Out

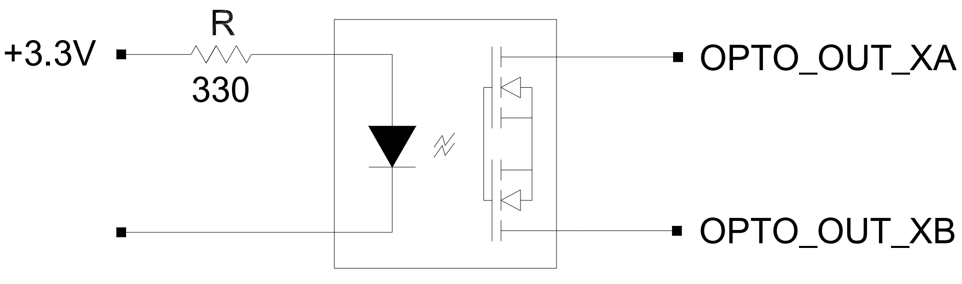
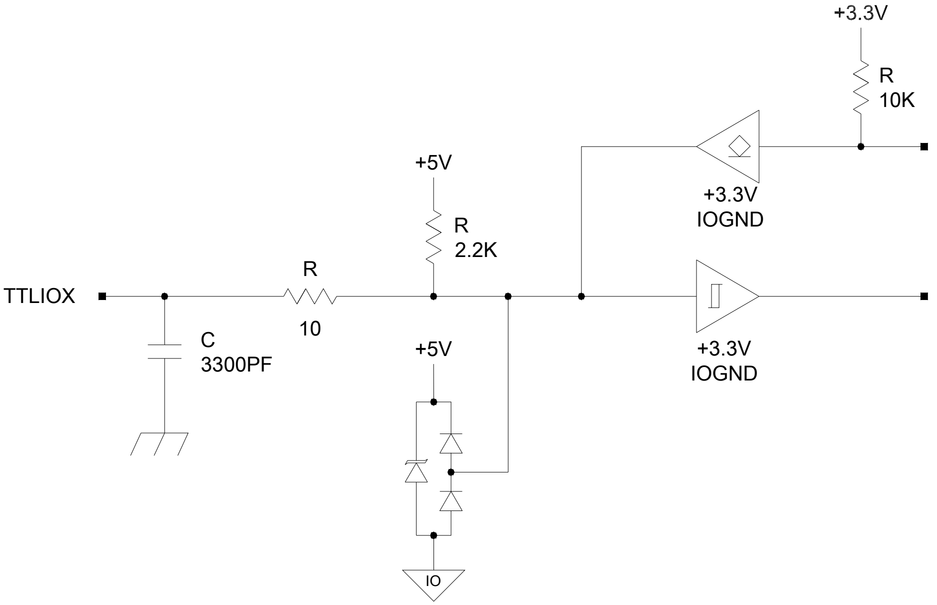


Figure 2–8 shows the TTL I/O equivalent circuit.

FIGURE 2–8. TTL I/O



Grounding Tab (Optional)

Normally, the HawkEye™ 1500 chassis is connected to ground via the serial cable or the I/O interface cable shield (assuming that the cable shield is connected to ground). If not, a Grounding tab kit is available for the HawkEye™ 1500 reader. The kit provides two grounding tabs, a Solder tab, and a 0.187 Faston tab that gets mounted to one of the HDB-15 hex standoffs (see Figure 2–9).

FIGURE 2–9. Grounding Tab Mounting Location

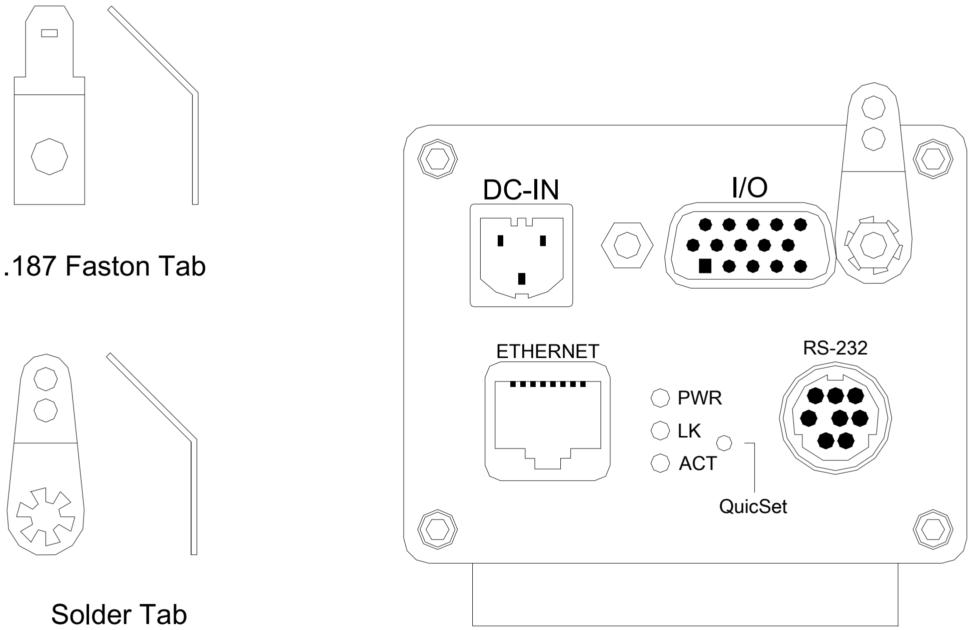


Table 2–9 lists the Grounding tab part numbers and descriptions for the HawkEye™ 1500.

TABLE 2–9. Grounding Tab Part Numbers

Part Number	Description
A1-40201-1	HawkEye™ Grounding Tab Kit – includes:
958-0039-1	0.187 Faston tab with 45 bend
958-0040-4	Solder tab with 45 bend

Use the following steps to install the grounding tab:

1. Remove one of the HDB-15 standoffs using a 3/16” Hex Driver.
2. Install the grounding tab (either Solder or Faston type) and standoff as shown in Figure 2–9.
3. Tighten the standoff.

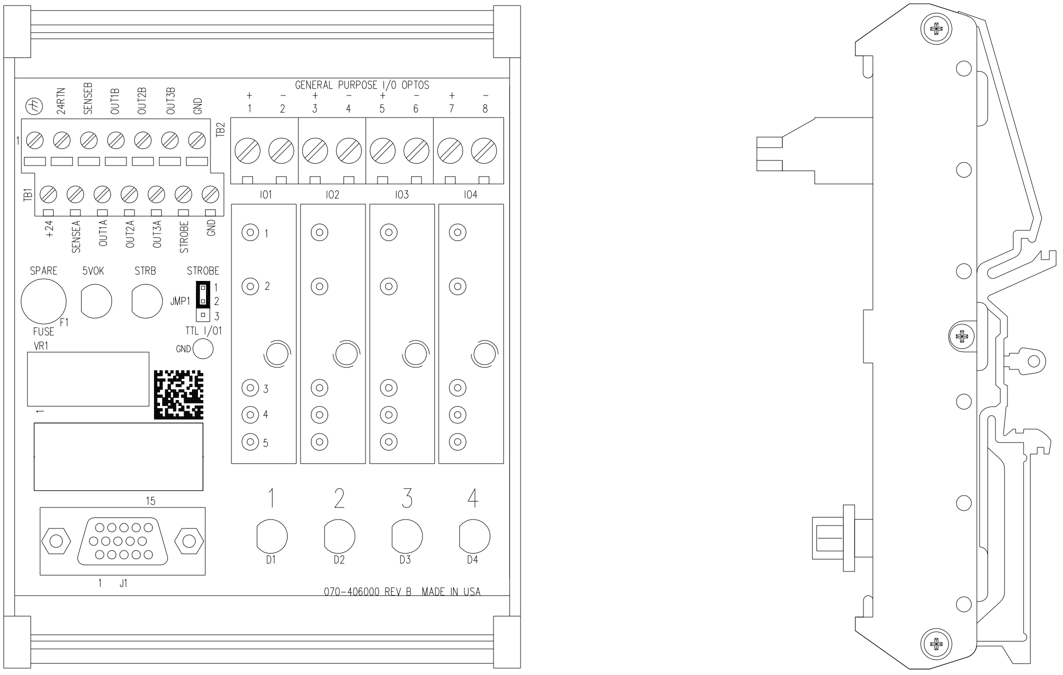
I/O Expansion Module (Optional)

The I/O Expansion Module provides a convenient method of connecting to the HawkEye™ camera I/O signals. It combines one dedicated optically isolated sensor input, three dedicated optically isolated outputs, and four digital I/O ports all on one circuit board, as shown in Figure 2–10, “I/O Expansion Module,” on page 2-14.

- All of the I/O connections from the camera to the I/O Expansion Module are connected via a 15-pin cable.
- One opto-isolated camera sensor input trigger is provided. The sensor input requires a bipolar isolated current input 5-24VDC at 1-5mA.
- Three opto-isolated outputs are provided. The outputs are bipolar isolated switches, Ron = 35 ohms max, Ion < 50ma., Voff <= 50VDC.
- Four digital I/O ports are provided (General Purpose I/O 1 - 4). The digital I/O ports are individually programmable as inputs or outputs from the camera using Opto 22™ G4-type isolator modules.
- General Purpose I/O 1 is also used to optionally control an external strobe light. Jumper, JMP1, is provided to select the strobe output or the isolator module.
- The board is provided with a Din-Rail mount.

Figure 2–10 shows the I/O expansion module.

FIGURE 2–10. I/O Expansion Module



Using the I/O Expansion Module

- Sensor Inputs — There is one sensor input. The sensor input is brought out directly from the 15-pin HD-Sub header (J1) to barrier strip (TB1) positions 4 and 5. The camera sensor trigger input is a bipolar isolated current input, and is located within the camera.

  A chassis ground connection is provided on terminal block “TB1” position 1 and is used for connecting the sensor cable shield to chassis ground.

- Opto Outputs — There are three opto outputs. The opto outputs are brought out directly from the 15-pin HD-Sub header (J1) to barrier strip (TB1) positions 6 through 11. The opto-isolated outputs represent the camera’s output signals. These signals are bipolar isolated output switches, and are located within the camera.

- General Purpose I/O — Four positions are available for industry-standard Opto 22™ G4 type input or output isolator modules. The field wiring to these modules is terminated on the barrier strip TB2. General Purpose I/O ports (isolator modules) 1 through 4 may be configured as either inputs or outputs. The TTL I/O signals from the camera enter via the 15-pin HD-Sub header (J1) on positions 5, 7, 8, & 9 corresponding to General Purpose I/O points 1 through 4. The TTL I/O signals are filtered with EMI “T” filters. There are four LED's that indicate whenever the TTL I/O signals are ON (low). The LED's operate whether or not an isolator module is plugged in.

Jumper (JMP1) must be set to positions 2&3 to use General Purpose I/O 1.

Notes: Setting JMP1 to positions 2&3 will disable the “Strobe” output.

In this position, the camera must not be set to External Strobe mode; refer to the camera’s user manual to set the camera illumination mode to something other than External Strobe mode.

+24 volts DC power (fused @ 100mA) is brought out directly from the 15-pin HD-Sub header (J1) to barrier strip (TB1) position 1. Internal +5 volts DC is generated from the 24 volt DC source and provides logic power to the 4 General Purpose I/O Optos IO1 through IO4.

The Smart Camera I/O board accepts only 5-volt-logic solid-state relays (Opto 22™ G4-type isolator modules).

Note: There is no connector for connecting external isolator module power (5 volts).

- Strobe — The external strobe output is a TTL signal from the camera that connects to the barrier strip (TB1) position 12 (STROBE) via the 15-pin HD-Sub header (J1) and jumper (JMP1). The strobe output signal has an in-line Lo-Pass filter with a 2Mhz cutoff. There is a Strobe LED (STRB) that indicates whenever the strobe signal is ON (high).

  Jumper (JMP1) must be set to positions 1&2 to use the strobe output.

Note: Setting JMP1 to positions 1&2 will disable General Purpose I/O 1.

To use External Strobe, the camera must be set to External Strobe illumination mode; refer to either of the following:

Chapter 4 of this manual.

The ILLUMINATION command in the HawkEye™ 1500 Series Reference & Programmers Manual.

- Spare Fuse & 5VOK LED — There is a spare fuse in the spare fuse socket which doubles as a fuse/+5 volt OK test circuit. When the green LED (5VOK) is ON it indicates that both +5 volts and the fuse are OK.

Note: There is no connector for connecting external 5 volts.

Table 2–10 lists the part numbers for the Smart Camera I/O board and interface cables.

TABLE 2–10. I/O Expansion Module Suppliers

Part Number	Description
003-406000	I/O Expansion Module - includes:
966-0183-1	Cable HD-Sub 15 connector M/M 10 Feet
Optional Cables	
966-0183-2	Cable HD-Sub 15 connector M/M 15 Feet
966-0183-3	Cable HD-Sub 15 connector M/M 25 Feet

I/O Expansion Module Connectors

I/O Interface Connector – J1

Figure 2–11 shows the pinout for the HDB-15S connector.

FIGURE 2–11. Field I/O (HDB-15S) Connector

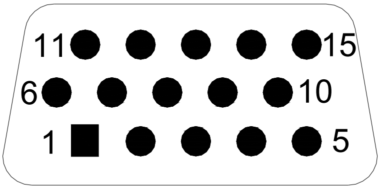


TABLE 2–11. Pinout I/O Interface Connector — J1

Pin	Description
1	Sensor Input A
2	Opto Output 1 A
3	Opto Output 2 A
4	Opto Output 3 A
5	General Purpose I/O 1 or Strobe
6	+24 volts
7	General Purpose I/O 2
8	General Purpose I/O 3
9	General Purpose I/O 4
10	24 volt return (Ground)
11	Sensor Input B
12	Opto Output 1 B
13	Opto Output 2 B
14	Opto Output 3 B
15	I/O Return (Ground)
Shell	Chassis Ground

Opto In, Opto Out, & Strobe Output Terminal Block – TB1

Figure 2–12 shows the I/O expansion module TB1.

FIGURE 2–12. **I/O Expansion Module — TB1**

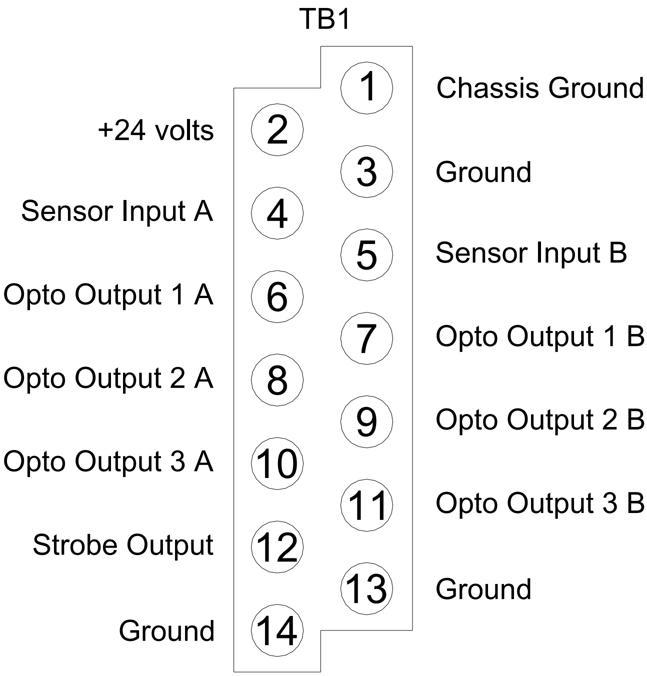


TABLE 2–12. **Pinout Opto In, Opto Out & Strobe Terminal Block — TB1**

Pin	Description
1	Chassis Ground
2	+24 volts
3	24 volt return (Ground)
4	Sensor Input A (Trigger)
5	Sensor Input B (Trigger)
6	Opto Output 1 A
7	Opto Output 1 B
8	Opto Output 2 A
9	Opto Output 2 B
10	Opto Output 3 A

TABLE 2–12. Pinout Opto In, Opto Out & Strobe Terminal Block — TB1 (Continued)

11	Opto Output 3 B
12	Strobe
13	Strobe Return (Ground)
14	Ground

General Purpose I/O Terminal Block – TB2

Figure 2–13 shows the I/O expansion module TB2.

FIGURE 2–13. I/O Expansion Module — TB2

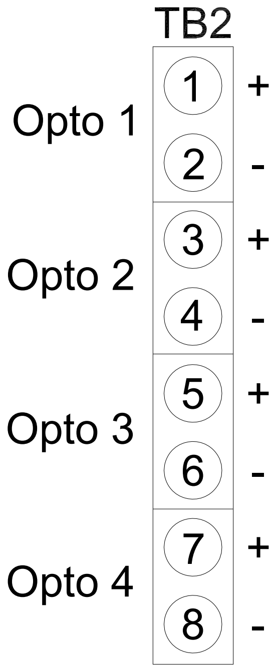


TABLE 2–13. Pinout General Purpose I/O Terminal Block — TB2

Pin	Description
1	G.P. Opto 1 Positive (+) Terminal
2	G.P. Opto 1 Negative (-) Terminal
3	G.P. Opto 2 Positive (+) Terminal
4	G.P. Opto 2 Negative (-) Terminal
5	G.P. Opto 3 Positive (+) Terminal
6	G.P. Opto 3 Negative (-) Terminal
7	G.P. Opto 4 Positive (+) Terminal
8	G.P. Opto 4 Negative (-) Terminal

Field I/O Wiring Examples

Input Opto Wiring

Sample wiring diagrams for trigger input A and B (opto input) are located on terminal block TB1 positions 4 and 5 (see Figure 2–14 and Figure 2–15).

FIGURE 2–14. Input Opto Wiring (Isolated NPN and PNP Sources)

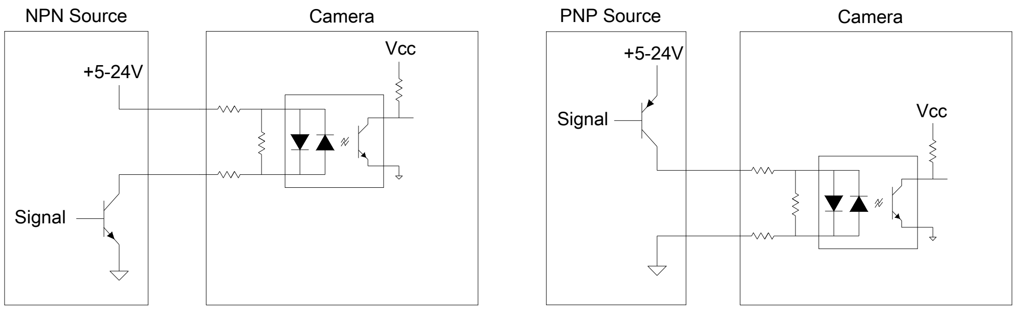
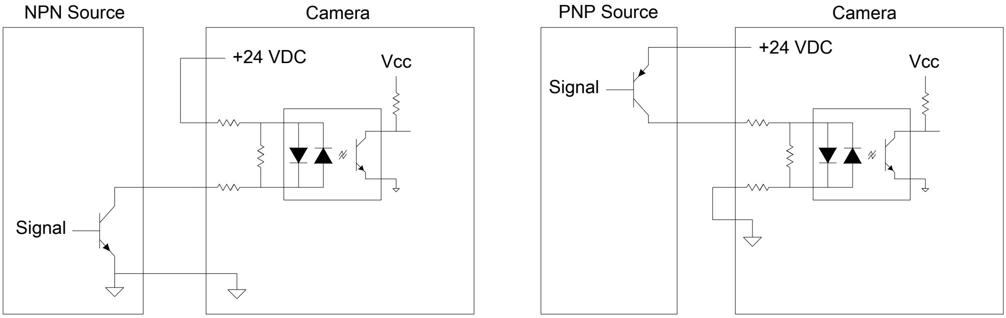


FIGURE 2–15. Input Opto Wiring (Non-Isolated NPN and PNP Sources)



Output Opto Wiring

Sample wiring diagrams for PASS, FAIL, and DATA VALID outputs (opto outputs 1, 2, and 3) are located on terminal block TB1 (see Figure 2–16 and Figure 2–17):

- PASS (opto output 1) — Positions 6 and 7
- FAIL (opto output 2) — Positions 8 and 9
- DATA VALID (opto output 3) — Positions 10 and 11

The assignment of PASS, FAIL, AND DATA VALID to these lines is the default factory setting. In V2.0, other outputs can be configured to come out on the OPTO_OUT1, OPTO_OUT2, and OPTO_OUT3 lines.

FIGURE 2–16. **Output Opto Wiring (Isolated Input)**

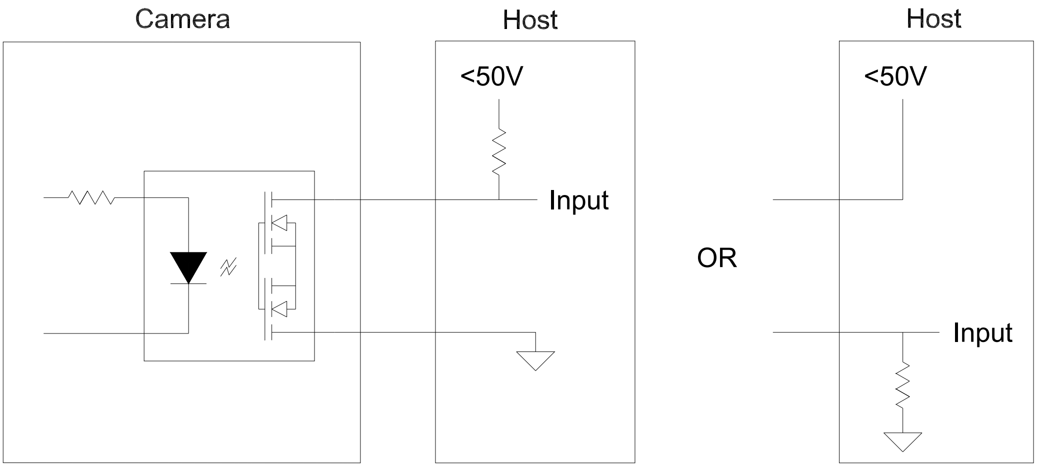


FIGURE 2–17. **Output Opto Wiring (Isolated Relay and PLC Inputs)**

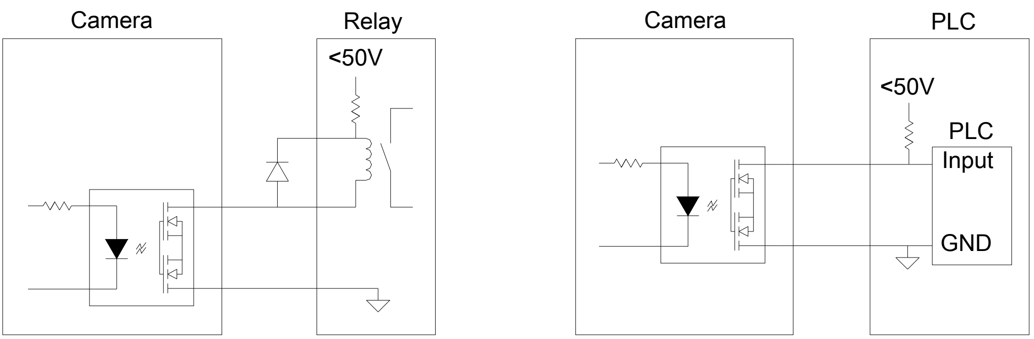
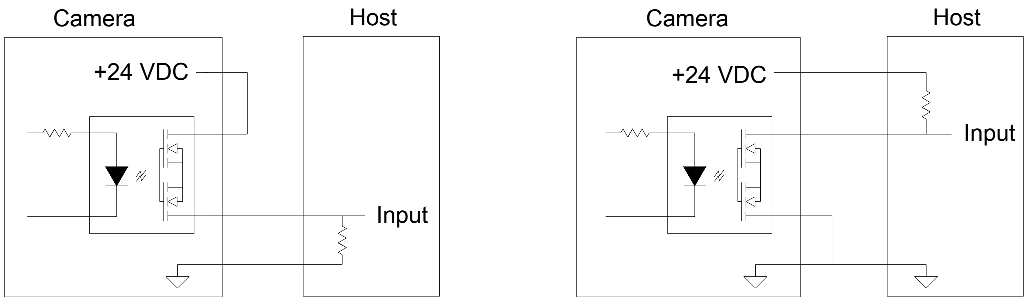


Figure 2–18 shows the output opto wiring for non-isolated inputs.

FIGURE 2–18. Output Opto Wiring (Non-Isolated Inputs)



The maximum current that can pass through the optoisolators is 50 mA.

Non-isolation setup can cause damage to the HawkEye™ 1500 if excessive voltage is applied to the optoisolators.

Sample wiring diagrams for strobe output (TTL I/O 1 in strobe mode) are located on terminal block TB1 positions 12 and 13 (GND) (see Figure 2–19 and Figure 2–20):

FIGURE 2–19. Equivalent Circuit of TTL IO 1 in Strobe Mode

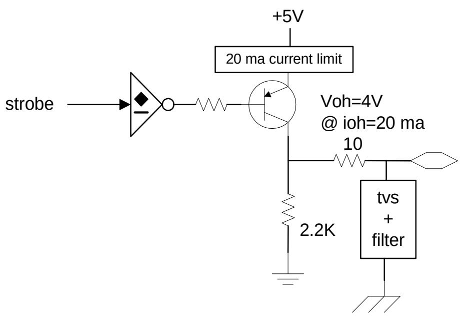
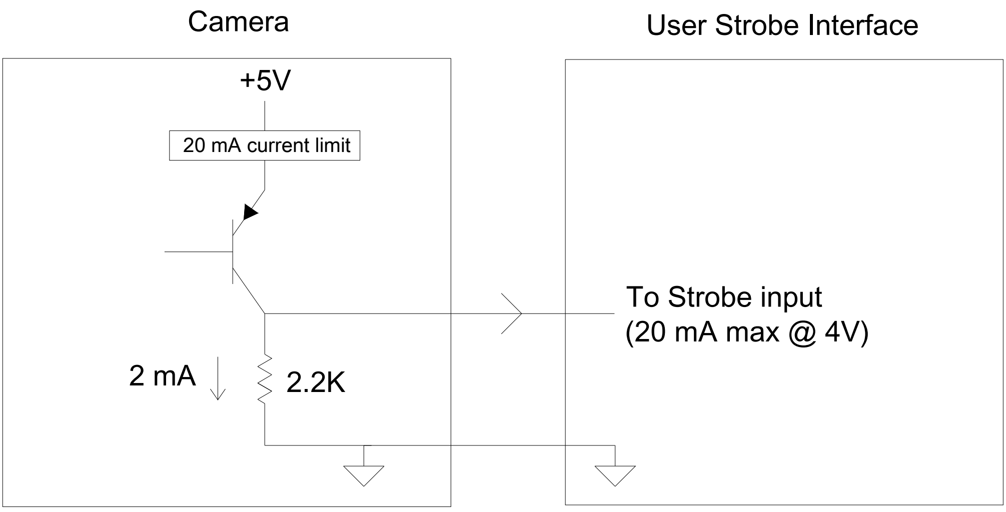


FIGURE 2–20. External DIO (TTL IO 1 Only) Wiring in Strobe Mode



Note: Jumper (JMP1) must be set to positions 1 and 2 to use the strobe output.

General Purpose I/O Wiring

Each Opto 22™ module has two associated screw terminals. Wire them according to the manufacturer's recommendations and your company's electrical standards. Typically, one side of all input modules connect to the hot power terminal with input devices (e.g., switches) wired between the other contact and the common terminal. Output modules are often wired to the common terminal, with loads returning to the hot terminal.

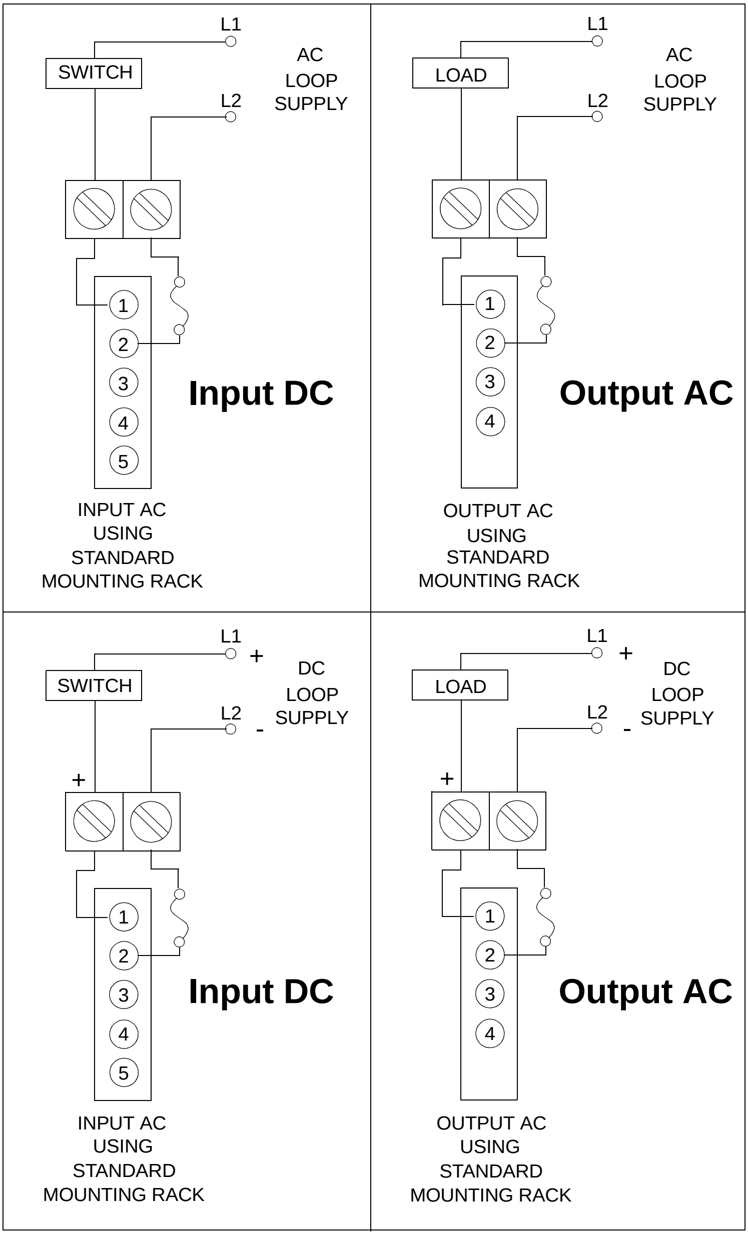
Figure 2–21 shows four sample wiring diagrams for: Input AC, Output AC, Input DC, and Output DC Opto modules on terminal block TB2.

- Opto 1 Positions 1 & 2
- Opto 2 Positions 3 & 4
- Opto 3 Positions 5 & 6
- Opto 4 Positions 7 & 8

Note: Correct polarity is critical for DC modules. The odd-numbered terminals on the I/O board are positive (+) and the even-numbered terminals are negative (-).

Jumper (JMP1) must be set to positions 2 and 3 to use opto 1 (TTL I/O 1) in I/O mode.

FIGURE 2–21. I/O Expansion Module G4 Opto I/O Wiring Examples

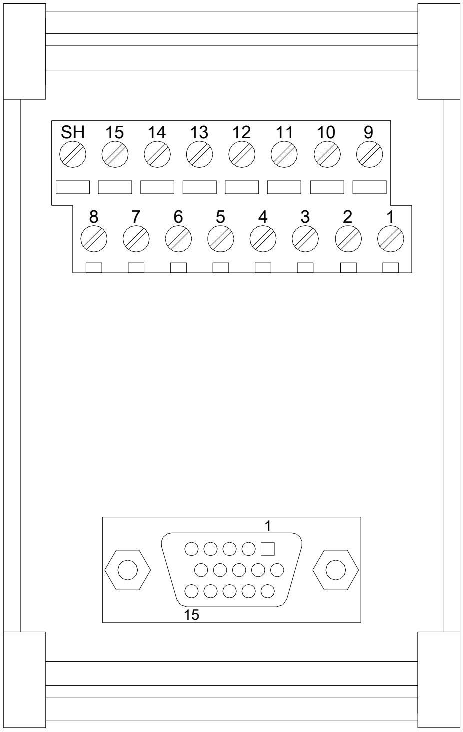


This gives you access to a combination of up to 4 input or output modules.

External I/O Terminal Block Adapter (Optional)

Figure 2–22 shows the I/O external terminal block adapter.

FIGURE 2–22. I/O External Terminal Block Adapter



External I/O Terminal Block Connectors

I/O Interface Connector – (15 Connector HDB-Sub)

Figure 2–23 shows the pinout for the HDB-15S connector.

FIGURE 2–23. Field I/O (HDB-15S) Connector

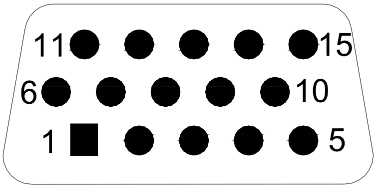


TABLE 2–14. Pinout I/O Interface Connector (15 Position HDB-Sub)

Pin	Description
1	Sensor Input A
2	Opto Output 1 A
3	Opto Output 2 A
4	Opto Output 3 A
5	General Purpose I/O 1 or Strobe
6	+24 volts
7	General Purpose I/O 2
8	General Purpose I/O 3
9	General Purpose I/O 4
10	24 volt return (Ground)
11	Sensor Input B
12	Opto Output 1 B
13	Opto Output 2 B
14	Opto Output 3 B
15	I/O Return (Ground)
Shell	Chassis Ground

Signal Distribution Terminal Block

Figure 2–24 shows the Signal Distribution Terminal Block.

FIGURE 2–24. Signal Distribution Terminal Block

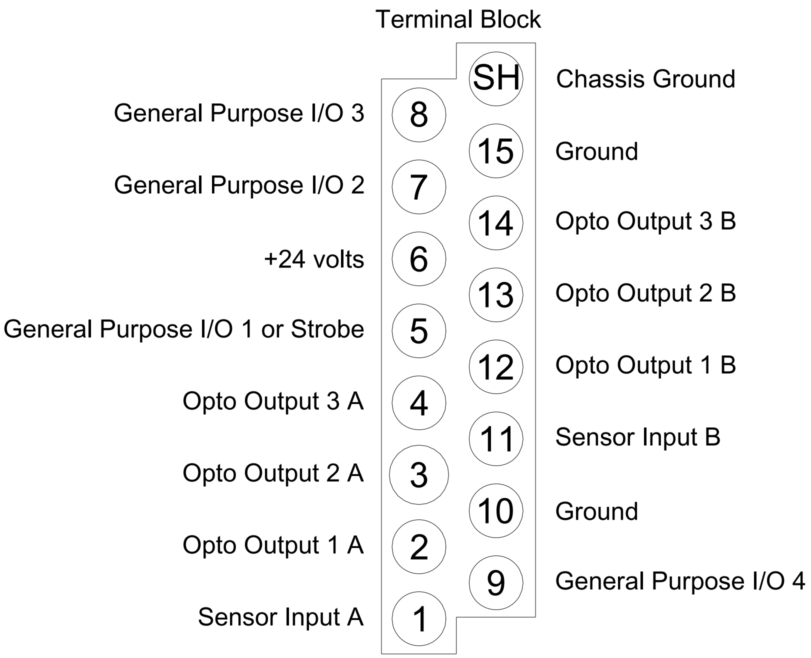


TABLE 2–15. Pinout Signal Distribution Terminal Block

Pin	Description
1	Sensor Input A
2	Opto Output 1 A
3	Opto Output 2 A
4	Opto Output 3 A
5	General Purpose I/O 1 or Strobe
6	+24 volts
7	General Purpose I/O 2
8	General Purpose I/O 3
9	General Purpose I/O 4
10	24 volt return (Ground)
11	Sensor Input B

TABLE 2–15. **Pinout Signal Distribution Terminal Block (Continued)**

12	Opto Output 1 B
13	Opto Output 2 B
14	Opto Output 3 B
15	I/O Return (Ground)

Field I/O Wiring Examples

Input Opto Wiring

Sample wiring diagrams for trigger inputs A and B (opto inputs) are located on terminal block TB1 positions 1 and 11 (see Figure 2–25 and Figure 2–26):

FIGURE 2–25. **Input Opto Wiring (Isolated NPN and PNP Sources)**

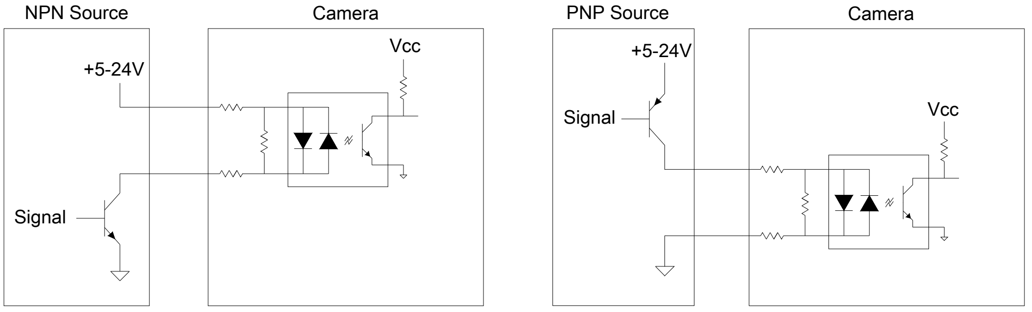
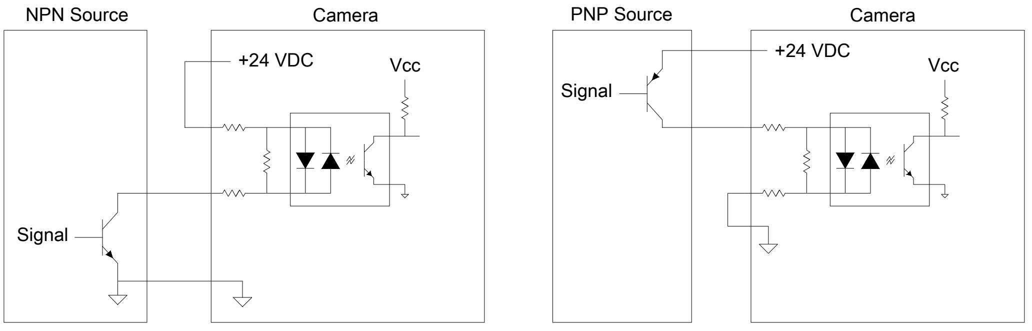


FIGURE 2–26. **Input Opto Wiring (Non-Isolated NPN and PNP Sources)**



Output Opto Wiring

Sample wiring diagrams for PASS, FAIL, and DATA VALID outputs (opto outputs 1, 2, and 3) are located on terminal block TB1 (see Figure 2–27, Figure 2–28 and Figure 2–29):

- PASS (opto output 1) — Positions 2 and 12
- FAIL (opto output 2) — Positions 3 and 13
- DATA VALID (opto output 3) — Positions 4 and 14

The assignment of PASS, FAIL, AND DATA VALID to these lines is the default factory setting. In V2.0. other outputs can be configured to come out on the OPTO_OUT1, OPTO_OUT2, and OPTO_OUT3 lines.

FIGURE 2–27. Output Opto Wiring (Isolated Input)

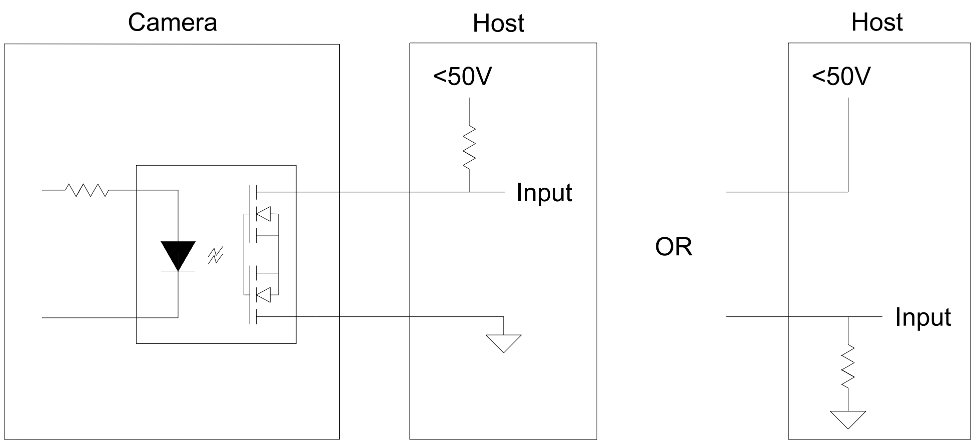


FIGURE 2–28. Output Opto Wiring (Isolated Relay and PLC Inputs)

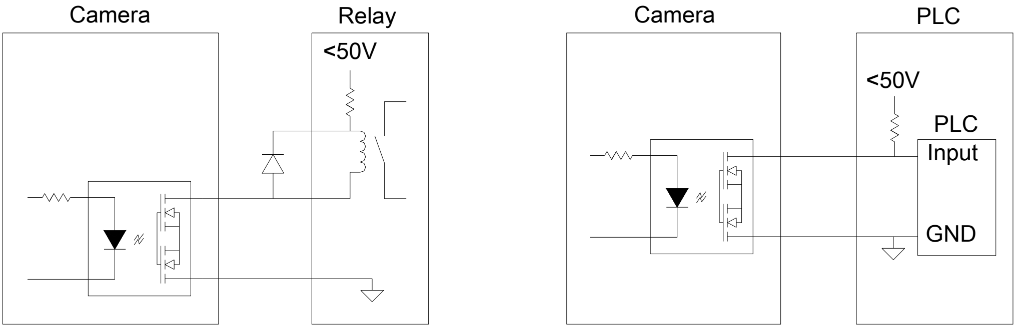
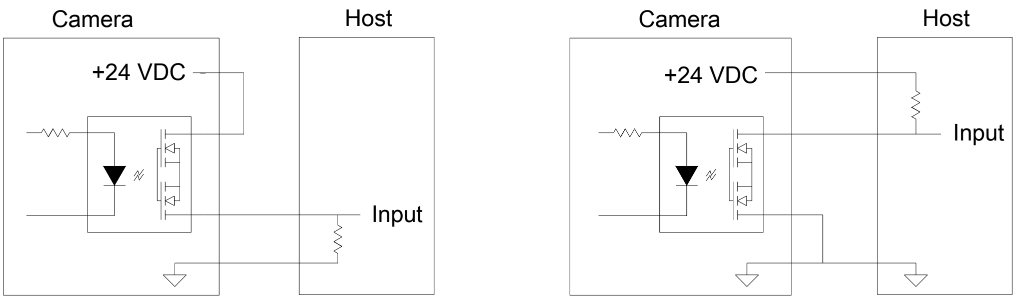


FIGURE 2–29. Output Opto Wiring (Non-Isolated Inputs)



The maximum current that can pass through the optoisolators is 50 mA.

Non-isolation setup can cause damage to the HawkEye™ 1500 if excessive voltage is applied to the optoisolators.

TTL I/O Wiring

Sample wiring diagrams for strobe output (TTL I/O 1 in strobe mode) are located on terminal block positions 5 and 15 (GND) (see Figure 2–30 and Figure 2–31):

FIGURE 2–30. Equivalent Circuit of TTL IO 1 in Strobe Mode

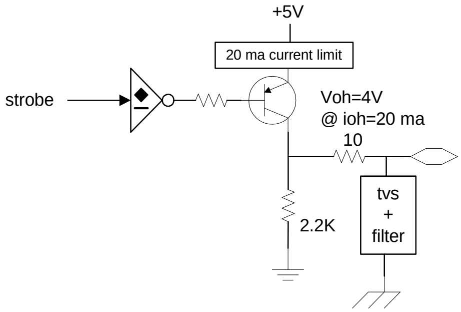
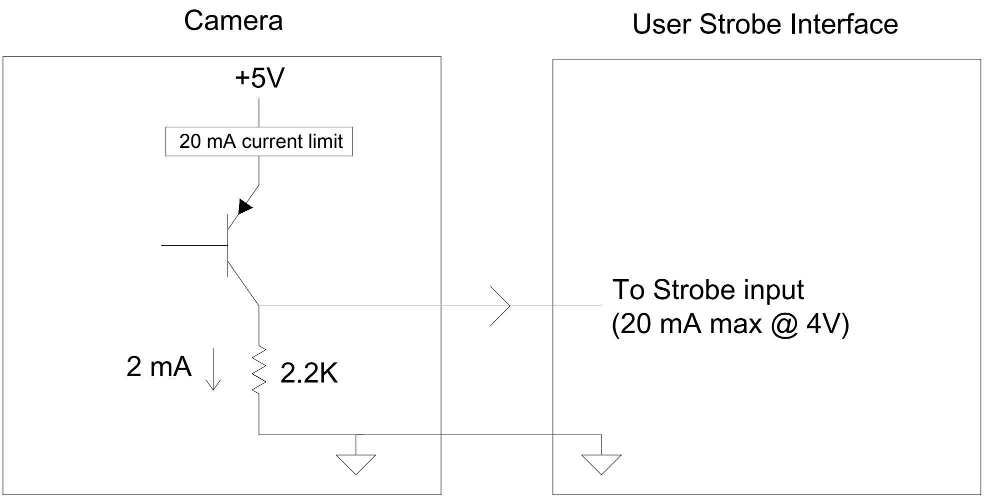


Figure 2–31 shows the equivalent DIO wiring in strobe mode.

FIGURE 2–31. External DIO (TTl IO 1 Only) Wiring in Strobe Mode



Sample wiring diagrams for external TTL I/O (including TTL I/O 1 in I/O mode) are located on the terminal block:

- TTL I/O 1 Position 5

- TTL I/O 2 Position 7
- TTL I/O 3 Position 8
- TTL I/O 4 Position 9

See Figure 2–32 and Figure 2–33.

FIGURE 2–32. Equiv. Circuit of TTL I/Os (including TTL IO 1 in I/O Mode)

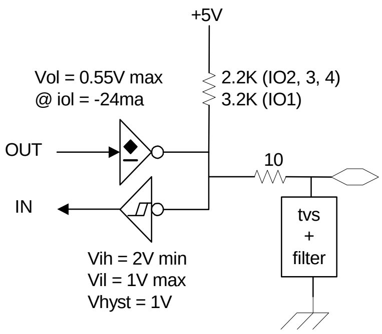
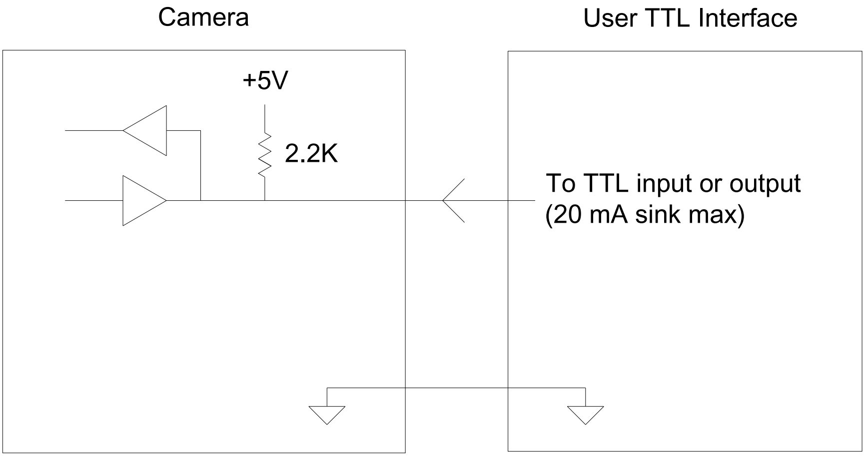


FIGURE 2–33. External DIO Wiring in I/O Mode



External I/O Terminal Block Adapter Cable (Optional)

Figure 2–34 shows the wiring for the external I/O terminal block adapter cable.

FIGURE 2–34. External I/O Adapter Cable

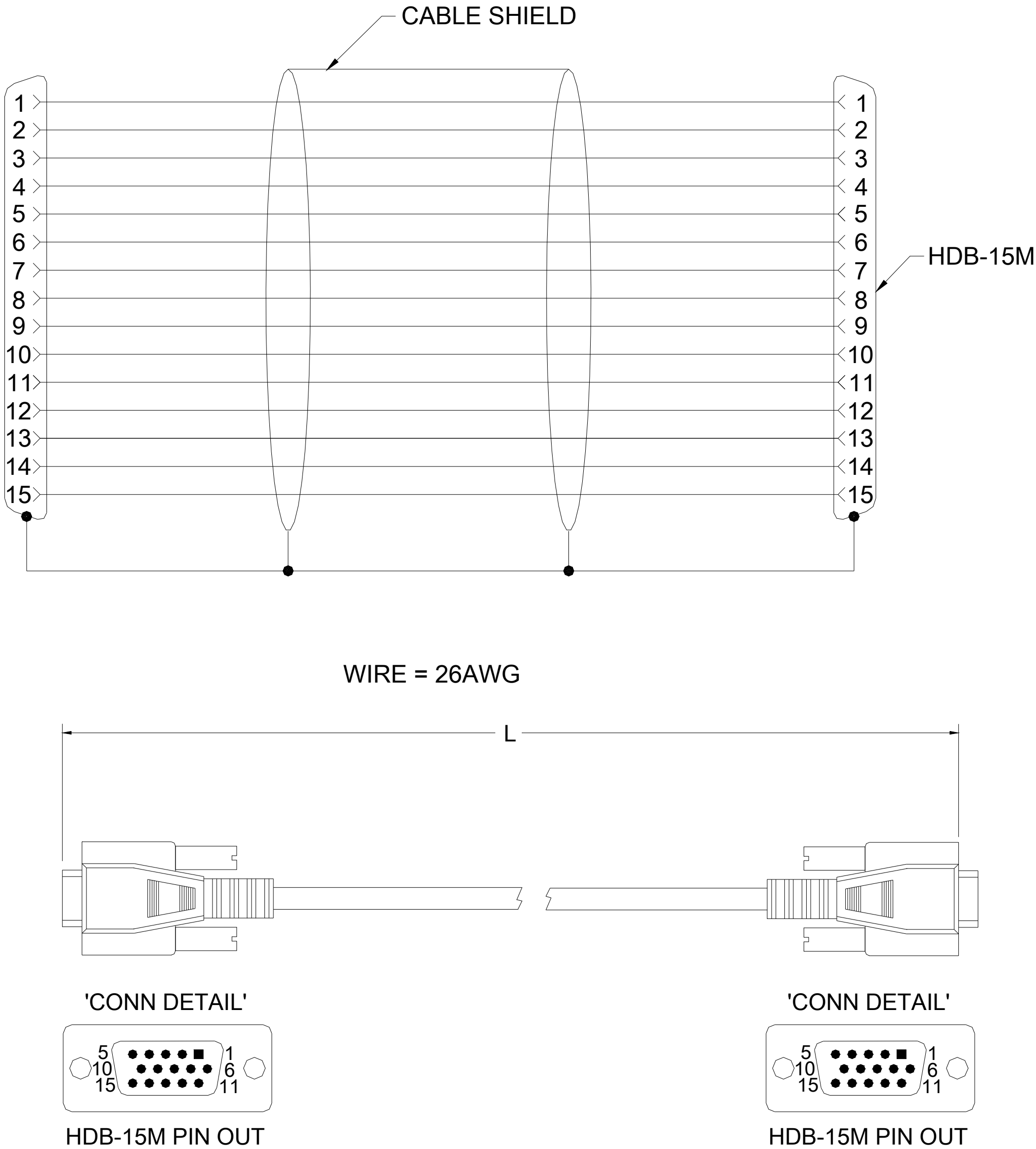


Table 2–16 shows the part numbers for the terminal block adapter and cables.

TABLE 2–16. I/O Terminal Block Adapter & Cable Part Numbers

Microscan P/N	Description
990-0056-1	Adapter, HD-SUB 15 connector to terminal block
966-0183-1	Cable HD-SUB 15 connector M/M 10 Feet
966-0183-2	Cable HD-SUB 15 connector M/M 15 Feet
966-0183-3	Cable HD-SUB 15 connector M/M 25 Feet

External Strobe & Sensor

For continuous motion or high-speed indexing applications, an external strobe and sensor may be required to freeze each part before the image can be acquired. The strobe unit may include a fiber-optic light pipe.

When choosing your part sensor, you must consider the time interval between the part passing into the sensing zone and an electrical signal being generated. When there is a large variation in process speed, considerable apparent motion of the part within the FOV may result. The HawkEye™ 1500 can compensate for this motion over a considerable range. However, the sensor should be fast enough to minimize this apparent shift.

To connect your sensor and strobe, refer to Figure 2–35 and Figure 2–36.

Figure 2–35 shows the I/O Expansion Module Sensor and Strobe Connections shown with a fiber-optic sensor, an SCM-1 strobe controller, and a 24 volt power supply.

FIGURE 2–35. I/O Expansion Module Sensor and Strobe Connections

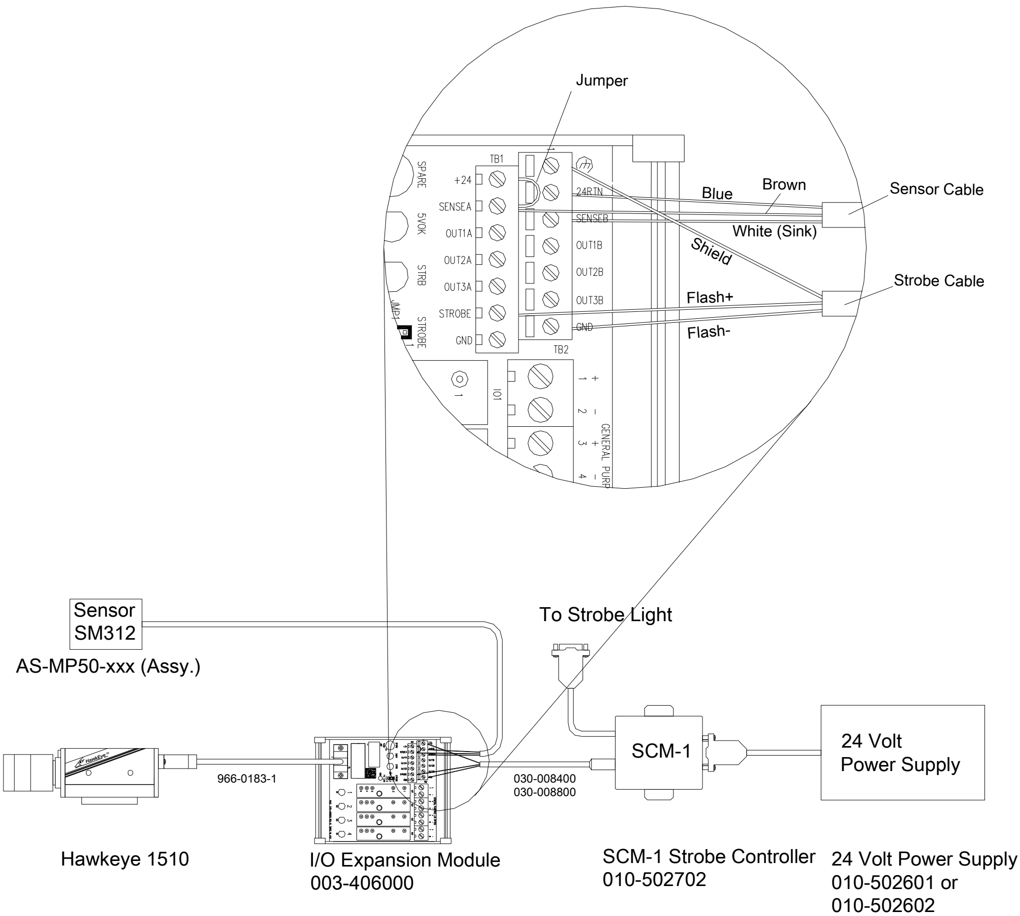
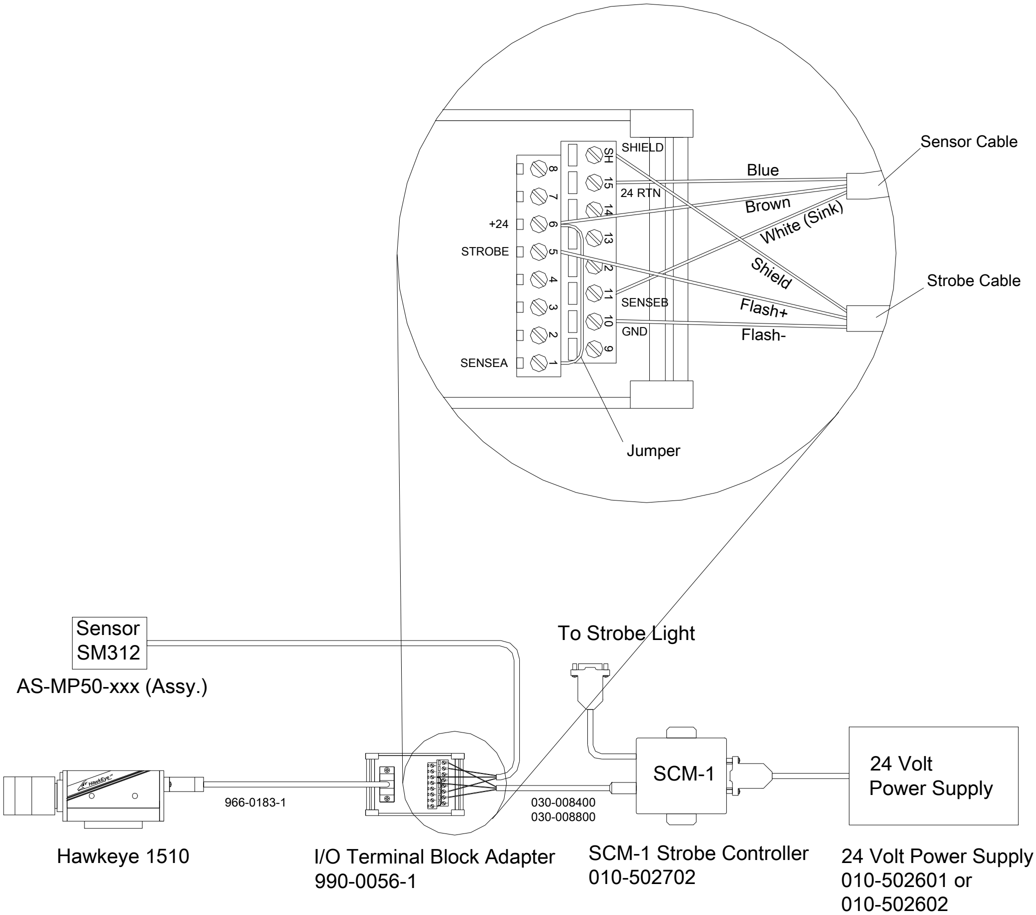


Figure 2–36 shows the I/O Terminal Block Adapter Sensor and Strobe Connections shown with a fiber-optic sensor, an SCM-1 strobe controller, and a 24 volt power supply.

FIGURE 2–36. **I/O Terminal Block Adapter Sensor & Strobe Connections**



Serial Connector & Serial Adapter Cable

Figure 2–37 shows the pinout for the mini-DIN 8 serial cable.

FIGURE 2–37. Serial Connector (mini-DIN 8)

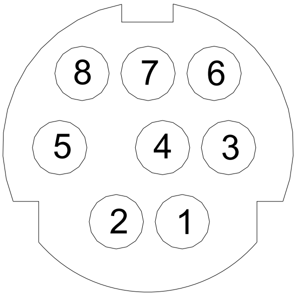


Table 2–17 lists the suppliers for the mini-DIN 8 mating connector.

TABLE 2–17. mini-DIN 8 Mating Connector Suppliers

Supplier	Part Number	Description
Kycon	KDMLA(x)-8P-G30	Mini-DIN Plug - 8 Position Snap and Lock Kit

Figure 2–38 shows the wiring for the HESC-006 serial adapter cable.

FIGURE 2–38. HESC-006 Serial Adapter Cable

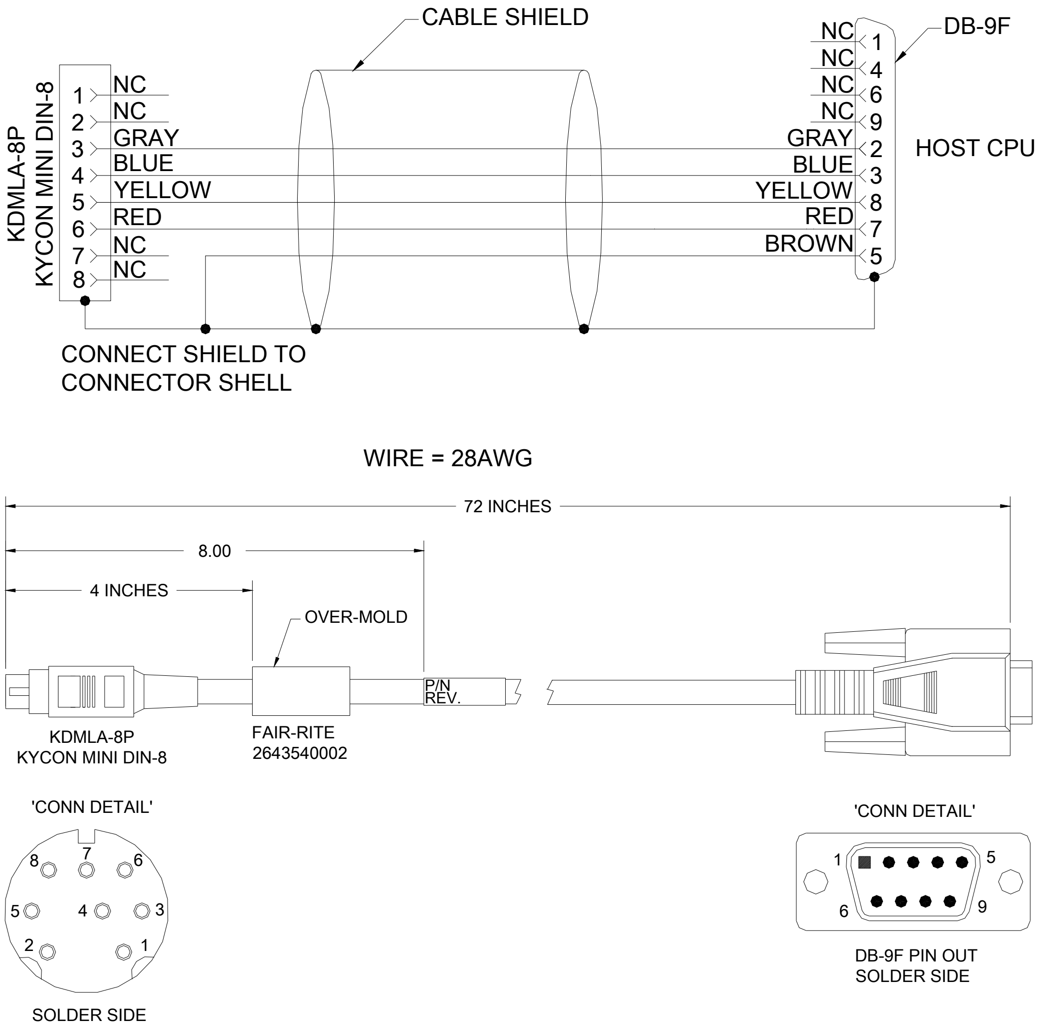


Table 2–18 describes the signals for the serial connector and serial cable.

TABLE 2–18. Serial Connector (mini-DIN 8), Serial Adapter Cable (DB9S)

DB9 Pin	Mini-DIN 8 Pin[1]	RS232 Signal	Direction	Description
1	n/c	DCD	n/u	Carrier detect
2	3	TXD	out	Transmit data
3	4	RXD	in	Receive data
4	n/c	DTR	n/u	Data terminal ready
5	7, 8	Gnd	Common	Signal ground
6	n/c	DSR	n/u	Data set ready
7	6	CTS	in	Clear to send
8	5	RTS	out	Ready to send
9	n/c	RI	n/u	Ring indicator
Shell	Shell	Shell	Common	Chassis ground

Note:

1. Mini-DIN 8 pins 1 & 2 are not connected.

Ethernet

Figure 2–39 shows the pinout for the RJ45 Ethernet connector.

FIGURE 2–39. RJ45 Ethernet Connector

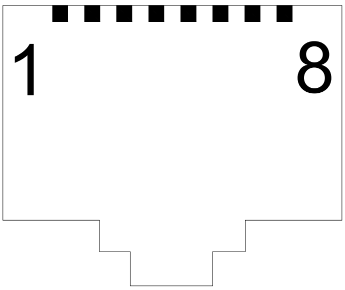


Table 2–19 describes the Ethernet connector signals.

TABLE 2–19. RJ45 Ethernet Connector Signals

Pin	Ethernet Signal
1	Ethernet TX (+)
2	Ethernet TX (-)
3	Ethernet RX (+)
4	
5	
6	Ethernet RX (-)
7	
8	

Power & Ethernet LEDs

Figure 2–40 shows the rear panel power and Ethernet LEDs.

FIGURE 2–40. Rear Panel Power & Ethernet LEDs

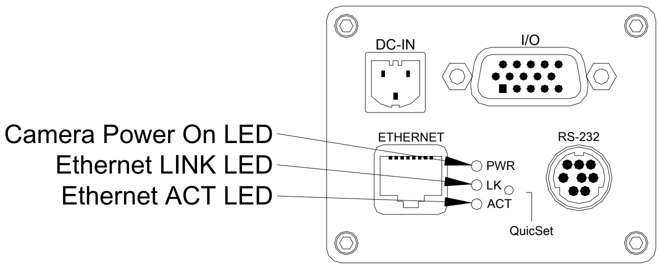


Table 2–20 describes the rear panel power and Ethernet LEDs.

TABLE 2–20. Rear Panel Power & Ethernet LEDs

Name	Color	Description
PWR	Green	Power on LED
LK	Green	Ethernet Link LED
ACT	Yellow	Ethernet Activity LED

Table 2–21 describes the Ethernet link LEDs.

TABLE 2–21. Ethernet Link LEDs

LK LED (Green)	ACT LED (Yellow)	Description
Solid	Solid	100 BaseT/Full Duplex
Solid	Flashing	100 BaseT/Half Duplex
Flashing	Solid	10 BaseT/Full Duplex
Flashing	Flashing	10 BaseT/Half Duplex

Mode/Status LEDs

Figure 2–41 shows the rear panel mode and status LEDs.

FIGURE 2–41. Mode/Status LEDs

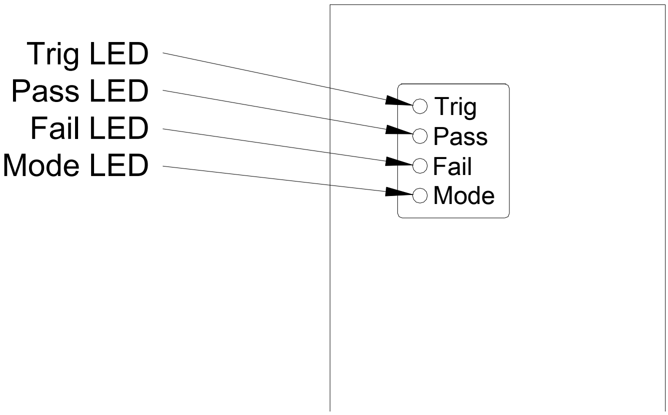


Table 2–22 describes the mode and status LEDs.

TABLE 2–22. Mode/Status LEDs

Name	Description	LED Color
Trig	Trigger LED	Yellow
Pass	Pass LED	Green
Fail	Fail LED	Red
Mode	Mode LED	Yellow

Verification LEDs

Table 2–23 shows the state of the LEDs when Verification is on.

TABLE 2–23. LEDs When Verification Is On

	Pass LED	Fail LED
Decode/Locate Failure	OFF	ON
POOR	OFF	ON
FAIR	ON	ON (halfway between GOOD and POOR)
GOOD	ON	OFF

Beeper

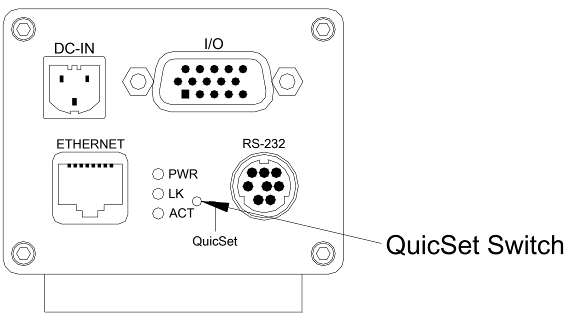
The HawkEye™ 1500 camera's beeper:

- Is user configurable
- Indicates Pass/Fail conditions
- Assists in the alignment of the symbol during setup

QuicSet®

Figure 2–42 shows the location of the QuicSet® recessed button.

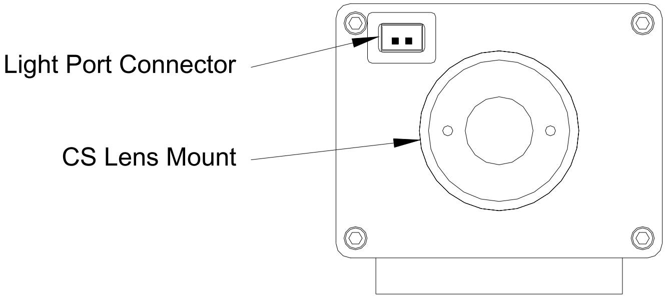
FIGURE 2–42. Location of QuicSet® Recessed Button



Front Panel HawkEye™ 1510

Figure 2–43 shows the layout of the front panel for the HawkEye™ 1510.

FIGURE 2–43. Front Panel Layout (HawkEye™ 1510 only)



- Light Port Connector – 12 VDC out
- CS Lens Mount – 5mm extension required to support C-Mount lenses (5 mm extension tube part number is 928-0047-1)

Light Port Connector

Figure 2–44 shows the pinout for the light port connector.

FIGURE 2–44. Light Port Connector

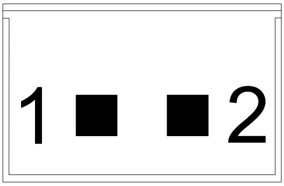


Table 2–24 lists the supplier for the light port mating connector.

TABLE 2–24. Light Port Mating Connector Supplier

Supplier	Part Number	Description
Molex	51090-0200 50212-8100	2 Position housing Crimp terminal

FIGURE 2–45. Optional Light Port Cable

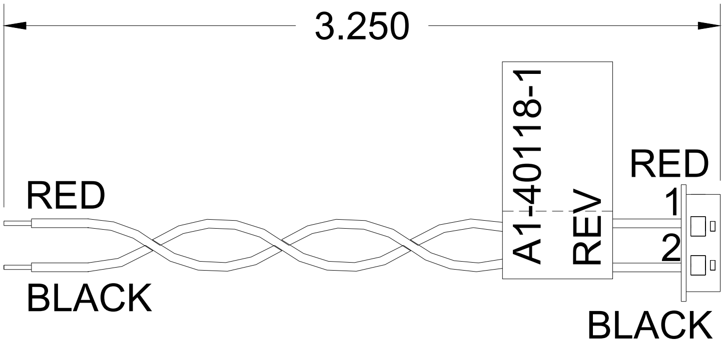


Table 2–25 lists the pinout for the light port.

TABLE 2–25. Light Port Pinout

Pin	Signal Name	State	Voltage
1	+12VDC	Off	Open
		On/Strobe	+12V
		Power Strobe	+24V
2	12V return		

Table 2–26 lists the part number for the optional light port cable.

TABLE 2–26. Optional Light Port Cable Part Number

Microscan P/N	Description
A1-40118-1	Cable, light port DC power to external light head

Do not exceed 0.5 amp continuous current draw or you may damage the HawkEye™ 1500.

Mounting Blocks

Table 2–27 contains the part numbers and descriptions of the four mounting blocks for the HawkEye™ 1500 camera.

TABLE 2–27. Mounting Blocks

Part Number	Description	Used on HawkEye™	Notes
HEBMA-5	Standard mounting block	1510, 1515, 1525	
HEBMA-2	Optional mounting block	1525	1
HEBMA-3	HE10 mounting block adapter	1510	2
HEBMA-4	HE15 mounting block adapter	1515	3

Notes:

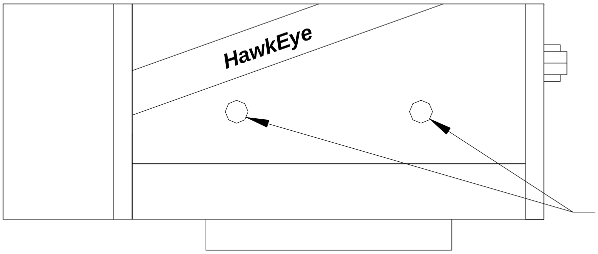
1. This mounting block is required for the HawkEye™ 1525 when mounting the reader where the mounting surface extends to the light. This mounting block is 3/8” thick vs. ¼” thick for the standard block.
2. Allows a HawkEye™ 1510 to mount in the same location when upgrading an HawkEye™ 10.
3. Allows a HawkEye™ 1515 to mount in the same location when upgrading an HawkEye™ 15.

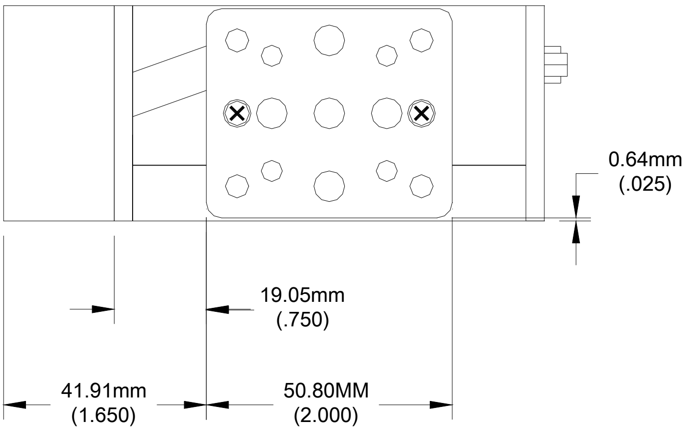
Note: Do not insulate the mounting block. The bottom of the HawkEye™ gets hot and metal-to-metal contact is necessary for heat dissipation. Refer to Appendix E, “Specifications,” for mounting block dimensions.

Optional Location for Mounting Block

Figure 2–46 shows the optional locations for the mounting block on the HawkEye™ 1500.

FIGURE 2–46. **Optional Location for Mounting Block (1515 Shown)**





Remove the two hole covers before installing mounting block in one of the two optional mounting locations, as shown in Figure 2–46. (A HawkEye™ 1515 is shown.) Use 4-40 x 5/16 pan head screws.

Using longer screws may damage the HawkEye™ 1500.

CHAPTER 3

HawkEye™ 1500 Series Overview

The chapter contains an overview of the HawkEye™ 1500 Series Smart Camera-Based Reader. It also contains information about application modes, lighting modes, retry modes, and I/O operations. For verification information, see the HawkEye™ 1500 Series Verification Manual.

The HawkEye™ 1500 has a wide variety of applications ranging from reading symbols on parts speeding by on a conveyer to stationary applications where each symbol is manually presented. The HawkEye™ 1500 has a rich set of options for dealing with different triggering, lighting, and I/O situations. ReadRunner, the HawkEye™ 1500 Graphical User Interface, provides a simple way of setting up all the various parameters. In particular, it presents a small number of application modes, each of which pre-sets the various parameters based on common usage scenarios. These application modes are merely starting points; any parameter can be subsequently altered, if necessary. For more information, see “Application Modes” on page 3-3.

Control over exposure of acquired images is referred to as **photometry**. The HawkEye™ 1500 provides both manual and automatic photometry. The auto-photometry mode is quite powerful and can adjust the exposure and gain separately for each image acquisition. For more information, see “QuicSet® Symbol Photometry” on page 3-35 and “Setting Up Photometry” on page 4-22.

For situations where the symbol may be moving or difficult to decode, several different **retry modes** are provided. The HawkEye™ 1500 can retry by count, time, or for the duration of an input signal. For more information, see “Retry Modes” on page 3-6.

There are several options available for configuring digital **I/O**. To interface to external equipment, you have the option of either using a timed data valid signal or a full handshake. For more information, see “HawkEye™ 1500 I/O Operations” on page 3-10.

The HawkEye™ 1500 also provides a rich set of ways to send data to external equipment, either using serial or network connections. The capabilities are very powerful but are also very simple to use. The options range from sending the decode data as a simple ASCII string via the serial port, up to sending a “**Cycle Report**” for each decode. The Cycle Report includes counts, timing data, and much more, and may even contain images. Libraries are included that allow custom data collection applications to be easily written using Visual Basic. For more information, see “Displaying Camera Report Information” on page 4-15.

The normal operational mode of the HawkEye™ 1500 is called “online”. In this mode, the results of each decode are sent to all connected client applications. Without taking the camera out of this mode, it is possible to program the camera and change parameters “on the fly.” This is called taking **control** of the unit. When a client takes control, it essentially has a private connection to the camera and results are not sent to other clients. For more information, see Chapter 4, “ReadRunner”.

Note: Use this device as specified by Microscan.

Unique Camera Names

At boot time, if a camera's network name is "HawkEye," the camera's software automatically changes "HawkEye" to "HawkEyexxyyzz" where xxyyzz is the last three fields of the camera's MAC address. For example:

TABLE 3–1. Unique Camera Names

Camera Name at Boot Time	MAC Address	Unique Camera Name
HawkEye	00:60:33:e0:00:11	HawkEyeE00011
HawkEye	00:60:33:e0:00:12	HawkEyeE00012
HawkEye	00:60:33:e0:00:13	HawkEyeE00013
HawkEye	00:60:33:e0:00:14	HawkEyeE00014
HawkEye	00:60:33:e0:00:15	HawkEyeE00015

Application Modes

To integrate the HawkEye™ 1500 with various scanning applications, one may need to change its operating parameters. As an aid to configuring the HawkEye™ 1500, ReadRunner has an Application Mode display available to you. This display contains four choices that generalize typical use for the HawkEye™ 1500. From the selection chosen, a number of pre-defined parameter selections are made. You may override any of these selections at any time. The four general application modes are:

- Demo
- Stop and Scan
- Motion
- Supermarket

Demo

Demo mode is the default application mode. It is fully automatic in operation. Illumination, targeting, automatic photometry, and continuous triggering are all on. The HawkEye™ 1500 will continuously read any symbol that is aligned to the laser target dots, that is, placed in the field of view.

Motion

This mode scans a symbol when it moves within the HawkEye™ 1500's field of view. Because the parts are in motion, there is no time for the automatic calculation of photometry values. Thus, automatic photometry is off. You must specify the exposure and gain settings manually. A trigger is supplied (through the I/O connector) to the camera by a sensor device. In addition, the Lighting selection will be Power-strobe. This selection allows the parts to be moving at a very high speed, where short exposure times are desired.

Stop and Scan

This mode scans a symbol when it stops within the HawkEye™ 1500's field of view. A trigger is supplied (through the I/O connector) to the camera by a sensor device when the symbol is in place. The illumination and laser targeting are set to strobe, and the automatic photometry control is on. When the trigger is supplied, the HawkEye™ 1500 will determine the correct gain and exposure and read the symbol.

Supermarket

This is often referred to as Presentation Mode. Parameters similar to those for Demo mode are used with the exception that an inter-symbol wait time (ISWT) is imposed. This means that, when a symbol is presented to the HawkEye™ 1500's field of view, it will be read once. The same symbol cannot be read again until the symbol is removed from the field of view for at least the indicated ISWT. A different symbol, when presented to the camera, will be read immediately.

Lighting Modes

Symbols or marks that are to be read by the HawkEye™ 1500 must be illuminated for optimum read performance. Depending on the type of surface texture or ambient light surrounding the symbol, the lighting may be customized. In addition, there are several models of the HawkEye™ 1500 available (Chapter 1, "Configurations"), and it is assumed that you have purchased the model that is optimal for the application that you are deploying. All models have the following configurations available to control the light installed on your particular unit:

- On/Off
- Strobe

- Power Strobe
- On With Power Strobe
- External (may be used with any of the other modes)

As detailed earlier in this chapter, when you select one of the four Application Modes, ReadRunner selects the appropriate lighting mode. You may override these selections as desired. The details of each are as follows:

Note: Strobing minimizes the heat produced when the light is on constantly.

WARNING! STROBING MAY HAVE ADVERSE EFFECTS ON HUMANS WHO ARE PRE-DISPOSED TO SEIZURES DUE TO PHOTOSENSITIVITY. PLACING THE HAWKEYE™ 1500 IN AN ENCLOSURE CAN MINIMIZE THE EFFECTS OF THE FLASHING OF A STROBE.

- Off — This selection is available when the ambient lighting provided is adequate for the optimal reading of the parts to be deployed. Therefore, the internal lighting will not be used during the exposure time.
- On — This selection turns the built in lighting on continuously. The lighting will be on at all times, and at a constant intensity. It will be the lighting used during the exposure time.
- Strobe — Strobed lighting is chosen to illuminate the symbol for just the selected exposure time. Typically, this is chosen when the symbol/part to be read is in motion and at relatively slow speeds. At all other times, the HawkEye™ 1500 turns off the lighting. This results in a flashing effect.
- Power Strobe — This selection is similar to the strobed selection, with the addition of a greater intensity value being supplied. This is employed when the part is in motion and at very high speeds. The burst of energy supplied for the duration of the exposure time allows parts to be in motion at much greater speeds. The maximum limit on exposure time is 1 ms.
- On With Power Strobe — This selection is provided to keep the lighting on at all times, but, when the camera is exposing, the Power Strobe intensity is supplied.

Note: This selection minimizes the effects of strobing on human vision.

- External — When external lighting is selected, the HawkEye™ 1500 will NOT illuminate its internal lighting, but will drive an external line to control an external source for the desired exposure time. This external line can be wired to an external lighting configuration that is optimal for the unique problem set. For more information about wiring, etc., see Chapter 2, “Connecting to the HawkEye™ 1500”. External lighting is allowed in conjunction with the other lighting modes to allow more flexibility in developing lighting solutions.

In addition, four other modes are available by combining the preceding modes:

- On & External
- Strobe & External
- PowerStrobe & External
- On/PowerStrobe/External

Note: When Verification is set to “AIM DPM-1-2006”, Lighting will be set to On.

Retry Modes

When you select the typical application via ReadRunner, there is no attempt at setting any retry capabilities. This is user dependent. The various Retry Modes are described here so you can apply them as needed.

- Time
- Count
- GPIO IN 4 Duration
- ISWT
- PID List
- PID List w/Acquire

- Light

Time

You may specify that, if the Decode attempt of a symbol fails, the HawkEye™ 1500 retry for up to 1 minute. The HawkEye™ 1500 will perform automatic photometry (if selected), and capture a new image, and then attempt to decode the symbol again. The HawkEye™ 1500 will report only one success or one failure no matter how many times the retry is attempted. In other words, if the retry time is set to 10 seconds, and the decode is successful on the 7th attempt, only one Success report (and signal) will be sent from the HawkEye™ 1500. If the HawkEye™ 1500 fails after 10 seconds, only one Failure message will be reported. The normal application to use retry attempts is in the Stop and Scan application mode. Basically, you are indicating that, when the triggered input occurs, the part may not exactly be in the field of view at exactly the time of the trigger. Retry attempts will be made and only one report will be issued. Typically, you would not have time to perform retries in a motion application.

Count

You may specify that, if the decode attempt of a symbol fails, the HawkEye™ 1500 will retry up to 15 times. The HawkEye™ 1500 will perform automatic photometry (if selected), capture a new image, and then attempt to decode the symbol again. The HawkEye™ 1500 will report only one success or one failure no matter how many times the retry is attempted. In other words, if the retry count is set to 10, and the decode is successful on the 7th attempt, only one Success report (and signal) will be sent from the HawkEye™ 1500. If the HawkEye™ 1500 fails on the 10th attempt, only one Failure message will be reported. Typically, the only time you would add retry attempts is in the Stop and Scan application mode. Basically, you are indicating that, when the triggered input occurs, the part may not exactly be in the field of view, so retry attempts will be made and only one report will be issued. Typically, you would not have time to perform retries in a motion application.

GPIO IN 4 Duration

You may specify that, if the Decode attempt of a symbol fails, the HawkEye™ 1500 will retry while the GPIO input line with the assigned tag “RETRY” (see “Advanced I/O” on page 4-54) is present. The HawkEye™ 1500 will perform automatic photometry (if selected), and capture a new image, and then attempt to decode the symbol again. The HawkEye™ 1500 will report only one success or one failure no matter how many times the retry is attempted. In other words, if the GPIO signal is on for 10 seconds, and the decode is successful on the 7th attempt, only one Success report (and signal) will be sent from the HawkEye™ 1500. If the HawkEye™ 1500 fails after 10 seconds and the signal is removed, only one Failure message will be reported. Typically, the only time you would add retry attempts is in the Stop and Scan application mode. Basically, you are indicating that, when the triggered input occurs, the part may not exactly be in the field of view, so retry attempts will be made and only one report will be issued. Typically, you would not have time to perform retries in a motion application.

ISWT (Inter-Symbol Wait)

Retry ISWT is the Presentation or Supermarket Mode. This duration selection is the Inter-symbol wait time. Inspections are performed continuously and only successful attempts are reported. In order not to report duplicate data decoded from the same image, a time limit (the Inter-symbol wait time) must be specified. This time limit ensures that the duplicate data is not reported until the specified time has elapsed since the last successful decode attempt.

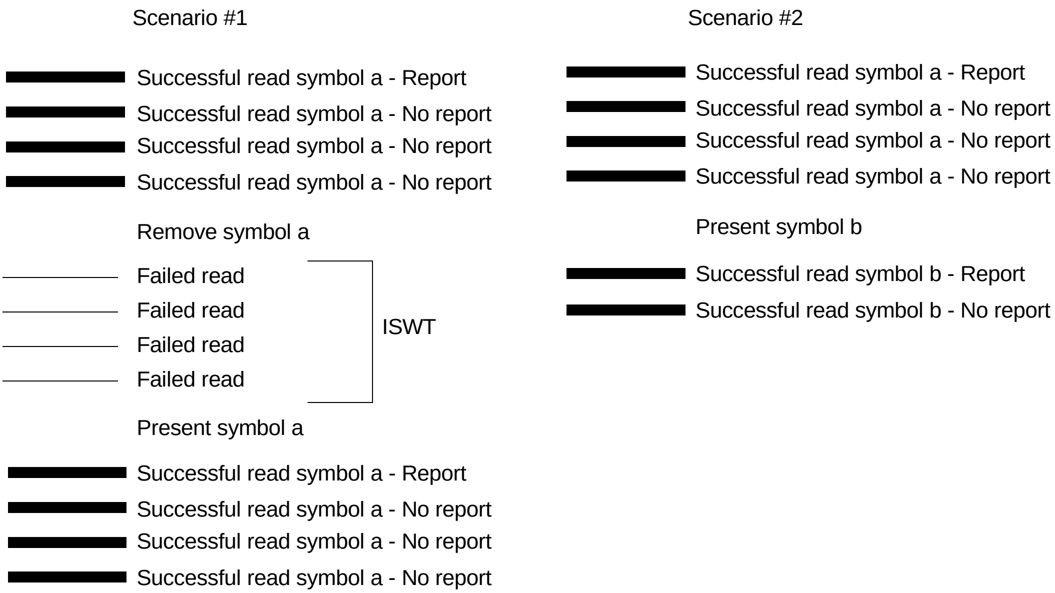
In other words, either of two things have to happen to get a report:

1. An ISWT of failures must occur between the presentation of duplicate symbols,

   OR

2. You must present a different symbol.

Graphically shown below, where a solid thick line indicates a decode read cycle, and a thin line indicated a no decode read cycle:



An ISWT equal to 0 indicates an infinite inter-symbol wait time, and no duplicate code will be reported. This is applicable to scenario #2 above.

Note: In order for the ISWT to work correctly, it must be greater than the Cycle Time reported on the ReadRunner camera report. ISWT = 0 indicates an infinite inter-symbol wait time, and no duplicate code will be reported.

PID List

The Retry List mode is differs considerably from those modes previously described. The most notable difference is that NO new image is acquired. All retry attempts are made on the same image. Therefore, you can use this mode in a Motion Application. Care must be taken that the retry list is not long enough to cause a trigger overrun.

The Retry List mode is associated with Product Identification, which is fully described in the HawkEye™ 1500 Technical Note titled “Jobs in the HawkEye™ 1500.” This Technical Note is on your HawkEye™ 1500 Series CD.

Each PID contains specific decoder settings for a given symbol. The decoder takes the one image and attempts to decode the with the settings in the listed PID until the list ends or a successful decode occurs.

PID List w/Acquire

This mode behaves the same as PID List except that a new image is taken for each PID using the photometry settings in the PID being tested.

Light

This retry mode allows a second lighting configuration to be tried while taking a second image. This might be useful if some codes are printed on glossy backgrounds and others are matte backgrounds requiring a different lighting strategy.

Note: When Verification is set to “AIM DPM-1-2006”, the Retry Mode will be disabled.

HawkEye™ 1500 I/O Operations

The HawkEye™ 1500 provides 9 discrete input/output (I/O) lines for your application. These I/O lines offer direct interaction between the HawkEye™ 1500 and the production environment via the HawkEye™ 1500 I/O connector.

Trigger Behavior

First, some definitions:

- Delay — The time after seeing the selected trigger edge until the camera image acquisition is started.
- Latch — The time after seeing the selected trigger edge during which new trigger edges will be ignored.
- Debounce — The ability to ignore switching noise that may look like multiple triggers. This is now handled by the LATCH time. Note that while the impact of noisy switching and spurious signals may be disabled during the LATCH time, addressing the root cause of these problems in the process is vital to assuring that only trigger signals are seen at the camera and that

these are the only signals that the camera responds to. Past attempts at trying to debounce the line in the camera by looking at the selected edge and seeing if the new level is maintained for a period of time, defeated the fundamental goal of triggering the acquisition when the selected edge was seen. Due to the sampling frequency of a digital line, debouncing made it possible to ignore real triggers or classifying noise as a valid trigger, both while deferring the trigger some indeterminate time into the future.

- Trigger Overrun — This results when the time between triggers is less than the time it takes the camera to complete an acquisition cycle. However, even with the time between triggers is greater than the acquisition time, it is possible to generate a trigger overrun condition by specifying a delay time that creates more than 100 pending triggers.

Trigger Diagrams

Figure 3–1 through Figure 3–6 illustrate different trigger scenarios; some produce a trigger overrun condition. All cases assume a trigger occurs on the rising edge. In the following diagrams:

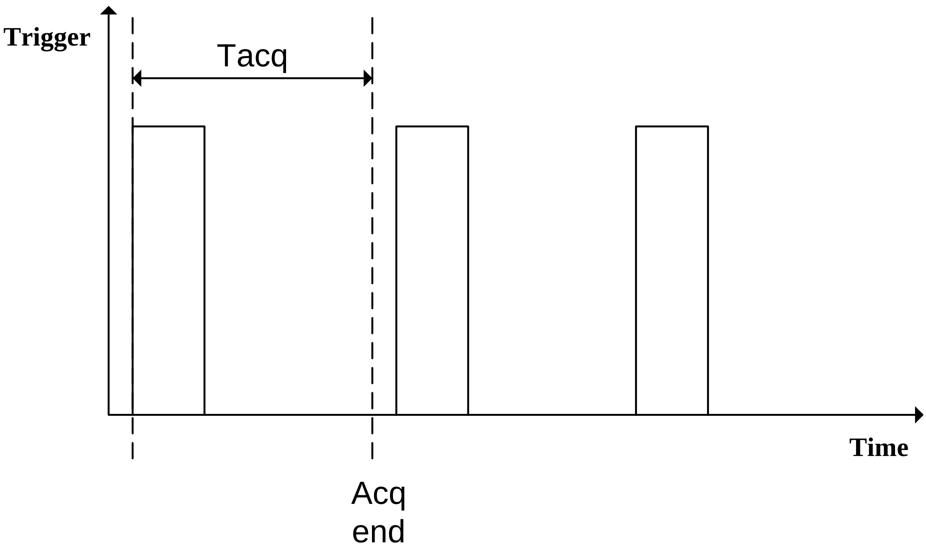
- T_d — Delay time
- T_l — Latch time
- T_{acq} — Acquisition time

Trigger Diagram 1

In this example, the acquisition cycle has time to complete before the next trigger is received. There is no trigger overrun in this case.

$T_d = 0$
$T_l = 0$
T_{acq} < time between triggers

FIGURE 3–1. Trigger Diagram

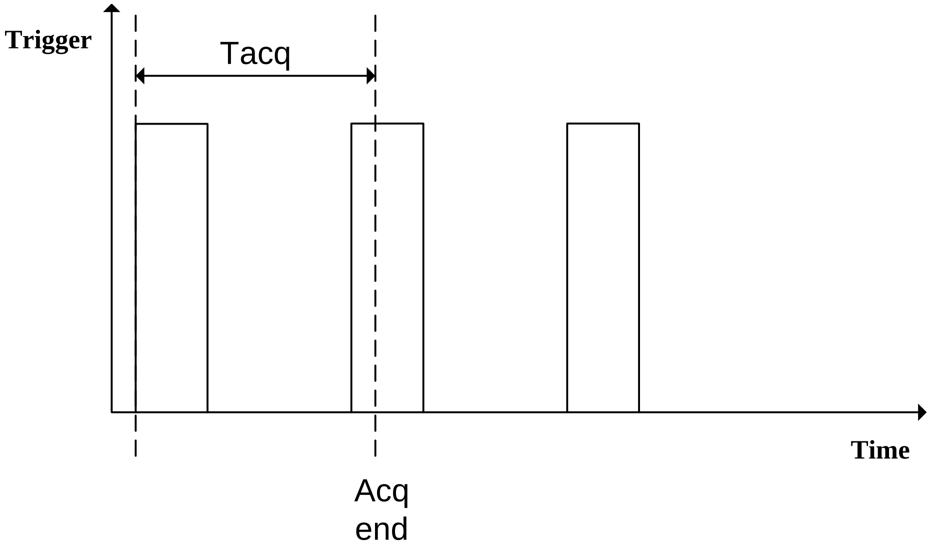


Trigger Diagram 2

This example illustrates a trigger overrun condition. The acquisition cycle has not completed before the next trigger is received.

$T_d = 0$
$T_l = 0$
T_{acq} > time between triggers

FIGURE 3–2. Trigger Diagram 2

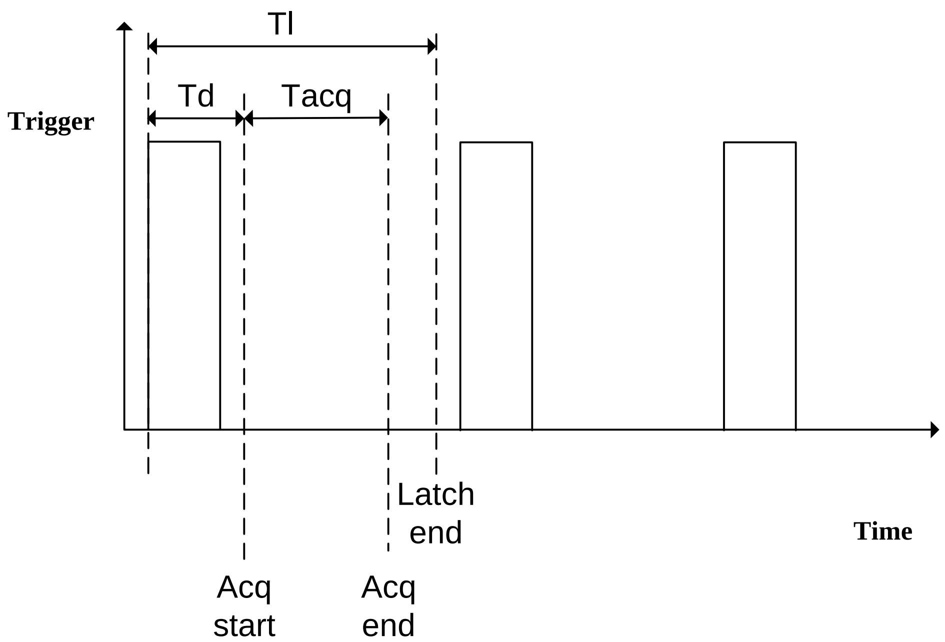


Trigger Diagram 3

In this example, both T_d and T_l are non-zero. Because T_l < time between triggers and $T_{acq} < T_l$ there is no trigger overrun condition.

T_d does not = 0
T_l < time between triggers
$T_{acq} < T_l$

FIGURE 3–3. Trigger Diagram 3

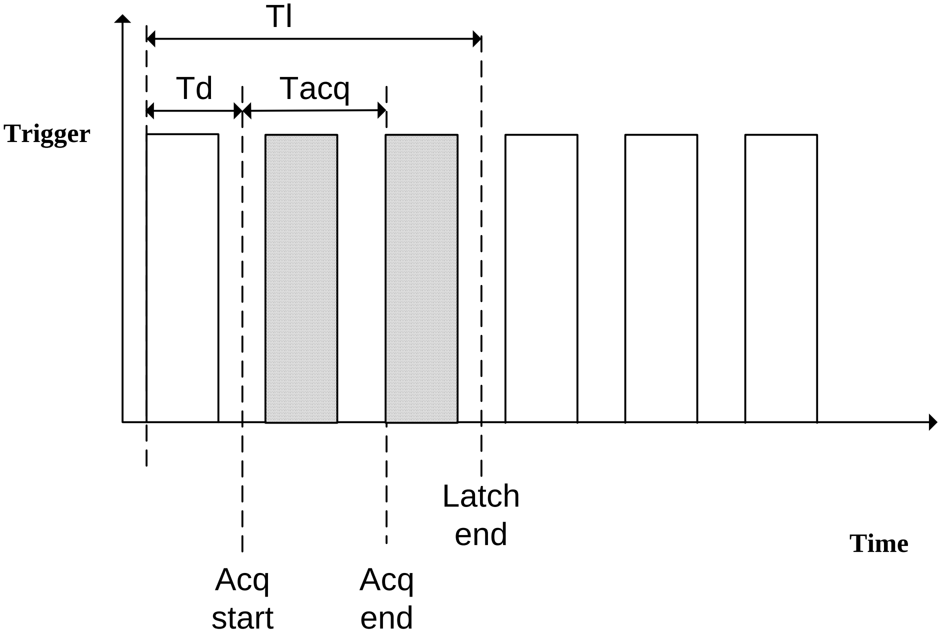


Trigger Diagram 4

This example illustrates how triggers are rejected when the latch time is greater than the time between triggers. Because T_l > time between triggers and $T_{acq} < T_l$, effectively T_{acq} < time between triggers so there is no trigger overrun.

T_d does not = 0
T_l > time between triggers
$T_{acq} < T_l$

FIGURE 3–4. Trigger Diagram

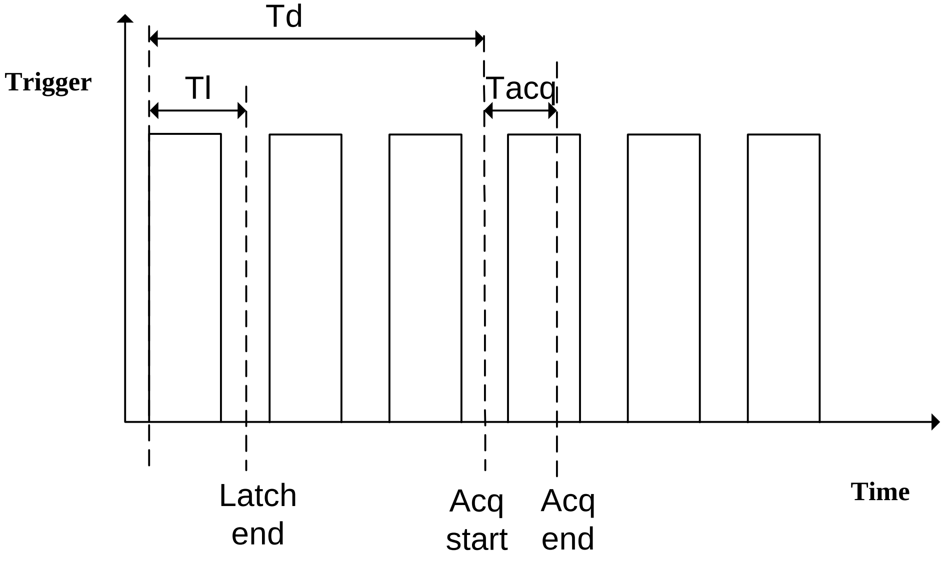


Trigger Diagram 5

This example produces no trigger overrun. Because T_l < time between triggers and $T_{acq} < T_l$ the acquisition cycle has time to complete before the next trigger. The diagram also illustrates how multiple triggers are pending in the trigger queue.

T_d > time between triggers
T_l < time between triggers
$T_{acq} < T_l$

FIGURE 3–5. Trigger Diagram

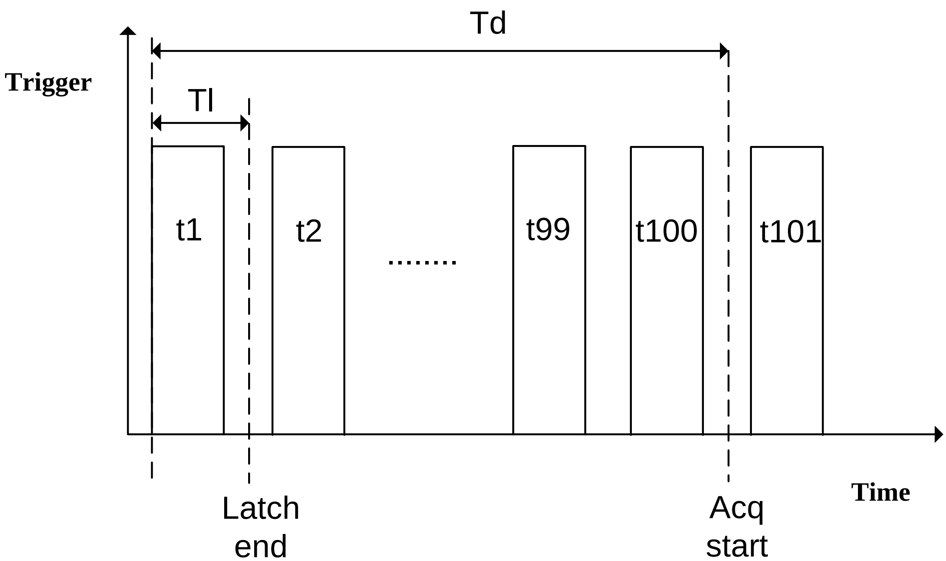


Trigger Diagram 6

This example generates a trigger overrun condition because the delay is set so long that the maximum number of pending triggers is exceeded.

T_d > time for max triggers
T_l < time between triggers
T_{acq} > time between triggers

FIGURE 3–6. **Trigger Diagram**



Triggering the Unit (Inputs)

The HawkEye™ 1500 requires a trigger to activate scanning. There are two types of triggers that control the unit: physical (hardware) and virtual (software) triggers. The selection of which trigger will be used is highly coordinated with the Application Mode selection provided by ReadRunner. When you select Demo mode or Supermarket mode, the system is virtually triggering itself continuously; once an image is taken and handled, it virtually triggers itself to initiate another sequence as soon as it completes the previous sequence. Therefore, the HawkEye™ 1500 is continuously running and requires no physical triggers to operate. When you select Motion mode or Stop and Scan mode, you are configuring the system such that a physical trigger event must occur to initiate the capture and decode sequence.

Physical Triggers

The input lines allow you to add product sensors or user-defined sources to sense products marked with the Data Matrix symbol. A product passes through the sensor mounted in proximity of the monitored manufacturing line. Once the sensor's path is broken, a trigger signal is sent to the HawkEye™ 1500 via the I/O connector.

The HawkEye™ 1500 receives the signal and tells the camera to capture the upcoming image. Then, the image is decoded and data is sent to the RS-232/Ethernet ports. At the same time, the camera sends signals to the I/O connector as output signals for you. For more information, see "Outputs" on page 3-19.

Typically, this trigger is an external device hardwired to the Opto Isolated pin provided on the HawkEye™ 1500 series connector. (See "Field I/O Connector" on page 2-8.)

When the trigger input is received, the HawkEye™ 1500 will initiate the sequence. If the trigger input is received too fast, such that the HawkEye™ 1500 cannot keep up, the camera will issue a RTE (Real Time Error) condition. This condition is reported in several ways: one is via the user configurable RTE Error message string (see "Output Format Strings" on page 4-71), and via the digital output line that has been assigned to report the "RTE" state of the HawkEye™ 1500 connector (see "Field I/O Connector" on page 2-8), and also to ReadRunner through the internal reporting mechanism (see Chapter 4, "ReadRunner").

Virtual Triggers

When configuring the HawkEye™ 1500 with your external equipment, you can virtually trigger the unit to simulate the Opto Isolated input to further debug and set up. This can be accomplished by depressing the Trigger button provided on ReadRunner (see Chapter 4, "ReadRunner"), or by issuing the following command:

```
'VT 2'

where 2 is the internal number associated with the Opto
Isolated Trigger input.
```

Alternately, you could also have an external piece of equipment issue the VT command at the desired rate in lieu of using the physical trigger.

Additional Physical Triggers Available

In addition to the Opto Isolated trigger provided on Pin pair 1, 11, you could also make use of the 4 TTL inputs to the camera. For more information, see “Field I/O Connector” on page 2-8, and “Advanced I/O” on page 4-54.

Outputs

Once an image is decoded, the data is sent to the RS-232/Ethernet ports. At the same time, the HawkEye™ 1500 sends signals to the I/O connector. The camera can be configured in such a way as to communicate its Read results to an external device via these I/O lines.

The HawkEye™ 1500 camera’s I/O connector has the following I/O lines:

- Pin pair 2, 12 corresponds to Pass signal (default assignment)
- Pin pair 3, 13 corresponds to Fail signal (default assignment)
- Pin pair 4, 14 corresponds to Data Valid signal (default assignment)

The following four I/O Modes are available (shown with their default assignment lines):

- Pass/Fail Only - Full Handshake — Handshake mode with OPTO_OUT1 = Pass, OPTO_OUT2 = Fail.
- Pass/Fail Only - Pipelined — PULSE/PIPELINED mode with OPTO_OUT1 = Pass, OPTO_OUT2 = Fail.
- Data Valid - Full Handshake — Handshake mode with OPTO_OUT1 = Pass, OPTO_OUT2 = Fail, OPTO_OUT3 = Data Valid.
- Data Valid - Pipelined — PULSE/PIPELINED mode with OPTO_OUT1 = Pass | Fail, OPTO_OUT3 = Data Valid.

In addition, with the Verification option, the following I/O modes are also supported:

- DV - 2 Line Verify - Full HS — GOOD, FAIR, POOR multiplexed on two assigned lines plus Data Valid
- DV - 3 Line Verify - Full HS — GOOD, FAIR, POOR, Data Valid, lines are assigned

- DV - 2 Line Verify - Pulse — GOOD, FAIR, POOR multiplexed on two assigned lines plus Data Valid
- DV - 3 Line Verify - Pulse — GOOD, FAIR, POOR, Data Valid, lines are assigned
- None – No external lines are driven to indicate any status

The mode selection can be made via ReadRunner IO Settings display or via serial command:

```
IO_MODE {mode} {duration}
```

For specifics, type:

```
HELP IO_MODE VERBOSE
```

In any of the I/O modes except None, the RTE signal may also be assigned. RTE signals a Real Time Error; in other words, Trigger Overrun on Pin #8 (default assignment on GPIO_OUT3).

In all of the modes, you can configure the polarity of the lines by using the SIGOUT command. In the following description, the term “drive” will be assumed to be in the direction you define. In addition to the I/O lines being driven, the results are always presented on the camera status LEDs.

If the Verification option is being used, the camera status LEDs translate to:

- GOOD = PASS LED
- FAIR = PASS + FAIL LED
- POOR = FAIL LED

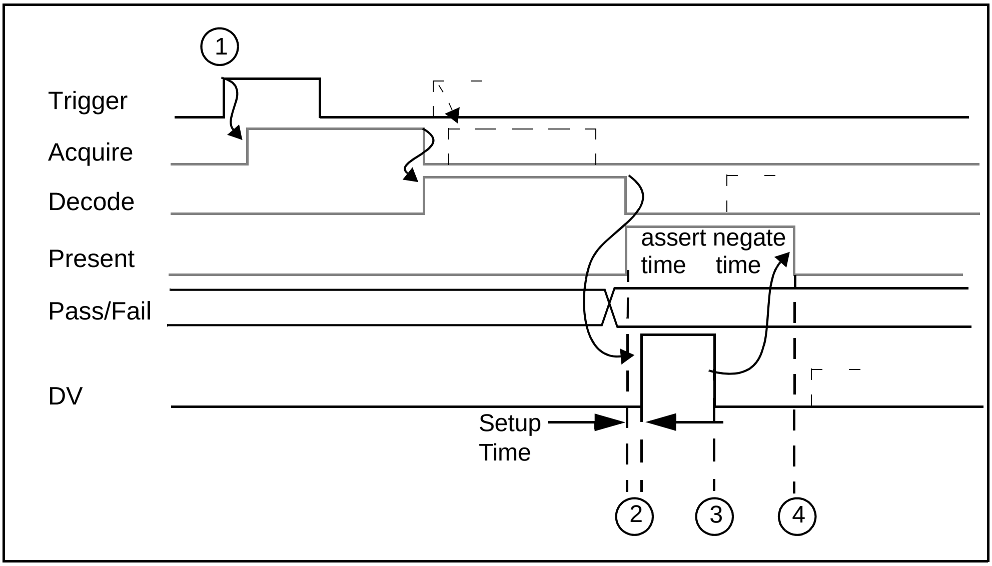
Data Valid — Pipelined

Note: The discussion of pin pairs assumes a default assignment of the I/O lines.

The Data Valid - Pipelined Mode drives pin pair 2, 12 when the results of the Decode are a success. It drives the same line to the opposite direction when the results of the Decode are a failure. It then drives pin pair 4, 14 for the pre-determined duration to indicate that the level on pin pair 2, 12 is valid. The line will be set to the polarity for an indicated duration. You can configure the duration of the signal by using the IO_MODE command.

SIGOUT used to have a delay that was not used by the code. It was left in as an option in the case of RTE to avoid rewriting all of the regression test. Also, SIGOUT applies to the line state and not the variable assigned to the line.

FIGURE 3–7. Data Valid - Pipelined Mode



Notes:

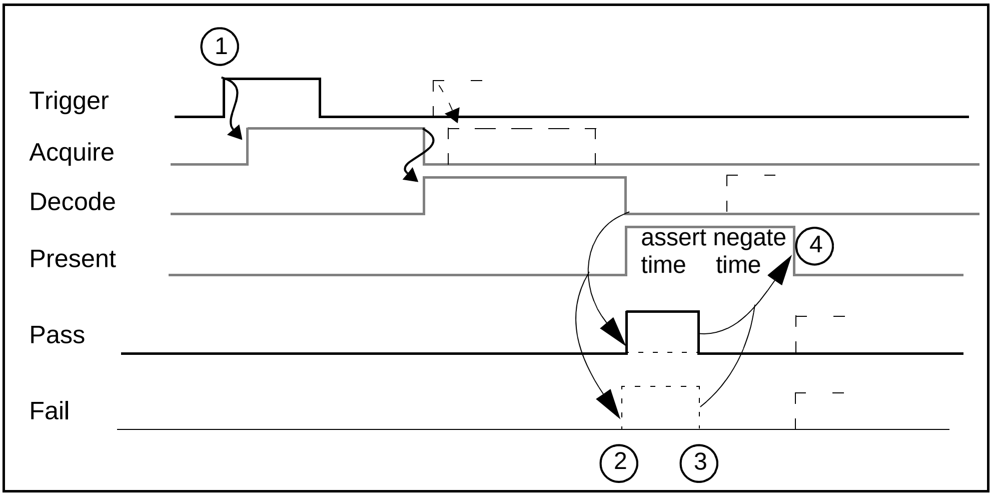
1. User asserts Trigger (user can re-trigger after acquisition is complete).
2. When read is complete, reader sets Pass/Fail and asserts DV after 2 ms set-up time.
3. Reader negates DV after user programmed “signal duration” time (new decode can start).
4. Reader can assert DV again only after an additional “signal duration time.”

Pass/Fail Only — Pipelined

Note: The discussion of pin pairs assumes a default assignment of the I/O lines.

The Pass/Fail Only - Pipelined Mode drives pin pair 2, 12 when the results of the Decode are a success and, at the same time, drives pin pair 3, 13 to the opposite direction. When the results of the Decode are a failure, it drives pin pair 3, 13 and, at the same time, drives pin pair 4, 14 to the opposite direction. The line will be set to the polarity for an indicated duration. You can configure the duration of the signal by using the IO_MODE command. Figure 3–8 depicts a Trigger occurring, which initiates an Image acquisition, which in turn is passed on to the Decoder for analysis. Once the outcome is determined, the Pass or Fail is presented on the I/O lines for the desired duration.

FIGURE 3–8. Pass/Fail Only - Pipelined



Notes:

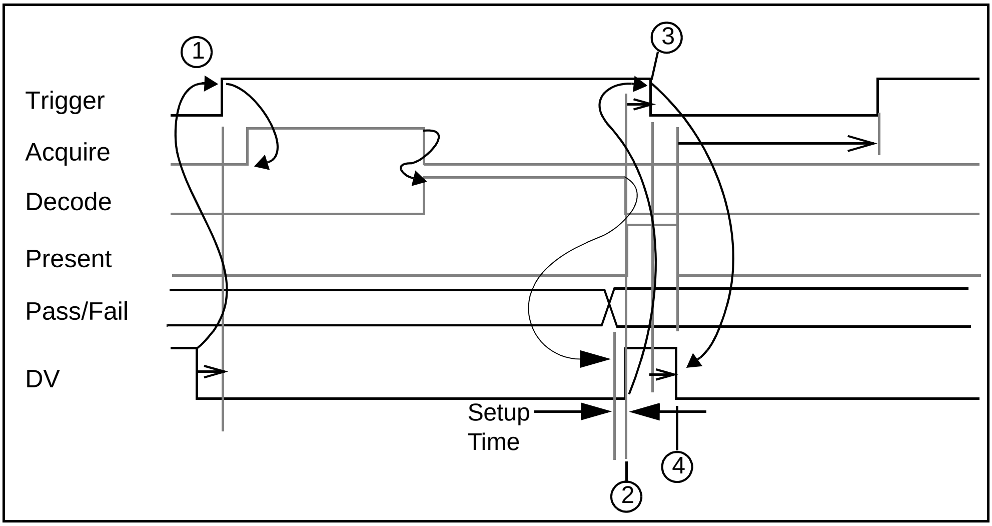
1. User asserts Trigger (user can re-trigger after acquisition is complete).
2. When read is complete, reader sets either Pass or Fail.
3. Reader negates Pass or Fail after user programmed “signal duration” time (new decode can start).
4. Reader can assert Pass or Fail again only after an additional “signal duration time.”

Data Valid — Full Handshake

Note: The discussion of pin pairs assumes a default assignment of the I/O lines.

The Data Valid - Full Handshake Mode drives pin pair 2, 12 when the results of the Decode are a success. It drives the same line to the opposite direction when the results of the Decode are a failure. It then drives pin pair 4, 14 to indicate that pin pair 2, 12 is valid and represents pass or fail. The line will be set to the desired polarity and will stay at that polarity until the next trigger initiates a new acquisition. Figure 3–9 depicts a Trigger occurring, which initiates an Image acquisition, which in turn is passed on to the Decoder for analysis. Once the outcome is determined, the Pass or Fail is presented on the I/O line until the next trigger occurs.

FIGURE 3–9. Data Valid - Full Handshake



Notes:

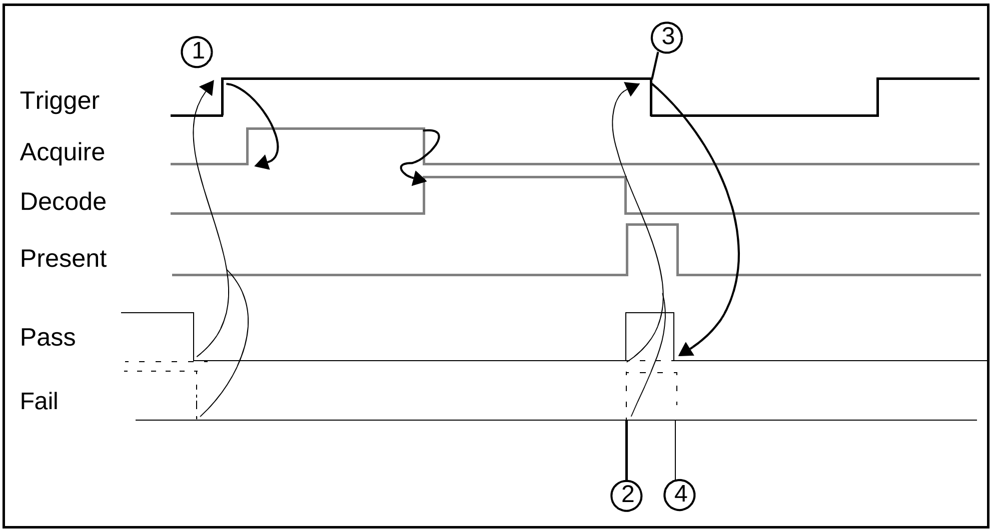
1. When DV is false, you can start read by asserting TRIG.
2. When read is completed, reader sets Pass/Fail and asserts DV after 2ms setup time.
3. User accepts Pass/Fail result and then negates TRIG.
4. Reader detects TRIG negated and then negates DV.

Pass/Fail Only — Full Handshake

Note: The discussion of pin pairs assumes a default assignment of the I/O lines.

The Pass/Fail Only - Full Handshake Mode drives pin pair 2, 12 when the results of the Decode are a success and, at the same time, drives pin pair 3, 13 to the opposite direction. When the results of the Decode are a failure, it drives pin pair 3, 13 and, at the same time, drives pin pair 2, 12 to the opposite direction. The line will be set to the desired polarity and will stay at that polarity until the next trigger initiates a new acquisition. Figure 3–10 depicts a Trigger occurring, which initiates an Image acquisition, which in turn is passed on to the Decoder for analysis. Once the outcome is determined, the Pass or Fail is presented on the I/O lines until the next trigger occurs.

FIGURE 3–10. Pass/Fail Only - Full Handshake



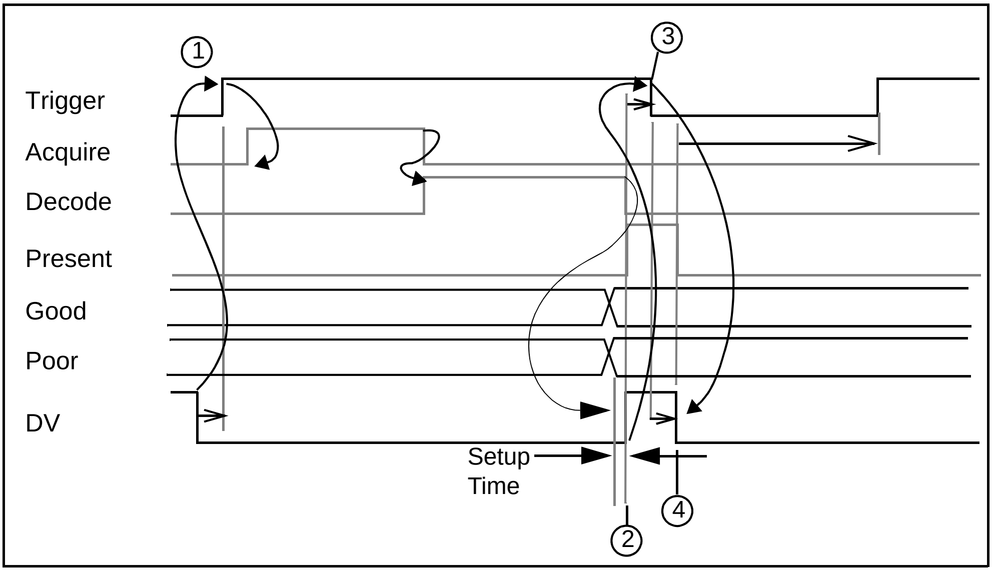
Notes:

1. When both Pass and Fail are false, you can start read by asserting TRIG.
2. When read is completed, reader sets either Pass or Fail.
3. User accepts Pass/Fail result and then negates TRIG.
4. Reader detects TRIG negated and then negates both Pass and Fail.

DV - 2 Line Verify — Full HS

The DV 2 Line Verify - Full HS Mode drives the Good and Poor lines when the results of the verification are completed (see Table 3–2 for the line states of Good, Fair, Poor, and Fail). This mode then drives the Data Valid line to indicate that the Good and Poor lines are valid and stable. The lines will be set to the desired polarity and will stay at that polarity until the next trigger initiates a new acquisition. Figure 3–11 depicts a Trigger occurring, which initiates an Image acquisition, which in turn is passed on to the Decoder for analysis. Once the outcome is determined, the Good, Fair, Poor, or Fail is presented on the I/O lines until the next trigger occurs.

FIGURE 3–11. DV - 2 Line Verify - Full HS



Notes:

1. When DV is false, you can start read by asserting TRIG.
2. When read is completed, reader sets the Good / Fair / Poor / Fail state and asserts DV after 2ms setup time.
3. User accepts the Good / Fair / Poor / Fail result and then negates TRIG.

4. Reader detects TRIG negated and then negates DV.

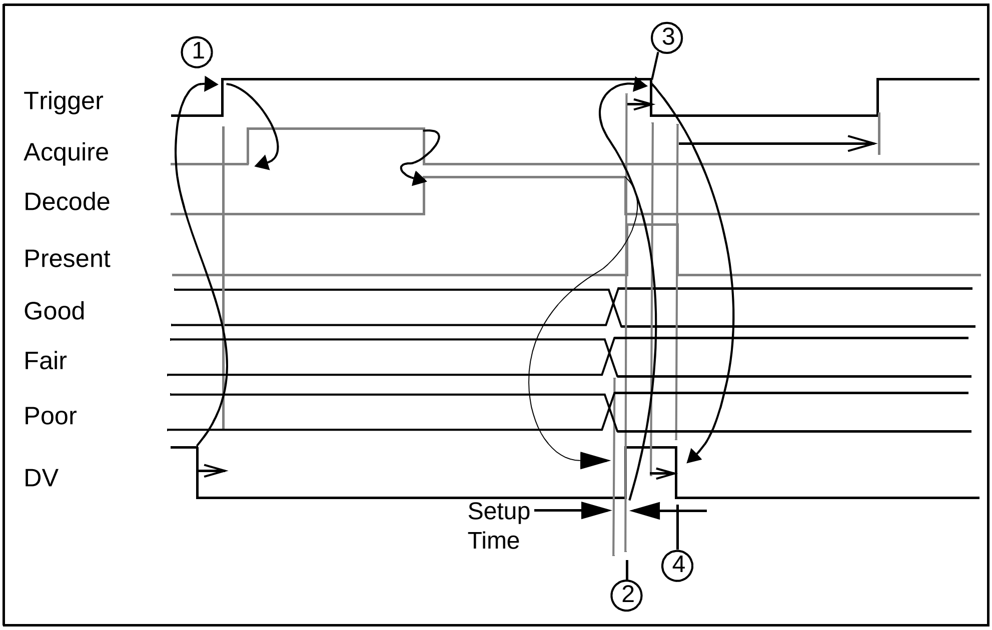
TABLE 3–2. Two Line Multiplex of Verification Results

	Line Keyword Name	
State	**Good**	**Poor**
Good	On	On
Fair	On	Off
Poor	Off	On
Fail	Off	Off

DV - 3 Line Verify — Full HS

The DV 3 Line Verify - Full HS Mode drives the Good, Fair, and Poor lines when the results of the verification are completed (see Table 3–3 for the line states of Good, Fair, Poor, and Fail). This mode then drives the Data Valid line to indicate that the Good, Fair, and Poor lines are valid and stable. The lines will be set to the desired polarity and will stay at that polarity until the next trigger initiates a new acquisition. Figure 3–12 depicts a Trigger occurring, which initiates an Image acquisition, which in turn is passed on to the Decoder for analysis. Once the outcome is determined, the Good, Fair, Poor, or Fail is presented on the I/O lines until the next trigger occurs.

FIGURE 3–12. DV - 3 Line Verify - Full HS



Notes:

1. When DV is false, you can start read by asserting TRIG.
2. When read is completed, reader sets the Good / Fair / Poor / Fail state and asserts DV after 2ms setup time.
3. User accepts the Good / Fair / Poor / Fail state result and then negates TRIG.
4. Reader detects TRIG negated and then negates DV.

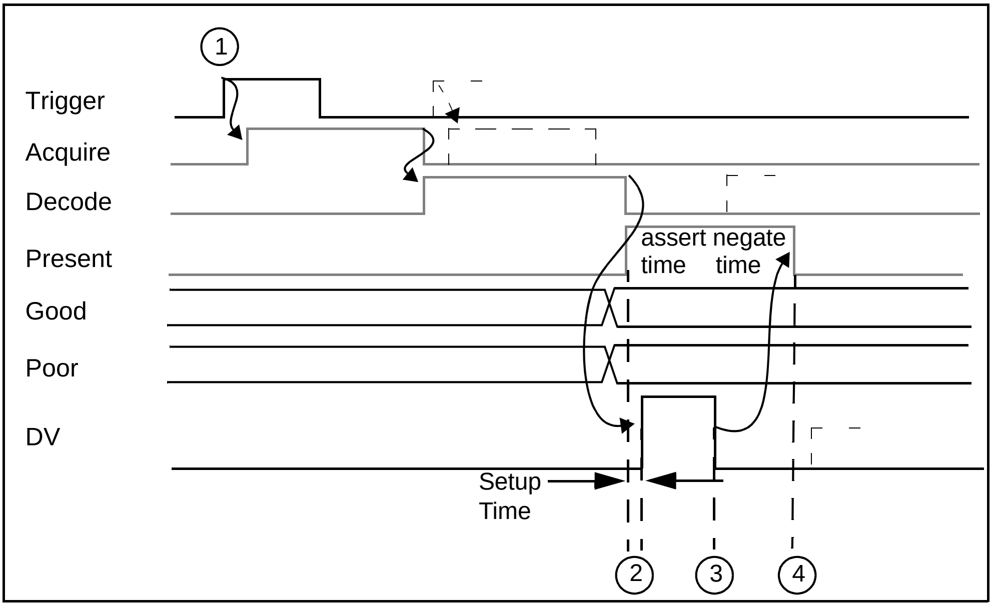
TABLE 3–3. Three Line Multiplex of Verification Results

	Line Keyword Name		
State	**Good**	**Fair**	**Poor**
Good	On	Off	Off
Fair	Off	On	Off
Poor	Off	Off	On
Fail	Off	Off	Off

DV - 2 Line Verify — Pulse

The Data Valid - Pipelined Mode drives the Good and Poor lines when the results of the verification are completed (see Table 3–4 for the line states of Good, Fair, Poor, and Fail). This mode then drives the Data Valid line for the pre-determined duration to indicate that the level on the Good and Poor lines is valid. The line will be set to the polarity for an indicated duration. You can configure the duration of the signal by using the IO_MODE command.

FIGURE 3–13. DV - 2 Line Verify - Pulse



Notes:

1. User asserts Trigger (user can re-trigger after acquisition is complete).
2. When read is complete, reader sets the Good / Fair / Poor / Fail state and asserts DV after 2 ms set-up time.
3. Reader negates DV after user programmed “signal duration” time (new decode can start).

4. Reader can assert DV again only after an additional “signal duration time.”

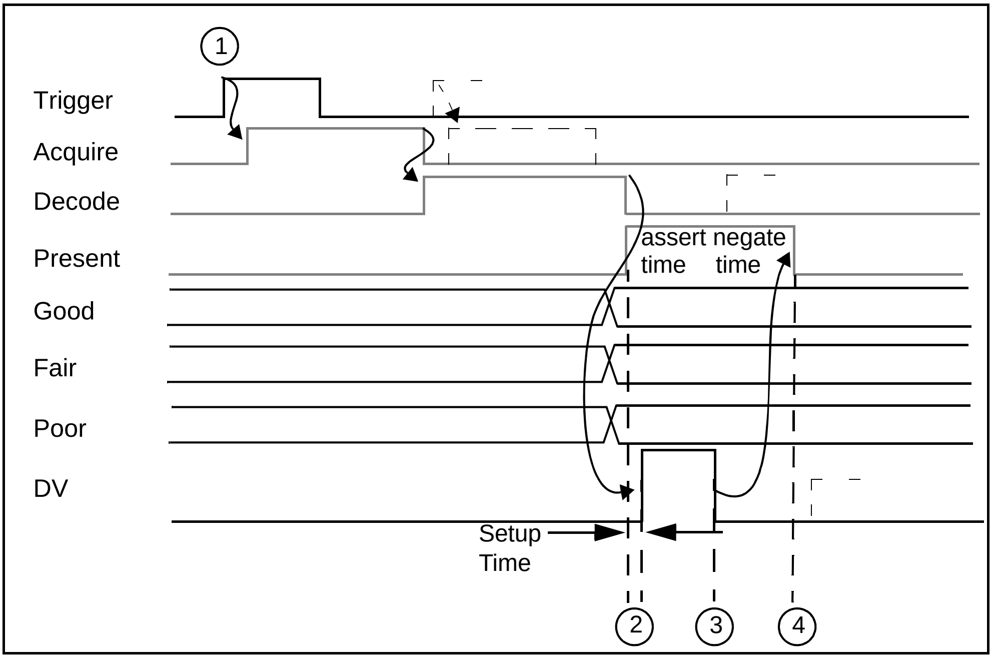
TABLE 3–4. Two Line Multiplex of Verification Results

	Line Keyword Name	
State	**Good**	**Poor**
Good	On	On
Fair	On	Off
Poor	Off	On
Fail	Off	Off

DV - 3 Line Verify — Pulse

The Data Valid - Pipelined Mode drives the Good, Fair, and Poor lines when the results of the verification are completed (see Table 3–5 for the line states of Good, Fair, Poor, and Fail). This mode then drives the Data Valid line for the predetermined duration to indicate that the Good, Fair, and Poor lines are valid and stable. The line will be set to the polarity for an indicated duration. You can configure the duration of the signal by using the IO_MODE command.

FIGURE 3–14. **DV - 3 Line Verify - Pulse**



Notes:

1. User asserts Trigger (user can re-trigger after acquisition is complete).
2. When read is complete, reader sets the Good / Fair / Poor / Fail state and asserts DV after 2 ms set-up time.
3. Reader negates DV after user programmed “signal duration” time (new decode can start).
4. Reader can assert DV again only after an additional “signal duration time.”

TABLE 3–5. Three Line Multiplex of Verification Results

	Line Keyword Name		
State	**Good**	**Fair**	**Poor**
Good	On	Off	Off
Fair	Off	On	Off
Poor	Off	Off	On
Fail	Off	Off	Off

Formatted Output & Audio

The HawkEye™ 1500 supports formatted output via the serial port as well as through TCP/IP. You can configure the camera to output custom text, optionally with headers and footers, based on pass/fail status and even specific failure status. This data is then transmitted to the serial port and any TCP connections on port 49098.

The camera supports different pass/fail criteria for outputting data. This includes Decode Pass, Decode Fail, Locate Fail, Match Pass, Match Fail, and Runtime Error. Runtime errors include trigger overruns, decoder overruns, acquisition timeouts, or network connection errors.

Each type of output has an associated format string. Within this string, you can output specific characters (ABCD, 1234, etc.), or non-printable characters (CR, LF, TAB).

Non-printable characters are denoted by using a backslash (“\”) followed by the hexadecimal value of the non-printable. Some common non-printable characters include:

- \00 — NULL character
- \09 — TAB character (TAB)
- \0A — Line Feed (LF)
- \0C — Form Feed (FF)
- \0D — Carriage Return (CR)

You can also use keywords to expand specific pieces of data within the output string. These keywords include: ANGLE, CHECKSUM, DATA, DATAHEX, DETAILED, PID#, TIMESTAMP, VERI_1_IAQG, VERI_AIMDPM, VERI_DETAIL, VERI_FORMATTED, VERI_GRADE, VERI_STATUS, UII_ONLY, and UII_WITH_INFO. For detailed information about these keywords, see “Format String Keywords” on page 4-77.

Reported Error Codes

Table 3–6 contains possible error codes.

TABLE 3–6. Report Error Codes

Error Number	Error Description
0	"Decode Successful"
3048	"Unable to decode"
4601	"No edge candidate found"
4604	"First edge not found"
4605	"Second edge not found"
4611	"Third edge not found"
4612	"Fourth edge not found"
4620	"Four corners not found"
4621	"Size varying too much"
4622	"Row / Column test failed"
4630	"Inspection timeout"
4633	“Border match test failed”
4650	"Verification contrast calibration failure"
4651	"Verification cellunit calibration failure"
4700	"Verification process error"
4701	"Verification unsupported"
4702	"Verification timeout"
4710	"ISO Verification ECC200 required"
4711	"ISO Verification aperture too small"
4712	"ISO Verification aperture too large"
4713	"ISO Verification insufficient space"
4714	"ISO Verification failed RDA STEP F 1"

TABLE 3–6. Report Error Codes (Continued)

Error Number	Error Description
4715	"ISO Verification failed RDA STEP F 2"
4716	"ISO Verification failed RDA STEP F 3"
4717	"ISO Verification failed RDA STEP A E"
4718	"ISO Verification failed final image adjustment"
4719	"ISO Verification failed RDA error correction"
4720	"ISO Verification DM row/column too large"
4721	"ISO Verification cell smaller than Min x"
4722	"ISO Verification cell larger than Max x"
6001	"QR Code general error"
6002	"QR Code ratio error"
6003	"QR Code finder error"
6004	"QR Code line fit error"
6005	"QR Code line intersect error"
6006	"QR Code corner error"
6007	"QR Code boundary error"
6008	"QR Code length ratio error"
6020	"QR Code decode unknown error"
6021	"QR Code rs level invalid"
6022	"QR Code format info failed"
6023	"QR Code version info failed"
6024	"QR Code rows cols invalid"
6025	"QR Code data codewrd invalid"
6026	"QR Code total codeword invalid"
6027	"QR Code mode indicator invalid"
6028	"QR Code mode unimplemented"
6029	"QR Code rs decode failed"
6030	"QR Code BCH15 5 undecodable"
6031	"QR Code model invalid"
6032	"QR Code timeout error"
6100	"QR Code design unimplemented"
6101	“NO UID ENABLE”

TABLE 3–6. Report Error Codes (Continued)

Error Number	Error Description
6102	"HEADER 1ST POSITION ERROR"
6103	"HEADER 2RD POSITION ERROR"
6104	"HEADER 3RD POSITION ERROR"
6105	"HEADER 4TH POSITION ERROR"
6106	"HEADER GS ERROR"
6107	"FORMAT INDICATOR ERROR"
6108	"TRAILER RS ERROR"
6109	"TRAILER EOT ERROR"
6110	"DOUBLE TRAILER ERROR"
6111	"DATA ELEMENT SEPARATOR ERROR"
6140	"SOFTWARE ERROR 1"
6190	"DATA QUALIFIER SPACE"
6192	"TEI DATA QUALIFIER"
6194	"TEI SPACE ERROR"
6195	"WRONG FORMAT INDICATOR"
6200	"DATA QUALIFIER ERROR"
6201	"NEED UII ELEMENT FIRST"
6202	"NO UII ELEMENT ERROR"
6203	"UII ELEMENT INCOMPLETE"
6204	"DATA ELEMENT CHARACTER"
6205	"LOWER CASE CHARACTER"
6206	"DATA ELEMENT TOO SHORT"
6207	"DATA ELEMENT TOO LONG"
6208	"ENTERPRISE NUM TOO LONG"
6209	"ORIGINAL PARTNUM TOO LONG"
6210	"SERIAL NUM TOO LONG ERROR"
6211	"REQUEST ELEMENT NUM ERROR"
6212	"UII TOO LONG"
6213	"SOFTWARE ERROR 2"
6301	"PNR TOO LONG"
6302	"PNR TOO SHORT"

TABLE 3–6. Report Error Codes (Continued)

Error Number	Error Description
6303	"PNR CHARACTER ERROR"
6311	"30P TOO LONG"
6312	"30P TOO SHORT"
6313	"30P CHARACTER ERROR"
6321	"240 TOO LONG"
6322	"240 TOO SHORT"
6323	"240 CHARACTER ERROR"
6331	"30T TOO LONG"
6332	"30T TOO SHORT"
6333	"30T CHARACTER ERROR"
6399	"NO CURRENT PART NUMBER"

QuicSet® Symbol Photometry

When the HawkEye™ is set up in triggered mode, Auto Photometry may not be appropriate. When the Photometry values need to be known ahead of time because the time cannot be spent determining Gain and Exposure values, the QuicSet® Symbol Photometry function can be invoked to determine the correct Gain and Exposure to be used for the part.

Perform **one** of the following procedures:

Procedure 1

If the part to be read is aligned, then follow this procedure:

1. Insert a paper clip into the hole marked QuicSet® and press once. Any time after hearing a beep, press the QuicSet button again to exit. The Photometry value (Gain and Exposure) have been determined.

OR

Procedure 2

If the part to be read is not aligned, then follow this procedure:

1. Insert a paper clip into the hole marked QuicSet® and press once.

2. Ensure that the yellow Mode light is flashing. Physically position the HawkEye™ 1500 until the laser dots are almost together for small Data Matrix symbols, or slightly farther apart on the horizontal axis for bar code symbols.

DO NOT LOOK AT THE LASER. WHEN READING PARTS THAT ARE HIGHLY REFLECTIVE, BE CAREFUL THAT THE LASER IS NOT REFLECTED INTO SOMEONE'S EYE.

When the laser beams are centered on the symbol, a series of beeps will be heard. The beeps have three tones:

– Low — Signals that the symbol is in the field of view

– Middle — Signals that the symbol is close to optimal read position

– High — Signals the optimal position

The three LEDs above the blinking Mode LED correspond to the three beeps.

– With poor alignment, there are no LEDs flashing and there is no beeping.

– As the alignment improves, the LEDs begin to illuminate, from bottom to top, and beeping begins.

– At optimal alignment, all the LEDs are flashing and beeping is at its highest intensity.

3. Slightly move the HawkEye™ 1500 until the high intensity beep is achieved, lock down the camera's position, then press the QuicSet® button once to exit QuicSet® mode.

You will hear three short beeps and the Mode LED will return to steady on. This indicates that the HawkEye™ is ready to read using the QuicSet® mode settings.

4. Insert a paper clip into the hole marked QuicSet® and press once. Any time after hearing a beep, press the QuicSet button again to exit. The Photometry value (Gain and Exposure) have been determined.

CHAPTER 4

ReadRunner

This chapter describes how to set up and use ReadRunner, the HawkEye™ 1500 Graphical User Interface. It also contains information about fine tuning and monitoring your application.

Setting Up Communications

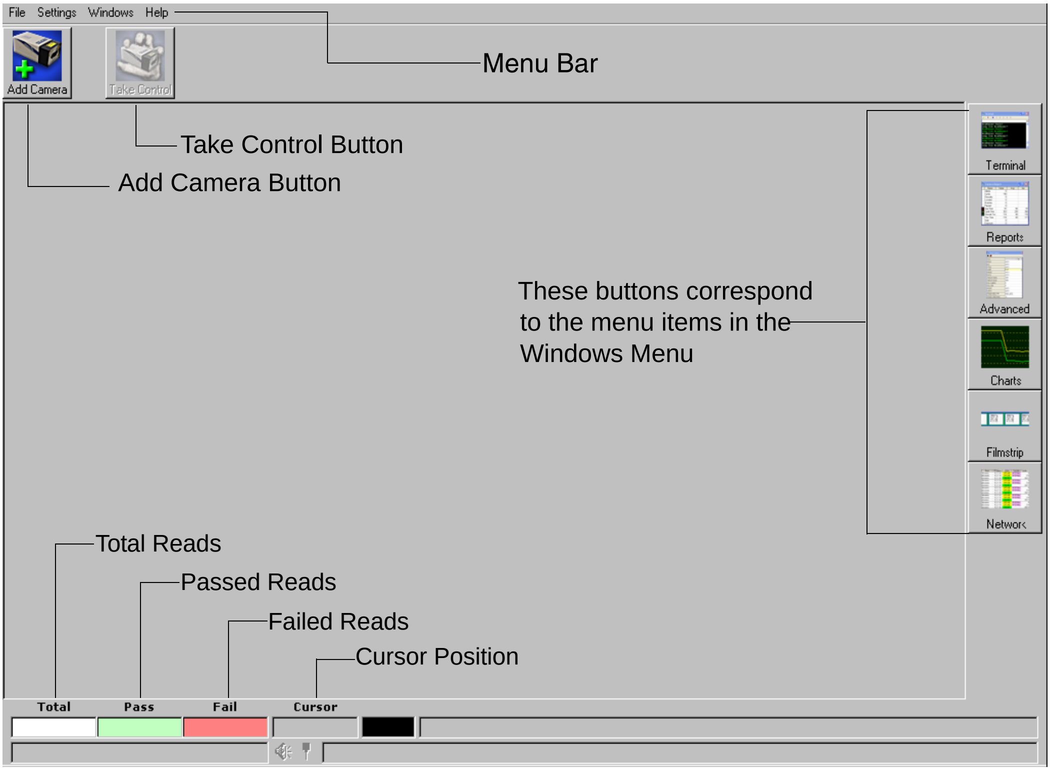
If you haven't already set up communications with the HawkEye™ 1500 Series Smart Camera-Based Reader, see the HawkEye™ 1500 Series Quick Start Guide.

Overview

ReadRunner is the HawkEye™ 1500 Graphical User Interface. From ReadRunner, you can access all the features and functionality of the HawkEye™ 1500 camera and monitor/control any camera on an Ethernet network.

Figure 4–1 displays the main ReadRunner window.

FIGURE 4–1. **ReadRunner Main Window**



ReadRunner contains menus and buttons that enable you to quickly and easily connect and set up the camera, and monitor and display information.

ReadRunner Menus

The ReadRunner menu bar contains the following menus:

- **File Menu** contains the following menu items:
  - Save Parameters on Camera (Ctrl+S), see page 4–89
  - Restore Defaults — Decoder, see page 4–89
  - Restore Defaults — Application Mode, see page 4–89
  - Restore Defaults — Both (Ctrl+Z), see page 4–90

- **Settings Menu** contains the following menu items:

- **Windows Menu** contains the following menu items:

- Advanced Tuning (Ctrl+A), see page 4–104
- Charting Window (Ctrl+H), see page 4–94
- Filmstrip Recorder (Ctrl+F), see page 4–131
- Network Overview (F12), see page 4–96
- Verification Report (F2), see page 4–90

• **Help Menu** contains the following menu item:

- About ReadRunner, see page 4–98

ReadRunner Shortcut Keys

Table 4–1 contains the ReadRunner shortcut keys.

TABLE 4–1. ReadRunner Shortcut Keys

Key	Description
Ctrl+A	Advanced Tuning, see page 4–104
Ctrl+B	Beep Control, see page 4–88
Ctrl+F	Filmstrip Recorder, see page 4–131
Ctrl+F6	IO Settings, see page 4–54
Ctrl+G	Output Settings, see page 4–71
Ctrl+H	Charting Window, see page 4–94
Ctrl+I	Load Image Files To Camera, see page 4–129
Ctrl+J	Serial/TCP Settings, see page 4–64
Ctrl+M	Application Mode, see page 4–46
Ctrl+O	Match Mode Settings, see page 4–29
Ctrl+P	Photometry, see page 4–22
Ctrl+Q	Use Camera for Acquisition, see page 4–130
Ctrl+R	Report Window, see page 4–15
Ctrl+S	Save Parameters on Camera, see page 4–89
Ctrl+T	Terminal Window, see page 4–91
Ctrl+U	Configure Part Queue, see page 4–120
Ctrl+V	Symbology and Verification, see page 4–25

TABLE 4–1. ReadRunner Shortcut Keys (Continued)

Key	Description
Ctrl+Z	Restore Defaults to Decoder & Application Mode, see page 4–89
F2	Verification Report, see page 4–96
F7	Save Camera Configuration To File, see page 4–18
F8	Load Camera Configuration To Camera, see page 4–18
F9	Save Current Image, see page 4–129
F12	Network Overview, see page 4–96
Shift+F5	Toggle Beeper, see page 4–87

ReadRunner Buttons

Just below the menu bar are two buttons:

- Add Camera — This button connects the camera you specify and adds a button containing the name of the newly-added camera. This shortcut makes it easy for you to move from one camera to another.

  `At boot time, if a camera's network name is "HawkEye," the camera automatically changes "HawkEye" to "HawkEyexxyyzz" where xxyyzz is the last three fields of the camera's MAC address. For example:`

TABLE 4–2. Unique Camera Names

Camera Name at Boot Time	MAC Address	Unique Camera Name
HawkEye	00:60:33:e0:00:11	HawkEyeE00011
HawkEye	00:60:33:e0:00:12	HawkEyeE00012
HawkEye	00:60:33:e0:00:13	HawkEyeE00013
HawkEye	00:60:33:e0:00:14	HawkEyeE00014
HawkEye	00:60:33:e0:00:15	HawkEyeE00015

- Take Control — This button denies other users access to a camera while you set it up and specify parameters for a job. After you take control of a camera, the Take Control button becomes the Release button.

By default, these two buttons appear on the main ReadRunner window.

Table 4–3 contains descriptions of other buttons that are common to ReadRunner.

TABLE 4–3. ReadRunner Buttons

Button	Description
	This is the Add Camera button. Click it to display a list of cameras that are currently on the network. For more information about adding a camera, see “Adding & Taking Control of a Camera” on page 4-7.
	This button indicates that the camera sqa001 has been added, but not selected (blue background and broken connection), and the connection is over TCP/IP.
	This button indicates that the camera sqa001 has been added and selected (green background and unbroken connection), and the connection is over TCP/IP.
	This button indicates that the camera has been added, but not selected (blue background), and that the connection is a serial connection.
	This button indicates that the camera has been added and selected (green background), and that the connection is a serial connection.
	This is the Take Control button. Click it to control a camera to adjust camera position, modify parameters, or reset statistics. For more information about controlling a camera, see “Adding & Taking Control of a Camera” on page 4-7.
	This button indicates that the camera sqa001 is being controlled by a user.
	This is the Release button. Click it to release control of a camera. For more information about releasing a camera, see “Releasing Control of a Camera” on page 4-21.

Setting Up Your Application

The procedures in this section explain:

- "Adding & Taking Control of a Camera" on page 4-7
- "Using Live Video to Align the Camera" on page 4-12
- "Using Learn During Image Optimization" on page 4-14
- "Displaying Camera Report Information" on page 4-15
- "Saving & Loading Configuration Files" on page 4-18
- "Releasing Control of a Camera" on page 4-21
- "Removing a Camera" on page 4-21

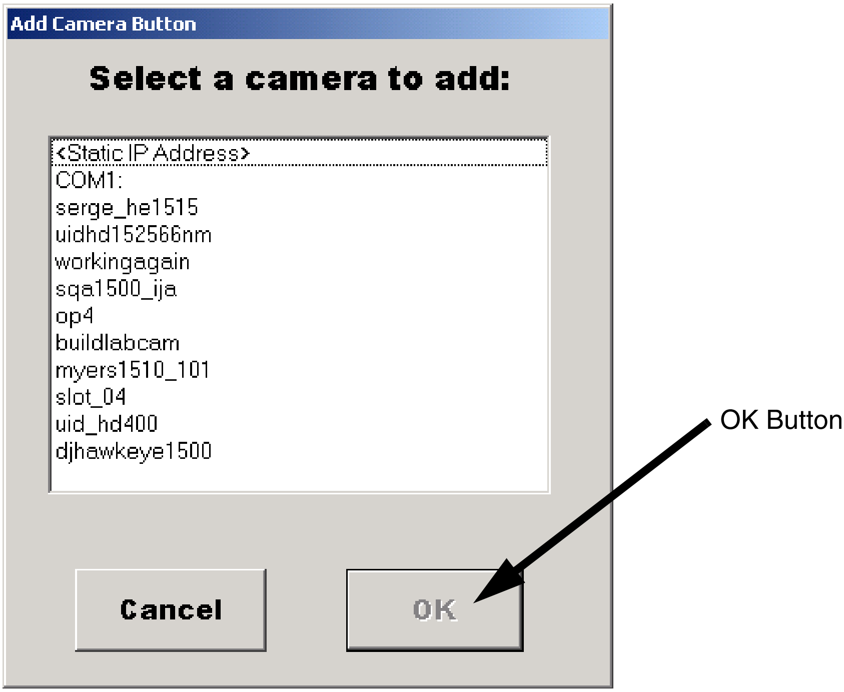
Adding & Taking Control of a Camera

Because the HawkEye™ 1500 camera is a network device, you must connect to it before adjusting parameters or displaying detailed information. To add (connect) a camera and take control of it:

1. Start ReadRunner by selecting Start, Programs, ReadRunner V2.4, and ReadRunner. The main ReadRunner window is displayed.
2. Click 

   ReadRunner displays the Add Camera Button window, as shown in Figure 4-2.

FIGURE 4–2. **Add A Camera Window**



3. Highlight the camera you want to add.

Note: In Figure 4–2, COM1 and COM2 are the COM ports on your PC. If a HawkEye™ 1500 camera is attached to COM1 of your PC, select COM1. If a HawkEye™ 1500 camera is attached to COM2 of your PC, select COM2.

If the camera is on a different subnet, see “Adding a Camera That is on a Different Subnet” on page 4-11.

---

4. Click OK.

ReadRunner adds a new camera button to the left of the Add Camera button, as shown in Figure 4–3.

FIGURE 4–3. New Camera Button Added

Result of “Add Camera” Button

5. Select the camera you want to control.

6. Click 

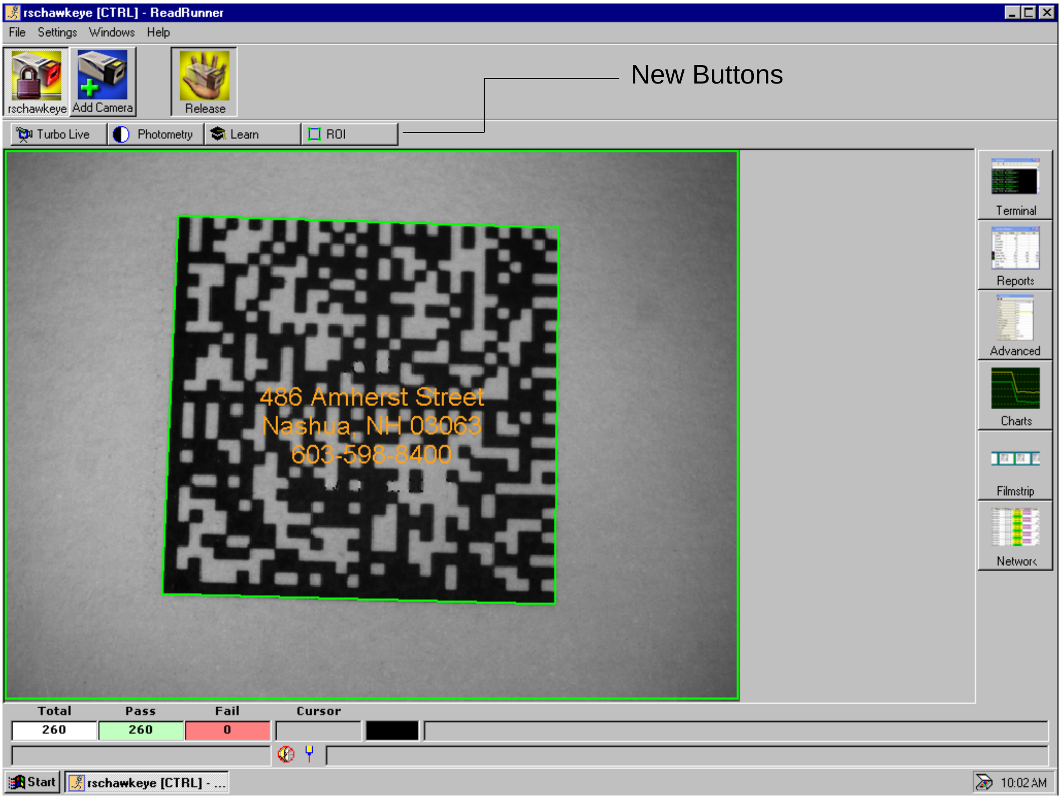
   `ReadRunner displays the following screen:`

FIGURE 4–4. “Do you wish to control the camera...” Window

7. Click Yes when asked if you wish to control camera sqa001.

   `ReadRunner displays a screen similar to the one in Figure 4–5.`

FIGURE 4–5. **Taking Control Window with New Buttons Displayed**



Notice that, after you take control of a camera, the Camera Selected button

becomes the Camera Locked button.

Also notice the new buttons displayed on the screen: Turbo Live, Photometry, Learn, and ROI.

Note: You must have control of the camera to set or modify parameters. This is assumed throughout the procedures in this chapter.

Adding a Camera That is on a Different Subnet

ReadRunner allows you to add a camera that is on a different subnet, assuming you know the IP address of that camera.

Note: The camera name of a camera that is on a different subnet will not appear in the Network Overview window.

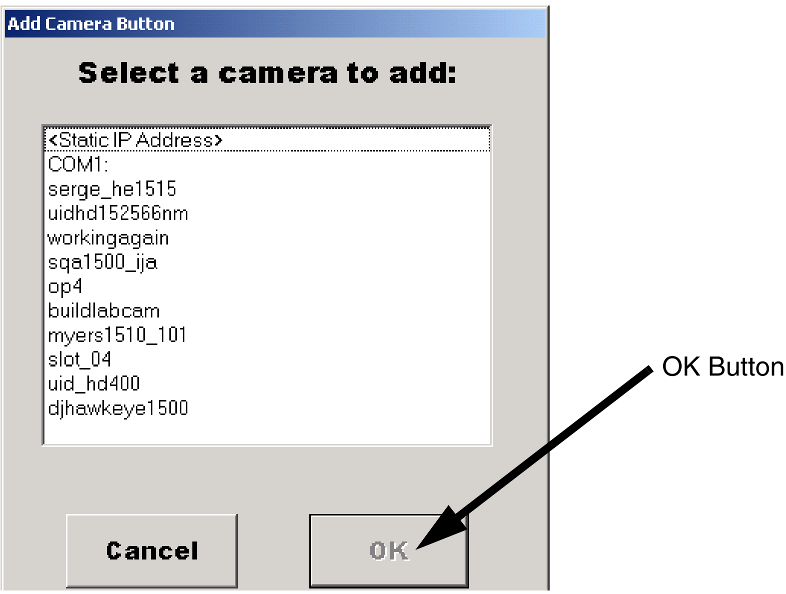
Use the following procedure to add a camera on another subnet:

1. Start ReadRunner by selecting Start, Programs, ReadRunner V#.#, and ReadRunner. The main ReadRunner window is displayed.

2. Click 

   ReadRunner displays the Add Camera Button window, as shown in Figure 4-6.

FIGURE 4–6. Add A Camera Window

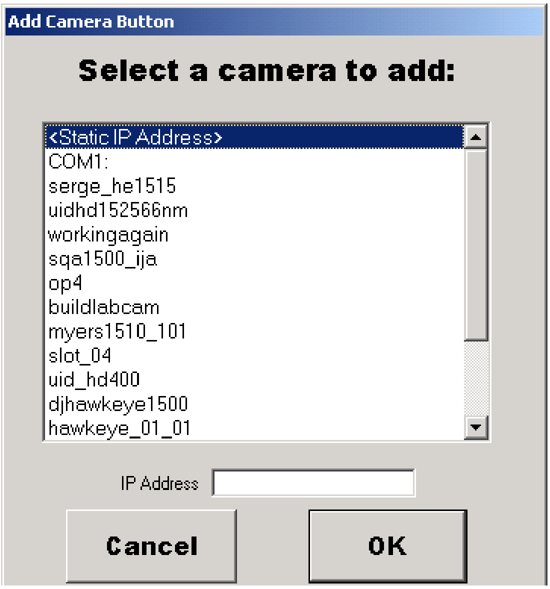


3. Select <Static IP Address>.

4. Click OK.

ReadRunner displays the Add Camera Button window with the IP Address field displayed, as shown in Figure 4–7.

FIGURE 4–7. Add A Camera Window



5. Enter the IP address of the camera and click OK.

Using Live Video to Align the Camera

After adding and controlling a camera, you must align and adjust its position. To align the camera:

1. Place a symbol in the camera's field of view.

2. Click 

to begin aligning and adjusting the camera.

Note: Clicking Turbo Live does not turn on the laser.

The camera stops decoding and increases the rate at which it takes pictures, which helps you to align and adjust the camera position.

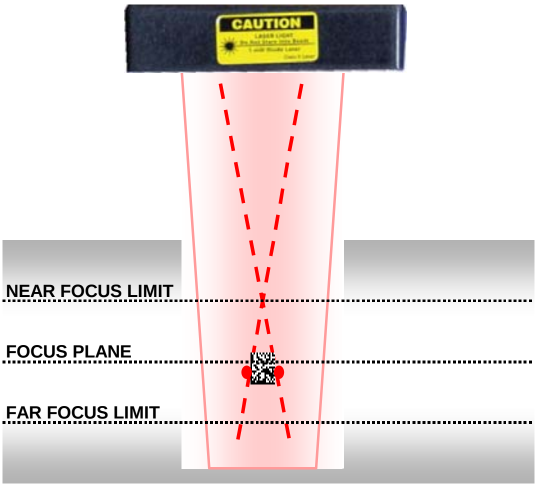
The camera should be correctly aligned and adjusted when the two laser dots are close together on the symbol.

Note: The point where the laser dots converge is when the camera is at “near focus.” (See Figure 4–8.) Moving the camera away from the symbol slightly will improve the focus.

Remember to keep the 20° tilt angle (HawkEye™ 1515) to the symbol surface to eliminate reflections.

DO NOT LOOK AT THE LASER. WHEN READING PARTS THAT ARE HIGHLY REFLECTIVE, BE CAREFUL THAT THE LASER IS NOT REFLECTED INTO SOMEONE’S EYE.

FIGURE 4–8. **Laser Targeting**



3. After you finish aligning and adjusting the camera, click 

DO NOT LOOK AT THE LASER. WHEN READING PARTS THAT ARE HIGHLY REFLECTIVE, BE CAREFUL THAT THE LASER IS NOT REFLECTED INTO SOMEONE’S EYE.

Note: HE1500-XL do not have lasers.

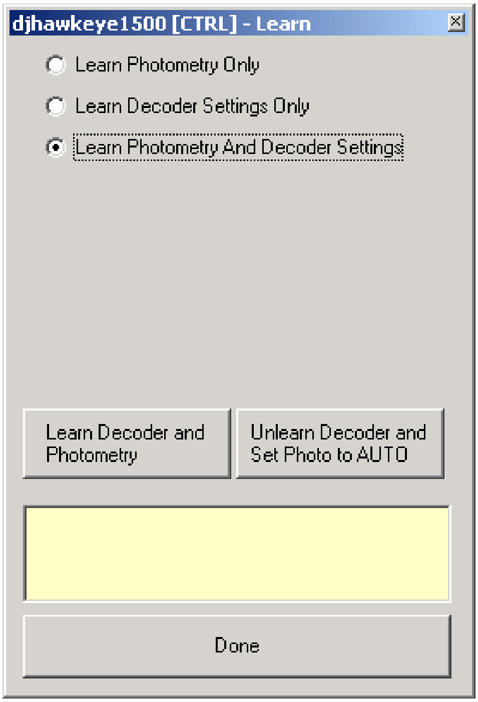
Note: When Verification is set to AIM DPM-1-2006, selecting Turbo Live will display concentric squares as video overlay to the live images. Live Video will perform automatic exposure control to improve the contrast of the image.

Using Learn During Image Optimization

Use the following procedure to learn photometry and decoder settings:

1. Click Learn to begin learning photometry and decoder settings. ReadRunner displays the Learn window, as shown in Figure 4–9.

FIGURE 4–9. Learn Window



2. Select Learn Photometry And Decoder Settings.

3. Click Learn Decoder and Photometry and then Done.

```
The camera learns the photometry and decoder settings.
```

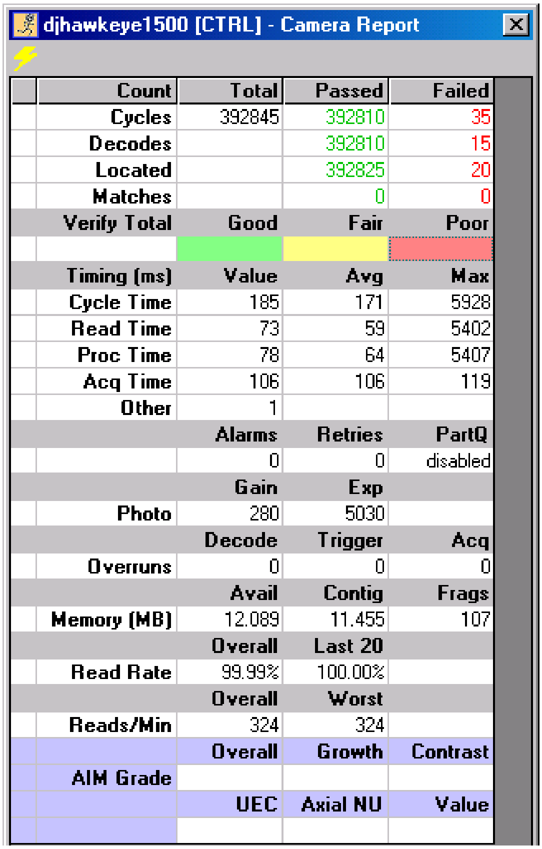
For more information about Learn, Unlearn, and Assisted Learn, see “Learning & Unlearning” on page 4-101.

Displaying Camera Report Information

The Camera Report window displays detailed information about what the HawkEye™ 1500 camera is reading. To display the Camera Report window:

1. Press Ctrl+R, or select Report Window from the Windows menu. ReadRunner displays the Camera Report window, as shown in Figure 4–10.

FIGURE 4–10. ReadRunner Camera Report Window



Count	Total	Passed	Failed
Cycles	392845	392810	35
Decodes		392810	15
Located		392825	20
Matches		0	0
Verify Total	Good	Fair	Poor
Timing (ms)	Value	Avg	Max
Cycle Time	185	171	5928
Read Time	73	59	5402
Proc Time	78	64	5407
Acq Time	106	106	119
Other	1		
	Alarms	Retries	PartQ
	0	0	disabled
	Gain	Exp	
Photo	280	5030	
	Decode	Trigger	Acq
Overruns	0	0	0
	Avail	Contig	Frags
Memory (MB)	12.089	11.455	107
	Overall	Last 20	
Read Rate	99.99%	100.00%	
	Overall	Worst	
Reads/Min	324	324	
	Overall	Growth	Contrast
AIM Grade			
	UEC	Axial NU	Value

- PartQ — Indicates whether or not the part queue is enabled (a number indicates the current setting for maximum records) or disabled.

  Range: Typically, 0 - 32 for reports with images and 0 - 5000 for reports without images. The maximum value is a function of available memory.

- Photo — Is the current photometry, indicating the current setting for Gain (in usec), and the current setting for Exposure (in dB).
- Memory — Is the current available memory, the largest contiguous block of memory available, and the number of fragmented blocks of memory.
- Read Rate
  - Overall — Is the overall read rate percentage.
  - Last 20 — Is the read rate percentage for the last 20 reads.
- Reads/Min
  - Overall — The overall number of reads per minute.
  - Worst — The lowest number of reads per minute.
- AIM Grade
  - Overall — When the verification is set to AIM mode, the Overall Grade is defined by AIM as the lowest grade of Symbol Contrast, Axial Nonuniformity, Print Growth, and Unused Error Correction. When verification is set to None, the Overall Grade is blank.
  - Growth — Measures how much the Data Matrix cells are over or under printed. Ideally, the dark cell and light cell should be of the same size, which is the Nominal Cell Size. For overprinted Data Matrix, the print growth is a positive value. For underprinted Data Matrix, the print growth is negative, corresponding to cell shrinkage. The print growth is based on the cells on the two timing borders. For example, for a dark on light Data Matrix, suppose the nominal cell size is 10 pixels and all dark cells have the average cell size 12 pixels, then the print growth is (12-10)/10= 20%. If we assume the average cell size of the dark cells is 8 pixels, then the print growth is (8-10)/10=-20%. The AIM print growth D value is

equal to growth multiplied by 3.3. The AIM grade for growth is:

A if -15% < Growth <= 15%
B if -21% < Growth <= 21%
C if -26% < Growth <= 26%
D if -30% < Growth <= 30%
F if Growth is < -30% or Growth > 30%

- Contrast — The difference in reflectance (measured by grayscale values) between the light and dark cells of the symbol. It is measured according to the AIM specification. The arithmetic mean of the darkest 10% of the pixels and that of the lightest 10% pixels within the Data Matrix area are computed. The Symbol Contrast is the difference of the two means divided by the full gray scale range. The Symbol Contrast Grade is given as:

  A if contrast is >= 70%
  B if contrast is >= 55%
  C if contrast is >= 40%
  D if contrast is >= 20%
  F if contrast is < 20%

- UEC — Unused Error Correction. A Data Matrix symbol has fixed error correction capacity. When a Data Matrix is decoded, the Error Correction (used) indicates how much of the error correction capability is consumed in order to decode the symbol. The more the error correction is used, the less the Unused Error Correction is left within the error correction capacity, which corresponds to the poorer print quality of the symbol. The grade (maximum is 1.0) is:

  A if UEC >= 0.62
  B if UEC >= 0.50
  C if UEC >= 0.37
  D if UEC >= 0.25
  F if UEC < 0.25

- Axial NU — The Axial Nonuniformity measures how different the average spacing between the center of adjacent cells in horizontal axis is from that of vertical axis. A square Data Matrix with same number of rows and columns tends to look like a rectangular matrix if Axial Nonuniformity is significant. This is measured according to AIM specification. Axial Nonuniformity is graded as:

A if AN <= 0.06
B if AN <= 0.08
C if AN <= 0.10
D if AN <= 0.12
F if AN > 0.12

- Value — The actual contrast value. This is a numeric value.

Resetting Camera Report Statistics

You can reset the camera statistics that are displayed in the Camera Report window (Figure 4–10) and at the bottom of ReadRunner's main window. To reset the statistics on camera X:

1. Select Reset Statistics from the Settings menu.
2. Click Yes when asked if you wish to control camera X.
3. Click Yes when asked if you want to reset the camera statistics.

    `ReadRunner resets the statistics in the Camera Report window and in the main ReadRunner window.`

Saving & Loading Configuration Files

A configuration file contains specific commands for reading a symbol. Once you have the settings you want, you can save them to the PC, and later reload them to the same camera or to another camera. Save Camera Configuration To File saves the entire registry to an .HCF (text-based) configuration file on you PC. This information includes:

- Separate photometry for each PID
- All PID settings
- Currently active PID in the JOB
- Whether the JOB should be restored on reboot

The following items are excluded from the list of saved configuration items:

- BEEP ?
- DHCPEX ?

- DOMAIN ?
- EIPENABLE ?
- IP ?
- MACADDR ?
- TARGET ?
- TTY ?

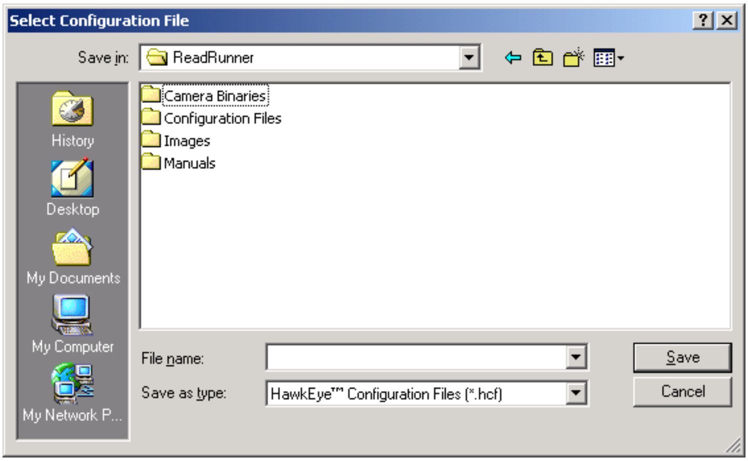
Note: The order of execution is important to ensure that PIDs are created before they may be used in a JOB related command.

Saving Configuration Files

To save the current camera configuration file:

1. Press F7, or select Save Camera Configuration To File from the File menu. ReadRunner displays the Select Configuration File window, as shown in Figure 4–11.

FIGURE 4–11. Select Configuration File Window — Saving



2. Select a location where you want to save the configuration file and click Save.

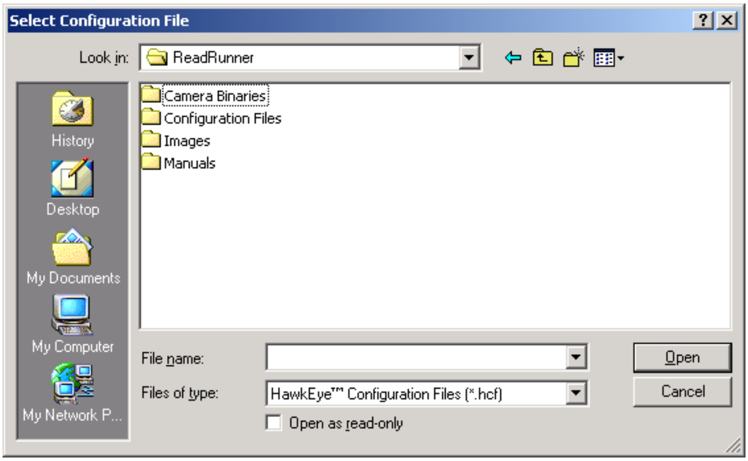
Loading Configuration Files

To load a configuration file to a camera:

1. Press F8, or select Load Configuration File To Camera from the File menu.

2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Select Configuration File window, as shown in Figure 4–12.

FIGURE 4–12. Select Configuration File Window

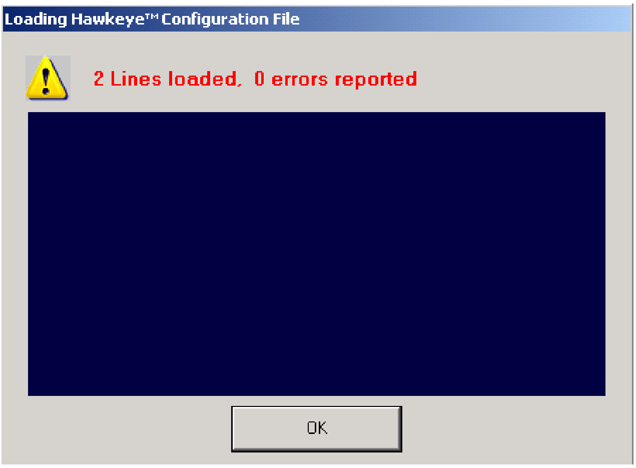


3. Double click on the Configuration Files folder.

4. Select the configuration file and click Open.

   ReadRunner displays a window similar to the in Figure 4-13 and loads the configuration file to the camera.

FIGURE 4–13. Select Configuration File Window — Loading



Releasing Control of a Camera

To release control of a camera:

1. Click Release.

ReadRunner releases control of the camera currently selected.

Removing a Camera

When you are finished using a camera and want to disconnect and remove it, use the following procedure:

1. De-select the button of the camera you want to remove.

Note: A button is de-selected when its background is blue.

2. Right click on the button of the camera you want to remove.
3. Highlight Remove Button.
4. Click Remove Button.

ReadRunner removes the camera from the main ReadRunner window.

Using ReadRunner

The procedures in this section explain:

- "Setting Up Photometry" on page 4-22
- "Preprocessing Images" on page 4-24
- "Setting Up Symbology" on page 4-25
- "Defining the Region of Interest" on page 4-26
- Specifying preferences, the application mode, serial and TCP/IP settings, and output format strings, beginning on page 4–44
- "Toggling the Target Laser" on page 4-86
- "Toggling the Beeper" on page 4-87
- "Saving Parameters on the Camera to Flash" on page 4-89
- "Restoring Defaults" on page 4-89
- Displaying reports and image over a serial connection, commands to and from the camera, timing and rate information, and information about cameras on the network, beginning on page 4–90

Setting Up Photometry

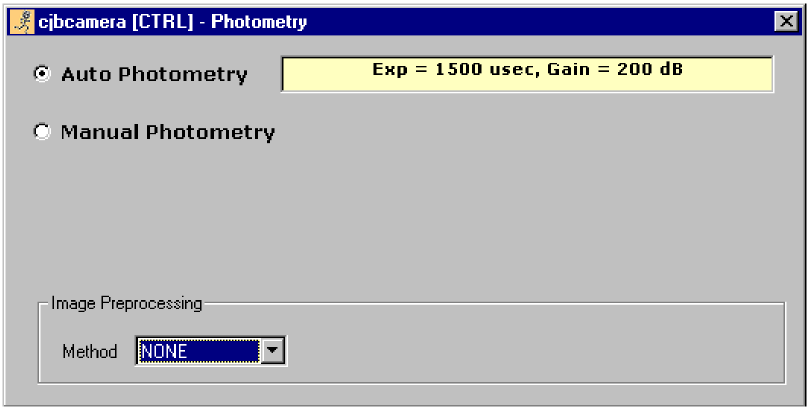
The HawkEye™ 1500 camera can use both automatic and manual photometry:

- Auto Photometry — Default (auto exposure). The HawkEye™ 1500 camera determines lighting control at each trigger and prior to each image scan.
- Manual Photometry — For faster read rates, you specify the exposure and gain settings.

To display the Photometry window:

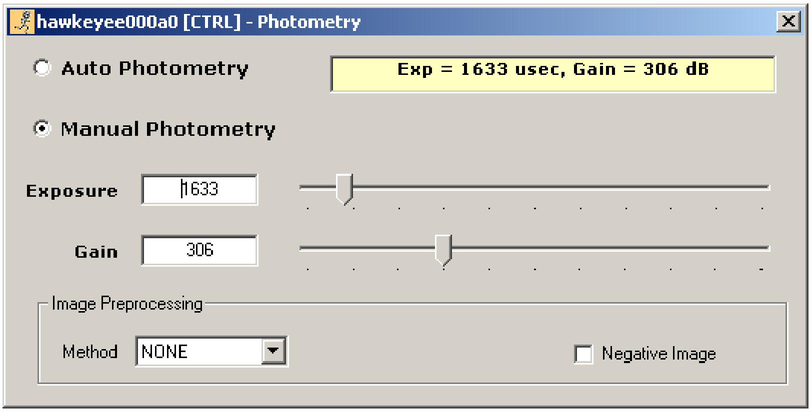
1. Press Ctrl+P, or select Photometry from the Settings menu.
2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Photometry window, as shown in Figure 4–14 (Auto Photometry selected) or Figure 4–15 (Manual Photometry selected).

FIGURE 4–14. Photometry Window — Auto Photometry



If you select Automatic Photometry, the HawkEye™ 1500 camera determines lighting control at each trigger and prior to each image scan.

FIGURE 4–15. Photometry Window — Manual Photometry



If you select Manual Photometry, you must specify the following:

– Exposure — Move the slider left to decrease or right to increase the exposure.

Range: 30 - 20000usec

Note: Higher values for Exposure mean a longer image acquisition time and total cycle time.

– Gain — Move the slider left to decrease or right to increase the gain.

Range: 0 - 1023dB

Note: When Verification is set to No Verification, Gain and Exposure can be adjusted by AutoPhotometry or manually.

When Verification is set to AIM DPM-1-2006, Gain and Exposure will be locked to the user and can only be updated by the kernel. Image preprocessing will be disabled.

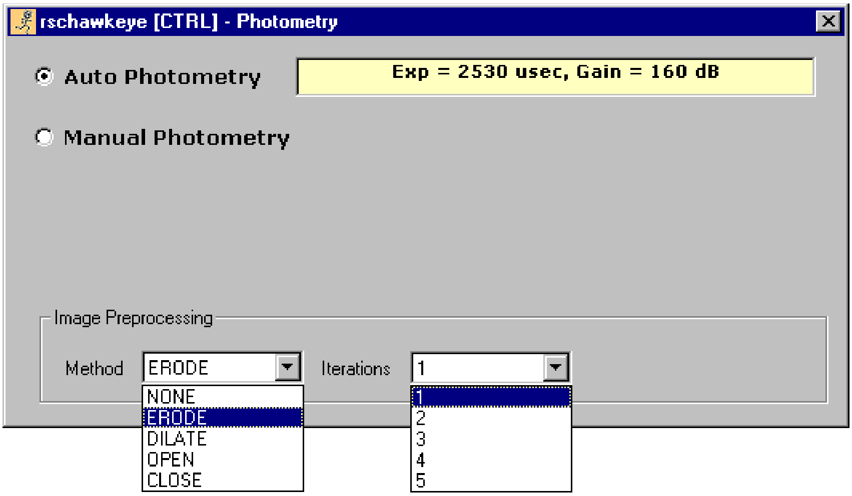
When Verification is set to anything other than No Verification or AIM DPM-1-2006, a Warning message will be displayed on the dialog box indicating that changing the Exposure or Gain will uncalibrate the camera. Image preprocessing will be disabled.

Preprocessing Images

The Photometry window contains the Image Preprocessing dialog area where you can select the method for processing images, and the number of iterations for the method, as shown in Figure 4–16.

Note: The Iterations dialog box is displayed after you select a method.

FIGURE 4–16. Photometry Window



The methods are:

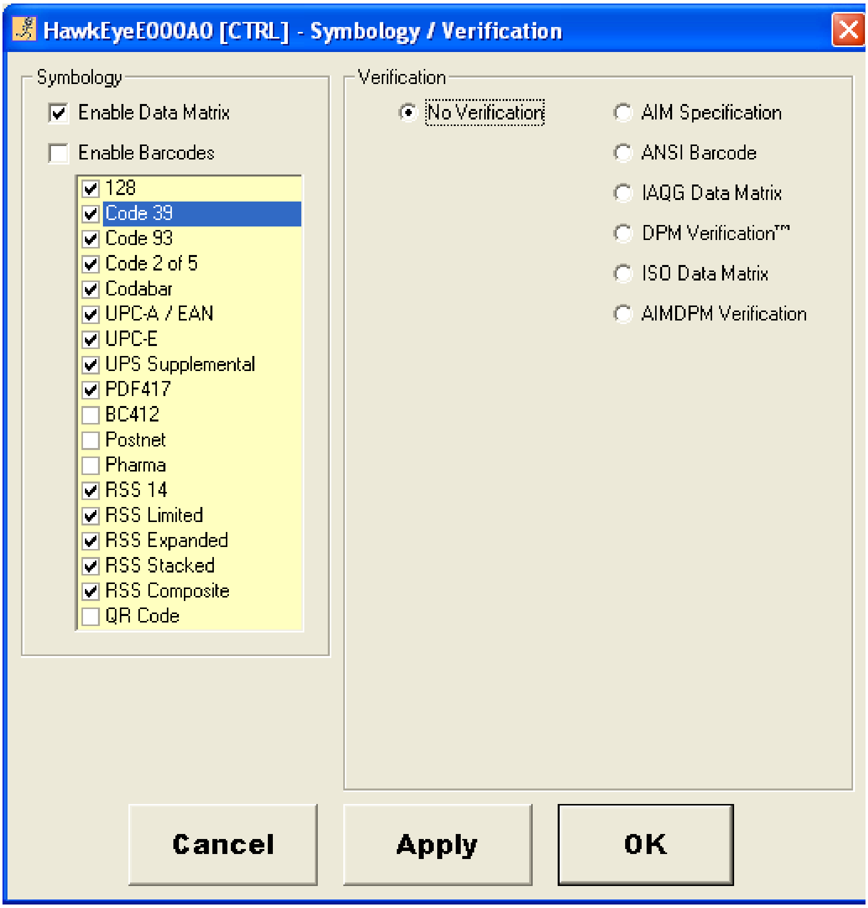
- NONE — This is the default.
- ERODE — This method increases the dark cell size. Use this method to increase the dark cell size of a dark on light Data Matrix.
- DILATE — This method increases the light cell size. Use this method to increase the light cell size of a light on dark Data Matrix.
- OPEN — This method removes minor light defects of dark cells.
- CLOSE — This method removes minor dark defects of light cells.

Setting Up Symbology

Symbology specifies what kind of symbols the HawkEye™ 1500 camera will read. If the decoder is “opened up” (that is, no learn has occurred), you can specify which symbols you want read. By de-selecting symbols, you can reduce decode time. To display the Symbology and Verification window:

1. Press Ctrl+V, or select Symbology and Verification from the Settings menu.
2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Symbology and Verification window, as shown in Figure 4–17.

FIGURE 4–17. **Symbology / Verification Window**



3. Specify the kinds of symbols the HawkEye™ 1500 camera will read:

   – Enable Data Matrix — Specifies that Data Matrix symbols will be read.

   – Enable Barcodes — Specifies that Barcode symbols will be read.

Defining the Region of Interest

Defining a region of interest (ROI) within the HawkEye™ 1500 camera's Field of View improves read rates and speeds up processing. To define an ROI:

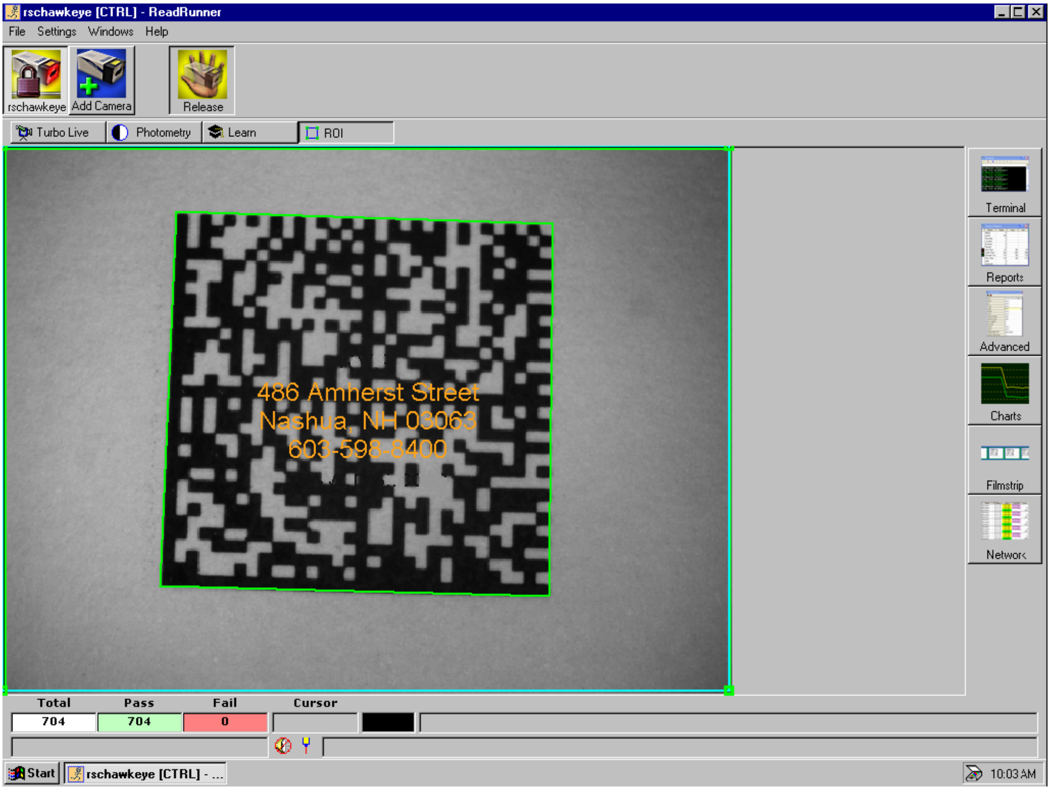
1. Click 

2. Click Yes when asked if you wish to control camera X.

3. Click 

   `ReadRunner displays a search ROI, as shown in Figure 4-18.`

FIGURE 4–18. ROI Window



4. Use the mouse to define a smaller search area in the camera's field of view. Adjust the ROI from any of its four corners.

Copying Current to PID

This menu item allows you to copy the current settings (always PID0) to a different PID (1 through 15). Each PID contains the following information:

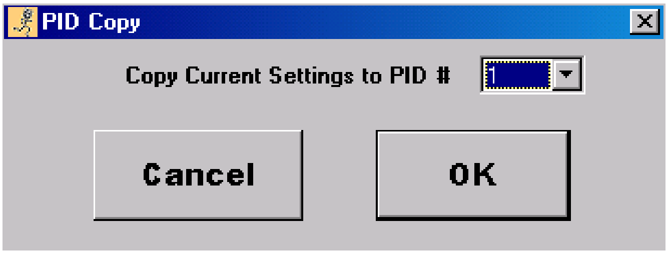
TABLE 4–4. Parameters Stored in Each PID

Parameter	Notes
Photometry	Gain & Exposure only. NOTE: If Autophotometry is ON (System parameters), then these PID settings are NOT used.
PreProcessing	Morphology operator and iterations
Decoder	All decoder parameters.

ReadRunner maintains the list of PIDs that are active (that were copied from Current) for use in the Advanced Retry modes.

Note: These are the only PIDs available to the RETRY commands (even though there are 15 open PID slots on the camera).

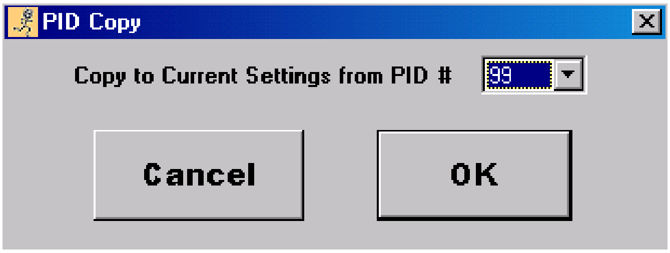
FIGURE 4–19. Copy Current Settings to PID # Dialog Box



Copying PID to Current

This menu item allows you to copy the PID settings (1 through 15 plus read-only 99) to the current settings (PID 0).

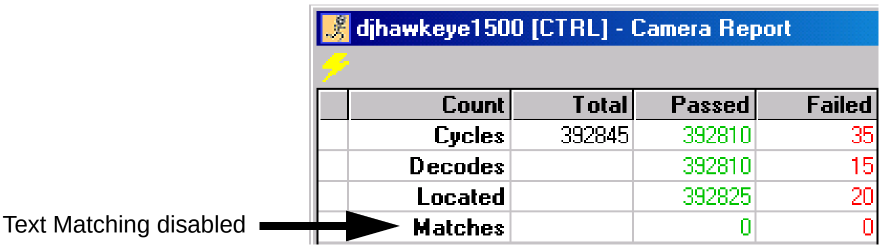
FIGURE 4–20. **Copy to Current Settings from PID # Dialog Box**



Setting Up Text Matching

The Camera Report window (Figure 4–21) displays detailed information (Cycles, Decodes, and Locates, and so on) about what the HawkEye™ 1500 camera is reading.

FIGURE 4–21. **ReadRunner Camera Report Window**



If you also want to display detailed information about the number of passed and failed text matches, you have to specify the text in the Match Mode Setup window. To set up text matching:

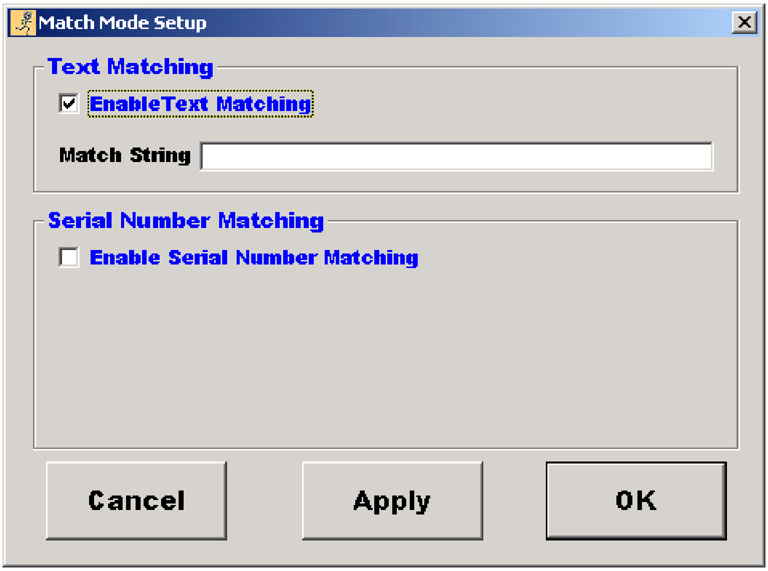
1. Press Ctrl+O, or select Match Mode Settings from the Settings menu.

   `ReadRunner displays the Match Mode Setup window, as shown in Figure 4-22.`

2. Click to enable Enable Text Matching.

   `ReadRunner displays the Match String text box (Figure 4-22).`

FIGURE 4–22. **Text Matching Window**



3. Enter the exact text (case sensitive) you want to match.

Note: You can use the * and ? wildcards in the match string definition.

The Match String should contain only printable characters.

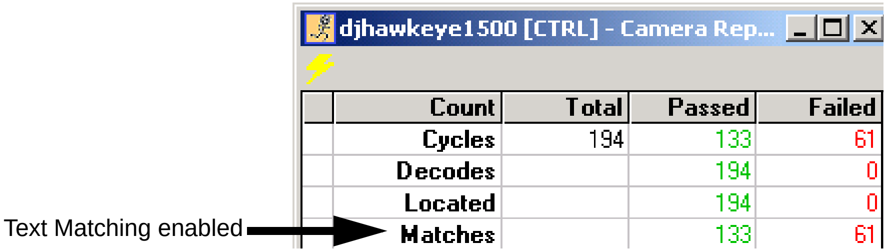
4. Click Apply and OK.

Graphically, in the image area, ReadRunner will display either of the following messages:

- For a successful match — Code Matched
- For an unsuccessful match — NO MATCH Error

After text matching is enabled, your Camera Report window will look similar to the one displayed in Figure 4–23.

FIGURE 4–23. ReadRunner Camera Report Window



Match List Triggered I/O

Match List Triggered IO allows you to match the decoded string against up to 15 match strings expressions containing optional single (?) and multiple (*) character wildcards. The resulting MATCH_INDEX is the index into the list of strings where the match was found. The MATCH_INDEX can be reported in text reports and as the 4-bit binary equivalent through the assignment of MATCH_BIT_n keywords to the IO lines.

Match List may be used in conjunction with "serial number sequence matching."

If discrete output is required (i.e., only one line set in the result), use only match strings 1, 2, 4, and 8. Use of other match strings will result in more than one output bit being set in the result. For example, if match string 7 (binary 0111) is matched, MATCH_BIT_1, MATCH_BIT_2, and MATCH_BIT_3 will be set.

Note: The Match List does not require image re-acquisition or re-decode using different PID settings, so it is much more efficient than other methods of determining which string is a match.

The HawkEye™ firmware supports a JOB based MATCH_LIST, which will contain up to 15 string expressions. The integer index of the string expression in the MATCH_LIST will correspond to a binary mask that will be applied to the Digital I/O lines of the camera.

At run time, following a decode, the application will operate as follows:

- If the MATCH_LIST is enabled, the application will search the non-empty entries in the MATCH_LIST from index 1 to index 15.

- When the first match between the decoded string and the string expression in the MATCH_LIST is encountered, the Digital I/O lines will be set in the bit pattern defined by the binary value of the index into the MATCH_LIST.
- Once a match has been found, the application will cease to search the list. This is an important point in that, if the first match string is “*”, all decoded strings will match it and the resulting index will always be 1.

Note: Always put the most specific match strings at the start of the Match List

---

Enable or disable the MATCH_LIST using the MATCH_LIST_ENABLE command. Enter entries to the MATCH_LIST using the MATCH_LIST command. For more information about these commands, see the HawkEye™ 1500 Series Reference and Programmers Manual.

Typical entries for a situation where four parts are to be identified by the last two entries in the decoded string using unique IO lines would be:

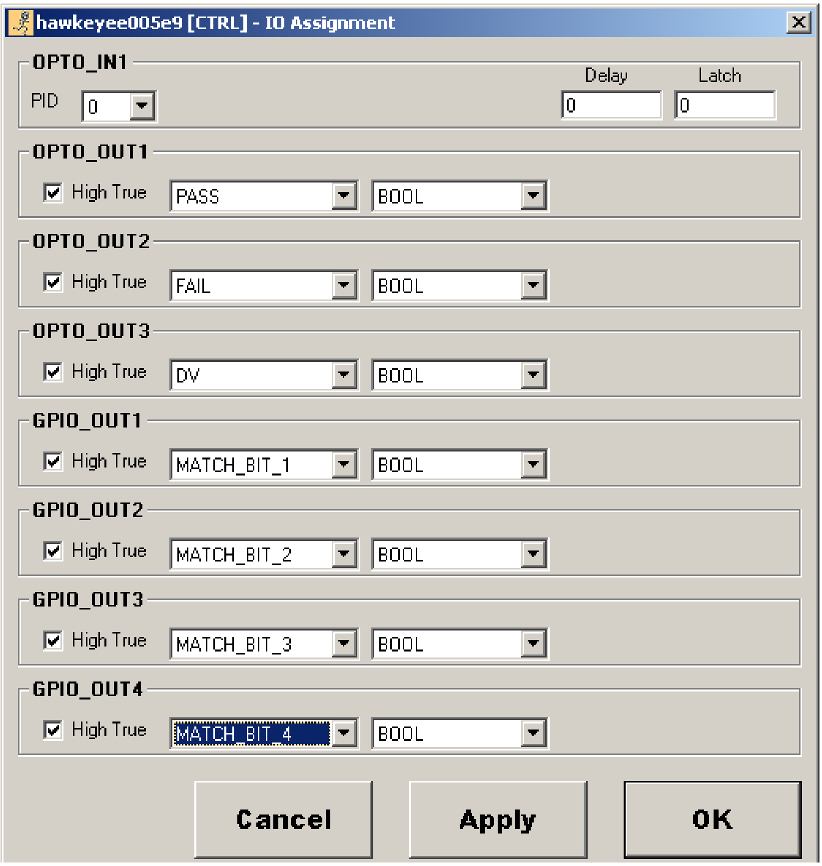
```
MATCH_LIST 1 “*AB”

MATCH_LIST 2 “*CD”

MATCH_LIST 4 “*EF”

MATCH_LIST 8 “*GH”
```

The following entry will turn on GPIO 1, 2, 3, and 4 when a match is found:

```
MATCH LIST 15 “*BOO”

MATCH_LIST n “”          Clears the entry at index n
```

The MATCH_LIST_ENABLE and MATCH_LIST settings are part of the configuration file (*hcf file) that can be saved or uploaded to the camera.

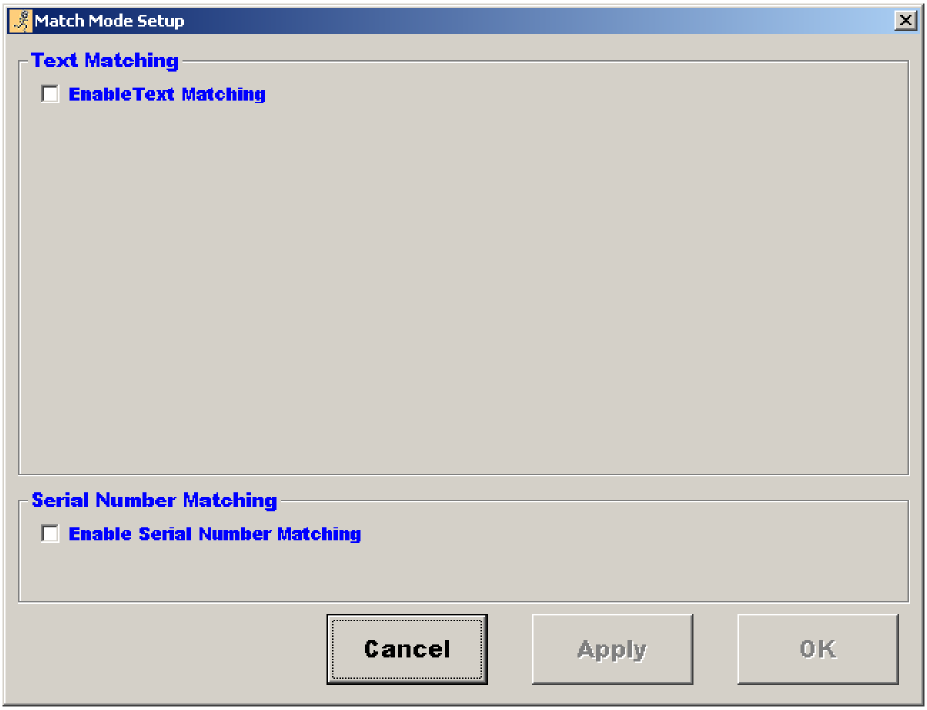
FIGURE 4–24. **MATCH_BIT Keyword Assignment**



The MATCH_BIT_1 through MATCH_BIT_4 keywords are available only when connected to a HawkEye™ camera running firmware newer than the released version of 2.1. Keyword assignments are performed with the following commands:

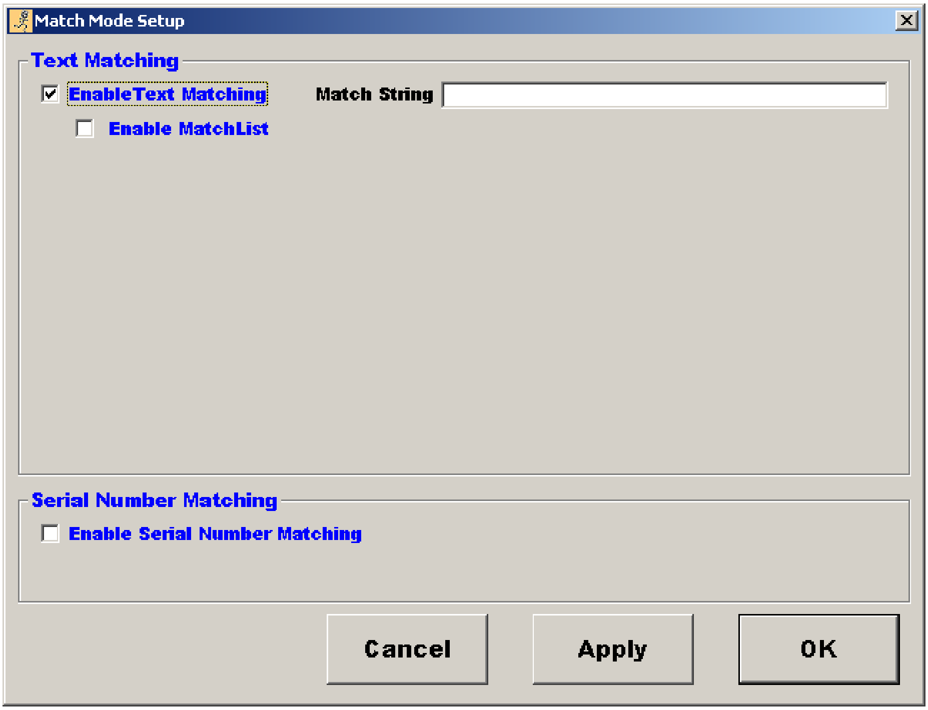
```
IOASSIGN OPTO_OUT1  PASS
IOASSIGN OPTO_OUT2  FAIL
IOASSIGN OPTO_OUT3  DV
IOASSIGN GPIO_OUT1  MATCH_BIT_1
IOASSIGN GPIO_OUT2  MATCH_BIT_2
IOASSIGN GPIO_OUT3  MATCH_BIT_3
IOASSIGN GPIO_OUT4  MATCH_BIT_4
```

FIGURE 4–25. **Enable Text Matching Screen**



The MatchList options will only be displayed for HawkEye™ cameras running firmware newer than the released version of 2.1.

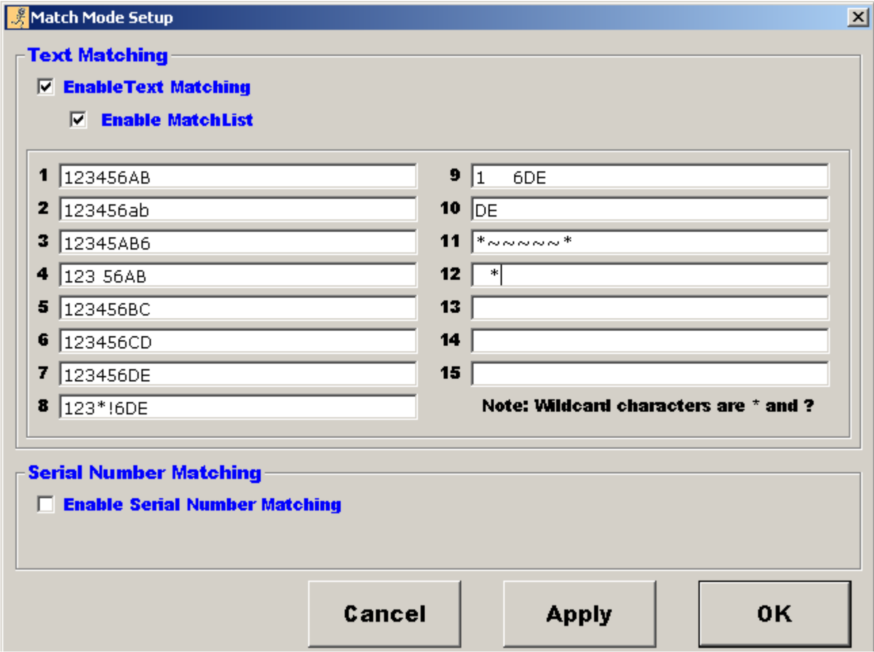
FIGURE 4–26. **Enable MatchList**



After you select Enable Text Matching, ReadRunner gives you the option to do single Match String or MatchList.

Note: After “Enable MatchList” is selected, the single Match String is removed from the dialog as it is no longer used.

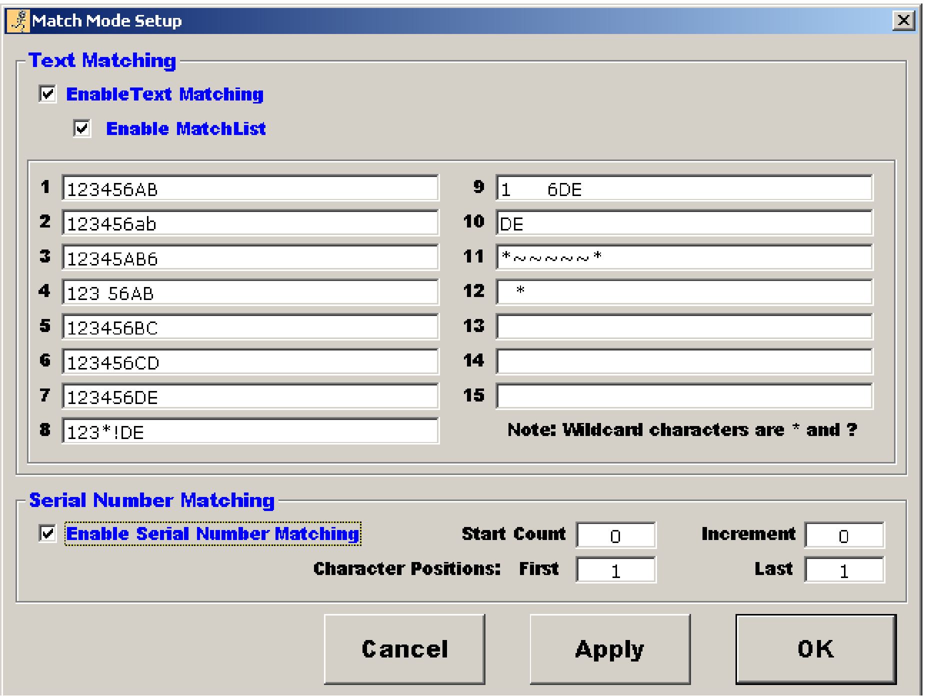
FIGURE 4–27. Adding Match Strings



The commands generated from the above settings are:

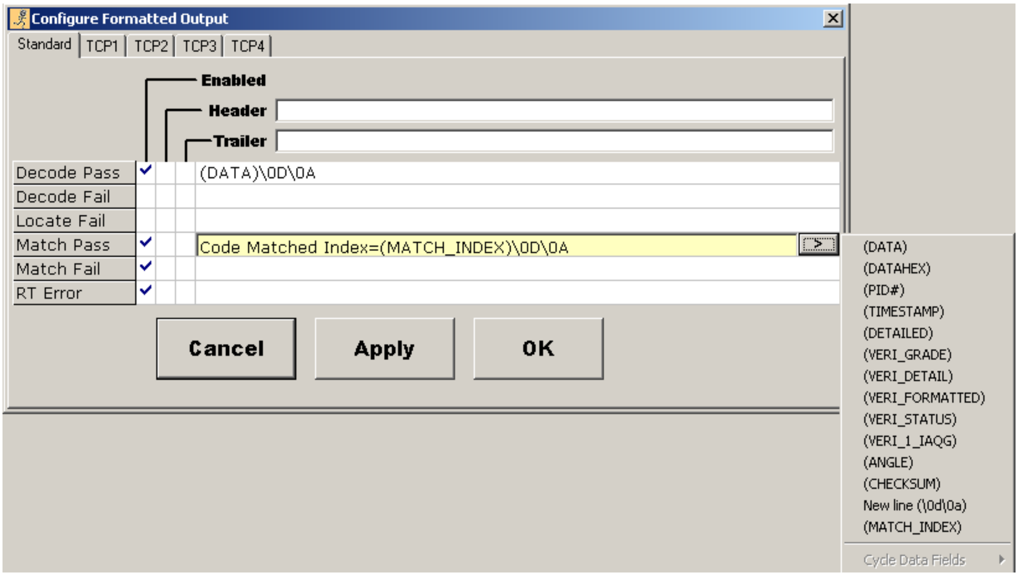
```
MATCHEX T “” 0 0 1 1
MATCH_LIST_ENABLE ON
MATCH_LIST 1 “123456AB”
MATCH_LIST 2 “123456ab”
MATCH_LIST 3 “12345AB6”
MATCH_LIST 4 “123 56AB”
MATCH_LIST 5 “123456BC”
MATCH_LIST 6 “123456CD”
MATCH_LIST 7 “123456DE”
MATCH_LIST 8 “123*!6DE”
MATCH_LIST 9 “1    6DE”
MATCH_LIST 10 “DE”
MATCH_LIST 11 “*~~~~~~*”
MATCH_LIST 12 “   *   “
MATCH_LIST 13 “”
MATCH_LIST 14 “”
MATCH_LIST 15 “”
```

FIGURE 4–28. Serial Number Matching



MatchList only replaces the single match string capabilities with multiple match string testing. All other functions, such as Serial Number Matching, remain unaffected and may be used in conjunction with Match List.

FIGURE 4–29. **Match Index for Text Reporting**



MATCH_INDEX has been added if you require text output instead of, or in addition to, Digital I/O output. MATCH_INDEX can be displayed in reports, graphic overlays, and as Digital I/O, as shown in Figure 4–30 through Figure 4–32.

FIGURE 4–30. **Example 1**

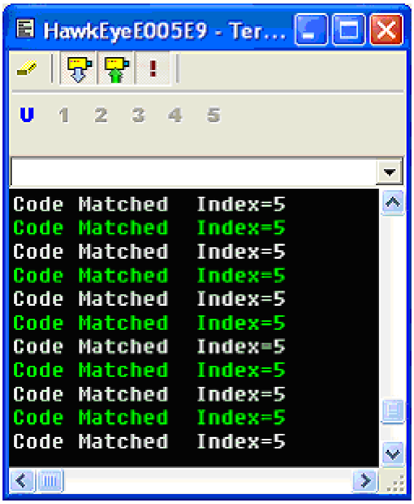


FIGURE 4–31. **Example 2**

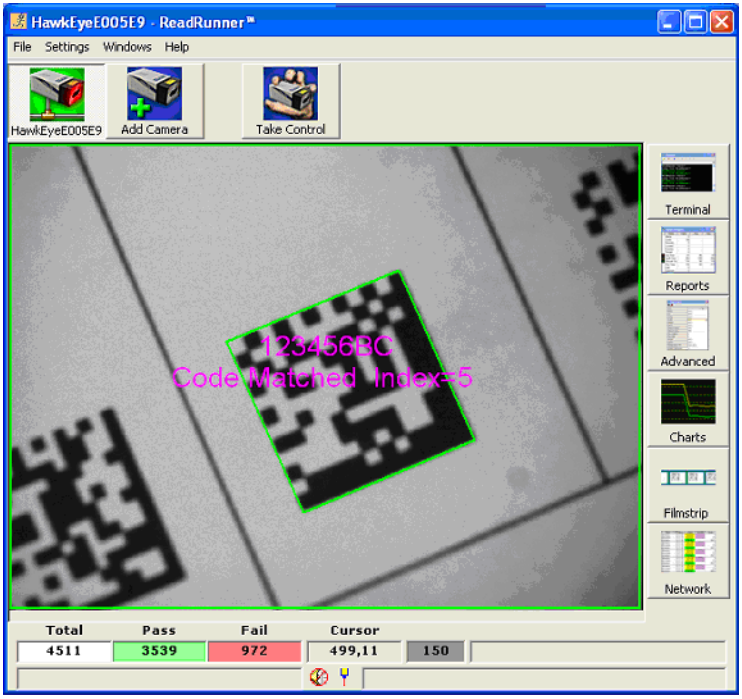
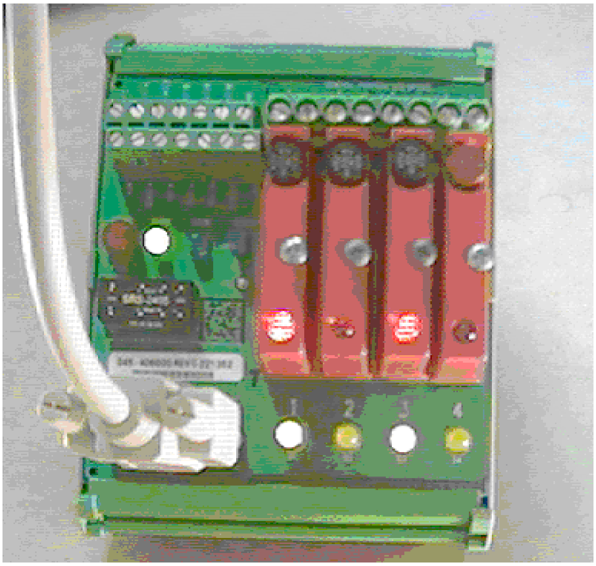


FIGURE 4–32. **Example 3**



Behavior of the Wildcard Match

The behavior of the wildcard match is:

- A case sensitive match is performed.
- * and ? are wildcard characters in the pattern as well as potentially valid characters in the string.
- * as a wildcard character represents 0 or more characters in a string.
- ? as a wildcard character represents 1 and only 1 character in a string.
- If * and ? are found in the string, the pattern can have either a * or ? describing that position, however, the pattern will also match any string with something at that position other than * or ? (i.e., the * and ? in the pattern are seen as wildcards first and characters second).

```
Examples of this behavior are:

"*string" matches "?string"

"*string" matches "*string"

"*string" matches "any string"
```

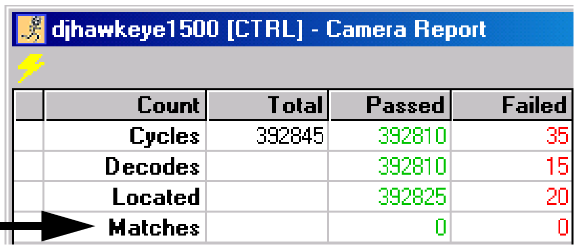
```
"*string" matches "string"


"?string" matches "?string"

"?string" matches "*string"

"?string" matches "Xstring"
```

Setting Up Serial Number Matching

The Camera Report window (Figure 4–33) displays detailed information (Cycles, Decodes, and Locates, and so on) about what the HawkEye™ 1500 camera is reading.

FIGURE 4–33. ReadRunner Camera Report Window

Serial Number Matching disabled



If you also want to display detailed information about the number of passed and failed serial number matches, you have to specify detailed information in the Match Mode Setup window. To set up serial number matching:

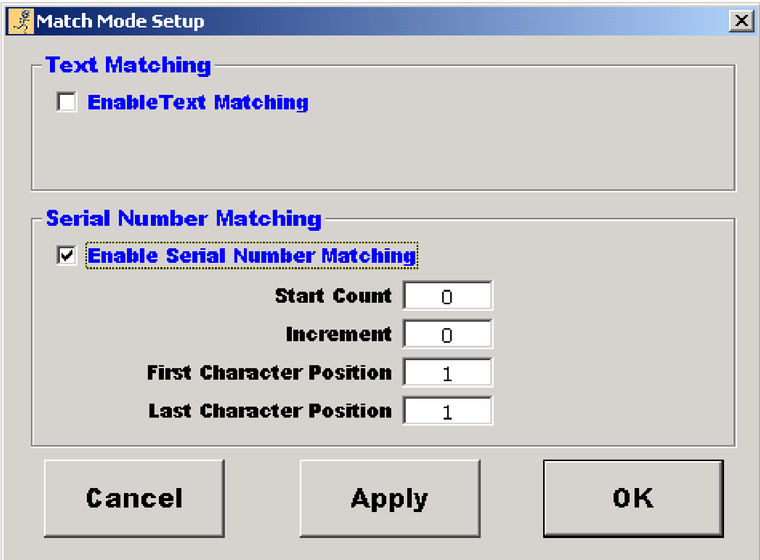
1. Press Ctrl+O, or select Match Mode Settings from the Settings menu.

   ReadRunner displays the Match Mode Setup window, as shown in Figure 4-34.

2. Click to enable Enable Serial Number Matching.

   ReadRunner displays the Enable Serial Number Matching text box (Figure 4-34).

FIGURE 4–34. **Serial Number Matching Window**



3. Specify the following:

   – Start Count — This is the starting serial number.

   – Increment — This indicates by how much the serial number will increase each time it is encountered.

   – First Character Position — This indicates where in a string the actual serial number begins.

   – Last Character Position — This indicates where in a string the actual serial number ends.

   For example, assume a part number looks like the following:

   ABCDEFG123456-101

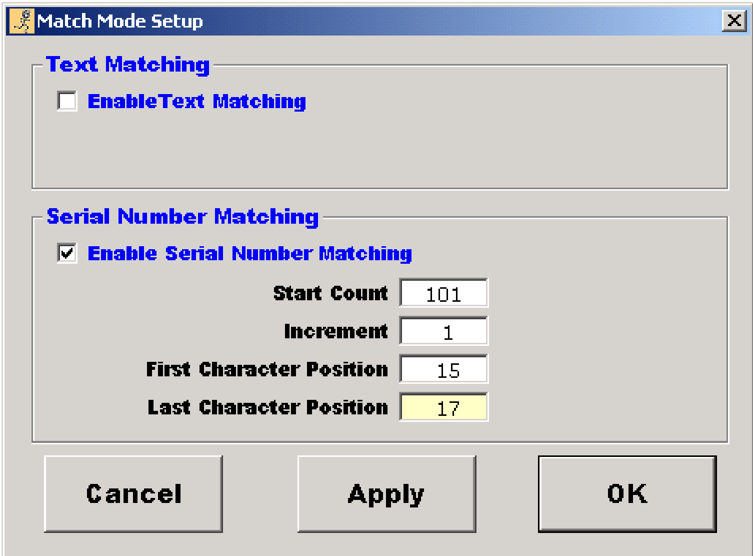
   Last Character Position

   First Character Position

Note: Serial match supports 9 digits.

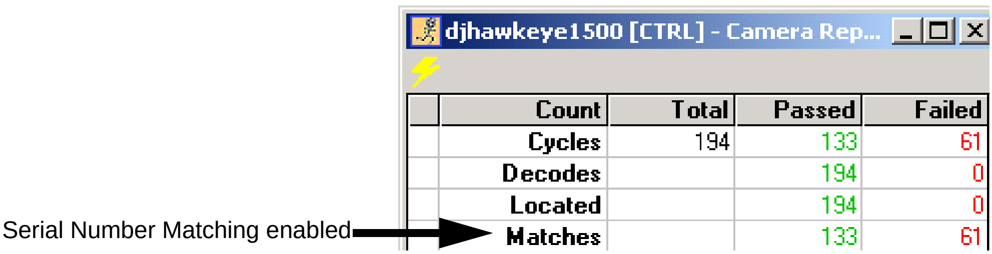
Figure 4–35 shows the correct settings for serial number matching, assuming a serial number that increments by one each time.

FIGURE 4–35. **Serial Number Matching Window**



After serial number matching is enabled, your Camera Report window will look similar to the one displayed in Figure 4–36.

FIGURE 4–36. **ReadRunner Camera Report Window**

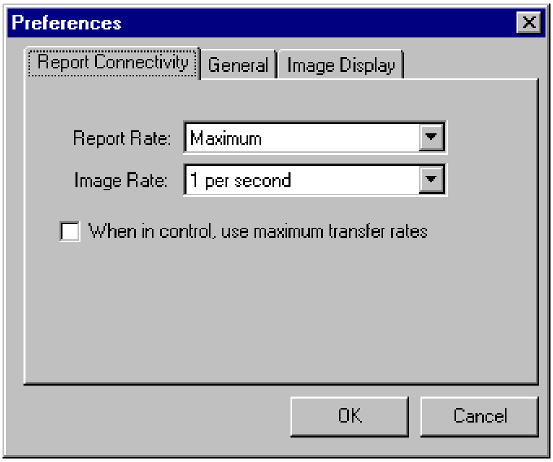


Specifying...

Preferences

The Preferences window contains three tabs. The Report Connectivity tab (Figure 4–37) allows you to specify the rate at which report information is sent to the Reports window.

FIGURE 4–37. Preferences Window — Report Connectivity Tab



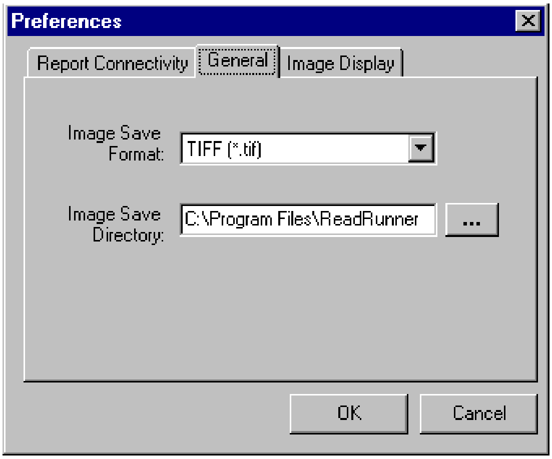
- Report Rate — Specifies the rate at which data is sent to the Reports window.
  - Maximum — Send report information as fast as possible.
  - 1 per second — At most, send report information once per second.
  - 2 per second — At most, send report information twice per second.
  - 4 per second — At most, send report information four times per second.
  - 8 per second — At most, send report information eight times per second.
- Image Rate — Specifies the rate at which images are sent to the camera.
  - Maximum — Send images as fast as possible.
  - 1 per second — At most, send an image once per second.

– 2 per second — At most, send an image twice per second.

- When in control, use maximum transfer rates — Send report information and images as fast as possible.

The General tab (Figure 4–38) allows you to specify the format and location of the save images.

FIGURE 4–38. Preferences Window — General Tab

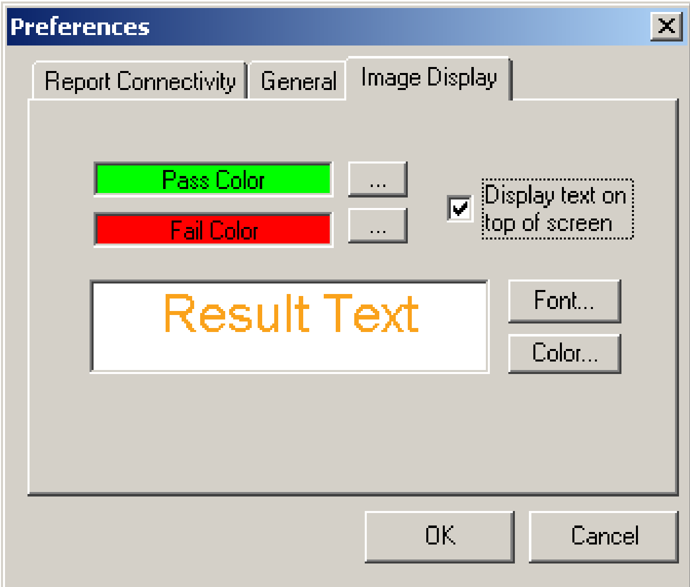


- Image Save Format — Specifies the format of the saved image, either TIFF (*.tif) or Bitmap (*.bmp).

- Image Save Directory — Specifies the directory for the saved image.

The Image Display tab (Figure 4–39) allows you to specify the color of Pass and Fail information, and the font, font style, and font size of result text.

FIGURE 4–39. Preferences Window — Image Display Tab



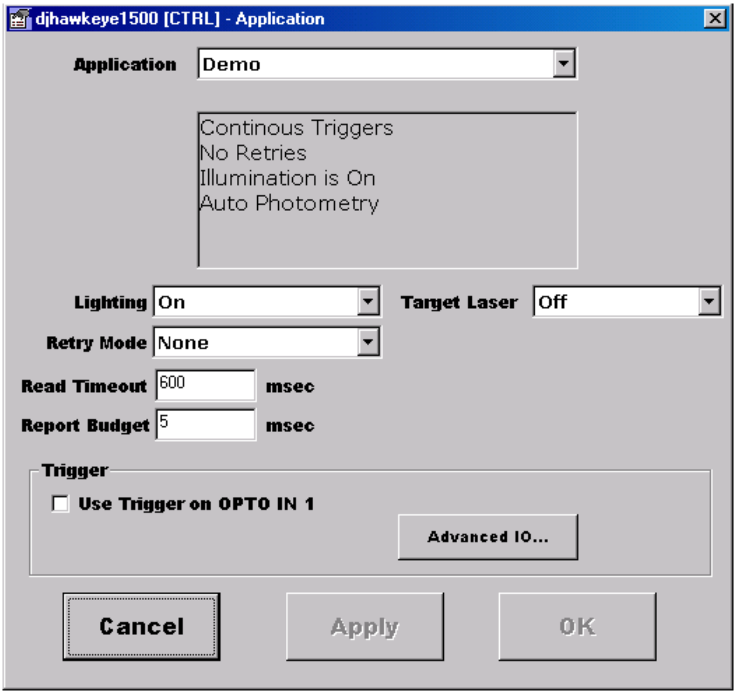
- Pass Color — Specifies the color of the information that passes. Click ... to display the color palette.
- Fail Color — Specifies the color of information that fails. Click ... to display the color palette.
- Result Text — When you click Font, ReadRunner displays fonts, font styles, and font sizes from which you can choose what the result text will look like. Click Color to display the color palette.
- Display text on top of screen — Click this option to optionally have the decode strings placed at the top of the image instead of over the decoded symbol. This allows images to be inspected for defects.

Application Modes

The application mode specifies the basic setup that the HawkEye™ 1500 camera will use for reading. To specify the application mode:

1. Press Ctrl+M, or select Application Mode from the Settings menu.
2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Application Mode window, as shown in Figure 4–40.

FIGURE 4–40. Application Mode Window



3. Select one of the following modes:

Note: When you change the Application mode, a dialog box will be displayed asking you to confirm the changes corresponding to the new Application mode.

– Demo — This mode demonstrates the capabilities of the HawkEye™ 1500 camera. Settings include continuous triggers, no retries, illumination constantly on, and auto photometry.

– Motion — This mode scans a symbol when it moves within the HawkEye™ 1500's field of view. Because the parts are in motion, there is no time for the automatic calculation of photometry values. Thus, automatic photometry is off. You must specify the exposure and gain settings manually. A trigger is supplied (through the I/O connector) to the camera by a sensor device. In addition, the Lighting selection will be

Power-strobe. This selection allows the parts to be moving at a very high speed, where short exposure times are desired.

– Stop and Scan — This mode scans a symbol when it stops within the HawkEye™ 1500's field of view. A trigger is supplied to the camera by a sensing device (not included as standard equipment) when the symbol is in place. The trigger signal is supplied through the I/O connector. The illumination is set to strobe, and the automatic photometry is on. When the trigger is applied, the HawkEye™ 1500 camera will set the exposure and read the symbol.

– Supermarket — This is often referred to as Presentation Mode. Parameters are set similarly to Demo mode, except that an inter-symbol wait time (ISWT) is imposed. This means that, when a symbol is presented to the HawkEye™ 1500's field of view, it will be read once. The same symbol cannot be read again until the symbol is removed from the field of view for at least the indicated ISWT. A different symbol, when presented to the camera, will be read immediately.

You may make modifications to any of these application modes.

Note: When Verification is set to AIM DPM-1-2006, Application, Lighting, and Retry Mode are disabled.

The remainder of this section on Application Mode describes:

- "Lighting" on page 4-48
- "Retry Modes" on page 4-49
- "Extended PID List" on page 4-51
- "Read Timeout" on page 4-52
- "Report Budget" on page 4-52
- "Triggers" on page 4-53
- "Advanced I/O" on page 4-54

Lighting

The lighting type specifies what kind of lighting the HawkEye™ 1500 camera will be using. To specify the lighting type:

1. Press Ctrl+M, or select Application Mode from the Settings menu.
2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Application Mode window.
3. Using the pull down menu, specify one of the following for lighting:
   - Off — Use ambient lighting.
   - On — Use constant lighting. This is the default.
   - Strobe — Use strobe lighting.
   - Power Strobe — Use power strobed lighting.
   - On w/Power Strobe — This selection keeps the lighting on at all times but, when the camera is exposing, the Power Strobe intensity is supplied. Usually, this selection is chosen to reduce the effects of strobing that can cause adverse effects to humans when in the physical presence of the camera.
   - External — Use external lighting with external strobe on GPIO 1.
   - On & External — Use constant lighting with external strobe on GPIO 1.
   - Strobe & Ext — Use strobe lighting with external strobe on GPIO 1.
   - Power Strobe & Ext — Use power strobe lighting with external strobe on GPIO 1.
   - On/PowerStrb/Ext — This selection keeps the lighting on at all times but, when the camera is exposing, the Power Strobe intensity is supplied. The external strobe is on GPIO 1. Usually, this selection is chosen to reduce the effects of strobing that can cause adverse effects to humans when in the physical presence of the camera.

Retry Modes

The retry mode specifies whether or not there will be multiple attempts, the number of attempts, and the time limit. To specify the retry mode:

1. Press Ctrl+M, or select Application Mode from the Settings menu.
2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Application Mode window.

3. From the Retry Mode pull down menu, specify one of the following:
   - None — Sets the read cycle to single attempt mode. This is the default.
   - Time — Set the read cycle to multiple attempts within the specified time limit.

     Range: 1500 - 60000ms
   - Count — Set the read cycle to a specified number of attempts.

     Range: 1 - 15
   - RETRY input Duration — Retries will continue while the state of the assigned RETRY input line remains high.

     Note: PID, Trigger Edge, Latch and Delay settings are ignored on this line.
   - Intersymbol Wait — When an Intersymbol Wait time is specified, leaving a symbol under the camera will produce a single result. A new result is produced only:
     - If the same symbol data is NOT read for at least as long as the intersymbol wait period

       OR
     - If a new symbol data is read
   - PID List — Allows each retry to use different decoder and preprocessing parameters without taking a new picture for each retry. Up to 15 PIDs can be used (5 retries). This mode retries in the order List 1 through List 15, and DOES NOT take a new picture each time.

     Note: See “Extended PID List” on page 4-51.
   - PID List w/Acquire — Allows each retry to use different decoder and/or preprocessing parameters while taking a new Picture for each retry. Up to 15 PIDs can be used (5 retries). This mode retries in the order List 1 through List 15, taking a new picture each time.

     Note: See “Extended PID List” on page 4-51.

– Light — Allows up to 2 retries using optionally different Decoder parameters. On the second retry, a new picture is taken using the same photometry parameters; however the lighting and/or strobe mode can be specified directly in the UI. This mode is a subset of the RETRY PID List w/Acquire mode. This mode retries in the order PID0 with Lighting ON, and then PIDn with External Light.

For additional information about Retry Modes, see "Retry Modes" on page 3-6.

Extended PID List

The PID List and PID List w/Acquire retry modes allow up to 15 PIDs with specific decoder settings to be tried. The additional PID list entry slots become visible in the user interface as they are populated. A value of 0, at any point, terminates the list

---

Note: The camera is sent the entire, defined list, but will only process retries on PIDs up to the point that it sees the first 0 in the list.

There are two modes of retry list operation:

- PID List — A single image acquisition is performed and decodes are attempted using the settings for each PID in the list, always starting at the first and working through to the end of the list. For performance reasons, it is important to order the list in terms of diminishing probability of decode success so that the chance of decode on the first one or two PIDs will be optimized. Subsequent reads proceed from the list position that the last read finished on.
- PID List w/Acquire — Cycles through the PID list, taking new images until a successful Acquire/Decode pair happens.

Both modes have process specific considerations that need to be taken into account before choosing which mode best suits a specific application.

For example, if the resulting Data Valid line controls when to move a conveyor belt to advance to the next part, you would not want to do a re-acquire, as the part would move on without trying the other PIDs. In this case, you would want to run without acquire so that all the PIDs are tried on the one image, and then give one final result.

PID List w/ Acquire is used in situations where various manual photometry settings are saved in different PIDs and a new image needs to be taken before the decoding process is started.

Read Timeout

This parameter specifies the time to spend searching and decoding a symbol (i.e., Data Matrix or Barcode). If Data Matrix reading is enabled, regardless of whether or not Barcode reading is enabled, the Read Timeout must be set in either of two ways. When Probe Direction for Data Matrix is set to:

- The default (HORI_VERT or VERT_HORI), you must set Read Timeout based on the following scenario. If the Data Matrix and Barcode reading takes x milliseconds to complete, set Read Timeout to at least 2 times the time in milliseconds. This allows the vertical probe to fail and the horizontal probe to complete successfully.
- HORIZONTAL only, VERTICAL only, or CRISS_CROSS, you must set Read Timeout based on the following scenario. If the Data Matrix and Barcode reading takes x milliseconds to complete, set Read Timeout to at least 1 time the time in milliseconds.

Note: When vertical is enabled, HORI_VERT and VERT_HORI are ignored internally and CRISS_CROSS is used for robustness.

Report Budget

This parameter specifies the amount of time in msec (default is 5) used to send results (decode string, image, report) over TCP/IP. It is scheduled in such a way that the read speed remains deterministic. This time gets added to the processing time for every part and, therefore, impacts the maximum reads per second of the camera.

This time can be reduced when sending results via a serial connection ONLY to gain a few more PPM, if necessary. However, be aware that the report budget is also used in Demo Mode to allow the camera to send images and reports to connected clients for the purpose of monitoring (Ethernet connections).

Setting this parameter to zero for very fast pipeline applications may have an adverse effect on ReadRunner responsiveness, and may cause it to stop receiving images and reports. Therefore, it is recommended that this value be set to at least 1 msec or more.

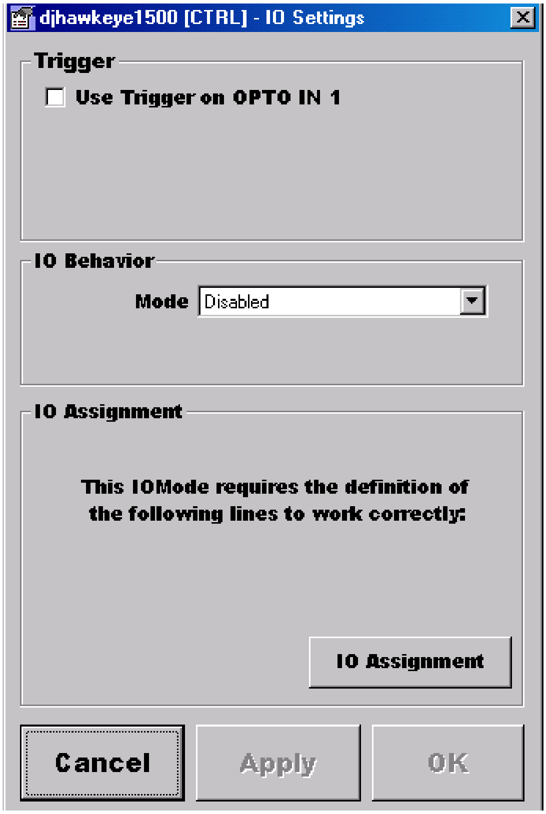
Triggers

To specify a trigger:

1. Press Ctrl+F6, or select IO Settings from the Settings menu.
2. Click Yes when asked if you wish to control camera X.

   ReadRunner displays the IO Settings window, as shown in Figure 4–41.

FIGURE 4–41. I/O Settings Window



3. Select Use Trigger on OPTO IN 1.
4. From the Triggering Edge pull down menu, select one of the following triggering edges:

- Low to High
- High to Low
- Both

5. Click Apply.
6. Click OK.

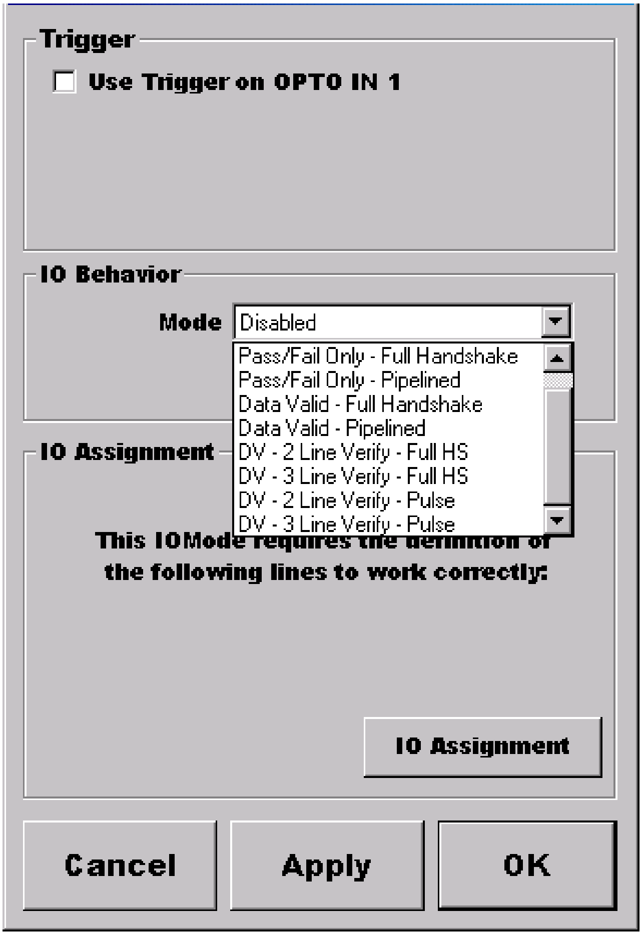
Advanced I/O

To specify the I/O behavior mode:

1. Press Ctrl+F6, or select IO Settings from the Settings menu.
2. Click Yes when asked if you wish to control camera X.

   ReadRunner displays the IO Settings window, as shown in Figure 4–42.

FIGURE 4–42. I/O Settings Window - Modes Displayed



3. From the IO Behavior Mode pull down menu, select one of the following:

   – Disabled

   – Pass/Fail Only - Full Handshake — Handshake mode with OPT1 = Pass, OPT2 = Fail.

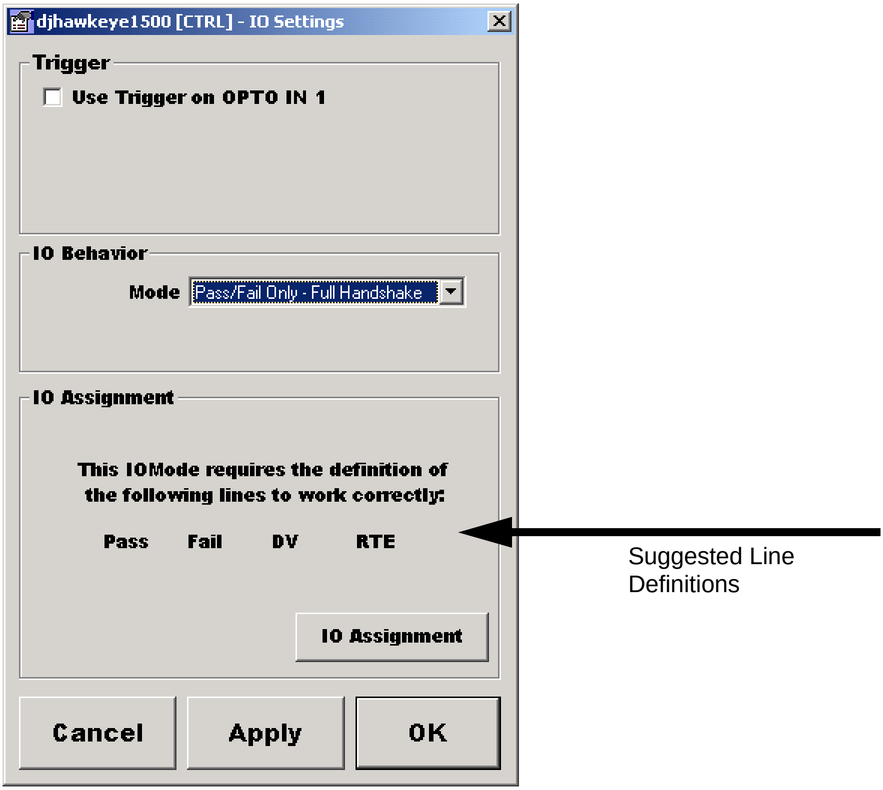
   – Pass/Fail Only - Pipelined — PULSE/PIPELINED mode with OPT1 = Pass, OPT2 = Fail.

- Data Valid - Full Handshake — Handshake mode with OPT1 = Pass, OPT2 = Fail, OPT3 = Data Valid.
- Data Valid - Pipelined — PULSE/PIPELINED mode with OPT1 = Pass | Fail, OPT3 = Data Valid.
- DV - 2 Line Verify - Full HS — GOOD, FAIR, POOR multiplexed on two assigned lines plus Data Valid.
- DV - 3 Line Verify - Full HS — GOOD, FAIR, POOR, Data Valid, lines are assigned
- DV - 2 Line Verify - Pulse — GOOD, FAIR, POOR multiplexed on two assigned lines plus Data Valid.
- DV - 3 Line Verify - Pulse — GOOD, FAIR, POOR, Data Valid, lines are assigned.

4. Click Apply.
5. Click OK.

`When you select an IO Mode, ReadRunner displays suggested line definitions to guide you through the next step (IO Assignment), as shown in Figure 4–43. See also Table 4–5.`

FIGURE 4–43. I/O Settings Window - Suggested Line Definitions



The default IO Assignment that has been pre-configured uses the same physical lines that were used in earlier versions of the HawkEye™ software.

Note: I/O points are NOT set automatically! You must set them manually.

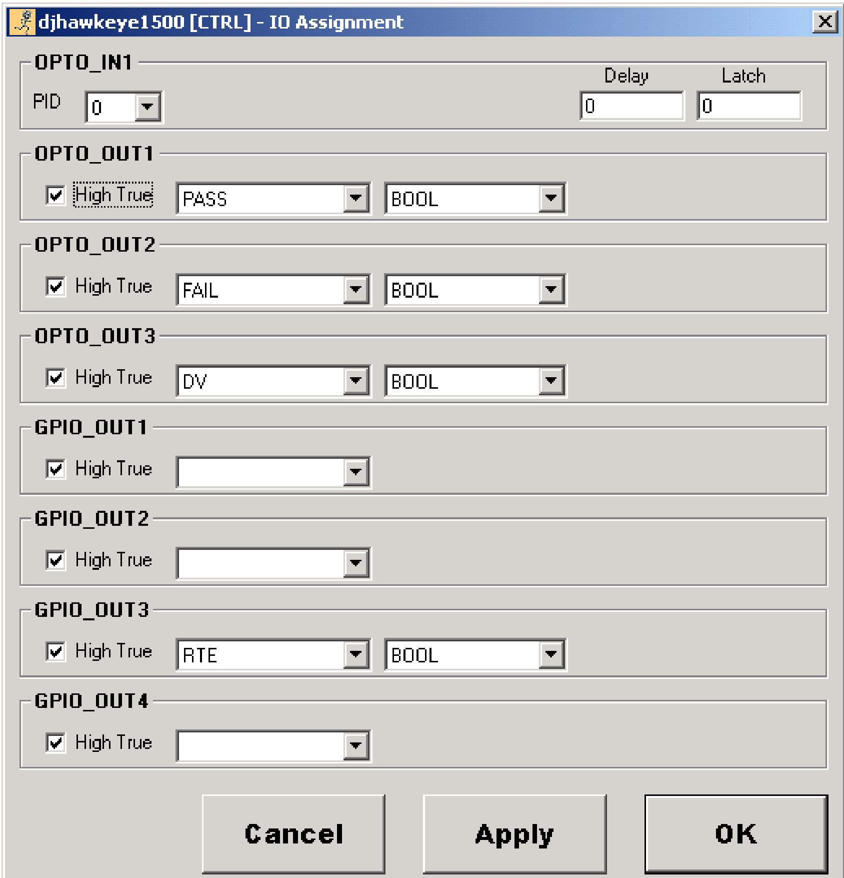
TABLE 4–5. Suggested Line Definitions

I/O Mode	Suggested Line Definitions
Data Valid - Full Handshake	Pass, Fail, DV, RTE
Data Valid - Pipelined	Pass, Fail, DV, RTE
DV - 2 Line Verify - Full HS	DV, RTE, Good, Poor
DV - 2 Line Verify - Pulse	DV, RTE, Good, Poor
DV - 3 Line Verify - Full HS	DV, RTE, Good, Fair, Poor
DV - 3 Line Verify - Pulse	DV, RTE, Good, Fair, Poor
Pass/Fail Only - Full Handshake	Pass, Fail, DV, RTE
Pass/Fail Only - Pipelined	Pass, Fail, DV, RTE

6. Click IO Assignment.

   ReadRunner displays the IO Assignment window, as shown in Figure 4–44.

FIGURE 4–44. **I/O Assignment Window**



Each of the configurable lines is displayed with the polarity, keyword name, and behavior desired.

Note: A given keyword name may only be configured for one physical line at a time.

Supported Keyword Names

The supported keyword names are:

- PASS — May also represent a MATCHED state if a Match Mode is enabled.

- FAIL — May also represent a NO MATCH state if a Match Mode is enabled.
- DV — Data Valid - This line is set on when results of a decode have been set and other lines are stable.
- RTE — Real Time Error - Indicates trigger overflows, timeout errors, and network failures.
- GOOD — (Used by the Verification option).
- FAIR — (Used by the Verification option).
- POOR — (Used by the Verification option).
- ANGLE_FAILURE — Signals when the measured angle is outside of your process limits.
  - Default Behavior = NOMINAL_DEV
  - Nominal Value = 0
  - Deviation Value = 15
- LOCATE_FAILURE — May be used as an overall indicator or masked to signal particular types of locate failures.
  - Default Behavior = MASK
  - Mask Value = &H7FFFFFF
- DECODE_FAILURE — If used when Match Mode is enabled, it allows you to distinguish between NO MATCH with a successful Decode and NO MATCH with an unsuccessful Decode.
- MATCH_FAILURE — When used in conjunction with the PASS signal when a Match Mode is enabled, provides two outputs that differ only in whether the match passed or failed. Both PASS and MATCH_FAILURE indicate that the symbol was successfully decoded.
- READY — This keyword indicates the online status of the camera. This output could be used to indicate to a PLC that the camera is ready to accept commands and trigger inputs.
- READ_A — This keyword configures the line as an input to trigger a read with a specified PID, trigger edge, latch, and delay values.

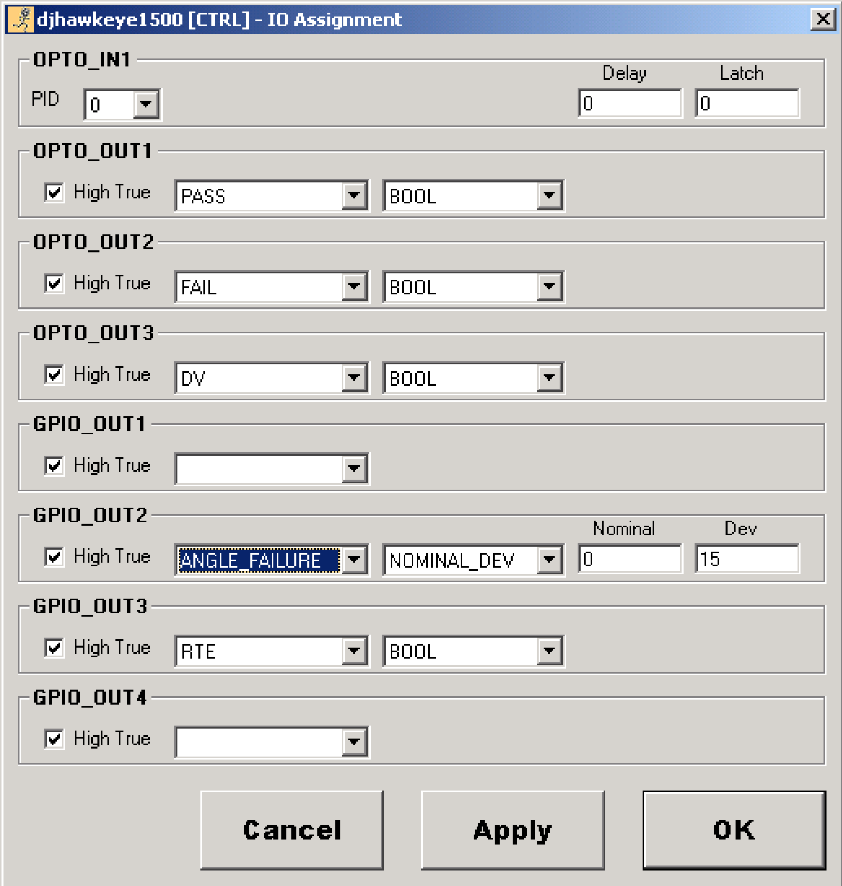
- READ_B — This keyword configures the line as an input to trigger a read with a specified PID, trigger edge, latch, and delay values.
- READ_C — This keyword configures the line as an input to trigger a read with a specified PID, trigger edge, latch, and delay values.
- READ_D — This keyword configures the line as an input to trigger a read with a specified PID, trigger edge, latch, and delay values.
- RETRY — This keyword configures an input line that is used with the retry mode “RETRY input Duration.” The level of the line is tested to determine when to stop retrying.
- LEARN — This keyword allows an external input to trigger an acquisition with a decoder and photometry learn.
- MATCH_SET — Triggers a “LEARN” of just the Match String; it does not affect the decoder or photometry settings.

Supported Behaviors

The supported behaviors for a given keyword name are:

- BOOL — A Boolean that is either ON or OFF. This is the default behavior for PASS, FAIL, DV, RTE, GOOD, FAIR, POOR, and DECODE_FAILURE keyword names.
- MASK — A bit mask is AND’ed to the keyword name’s value. If the result is 0, then the line is turned OFF. Otherwise, the line is turned ON. This is the default behavior for the LOCATE_FAILURE.
- NOMINAL_DEV — If the keyword name’s value is within the range of a nominal value, plus or minus a deviation value, the line is turned OFF. Otherwise, the line is turned ON. This is the default behavior for ANGLE_FAILURE.

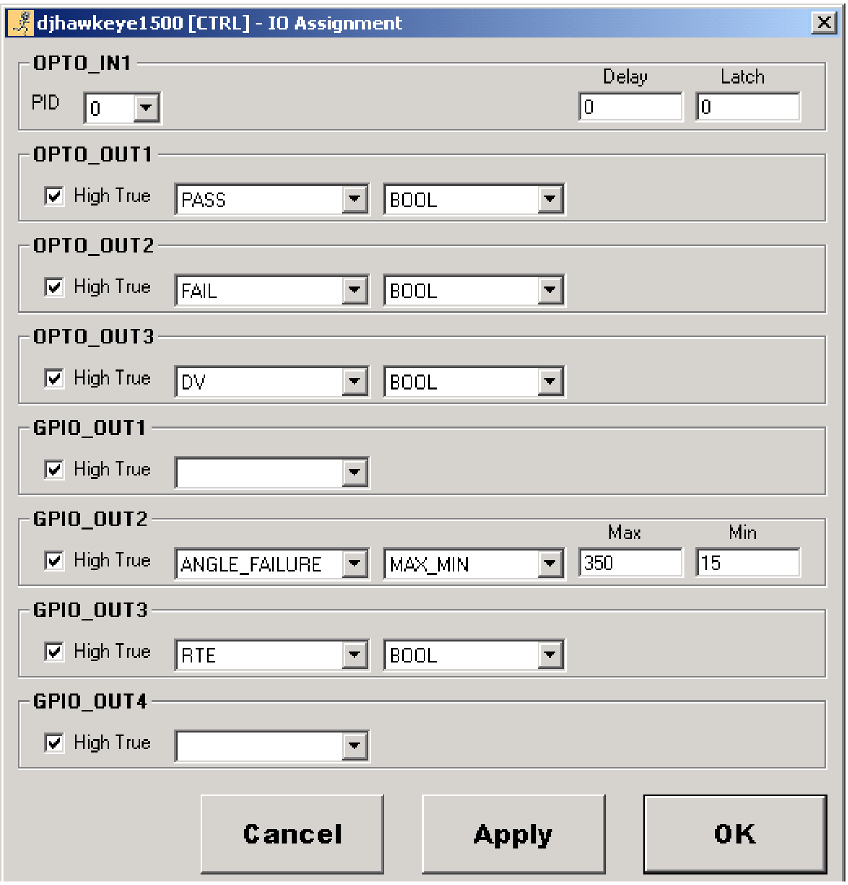
FIGURE 4–45. I/O Assignment Window - NOMINAL_DEV Behavior



When used with the ANGLE_FAILURE keyword name, the NOMINAL_DEV behavior understands that angles are in the range of 0° through 359°. The example in Figure 4–45 will consider that there is an ANGLE_FAILURE if the measured angle is less than 345° and greater than 15°. The deviation value may be 0°.

- MAX_MIN — If the keyword name’s value is within the range of a maximum value to a minimum value, the line is turned OFF. Otherwise, the line is turned ON. This is an alternate behavior for ANGLE_FAILURE.

FIGURE 4–46. I/O Assignment Window — MAX_MIN Behavior



When used with the ANGLE_FAILURE keyword name, the MAX_MIN behavior understands that angles are in the range of 0° through 359°. The example in Figure 4-46 will consider that there is an ANGLE_FAILURE if the measured angle is less than 350° and greater than 15°. MAX value equals MIN value, if no range is desired, and the measured angle must be equal to a single value. Otherwise, the ANGLE_FAILURE line will be ON.

- INPUT_POS — This behavior determines the trigger direction of an input; triggering is on a rising edge.
- INPUT_NEG — This behavior determines the trigger direction of an input; triggering is on a falling edge.

4 ReadRunner

- INPUT_BOTH — This behavior determines the trigger direction of an input; triggering is on both edges.

For additional information about IO Behavior Modes, see “Outputs” on page 3-19.

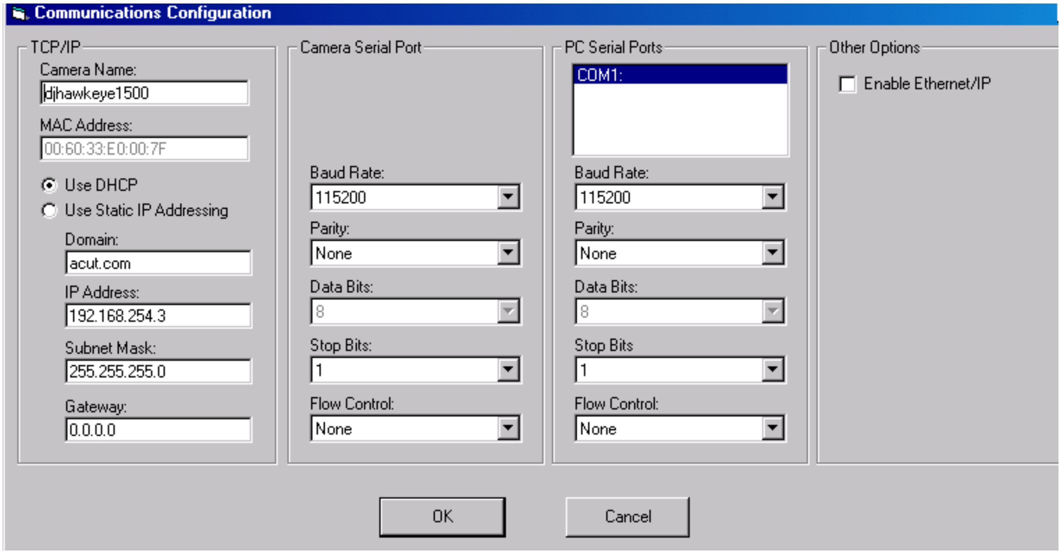
Serial/TCP Settings

The HawkEye™ 1500 camera can communicate using either RS-232 serial communications or TCP/IP.

When using TCP/IP, the HawkEye™ 1500 camera can utilize Dynamic Host Configuration Protocol (DHCP) for dynamic IP addressing. You can also configure the camera to use a “static” IP address. If you do not have a network connection available, you must use RS-232 communications to configure and control the camera. If you do have a network connection available, please contact your MIS administrator to determine if your network uses DHCP addressing, or to obtain a unique static IP address. To display the Communications Configuration window:

1. Press Ctrl+J, or select Serial/TCP Settings from the Settings menu.
2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Communications Configuration window, as shown in Figure 4–47.

FIGURE 4–47. Communications Configuration Window



3. Specify the type of IP addressing that you will be using, either DHCP or static IP addressing.

Note: If you select Use DHCP, you only have to specify Camera Name and Domain.

If you select Use Static IP Addressing, you also have to specify IP Address, Subnet Mask, and Gateway.

– TCP/IP

- Camera Name — The name of the camera, which cannot exceed 15 characters.

  At boot time, if a camera’s network name is “HawkEye,” the camera automatically changes “HawkEye” to “HawkEyexxyyzz” where xxyyzz is the last three fields of the camera’s MAC address. For example:

TABLE 4–6. Unique Camera Names

Camera Name at Boot Time	MAC Address	Unique Camera Name
HawkEye	00:60:33:e0:00:11	HawkEyeE00011
HawkEye	00:60:33:e0:00:12	HawkEyeE00012
HawkEye	00:60:33:e0:00:13	HawkEyeE00013
HawkEye	00:60:33:e0:00:14	HawkEyeE00014
HawkEye	00:60:33:e0:00:15	HawkEyeE00015

- MAC Address — The burned in address of the camera's network adapter. This is set in the factory; you cannot change it.

Note: Each camera's unique mac address is labeled on the bottom of the unit.

- Domain — The domain. For example, acut.com
- Use DHCP — Use dynamic IP addressing (DHCP) instead of static addressing. Contact your MIS administrator for information about DHCP.
- Use Static IP Addressing — Use static IP addressing instead of dynamic addressing (DHCP). Contact your MIS administrator for information about obtaining a static IP address. You must specify the following:

  IP Address — The IP address of the HawkEye™ 1500 camera.

  Subnet Mask — The subnet mask for the HawkEye™ 1500 camera.

  Gateway — The gateway address.

– Camera Serial Port

- Baud Rate — Specifies the speed at which the camera will operate.

  Range: 600 to 115200

- Parity — Is None, Odd, or Even.

- Data Bits — Is 7 or 8.
- Stop Bits — 1 or 2.
- Flow Control — Is None, Xon/Xoff, or Hardware.

– PC Serial Ports — Select a COM port before setting the following:

- Baud Rate — Specifies the speed at which the selected COM port will operate.

  Range: 600 to 115200
- Parity — Is None, Odd, or Even.
- Data Bits — Is 7 or 8.
- Stop Bits — 1 or 2.
- Flow Control — Is None, Xon/Xoff, or Hardware.

Ethernet/IP Connectivity

Note: Enabling Ethernet/IP directly impacts the network and should not be enabled casually on a non-manufacturing network. Contact your manufacturing network administrator before adding Ethernet/IP devices.

The HawkEye™ 1500 supports Ethernet/IP connectivity using an implementation of a Generic Device according to the Ethernet/IP Specifications from the Open DeviceNet Vendor Association. The corresponding terminal command is:

EIPENABLE {Y | N | ON | OFF}

This option is stored in the BOOTPARAMS section of flash.

To enable/disable Ethernet/IP, click to select/de-select the Enable Ethernet/IP checkbox (see Figure 4–47).

Note: You must reboot the camera for this action to take effect.

The camera identifies itself as a Generic Device (type 0). It supports a series of objects that reflect the physical I/O, as well as the CycleReport data, of the camera. The Ethernet/IP objects allow access to individual points or to groups of points. The camera has 8 I/O points, as shown in Table 4–7.

TABLE 4–7. Camera I/O Point Assignments

Bit/Instance	I/O Point
1	Opto In 1 (trigger)
2	Opto Out 1
3	Opto Out 2
4	Opto Out 3
5	TTL I/O 1
6	TTL I/O 2
7	TTL I/O 3
8	TTL I/O 4

The I/O static objects are:

- Discrete Output Object (class code 9, instance 1-8) — This object supports the Get_Attribute_Single/Set_Attribute_Single common services and the "Value" attribute (ID 3). "Status" data is not supported. This object allows the client to access specific values of I/O. Data is read/returned as individual bytes (zero or non-zero). For I/O points programmed as "inputs" on the camera, values are virtually represented on the camera, but the physical inputs themselves remain unchanged. You cannot write the camera outputs, only the camera inputs.
- Assembly Object (class code 4, instance 44) — This object gets the current state of the I/O points as a single byte as listed in Table 1. It supports Get_Attribute_Single service (ID 0x0E) for attribute ID 3.
- Assembly Object (class code 4, instance 34) — This object gets or sets the current state of the I/O points to/from a single byte as listed in Table 1. It supports Get_Attribute_Single service (ID 0x0E) or the Set_Attribute_Single service (ID 0x10) for attribute ID 3. As with the DIO object above, I/O points that are programmed as physical inputs to the camera are represented in software and can be changed by you. For example, if the s/w detects a transition on Opto In 1 consistent with a trigger signal, it will virtually trigger the acquisition.

The camera also supports a vendor-specific static Assembly object to represent camera results. The ID of this object is 300 (vendor-specific) and supports the Get_Attribute_Single service (ID 0x0E) or the Set_Attribute_Single service (ID 0x10) for attribute ID 3. The attribute gets/sets an array of bytes. The total structure size is 376 bytes or 94 long words (30 long words + 256 bytes of decode data). The bytes are described in Table 4–8.

TABLE 4–8. CycleReport Assembly Data Bytes

Long Word	Bytes	Meaning
1	1-4	User-defined keyword value
2	5-8	Count of triggers
3	9-12	Count of passes
4	13-16	Count of failures
5	17-20	Count of successful locates
6	21-24	Count of failed locates
7	25-28	Count of successful decodes
8	29-32	Count of failed decodes
9	33-36	Count of successful matches
10	37-40	Count of failed matches
11	41-44	Count of alarms
12	45-48	Count of trigger overruns
13	49-52	Count of decode overruns
14	53-56	Count of acquisition timeouts
15	57	Pass (1) /Fail (0)
	58	Located (1) /Not Located (0)
	59	Decoded (1) /Not Decoded (0)
	60	Matched (1) /Not Matched (0) (0 if matching is not active)
16	61-64	Detailed Decode Failure
17	65-68	Overall AIM Verification Grade

TABLE 4–8. CycleReport Assembly Data Bytes (Continued)

Long Word	Bytes	Meaning
18-21	69-87	Decode Data as a set of numbers (when possible). Numbers are converted as sets of 9 digit decimals, up to four total numbers
22-29	88-128	Reserved for future use
30	120	Length of decode data
31-94	124-376	256 bytes of formatted decode data

When setting the attribute, only the user-defined keyword value is persistent. The data is restored on the next cycle.

The camera also supports custom assembly objects for combining camera I/O and cycle data as well as for receiving camera commands. These objects are:

- Assembly Object (class code 4, instance 1) — This object gets the current state of the I/O points as well as all the cycle data. This object supports I/O connectivity and update with the PLC implicitly. The data format is a series of DINTs. The first byte of the first DINT represents the I/O as listed in Table 1. The other bytes are currently ignored. The rest of the DINT data represents the cycle data as listed in Table 2 except for the formatted decode data and its length. This object thus has 30 DINTs of data. It also support Get_Attribute_Single service (ID 0x0E) for attribute ID 3.
- Assembly Object (class code 4, instance 2) — This object sets the current state of the I/O points and also allows you to program the “user keyword” in the cycle data structure. This object has 2 DINTs. The first byte of the first DINT represents the I/O as listed in Table 1. The other bytes are currently ignored. The second DINT is the user-defined keyword. This object supports implicit I/O connectivity and the Get_Attribute_Single service (ID 0x0E) or the Set_Attribute_Single service (ID 0x10) for attribute ID 3. As with the DIO object above, I/O points that are programmed as physical inputs to the camera are represented in software. For example, if the s/w detects a transition on Opto In 1 consistent with a trigger signal, it will virtually trigger the acquisition.
- Assembly Object (class code 4, instance 301) — This object sends commands to the camera via Ethernet/IP. The object supports 260 bytes of data. The first four is a DINT that contains the command length. The remaining 256 bytes is the command string. This object does not support implicit I/O connectivity. It supports the Get_Attribute_Single service (ID

0x0E) or the Set_Attribute_Single service (ID 0x10) for attribute ID 3. When the Set_Attribute_Single service is called, the command is executed. The command is executed asynchronously and there is no specific command handshake.

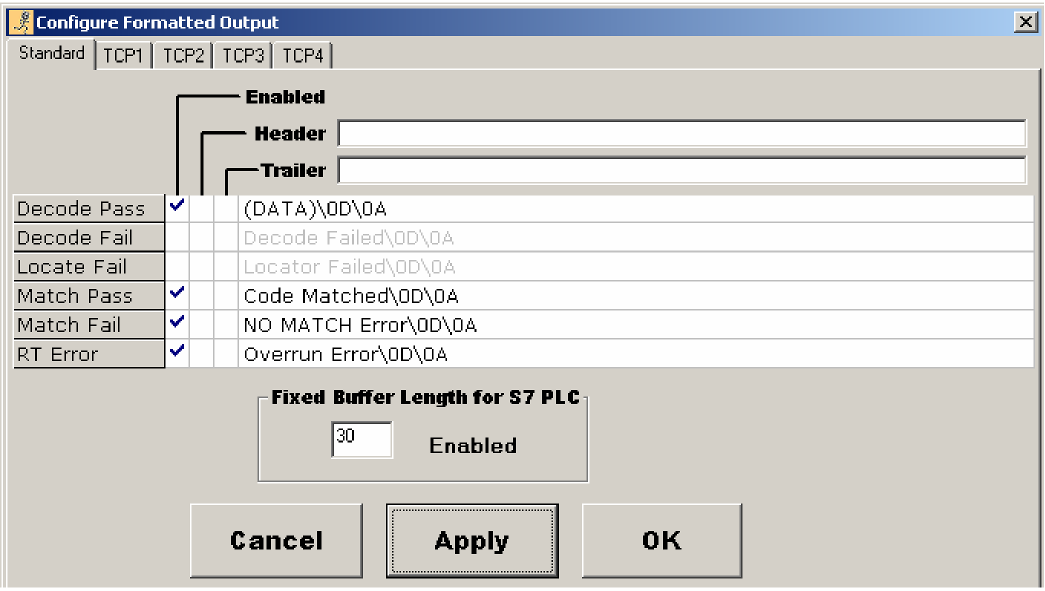
A detailed technote (HawkEye™ 1500 Tech Note - EIP Communicating with ControlLogix PLC.pdf) can be found on the Release CD. This document provides information on how to set up the ControlLogix PLC to communicate with the Camera via Ethernet/IP.

Output Format Strings

The output format string specifies the text that the camera sends out the serial port and over TCP for various conditions (Pass, Fail, Error), and whether or not the camera beeps with each condition. To display the Output Format String window:

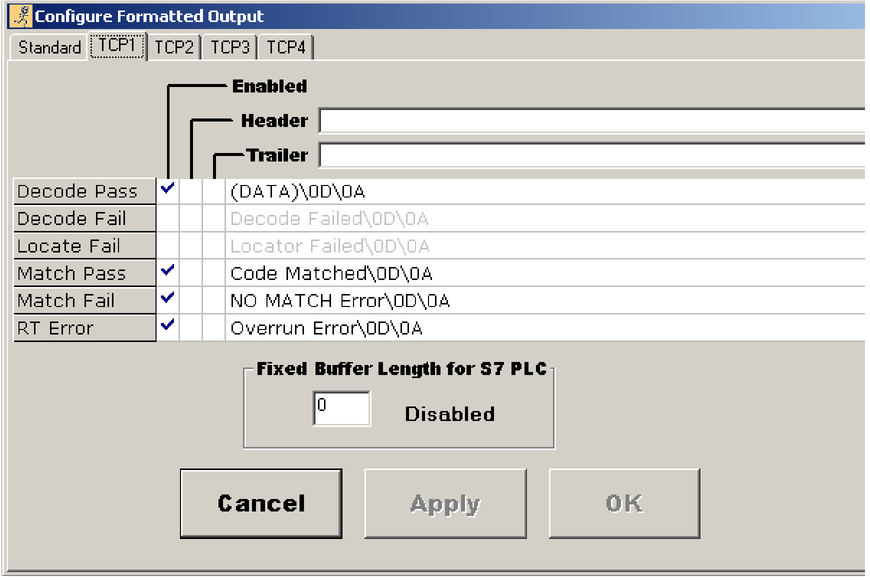
1. Press Ctrl+G, or select Output Settings from the Settings menu.
2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Communications Configuration window, as shown in Figure 4–48.

FIGURE 4–48. **Output Format Strings Window (Standard)**



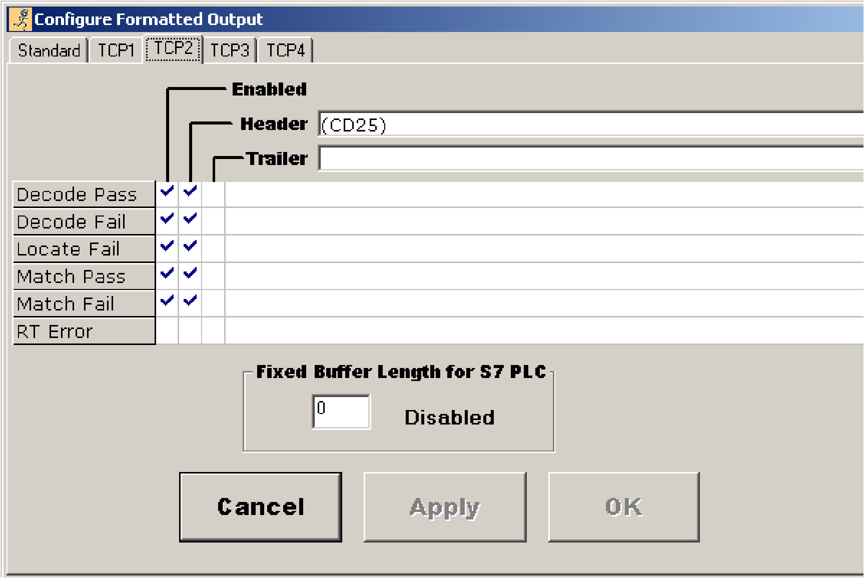
- Standard (Camera Serial Port) — By default, this port is programmed as shown in Figure 4–49.
- TCP1 — By default, this port is programmed as shown in Figure 4–49.

FIGURE 4–49. Output Format Strings Window (TCP1 - Port 49098)



- TCP2 — By default, this port is programmed to send the raw image data (CD25), as shown in Figure 4–50.

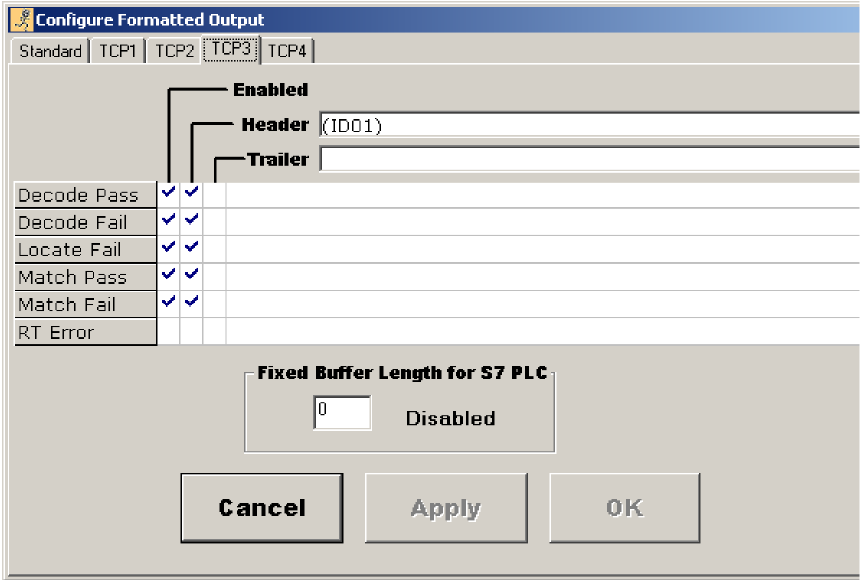
FIGURE 4–50. Output Format Strings Window (TCP2 - Port 49099)



- TCP3 — By default, this port is programmed to send the image data as a bmp binary stream (ID01) (32 bit length/data), as shown in Figure 4–51.

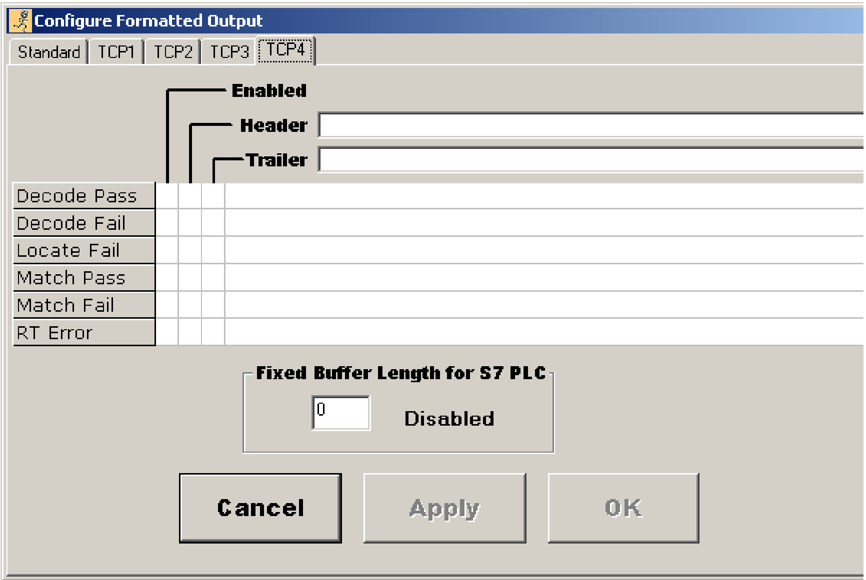
Note: The keyword is programmed in the Header so that it can be entered only once for all programmable output strings.

FIGURE 4–51. **Output Format Strings Window (TCP3 - Port 49100)**



- TCP4 — By default, this port is not programmed, as shown in Figure 4–52.

FIGURE 4–52. Output Format Strings Window (TCP4 - Port 49101)



- Fixed Buffer Length for S7 PLC — The fixed length data string output can be enabled from the ReadRunner UI or through a command. To enable it from the UI, set the Fixed Buffer Length parameter in the Configure Formatted Output dialog to a value greater than 1 but no more than 254. To disable the mode, set the Fixed Buffer Length to 0 or 1.

  `Alternatively, use the following command to enable/disable the fixed length output:`

  ```
  OUTPUT {port} FIXED {count}
  ```

  `where:`

  – port is one of the following:

    - STANDARD
    - TCP1
    - TCP2
    - TCP3
    - TCP4

– count = 0 - 254

The fixed length output is enabled when the count is between 2 and 254 (inclusive) and disabled when the count is 0 or 1.

All five ports (STANDARD for serial and four TCP ports) can be configured individually to output data in a different fixed length or to not enable the fixed length mode at all.

When the fixed length string output mode is enabled, the formatted output data (up to 252 characters) will be pre-pended with two characters as follows:

char1 char2 “formatted output data”

The first character in the string (char1) has the “type” information which contains the ASCII value of the following characters:

– O — Indicates that the formatted output data is the result of Decode Pass event 'O'KDEC

– D — Indicates that the formatted output data is the result of Decode Fail event 'D'ECFL

– L — Indicates that the formatted output data is the result of Locate Fail event 'L'OCFL

– M — Indicates that the formatted output data is the result of Match Pass event OK'M'ATCH

– N — Indicates that the formatted output data is the result of Match Fail event MATCHFL ('N'o Match)

– R — Indicates that the formatted output data is the result of RT Error event 'R'TE

The second character in the string (char2) contains the length value of the formatted output data. If the length of the formatted output data is less than (count - 2), then the characters after the last character of the formatted output will be replaced by '@'. If the length of the formatted output data is more than
(count - 2), then the formatted output data will be truncated such that only the first (count-2) characters will be outputted with the char2 containing the length value of (count - 2).

Each channel can be setup individually, each with their own Header and Trailer strings.

A detailed list of all keywords can be found in the next section.

Format String Keywords

- (ANGLE) — This keyword expands to the orientation in degrees (0 to 359) of the code that was read.
- (CHECKSUM) — This keyword expands to the Checksum of the characters in the appropriate Output Format String where the (CHECKSUM) keyword is inserted. The two character ASCII value representation of the Hexadecimal checksum (for example, 2E) is substituted in the Output string. The checksum is calculated as the Exclusive OR (XOR) of all the characters up to the (CHECKSUM) keyword, including the Header and the Trailer characters if (CHECKSUM) is placed at the end of the trailer string.
- (DATA) — This keyword will be substituted with the actual decoded string. For example, if “4567321” were decoded, and the event were set as “OKDEC: This is the decoded data: (DATA)\0d\0a”, then the one line of output will be followed by carriage return and line feed.
- (DATAHEX) — This keyword converts the decode data to a hex character string, similar to the previous “OUTPUT FORMAT HEX” functionality.
- (DETAILED) — This keyword returns a detailed failure string.
- (PID#) — This keyword returns the PID number of the PID that successfully decoded the symbol.
- (TIMESTAMP) — This keyword return the time stamp of the trigger. This expands to a date/time string in GMT (e.g., “THU APR 03 15:42:52 2003”).
- (VERI_1_IAQG) — This keyword provides a DMx AutoID compatibility mode for the IAQG Verification 1 report. This allows the HawkEye™ 1500 to be used as a “drop in” replacement for a DMX AutoID.

```
Verification 1 reporting in DMxAutoID+ v1.5.2:

0236514397793412001;F;65;24;B;19;0;A;0,9;1,00

0236514397793412001; is the decoded string
F;65;24; are Dot Size Grade, Dot Size Count1, and Dot Size
Count2
```

4

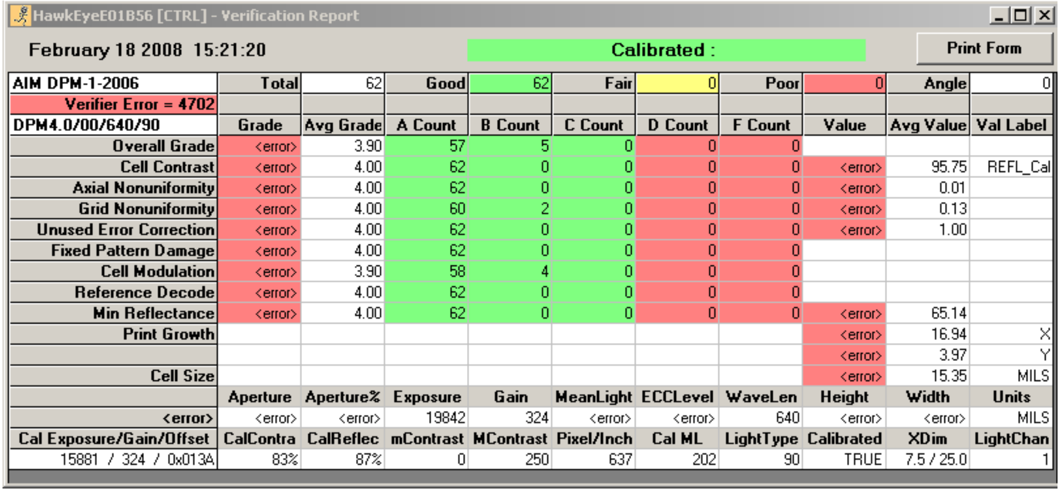
ReadRunner

```
B;19;0; are Dot Center Grade, DC Count1, and DC Count2
A;0,9; are Angle of Distortion Grade and angle (note
representation of 0.9)
1,00 is the Average cell fill (note representation of 1.00)
```

- (VERI_DETAIL) — This keyword return the detailed verification data, separated by semicolons (“;”), but only when Verification is enabled. This data includes the overall grade, grade contrast, contrast, grade axial number, AIM axial non-uniformity value, grade print growth, AIM print growth X, AIM print growth Y, grade error correction, number of error bits, and the UEC value.
- (VERI_FORMATTED) — This keyword returns the formatted verification data. This is similar to VERI_DETAIL, but each value is identified by an acronym title and is separated by spaces.
- (VERI_GRADE) — This keyword returns the overall verification grade as a number, but only when Verification is enabled.
- (VERI_STATUS) — This keyword returns the verification status available when Verification is enabled. The possible values are:
  - 3 - Good
  - 2 - Fair
  - 1 - Poor
  - 0 - No Verification Enabled
  - 4700 - Verification Process Error — The decode was successful but possibly, because the code was seen too close to the edge of the image, all of the verification calculations could not be completed.

    The value is also displayed in the upper left corner of the Verification Report when it occurs, as shown in Figure 4–53.

FIGURE 4–53. Value Displayed in Verification Report



HawkEyeE01B56 [CTRL] - Verification Report

February 18 2008 15:21:20 — Calibrated : — Print Form

AIM DPM-1-2006	Total	62	Good	62	Fair	0	Poor	0	Angle	0
Verifier Error = 4702										
DPM4.0/00/640/90	Grade	Avg Grade	A Count	B Count	C Count	D Count	F Count	Value	Avg Value	Val Label
Overall Grade	<error>	3.90	57	5	0	0	0			
Cell Contrast	<error>	4.00	62	0	0	0	0	<error>	95.75	REFL_Cal
Axial Nonuniformity	<error>	4.00	62	0	0	0	0	<error>	0.01	
Grid Nonuniformity	<error>	4.00	60	2	0	0	0	<error>	0.13	
Unused Error Correction	<error>	4.00	62	0	0	0	0	<error>	1.00	
Fixed Pattern Damage	<error>	4.00	62	0	0	0	0			
Cell Modulation	<error>	3.90	58	4	0	0	0			
Reference Decode	<error>	4.00	62	0	0	0	0			
Min Reflectance	<error>	4.00	62	0	0	0	0	<error>	65.14	
Print Growth								<error>	16.94	X
								<error>	3.97	Y
Cell Size								<error>	15.35	MILS
	Aperture	Aperture%	Exposure	Gain	MeanLight	ECCLevel	WaveLen	Height	Width	Units
<error>	<error>	<error>	19842	324	<error>	<error>	640	<error>	<error>	MILS
Cal Exposure/Gain/Offset	CalContra	CalReflec	mContrast	MContrast	Pixel/Inch	Cal ML	LightType	Calibrated	XDim	LightChan
15881 / 324 / 0x013A	83%	87%	0	250	637	202	90	TRUE	7.5 / 25.0	1

– 4701 - Verification Unsupported — The decode was successful but verification is not supported for the type of barcode seen.

– 4702 - Verification Timeout — The decode was successful but the verification calculations could not be completed before a new decode request was received.

- (UII_ONLY) — UII/CPN only can be enabled with the keyword UII_ONLY for formatted output. When this option is enabled, the UII/CPN data will be added to the formatted output based on the position of the keyword. The UII/CPN output has the following format: UII/CPN.

TABLE 4–9. UII/CPN

Content of Decoded Data Matrix	UII/CPN
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 & CPN Valid UII & 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 & invalid CPN Valid UII & invalid LBN	UII:UII_data (CPN ERROR) Example: UII:12345678 (CPN ERROR) UII:UII_data (LBN ERROR) Example: UII:12345678 (LBN ERROR)
Invalid UII & valid CPN Invalid UII & 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 & Invalid CPN; Invalid UII & Invalid LBN)	No output data

- (UII_WITH_INFO) — UII/CPN with Data Fields can be enabled with the keyword UII_WITH_INFO for formatted output. When this option is enabled, the output data corresponding to this keyword will have the following format: UII/CPN; DF0; DF1; DF2; DF3; DF4; DF5; DF6; DF7.

TABLE 4–10. UII/CPN with Data Fields

Content of Decoded Data Matrix	UII/CPN	DF0
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 & CPN Valid UII & 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 & Invalid CPN Valid UII & 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 & Valid CPN Invalid UII & 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 & Invalid CPN Invalid UII & 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

Notes:

1. The error messages, xxxx, are listed in Table 3–6, “Report Error Codes,” on page 3-32.

2. 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 less than seven data elements, an empty string is filled in at the end. If there are more than seven elements, only first seven elements are displayed.

3. There is a space between UII and CPN in both tables (UII:12345678 CPN:87654321).

4. The constructed UII type can be Contruct_1, Contruct_2, Construct_1_2, or IUID_EQUIVALENT.

5. The constructed CPN type can be PNR, 30P, or 240. The constructed LBN type can be 30T.

Keyword Example

Assume that you want to display formatted verification data every time there is a successful decode. You would do the following:

1. Start ReadRunner.

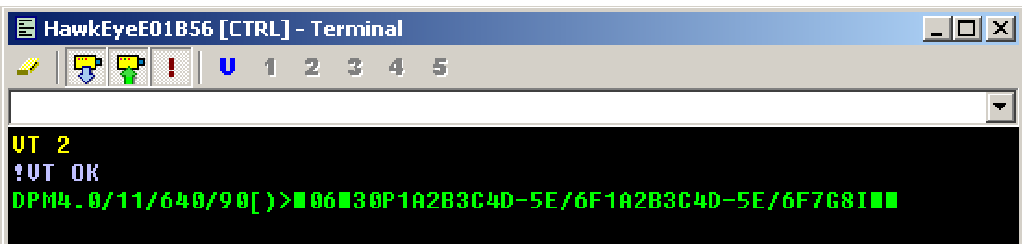
2. Click Add Camera to connect to a camera.

3. From the Settings menu, click Output Settings.

4. Click Yes when asked if you wish to control camera X.

```
With the Decode Pass as (VERI_AIMDPM)(DATA)\0D\0A in
Figure 4-54, ReadRunner displays the Output setting window
shown in Figure 4-54.
```

FIGURE 4–54. Output Settings Window



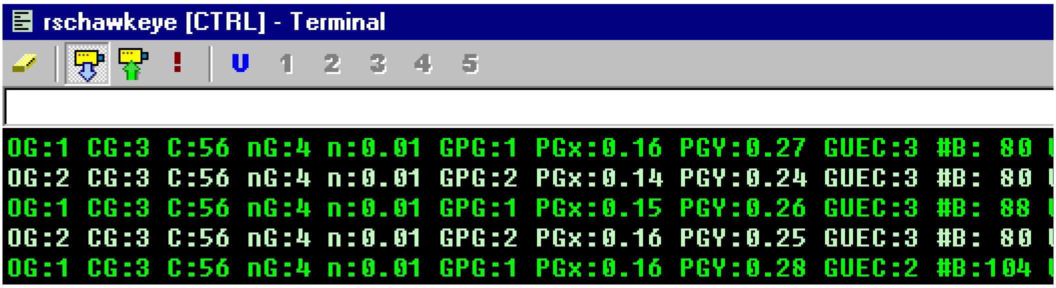
5. In the Decode Pass text box, type:

```
(VERI_FORMATTED)
```

6. Click Apply and OK.
7. From the Settings menu, click Symbology and Verification to display the Symbology / Verification window.
8. Select AIM (ISO 16022).
9. Click Apply and OK.
10. From the Windows menu, click Terminal Window.
11. Click the Display Report Output From Camera button.

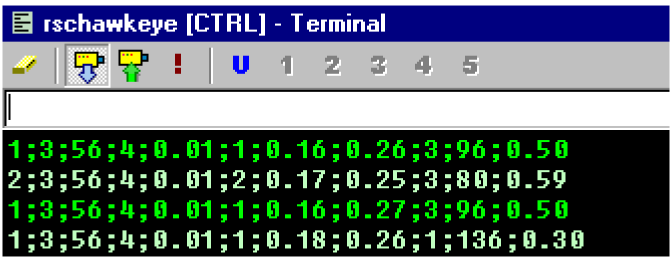
    `You will see a screen containing information similar to the following:`

FIGURE 4–55. Decode Pass Text Box Contains (VERI_FORMATTED)



12. If you replace (VERI_FORMATTED) with (VERI_DETAIL) in the Decode Pass text box of the Output Setting window, the output would look similar to the following:

FIGURE 4–56. Decode Pass Text Contains (VERI_DETAIL)



Finally, you can also customize the number of beeps that the camera emits on these pass/fail conditions. By default, the camera beeps once when the decode passes. You can configure each pass/fail condition with up to 3 beeps.

The format string keywords are furthermore expanded to allow custom binary data to be output from the camera. The camera support the new keywords “(CDxx)” and “(IDxx)” where xx is a 2-digit decimal number. The number corresponds to a specific field of data, as listed in Table 4–11 and Table 4–12. All the data that was available in the Camera report and could only be accessed by programming can now be programmed into the Formatted Output strings sent by the camera over TCP. They are not available on standard output; just TCP1 through TCP4.

TABLE 4–11. CD Fields

Field	Description
00	All Data
01	Overall Pass/Fail
02	Count of triggers
03	Count of passes
04	Count of failures
05	Count of successful locates
06	Count of failed locates
07	Count of successful decodes
08	Count of failed decodes
09	Count of successful matches
10	Count of failed matches
11	Count of alarms
12	Count of trigger overruns
13	Count of decode overruns
14	Count of acquisition timeouts
15	PID
16	Symbol Pass/Fail
17	Located / Not Located
18	Decoded / Not Decoded
19	Matched / Not Matched (0) (0 if matching is not active)
20	Detailed Decode Failure

TABLE 4–11. CD Fields (Continued)

Field	Description
21	Overall AIM Verification Grade
22	Unformatted decode data
	Length/raw data
23	Photometry Settings
	Auto|Manual/Gain/Exposure
24	Image Timestamp
25	Image Data Raw
	Width/Height/Gray-scale Values

TABLE 4–12. ID Fields

Field	Description
01	Image Data Bmp 32-bit Length/Binary Values
02	Image Data Tiff 32-bit Length/Binary values

All data is written in binary format on the corresponding output port (TCP only). Any 32-bit data is first converted to network order before being sent.

The Enabled check box specifies whether or not the data in each text box gets sent out the serial port:

- Header — Enter the header information you want displayed. When you select the checkbox for a given string, the header information precedes the text.
- Trailer — Enter the trailer information you want displayed. When you select the checkbox for a given string, the trailer information follows the text.
- Decode Pass — When checked, send the data in the text box out the serial port on every successful decode.
- Decode Fail — When checked, send the data in the text box out the serial port on every unsuccessful decode.

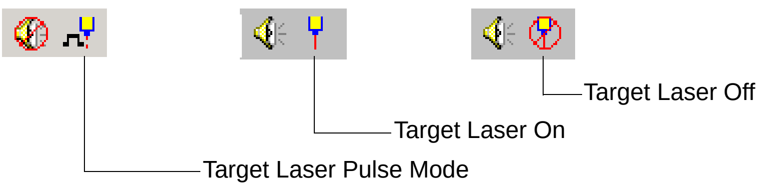
- RT Error — A Runtime error, which includes any alarms, trigger overruns or process overruns. Currently, any alarm that is set is network related: connection was dropped, connection timed out, etc.
- Locate Fail — When checked, send the data in the text box out the serial port on every unsuccessful locate.
- Match String — A string to use for matching a decoded symbol. For example, if you are reading a serial number, you would use matching to validate that the decoded symbol matched against a certain string. Wildcards are supported:
  - ? — Any single character
  - * — Any preceding or following characters
- Match Pass — When checked, send the data in the text box out the serial port on every successful match.
- Match Fail — When checked, send the data in the text box out the serial port on every unsuccessful match.

Toggling the Target Laser

The target laser helps you to align and adjust the position of the HawkEye™ 1515 and 1525 cameras. To toggle the state of the target laser, do one of the following:

- Click the target laser icon (see Figure 4–57) at the bottom of the main ReadRunner window.

FIGURE 4–57. Toggling Target Laser



- From the Application Mode window, select or de-select one of the following from the Target Laser pulldown menu:
  - Off

- On
- Strobe (2 sec)

• From the Settings menu, select Targeting Laser, and then select or de-select one of the following:

- Off
- On
- 2 Second Strobe

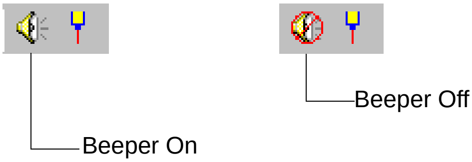
DO NOT LOOK AT THE LASER. WHEN READING PARTS THAT ARE HIGHLY REFLECTIVE, BE CAREFUL THAT THE LASER IS NOT REFLECTED INTO SOMEONE'S EYE.

Toggling the Beeper

You can toggle the beeper so that you hear/don't hear beeps from the HawkEye™ 1500 camera. This overrides any settings in the Output Settings window (see "Output Format Strings" on page 4-71). To toggle the state of the beeper, do one of the following:

1. Click the beeper icon (see Figure 4–58) at the bottom of the main ReadRunner window.

FIGURE 4–58. Toggling Beeper



2. From the Settings menu, select Beeper.

Note: A checkmark preceding Beeper in the Settings menu indicates that the beeper is on.

4 ReadRunner

3. From the keyboard, press Shift+F5.

Controlling the Beep

You can control the beeper so that you hear/don't hear beeps from the HawkEye™ 1500 camera under various conditions. To control the beeper:

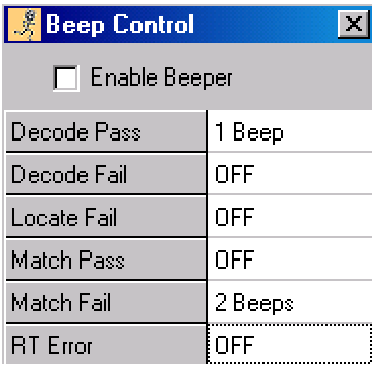
1. Do either of the following:

   a. From the Settings menu, select Beep Control.

   b. From the keyboard, press Ctrl+B.

   ReadRunner displays the Beep Control dialog box, as shown in Figure 4–59.

FIGURE 4–59. Beep Control Dialog Box



2. Specify the number of beeps, if any, you want for each condition. Your choices are:

   – OFF

   – 1 Beep

   – 2 Beeps

   – 3 Beeps

3. Select the Enable Beeper checkbox.

4. Close the Beep Control dialog box.

Saving Parameters on the Camera to Flash

This menu item saves the current memory settings to flash so that the camera is properly restored when it reboots. To save the parameters on the camera to flash:

1. Press Ctrl+S, or select Save Parameters on Camera from the File menu.
2. Click Yes when asked if you wish to control camera X. ReadRunner displays the message:

   `Camera Parameters are saved to flash`

   `ReadRunner also saves the parameters to the ReadRunner.ini file.`

3. Click OK.

Restoring Defaults

Decoder

Restoring defaults to the decoder is the same as doing an unlearn. To restore the default settings to the decoder:

1. From the File menu, select Restore Defaults and then Decoder.
2. Click Yes when asked if you wish to control camera X.

   `ReadRunner restores the decoder settings to their defaults and saves them to Flash.`

Application Mode

Restoring defaults to the Application Mode options is the same as selecting Demo Mode:

- Continuous Triggers
- No Retries
- Illumination is on
- Auto Photometry

To restore the default settings for the Application Mode:

1. From the File menu, select Restore Defaults and then Application Mode.

2. Click Yes when asked if you wish to control camera X.

   `ReadRunner restores the Application Mode options to the default information listed above and saves them to Flash.`

Decoder & Application Mode

To restore the default settings to both the decoder and the Application Mode options:

1. Press Ctrl+Z, or select Restore Defaults and then Both from the File menu.

2. Click Yes when asked if you wish to control camera X.

   `ReadRunner restores the decoder settings and the Application Mode options to their defaults and saves them to Flash.`

Displaying...

Verification Report

To display a Verification Report:

1. Press F2, or select Verification Report from the Windows menu. ReadRunner displays the Verification Report window specific to the Verification Mode selected (DPM Verification in this example), as shown in Figure 4–60.

FIGURE 4–60. Verification Report Window

DPM Verification™	Total	1944	Good	1944	Fair	0	Po
	Grade	Avg Grade	A Count	B Count	C Count	D Count	F Coun
Overall Grade	B	3.00	0	1944	0	0	
Cell Size	A	4.00	1944	0	0	0	
Center Offset	A	4.00	1944	0	0	0	
Size Offset	B	3.00	74	1870	0	0	
Cell Modulation	B	3.00	0	1944	0	0	
Border Match	A	4.00	1944	0	0	0	
Symbol Contrast	B	3.00	0	1944	0	0	
Axial Nonuniformity	A	4.00	1944	0	0	0	
Print Growth	A	4.00	1944	0	0	0	
Unused Error Correction	A	4.00	1944	0	0	0	
Angle of Distortion	A	4.00	1944	0	0	0	

Reports & Images Over A Serial Connection

You can connect to the HawkEye™ 1500 camera serially or over TCP/IP. When you connect serially, ReadRunner displays different buttons than the buttons you see when you are connected over TCP/IP, as shown in Figure 4–61.

FIGURE 4–61. Buttons When Connected Serially

Unlike when connected over TCP/IP, when connected over a serial connection you have to request that ReadRunner display reports and images. You can display both reports and images at the same time. To display reports:

1. Click Hook Reports.
2. From the pull down menu, select All or Failures.

To display images:

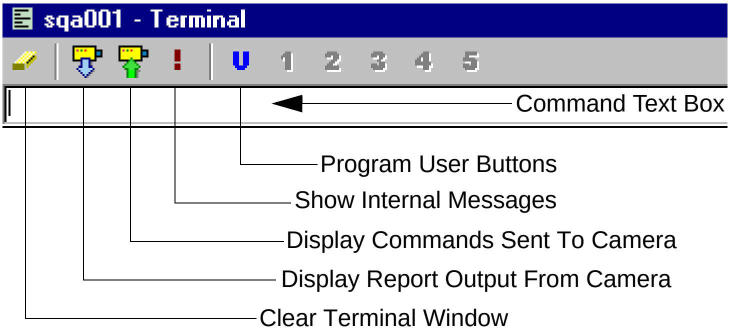
1. Click Hook Images.
2. From the pull down menu, select the amount of compression each image should have:
   - No Compression
   - 4:1 Compression
   - 8:1 Compression
   - 16:1 Compression
   - 32:1 Compression

Commands Sent To and Output From the Camera

From the Terminal window, you can display commands sent to and received from the camera, display internal commands, program user buttons, and send remote commands to the camera. To display the Terminal window:

1. Press Ctrl+T, or select Terminal Window from the Windows menu. ReadRunner displays the Terminal window, as shown in Figure 4–62.

FIGURE 4–62. **ReadRunner Terminal Window**



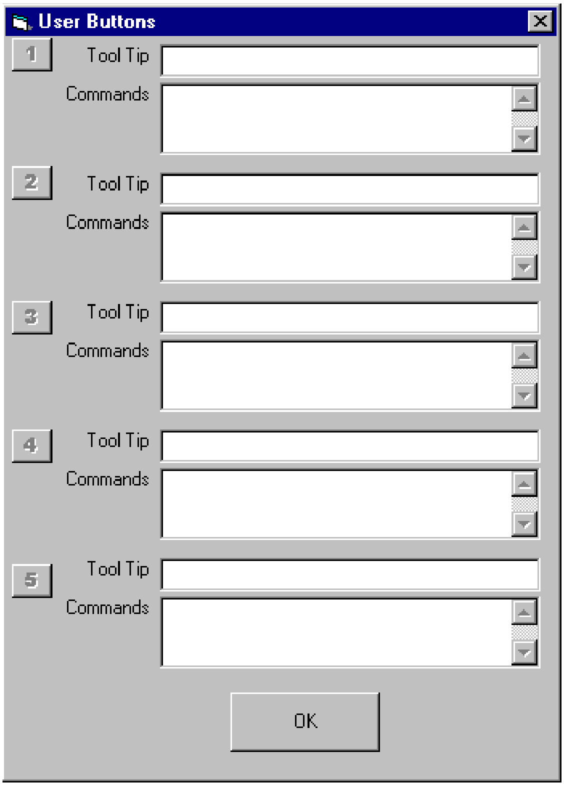
The ReadRunner Terminal window displays output from the camera, commands sent to the camera, and internal messages. From this window, you can also program five user buttons (see “Programming User Buttons” on page 4-92).

- Clear Terminal Window — This button clears the terminal window.
- Display Report Output From Camera — When selected, this button displays output from the camera. By default, this button is selected when you open the Terminal window.
- Display Commands Sent To Camera — When selected, this button displays commands that you send to the camera. By default, this button is selected when you open the Terminal window.
- Show Internal Messages — When selected, this button enables the display of every response (or “ack”) from a command sent to the camera. These responses are denoted with a “!”. By default, the display of these responses is disabled.

Programming User Buttons

User buttons contain commands that are generated when you click one of the five user buttons in the Terminal window. When you click U, ReadRunner displays the User Buttons window, as shown in Figure 4–63.

FIGURE 4–63. User Buttons Window



Specify a tool tip and the commands (you can specify more than one) for a user button. When you are finished, click OK.

Note: A user button is enabled when both the Tool Tip text box and the Commands text box contain valid information.

- Tool Tip — This text box contains the message that is displayed when the mouse pointer hovers over a user button. For example, if the Tool Tip for user button 1 contains “Virtual Trigger 1,” then “Virtual Trigger 1” is displayed when you put the mouse pointer on the 1.
- Commands — This text box contains the actual remote commands that are generated when you click a user button. For more information about remote

commands, see the HawkEye™ 1500 Series Reference & Programmers Manual.

Sending Remote Commands to the Camera

From the Terminal window, you can send remote commands to the camera in the Command Text Box area (see Figure 4–62).

Note: De-select Display Report Output From Camera before you send remote commands to the camera.

To display a list of remote commands, type:

```
help
```

To display help for a specific command, type:

```
help <command>
```

```
Where:
```

```
<command> is the name of the command
```

To display detailed information for a specific command, type:

```
help <command> verbose
```

```
Where:
```

```
<command> is the name of the command
```

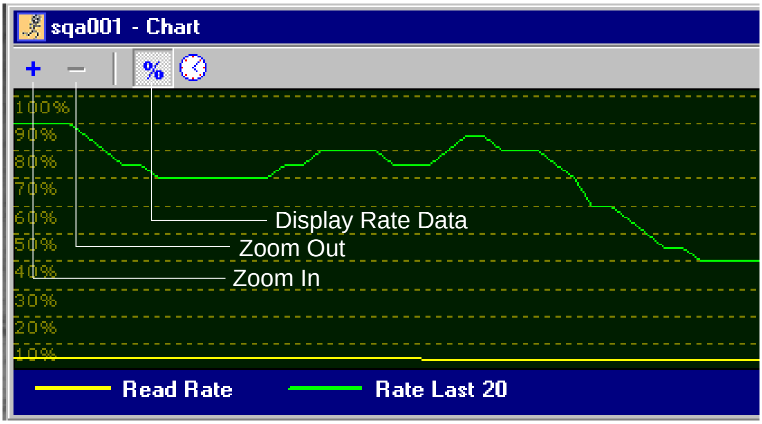
```
verbose specifies that you want to see detailed information
```

Timing & Rate Information

The Chart window displays the Timing rate and Read rate of the camera. To display the Charts window:

1. Press Ctrl+H, or select Charting Window from the Windows menu. ReadRunner displays the Chart window, as shown in Figure 4–64 (Rate data displayed) and Figure 4–65 (Timing data displayed).

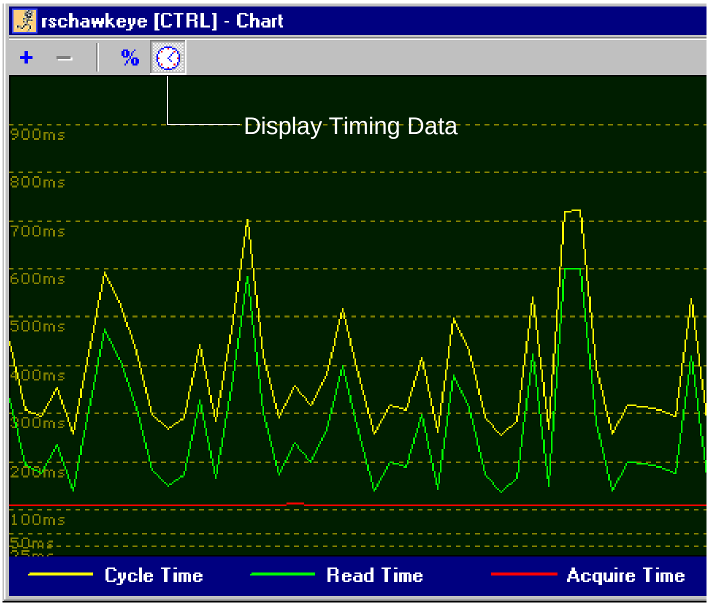
FIGURE 4–64. ReadRunner Charts Window — Rate Data



This window displays rate information about the current images that the HawkEye™ 1500 camera is reading.

- Zoom In/Out — Zoom In displays more detail; Zoom Out displays less detail.
- Display Rate Data — Displays graphically how well the camera is able to read symbols as a percentage of successful reads (100% means zero no-reads). Yellow indicates overall read rate since the camera was powered up. Green indicates read rate for the last 20 reads.

FIGURE 4–65. **ReadRunner Charts Window — Timing Data**



This window displays timing information about the current images that the HawkEye™ 1500 camera is reading.

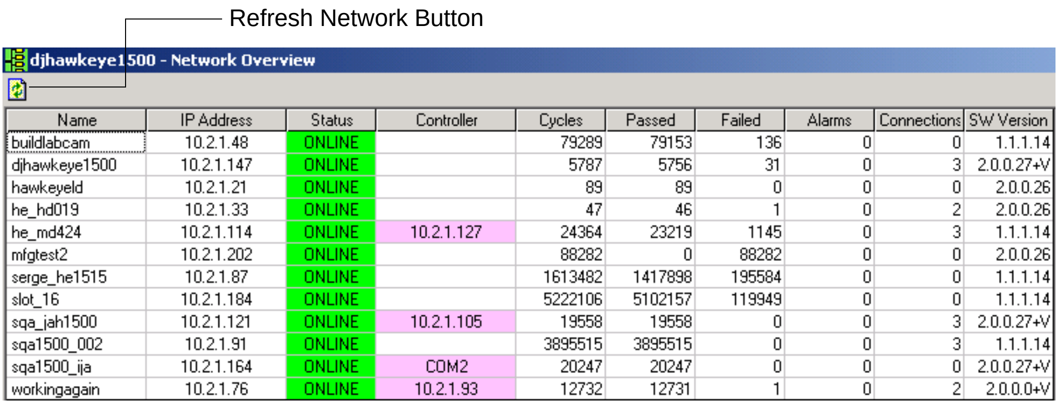
- Display Timing Data — Displays cycle time, read time, and acquire time graphically. Yellow indicates cycle time. Green indicates decode time. Red indicates acquisition time.

Information About Cameras on the Network

The Network Overview window displays information about HawkEye™ 1500 cameras on the network. To display the Network Overview window:

1. Press F12, or select Network Overview from the Windows menu. ReadRunner displays the Network Overview window, as shown in Figure 4–66.

FIGURE 4–66. ReadRunner Network Overview Window



Name	IP Address	Status	Controller	Cycles	Passed	Failed	Alarms	Connections	SW Version
buildlabcam	10.2.1.48	ONLINE		79289	79153	136	0	0	1.1.1.14
djhawkeye1500	10.2.1.147	ONLINE		5787	5756	31	0	3	2.0.0.27+V
hawkeyeld	10.2.1.21	ONLINE		89	89	0	0	0	2.0.0.26
he_hd019	10.2.1.33	ONLINE		47	46	1	0	2	2.0.0.26
he_md424	10.2.1.114	ONLINE	10.2.1.127	24364	23219	1145	0	3	1.1.1.14
mfgtest2	10.2.1.202	ONLINE		88282	0	88282	0	0	2.0.0.26
serge_he1515	10.2.1.87	ONLINE		1613482	1417898	195584	0	0	1.1.1.14
slot_16	10.2.1.184	ONLINE		5222106	5102157	119949	0	0	1.1.1.14
sqa_jah1500	10.2.1.121	ONLINE	10.2.1.105	19558	19558	0	0	3	2.0.0.27+V
sqa1500_002	10.2.1.91	ONLINE		3895515	3895515	0	0	3	1.1.1.14
sqa1500_ija	10.2.1.164	ONLINE	COM2	20247	20247	0	0	0	2.0.0.27+V
workingagain	10.2.1.76	ONLINE	10.2.1.93	12732	12731	1	0	2	2.0.0.0+V

Note: Click on a column heading to sort the column.

- Refresh Network — This button clears the screen and displays the current cameras on the network.
- Name — The network name of the camera.
- IP Address — The IP address of the camera.
- Status — The status (ONLINE or OFFLINE) of the camera.
- Controller — Identifies which connection (INTERNAL, COM2, an IP address) is “controlling” the camera.
- Cycles — The total number of reads the camera has performed.
- Passed — The number of successful reads the cameras has performed.
- Failed — The number of unsuccessful reads the cameras has performed.
- Alarms — The number (in red) of serious conditions that need attention.
- Connections — The number of connections to the camera.

  Note: ReadRunner establishes three connections (image, result, command). You cannot change the number of connections other than through the process of performing some task using ReadRunner.

– SW Version — The version of software running on the camera. A +V after a version indicates that the camera has a valid Verification License key installed.

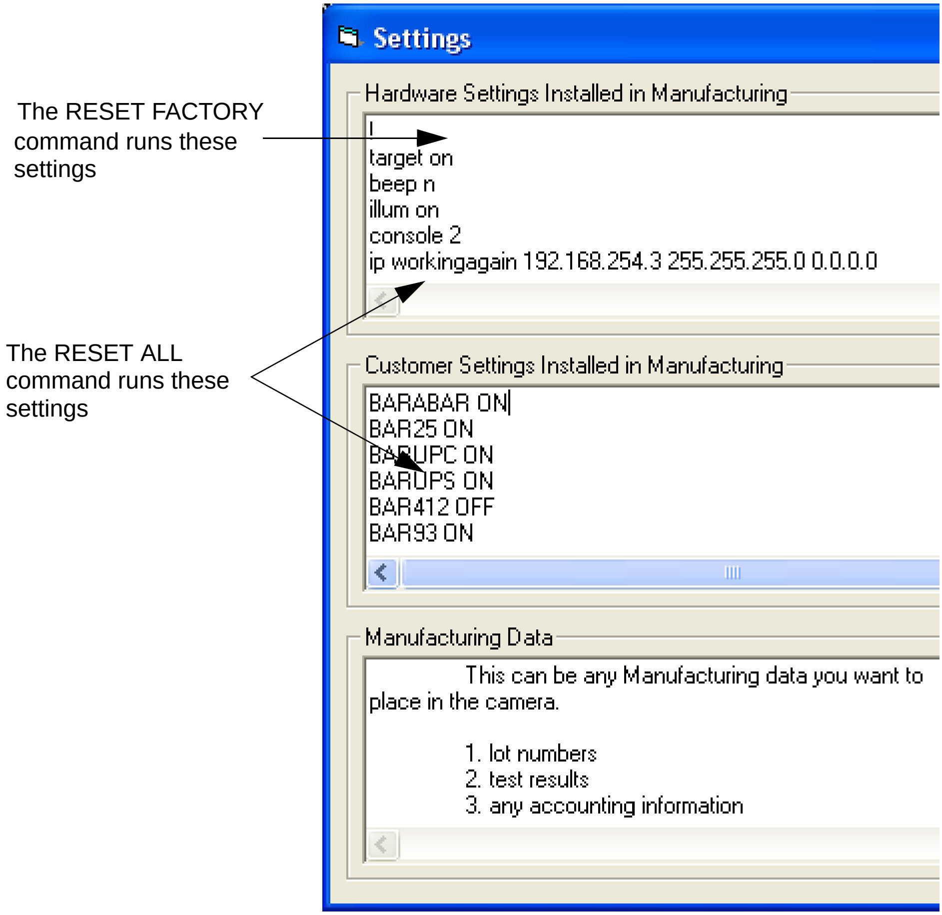
ReadRunner Version Number

To display the ReadRunner version number, select About ReadRunner from the Help menu. ReadRunner displays the About ReadRunner window.

Note: For proper operation, make sure the version of the software displayed in the Network Overview Window matches the ReadRunner revision used to connect to the camera. Contact Microscan to ensure you have the latest copy of software.

- Details — This button displays the Settings screen, as shown in Figure 4–67. When a camera is selected, the detail button displays the settings screen.

FIGURE 4–67. Settings Window



– Hardware Settings Installed in Manufacturing — These settings depend on the hardware purchased by the customer.

– Customer Settings Installed in Manufacturing — These settings depend on the customer’s specifications.

– Manufacturing Data — These are such things as MAC address, serial numbers, and so on.

Note: The RESET FACTORY command restores the camera to factory settings and runs the stored hardware specific commands.

The RESET ALL command restores the factory settings, runs the stored hardware commands, and runs the stored customer commands.

For more information about either command, see the HawkEye™ 1500 Series Reference and Programmers Manual.

Fine Tuning & Monitoring Your Application

The procedures in this section explain:

- “Selecting Symbologies (1D or 2D)” on page 4-100
- “Learning & Unlearning” on page 4-101
- “Modifying Decoding Parameters” on page 4-104
- “Debugging Images” on page 4-120

Selecting Symbologies (1D or 2D)

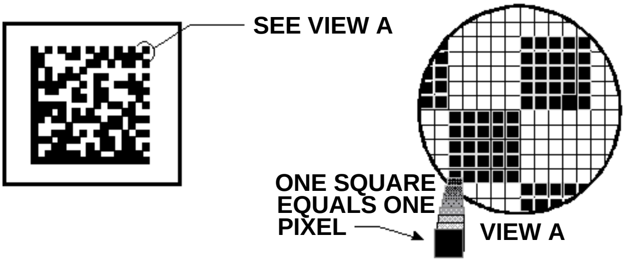
Camera Resolution and Pixels

The number of pixels within the CCD array determines camera resolution. The more pixels in the captured image, the higher the resolution or quality of the image.

Determining the correct camera resolution for the application is an important part of the HawkEye™ 1500 camera setup. The HawkEye™ 1500 requires that each cell in a Data Matrix symbol have a minimum pixel grid of 5x5 to obtain maximum performance.

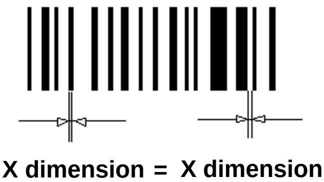
Ensure that you select a matrix size that allows you to obtain this pixel-to-cell ratio. The example in Figure 4–68 depicts a magnified view of a partial image. Notice that each cell contains the minimum number of pixels.

FIGURE 4–68. **Partial Data Matrix Image**



The X dimension, which refers to the narrowest bar and narrowest space, must consist of at least 2 pixels. In the case of a linear or stacked barcode, the minimum pixel width applies to the X dimension, as shown in Figure 4–69.

FIGURE 4–69. **X Dimension**



Learning & Unlearning

A learn sequence narrows down the parameters associated with a symbol, resulting in a more reliable decode of the same type of symbol. In most cases, learning speeds up the decoding time.

- Learn Photometry — Locks in the exposure and gain settings, and speeds up image acquisition.
- Learn Decoder Settings — Locks the Data Matrix size, polarity, etc., and speeds up decoding. Otherwise, the camera looks for all types of symbols.

Once a learn sequence is initiated, all symbols that the HawkEye™ 1500 camera reads must be of the same type and size in order to be decoded. To display the Learn window:

1. Click 

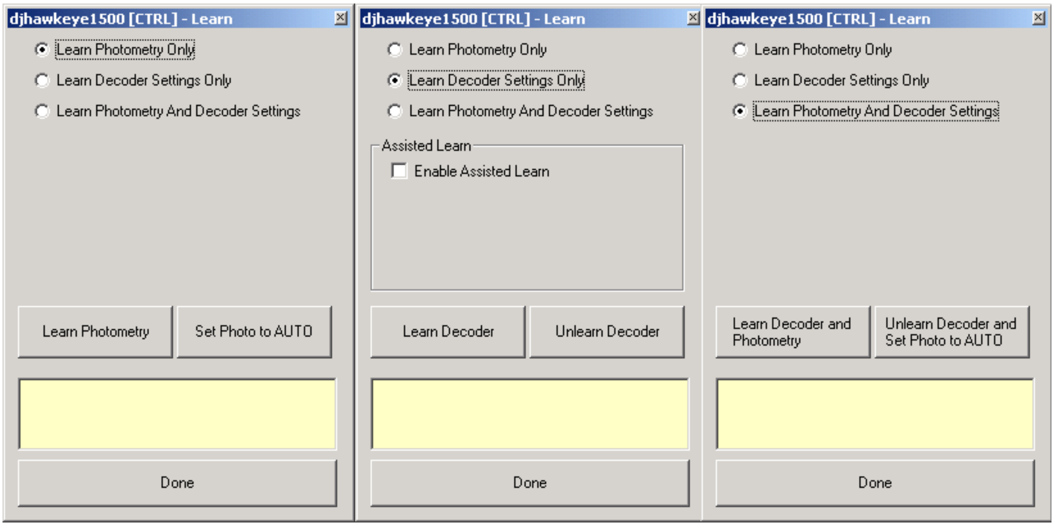
2. Click Yes when asked if you wish to control camera X.

3. Click Learn

ReadRunner displays one of the three Learn windows shown in Figure 4–70.

Note: Notice how the text changes for each button depending on the option chosen.

FIGURE 4–70. Three Learn Windows



4. Select one of the following:

   – Learn Photometry Only — Specifies that the camera should learn only lighting information.

   – Learn Decoder Settings Only — Specifies that the camera should learn only decoder settings.

   – Learn Photometry and Decoder Settings — Specifies that the camera should learn both lighting information and decoder settings. This is the default.

After you click Learn in the Learn window, the HawkEye™ 1500 camera “learns” the information you specified.

Enabling Assisted Learn

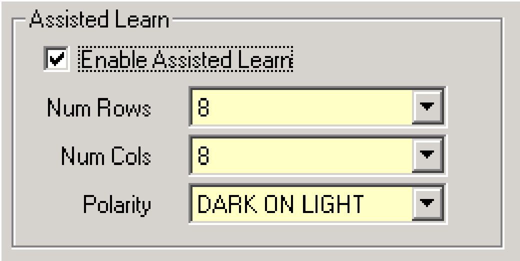
This command narrows down the parameters associated with a symbol by explicitly specifying the number of rows and columns, and the polarity. As with a learn, this results in a more reliable decode of the same type of symbol. In most cases, it also speeds up the decoding time.

1. From the Learn window, select Learn Decoder Settings Only.

2. Select (check) Enable Assisted Learn.

   `ReadRunner displays the Enable Assisted Learn dialog box, as shown in Figure 4-71.`

FIGURE 4–71. Enable Assisted Learn Dialog Box



3. Specify the following:

   – Num Rows — The number of rows expected in the Data Matrix.

     Range: 8 - 144

   – Num Cols — The number of columns expected in the Data Matrix.

     Range: 8 - 144

   – Polarity — The expected polarity of the Data Matrix. Valid values are DARK ON LIGHT or LIGHT ON DARK

4. ReadRunner also displays a box that you must place around the symbol. Use the four corners (one at a time) to move the box around the symbol.

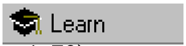
5. Click Done.

Unlearning

To “unlearn” the learned information:

1. Click 

2. Click Yes when asked if you wish to control camera X.

3. Click 

ReadRunner displays the Learn window (Figure 4–70).

4. Click Unlearn.

5. Click Done.

Note: Unlearn “unlearns” what was set in a previous Learn.

Modifying Decoding Parameters

This section describes modifying the following:

- “Data Matrix Parameters” on page 4-104
- “Barcode Parameters” on page 4-109
- “Expert Settings” on page 4-113
- “Data Matrix Fine Tune” on page 4-115
- “BC412 Parameters” on page 4-116
- “QR Code Parameters” on page 4-117
- “Code 39 Parameters” on page 4-118
- “I2of5 Parameters” on page 4-119
- “UPC Parameters” on page 4-119

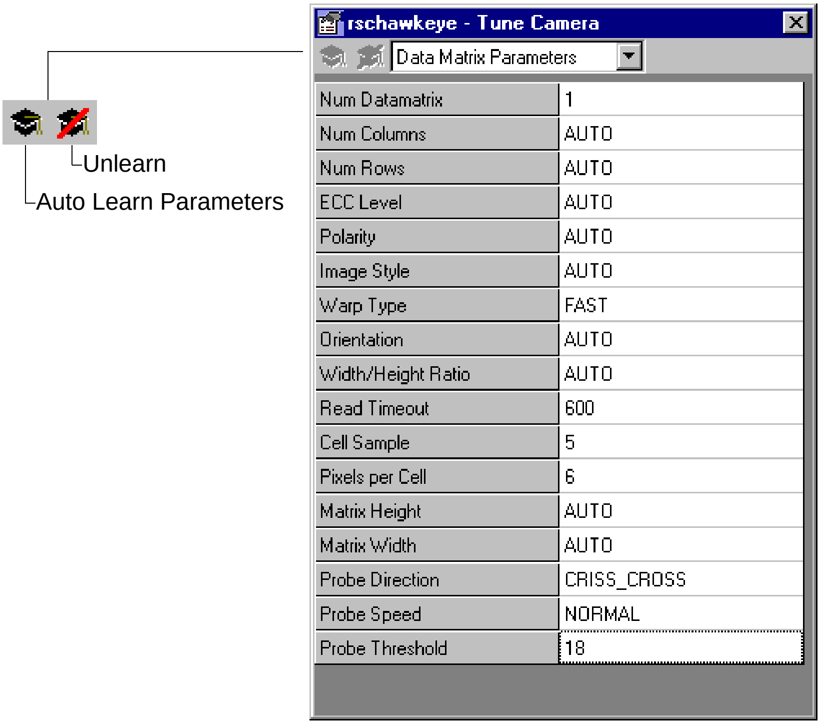
Data Matrix Parameters

To modify Data Matrix parameters:

1. Press Ctrl+A, or select Advanced Tuning from the Windows menu.

2. Select Data Matrix Parameters in the pull-down window. ReadRunner displays the window shown in Figure 4–72.

FIGURE 4–72. Advanced Window - Data Matrix Parameters



– Auto Learn Parameters — This button automatically learns the parameters for the current image.

– Undo Auto Learn — This button unlearns parameters and resets them to their defaults.

– Num Datamatrix — This pull down menu specifies the maximum number of Data Matrix symbols to find.

Range: 0 (disable/no look) or 1
Default: 1

– Num Columns — This pull down menu specifies the number of columns to expect in the Data Matrix symbol.

Range: 8 - 144, or AUTO (autoselect)

4

ReadRunner

– Num Rows — This pull down menu specifies the number of rows to expect in the Data Matrix symbol.

  Range: 8 - 144, or AUTO (autoselect)

– ECC Level — This pull down menu specifies the expected Data Matrix Error Correction level. Valid values are AUTO, SPEC (default), 0, 50, 80, 100, 140, or 200. An ECC Level of 200 is the most common.

– Polarity — This pull down menu specified the polarity. Valid values are:

  - DL — The Data Matrix is dark on light in the image.
  - LD — The Data Matrix is light on dark in the image.
  - AUTO — The Data Matrix is either DL or LD.

– Image Style — This pull down menu selects the Data Matrix symbol image style. Valid values are:

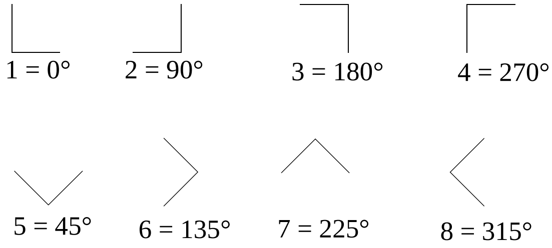
  - NORMAL — Data Matrix is imaged normally as opposed to mirrored.
  - MIRROR — A mirrored Data Matrix image is obtained.
  - AUTO — The Data Matrix is either NORMAL or MIRROR. This is the default.

– Warp Type — This pull down menu specifies the matrix warping speed. Valid values are:

  - Fast — The warping algorithm uses fast but less precise implementation. This is the default.
  - Slow — The warping algorithm uses slow but more precise implementation.

– Orientation — This pull down menu specifies the matrix orientation in READ mode. When a Learn is performed, the orientation of the Data Matrix is set to one of the orientation values shown below. In Read mode, the controller is optimized to find the Data Matrix whose orientation is similar to the orientation determined in Learn. More specifically, if the Data Matrix orientation in Read mode stays within the +/-23° from the “learned” orientation, the controller will perform more reliably. If the orientation of the Data Matrix can be arbitrary, then you should change the orientation setting after the Learn by setting the orientation to AUTO. All measurements are assumed to be rotated in a CCW direction.

Range: 1 - 8, or AUTO (default)



– Width/Height Ratio — This pull down menu specifies the expected width to height ratio when AUTO HEIGHT and WIDTH are selected.

  Range: 3 (3 = 0.3) - 50 (50 = 5.0), or AUTO (default)

– Read Timeout — This parameter specifies the time to spend inspecting a symbol. For good, easy to read symbols, you might set this value this value lower than the default. For harder to read symbols, you might set this value higher. The purpose of this parameter is to minimize the time the camera takes trying to read a symbol that is bad before it simply fails the symbol and moves on to reading other symbols. For additional information, see “Read Timeout” on page 4-52.

  Range: 0 - 9999ms
  Default: 600ms

– Cell Sample — This pull down menu specifies cell sampling.

  Range: 1 - 7
  Default: 5

– Pixels per Cell — This pull down menu specifies the expected cell width. It is recommended that the Field of View (FOV) of the camera be chosen such that the Cell Size of the Data Matrix is between 5 and 10 pixels. The cell size can be estimated as follows using the values obtained after a successful learn:

$$\frac{\text{Matrix Height (in pixels) + Matrix Width (in pixels)}}{\text{Number of Rows + Number of Columns}}$$

Typically, the decode algorithm operates with maximum robustness when Pixel Per Cell is set to the estimated cell size. In the HawkEye™ 1500, the Pixel Per Cell is always set to the default value of 6 pixels after a learn in order to speed up the decode process for most of the applications. You can enhance robustness by adjusting this parameter.

On the other hand, if the quality of the Data Matrix is high, then lowering the cell size value slightly will not affect the read rates but will further speed up the Read cycle. When the cell size drops to 3 pixels or less, however, the locate/decode time will increase and the robustness of the algorithm may be reduced.

Range: 2 - 20
Default: 6

– Matrix Height — This pull down menu specifies the Data Matrix height.

  Range: 20 - 1024, or AUTO (default)

– Matrix Width — This pull down menu specifies the expected matrix width.

  Range: 20 - 1024, or AUTO (default)

– Probe Direction — This pull down menu specifies the Probe direction. Valid values are:

  - CRISS_CROSS — The first probe searches horizontally; the second probe searches vertically; the third probe searches horizontally at a different location; the fourth probe searches vertically at a different location and so on until (a) a Data Matrix is decoded, or (b) the search is completed within the ROI, or (c) the Read Time-out is reached. This is the default.

  - HORIZONTAL — All probes search horizontally at different locations until (1) a Data Matrix is decoded, or (b) the search is completed within the ROI, or (c) the Read Time-out is reached.

- VERTICAL — All probes search vertically at different locations until (1) a Data Matrix is decoded, or (b) the search is completed within the ROI, or (c) the Read Time-out is reached.
- HORI_VERT — First performs HORIZONTAL search. The VERTICAL search will not be invoked unless no Data Matrix is decoded in HORIZONTAL search. Note that half of the total Read Time-out value is allocated for HORIZONTAL search and half for VERTICAL search. This is the default.
- VERT_HORI — First performs VERTICAL search. The HORIZONTAL search will not be invoked unless no Data Matrix is decoded in HORIZONTAL search. Note that half of the total Read Time-out value is allocated for HORIZONTAL search and half for VERTICAL search.

– Probe Speed — This pull down menu specifies the Probe speed. Valid values are NORMAL (default), OVERDRIVE, or TURBO.

– Probe Threshold — This pull down menu specifies the Probe threshold.

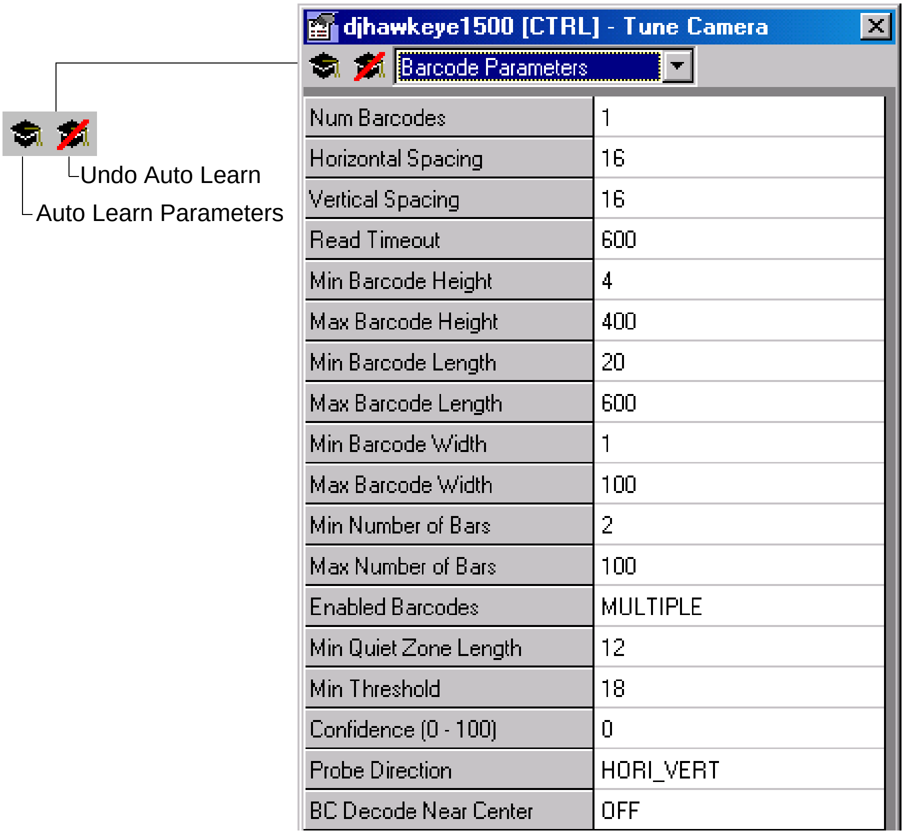
Range: 5 - 100
Default: 18

Barcode Parameters

To modify Barcode parameters:

1. Press Ctrl+A, or select Advanced Tuning from the Windows menu.
2. Select Barcode Parameters in the pull-down window. ReadRunner displays the window shown in Figure 4–73.

FIGURE 4–73. Advanced Window - Barcode Parameters



– Auto Learn Parameters — This button automatically learns the parameters for the current image.

– Undo Auto Learn — This button unlearns parameters and resets them to their defaults.

– Num Barcodes — This pull down menu specifies the number of barcodes to decode.

Range: 0 or 1
Default: 1

– Horizontal Spacing — This pull down menu specifies the horizontal spacing.

Range: 8 - 256
Default: 16

– Vertical Spacing — This pull down menu specifies the vertical spacing.

  Range: 8 - 256
  Default: 16

– Read Timeout — This parameter specifies the time to spend inspecting a symbol. For more complete information, see “Read Timeout” on page 4-52.

  Range: 0 - 9999ms
  Default: 600ms

– Min Barcode Height — This pull down menu specifies the minimum height of the barcode.

  Range: 4 - 400
  Default: 4

– Max Barcode Height — This pull down menu specifies the maximum height of the barcode.

  Range: 4 - 400
  Default: 400

– Min Barcode Length — This pull down menu specifies the minimum length of the barcode.

  Range: 20 - 600
  Default: 20

– Max Barcode Length — This pull down menu specifies the maximum length of the barcode.

  Range: 20 - 600
  Default: 600

– Min Barcode Width — This pull down menu specifies the minimum barcode width.

Range: 1 - 100
Default: 1

– Max Barcode Width — This pull down menu specifies the maximum barcode width.

  Range: 1 - 100
  Default: 100

– Min Number of Bars — This pull down menu specifies the minimum number of the bars.

  Range: 2 - 100
  Default: 2

– Max Number of Bars — This pull down menu specifies the minimum number of the bars.

  Range: 2 - 100
  Default: 100

– Enable Barcodes — When you click this parameter, ReadRunner displays the Symbology/Verification Window. Select the barcodes you want enabled.

– Min Quiet Zone Length — The minimum length of the Quiet Zone. If for any reason the quiet zone is less than the default value in the image due to camera setup restrictions, the Min Quiet Zone Length parameter can be decreased in order for the algorithm to accept shorter space as quiet zone. On the other hand, if the barcode is big in the FOV such that some space between two adjacent bars exceeds the value specified in the Min Quiet Zone Length, then there is a possibility that this space will be detected as possible quiet zone. To avoid this situation, the Min Quiet Zone Length can be increased. (Supported with the BARQZ terminal command.)

  Range: 5 - 100
  Default: 12

  Note: Do not adjust Min Quiet Zone Length unless you understand how the change will affect the system performance. Since Learn Barcode uses the parameter values specified in above dialog box, setting these values inappropriately may cause the Learn Barcode to fail.

– Min Threshold — The barcode threshold out of 255 grayscale. (Supported with the BARTHRESH terminal command.)

  Range: 5 - 100
  Default: 18

– Confidence (0 - 100) — For barcodes that do not use checksum such as Code 39, I 2 of 5, Codabar, and BC412, the threshold value can help reduce potential misdecodes. The default value is 0, meaning that any possible decode is accepted. When the value is raised, a certain number of decodes must agree before the decode is declared successful. On the other hand, if the value is set too high, then a normally decodable but imperfect barcode may not be decoded. (Supported with the BARCONF terminal command.)

  Range: 0 - 100
  Default: 0

– Probe Direction — Enables you to set the direction of the search:

  - HORI
  - VERT
  - HORI_VERT (default)
  - VERT_HORI

  (Supported with the BARPRB terminal command.)

– BC Decode Near Center — A barcode will not be decoded unless it is near the center of the FOV. (Supported with the BC_DECODE_NEAR_CENTER terminal command, which is part of the FINETUNE terminal command.)

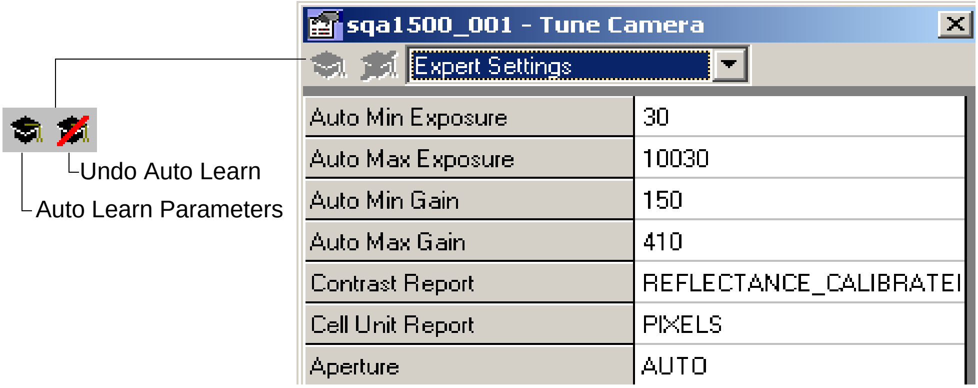
  Default: Off

Expert Settings

To modify expert settings parameters:

1. Press Ctrl+A, or select Advanced Tuning from the Windows menu.
2. Select Expert Settings in the pull-down window. ReadRunner displays the window shown in Figure 4–74.

FIGURE 4–74. **Advanced Window - Expert Settings**



– Auto Min Exposure — The minimum exposure time for auto photometry.

  Default: 30

– Auto Max Exposure — The maximum exposure time for auto photometry.

  Default: 10030

– Auto Min Gain — The minimum gain for auto photometry.

  Default: 150

– Auto Max Gain — The maximum gain for auto photometry.

  Default: 410

– Contrast Report— This is no longer used as a customer-settable parameter. If contrast is calibrated, it is reported as REFLECTANCE_CALIBRATED; if not calibrated, it is reported as UNCALIBRATED.

– Cell Unit Report — The units to report cell size in MILS or PIXELS.

  Note: MILS is only valid if the camera is calibrated. See the HawkEye™ 1500 Series Verification Manual.

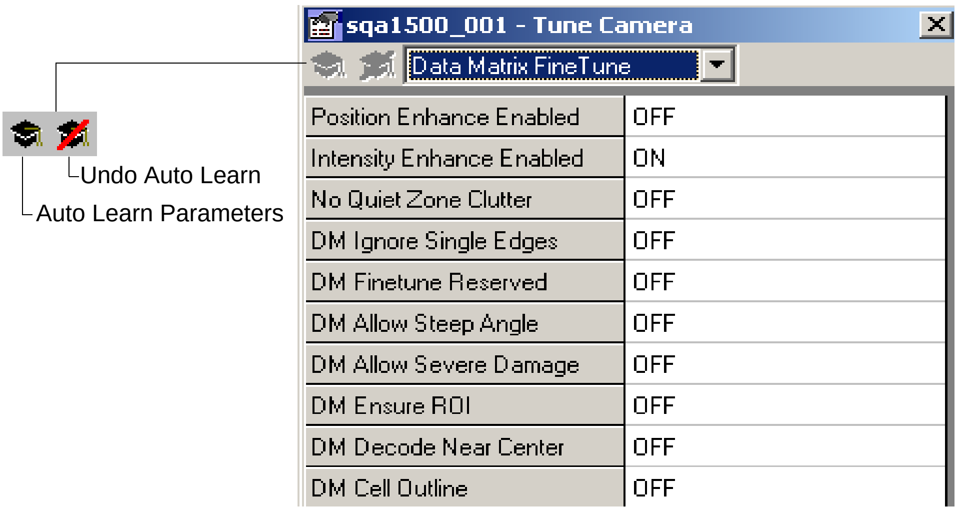
– Aperture — Used with ISO15415 calibration. See the HawkEye™ 1500 Series Verification Manual.

Data Matrix Fine Tune

To fine tune Data Matrix parameters:

1. Press Ctrl+A, or select Advanced Tuning from the Windows menu.
2. Select Data Matrix Fine Tune in the pull-down window. ReadRunner displays the window shown in Figure 4–75.

FIGURE 4–75. Advanced Window - Data Matrix Fine Tune



– Position Enhance Enabled — Enabling this will instruct the algorithm to locate the four corners of the Data Matrix more precisely and in general reduce the number of error correction used for severely underprinted or overprinted symbols. The default is Off.

– Intensity Enhanced Enabled — Designed to overcome dramatic intensity variation over the matrix border area. For example, certain poorly marked Data Matrix may have some cells that are almost invisible compared to the rest of the cells. Without enabling the option, the system may issue a status code indicating a certain edge cannot be found. Enabling the option will help read this type of Data Matrix more consistently. The default is Off.

– No Quiet Zone Clutter — Enabling the option will speed up the reading process when Data Matrix has sufficient quiet zone. With sufficient quiet zone, a Data Matrix with irregular, curved, or distorted border(s) can be read more efficiently with the option checked. The border problem typically is associated with low quality inkjet or dot peen marks or when the Data Matrix is imaged at an angle.

– DM Ignore Single Edges — Allow a more severely damaged Data Matrix to be read by sacrificing speed.

– DM Finetune Reserved — Reserved for future use.

– DM Allow Steep Angle — In some Data Matrix reading applications it is not possible to set up the camera such that the focal plane is in parallel with the surface of the Data Matrix label. When the focal plane and the label surface form a steep angle, the Data Matrix in the image will have severe geometrical distortion. Enable the flag to read the distorted Data Matrix symbols.

– DM Allow Severe Damage — Enabling the option will increase the robustness of the software in reading Data Matrix with severe border damages. To use it, first perform a successful Learn on a less damaged label. Then enable (check) the option to read labels with more border damages in Run mode.

– DM Ensure ROI — Enabling the option will ensure that no Data Matrix is located unless it is fully inside the ROI.

– DM Decode Near Center — A Data Matrix will not be decoded unless it is near the center of the FOV.

  Default: Off

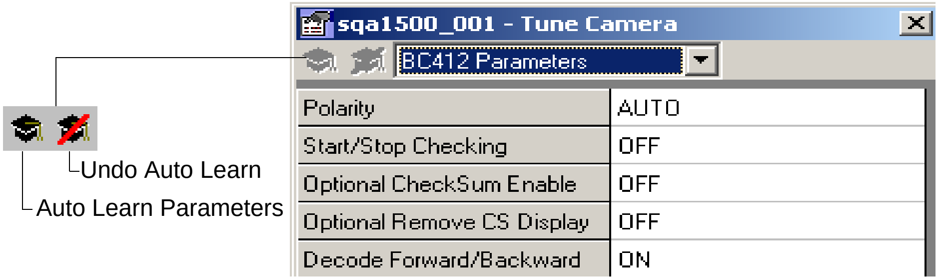
– DM Cell Outline — Enabling the option will help the algorithm decode a Data Matrix with outlined cells only. In this case, the On and Off cells have little or no contrast but they are separated by edges of the cells.

BC412 Parameters

To modify BC412 parameters:

1. Press Ctrl+A, or select Advanced Tuning from the Windows menu.
2. Select BC412 Parameters in the pull-down window. ReadRunner displays the window shown in Figure 4–76.

FIGURE 4–76. Advanced Window - BC412 Parameters



- Polarity — When the polarity is known, select either Dark on Light, or Light on Dark from the drop-down list. When polarity is not known, select AUTO (the default).
- Start/Stop Checking — When ON, the HawkEye™ 1500 looks for a start pattern and a stop pattern on the BC412 barcode.

  Default: Off
- Optional Check Sum Enable— Enable this item if the BC412 barcode contains a checksum character.
- Optional Remove CS Display — If the CheckSum is present, should it be displayed?
- Decode Forward/Backward — This is only useful when Start/Stop Checking is OFF. When Decode Forward/Backward is ON (the default) the HawkEye™ 1500 decodes data left-to-right. When Decode Forward/Backward is OFF, the HawkEye™ 1500 decodes data from right to left. This mode should not be used for SEMI BC412 barcode that contains both Start/Stop patterns and Check Sum.

  Default: On

QR Code Parameters

To modify QR Code parameters:

1. Press Ctrl+A, or select Advanced Tuning from the Windows menu.
2. Select QR Code Parameters in the pull-down window. ReadRunner displays the window shown in Figure 4–77.

FIGURE 4–77. Advanced Window - QR Code Parameters

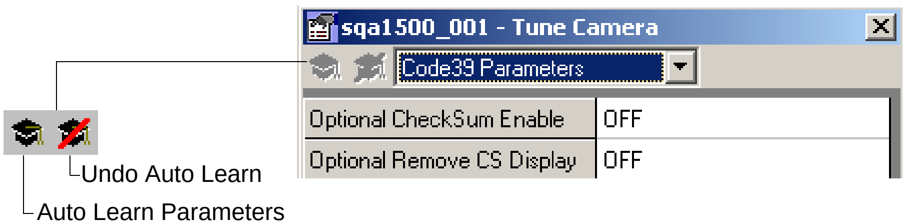
- Finder Pattern Misalignment — Select the appropriate value to handle finder pattern alignment error when the cells in the finder pattern and the remaining cells do not line up correctly.
- Polarity — Select Dark on Light, Light on Dark, or AUTO, depending on how the code is presented to the camera. If AUTO is not used, some performance enhancement should be seen by giving a hint to the locator.

Code 39 Parameters

To modify Code 39 parameters:

1. Press Ctrl+A, or select Advanced Tuning from the Windows menu.
2. Select Code 39 Parameters in the pull-down window. ReadRunner displays the window shown in Figure 4–78.

FIGURE 4–78. Advanced Window - Code 39 Parameters



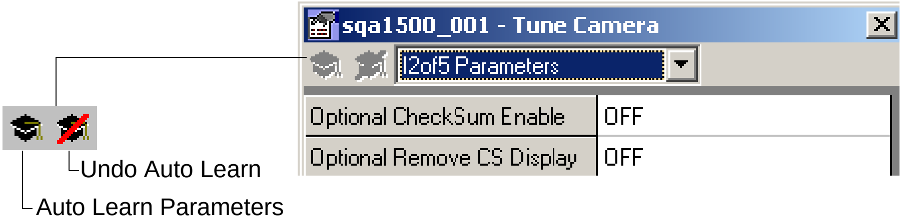
- Optional Check Sum Enable— Enable this item if the Code 39 barcode contains a checksum character.
- Optional Remove CS Display — If the CheckSum is present, should it be displayed?

I2of5 Parameters

To modify I2of5 parameters:

1. Press Ctrl+A, or select **Advanced Tuning** from the Windows menu.
2. Select **I2of5 Parameters** in the pull-down window. ReadRunner displays the window shown in Figure 4–79.

FIGURE 4–79. Advanced Window - I2of5 Parameters



– Optional Check Sum Enable— Enable this item if the I2of5 contains a checksum character.

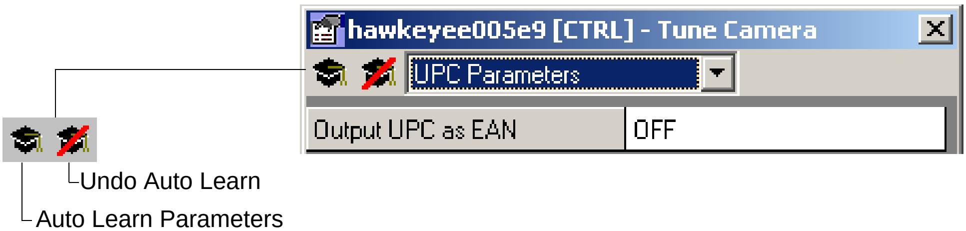
– Optional Remove CS Display — If the CheckSum is present, should it be displayed?

UPC Parameters

To modify UPC parameters:

1. Press Ctrl+A, or select **Advanced Tuning** from the Windows menu.
2. Select **UPC Parameters** in the pull-down window. ReadRunner displays the window shown in Figure 4–80.

FIGURE 4–80. Advanced Window - UPC Parameters



– Output UPC as EAN — Specifies whether or not UPC data is displayed as EAN.

Debugging Images

Configuring the Part Queue

The Part Queue saves the result data to a file path (local or remote). You can configure the camera to save the data when it occurs (rather than in memory), or save the data on demand with a command. The records are stored as a re-usable queue; that is, when the queue is full, the newest record replaces the oldest record. Part Queue is loss-less; it will store a record for every image it sees.

Note: The Part Queue uses the default ASCII Command port, which is port 49095.

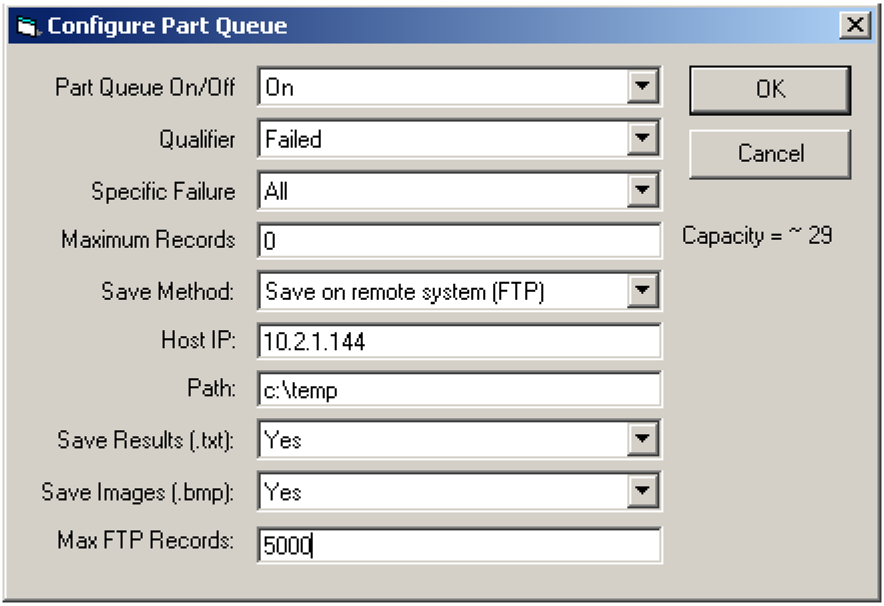
Telnet (which, by default, communicates on port 23) provides a connection to the Camera's low-level Monitor Shell for diagnostics purposes only. A special Shell command is available in this environment for sending HE15xx commands to the Camera via Telnet. The command has the following format:

cmd “<HE15xx Reader ASCII Command String>”

To configure the Part Queue:

1. Press Ctrl+U, or select Configure Part Queue from the Settings menu.
2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Configure Part Queue window, as shown in Figure 4–81.

FIGURE 4–81. Configure Part Queue Window



3. Specify parameters from the following pull down menus:

   – Part Queue On/Off — Enables (On) or disables (Off) the part queue.

   – Qualifier — Specifies what you want to store in the part queue:

     - All
     - Passed
     - Failed
     - Verify Good, Fair, Poor, Failed
     - Verify Good, Fair, Poor
     - Verify Good
     - Verify Fair
     - Verify Poor
     - Verify Failed

  - Verify Poor, Failed
  - Verify Fair, Poor, Failed
- Specific Failure — Specifies what kind of failures you want to store:
  - All
  - Location Failure
  - Decode Failure
  - Match Failure
  - Runtime Error

  Note: This field is activated when you select Failed for the Qualifier parameter.
- Maximum Records — Specifies the maximum number of images you want to store in the part queue. The approximate limit will be displayed next to this field, and depends on the amount of memory on the camera and the PartQ options selected.
- Save Method — The Part queue must be activated for either mode to function. Then, you can specify the qualifier and number of records to save.

  Note: When left at zero, Maximum Records has a different meaning for each mode.
  - Save in camera memory — This is the default. All available memory stores read records. Typically, there is enough space for the last 30 records (when programmed to store Images as well), and about 5000 records (when programmed to store reports without images). Older records are dropped to make room for new ones over time such that the last N records are always available in memory.

    Note: When programmed in this mode, records saved on the camera can be retrieved by a remote client at any time over TCP or Serial. Retrieving records over TCP does not impact performance of the Reader; however retrieving reports over Serial will degrade the performance of the Reader.

You can use the part queue mechanism to buffer Read cycle reports over time (30 cycles with images, 5000 cycles without) such that, if the communication to the camera over TCP is lost, past read data can be siphoned out of the camera once the client reconnects. This logic can be easily initiated by the client on reconnects by sending the PARTREQ command.

- Save on remote system (FTP) — Records are written to the remote host with a unique cycle number in their file name. There is no wrap-around, so, if they are not purged on the remote host, this could potentially fill-up the remote disk file system. When the Maximum Records is non-zero, the records will wrap around after the Nth one has been saved, re-using the file names.

When you select Save on remote system (FTP), the Host IP (i.e., xx.xx.xx.xx, for example 10.2.1.98) and path needs to be specified. The Path uses the UNIX directory convention "/" as a directory separator. Optionally, Results and/or Images can be saved.

An FTP server must be running on the destination host for this option to function. The FTP Server must be configured with a "Camera" Account as follows:

User Name: target

Password: password

The credentials of that special user must be set to have write permissions to the destination PATH entered in the Configure Part Queue form. The root of the path should be restricted by the FTP Server for that account if desired.

Note: Selecting this option will drastically impact the performance of the camera. The Read Rate (i.e. Reads/sec) will significantly decrease; therefore, this option should only be used in special tuning or debugging situation. If the Reader job is triggered, the camera may overrun; you need to make sure that there is enough idle time between triggers to save the data via FTP. For a clean 100-BaseT Ethernet Network, it should take in the order of 100 to 200 msec to send an image via FTP.

This mechanism does not guarantee that every read cycle will be saved via FTP. Use Save to Memory if capturing the last N records is important without missing one.

Uploading using FTP with QueueView can cause lost data. It was discovered that a customer with more than one HawkEye 1500 Smart Camera reader connected to an FTP server and using high trigger rates, could lose a text or image file due to the limitations of a network data throughput. A FTP server, while capable of supporting multiple connections, is not always able to handle a new connection request at a particular time and returns an ECONNREFUSED status to the HawkEye 1500 Smart Camera reader. This error is detected and multiple retries are attempted to provide a very high likelihood of success.

We therefore recommend the following guidelines be considered when using the QueueView FTP feature:

– The transfer of files becomes part of the decision process for determining worst-case HawkEye 1500 Smart Camera reader trigger rates. We recommend no less than 3 seconds between triggers with 1 HawkEye 1500 Smart Camera. Triggering faster than the files can be successfully transferred will cause “Decode Process Overruns” with loss of data.

– The number of HawkEye 1500 Smart Camera readers on a network segment communicating to a single FTP server also should be limited to prevent contention for access to the FTP server. We recommend that the minimum trigger interval, on a single HawkEye 1500 Smart Camera application using the QueueView FTP feature, be 3 seconds. For additional HawkEye 1500 Smart Cameras, this minimum trigger interval should then be increased by 1 second for each HawkEye 1500 Smart Camera added. For example, when using 2 HawkEye 1500 Smart Cameras, the interval between triggers should be a total of 4 seconds (3 seconds plus 1 second for the additional HawkEye 1500 Smart Camera reader equals 4 seconds). When using 5 HawkEye 1500 Smart Cameras, the interval between triggers should be a total of 7 seconds (3 seconds plus 1 second for each of the 4 additional HawkEye 1500 Smart Camera readers = 7 seconds). Adding additional FTP servers and network segments should be considered to keep minimum trigger rates low.

- The worst-case bandwidth requirements on the network segment should never exceed 40% of the theoretical bandwidth. If the network is 10BaseT, 40% x 10,000,000 bits per second / 8 bits per byte = 500Kbytes/second. A HawkEye 1500 Smart Camera reader image is 326Kbytes and a text report is 1Kbyte. With FTP protocol overhead added, one HawkEye 1500 Smart Camera reader's image and text report takes nearly all of the available bandwidth for 1 second.

- The HawkEye 1500 Smart Camera will attempt to connect to the FTP server for each text or image file result. In the event that a connection fails, the HawkEye 1500 Smart Camera will retry this connection up to 100 times, spaced from 1 to 10 ms apart. In the event of an unsuccessful FTP connection, the HawkEye 1500 Smart Camera will stop retrying and get ready to accept the next trigger. This will result in data loss. If additional triggers come in before the FTP transfer is done on the buffered data, it will result in data loss. If the network is broken, it will result in data loss. Note that there is no non-volatile storage on the HawkEye 1500 Smart Camera so a power failure will result in lost data.

- Host IP — The IP address of the system running the FTP server.
- Path — The directory on the FTP server where the files will be written.
- Save Results (.txt) — Yes or No. This only applies to FTP reports.
- Save Images (.bmp) — Yes or No. This also applies to “Save in camera memory” save method.
- Max FTP Records — Is either of the following:
  - 0 = Records with the file names containing a 9 digit decimal representation of the cycle count “_000000000_”.
  - 1-9999 = Sequentially labeled records with the file names containing a 4 digit representation “_0000_”, wrapping at N records and overwriting old records

4. Click OK.

Uploading Images Using QueueView

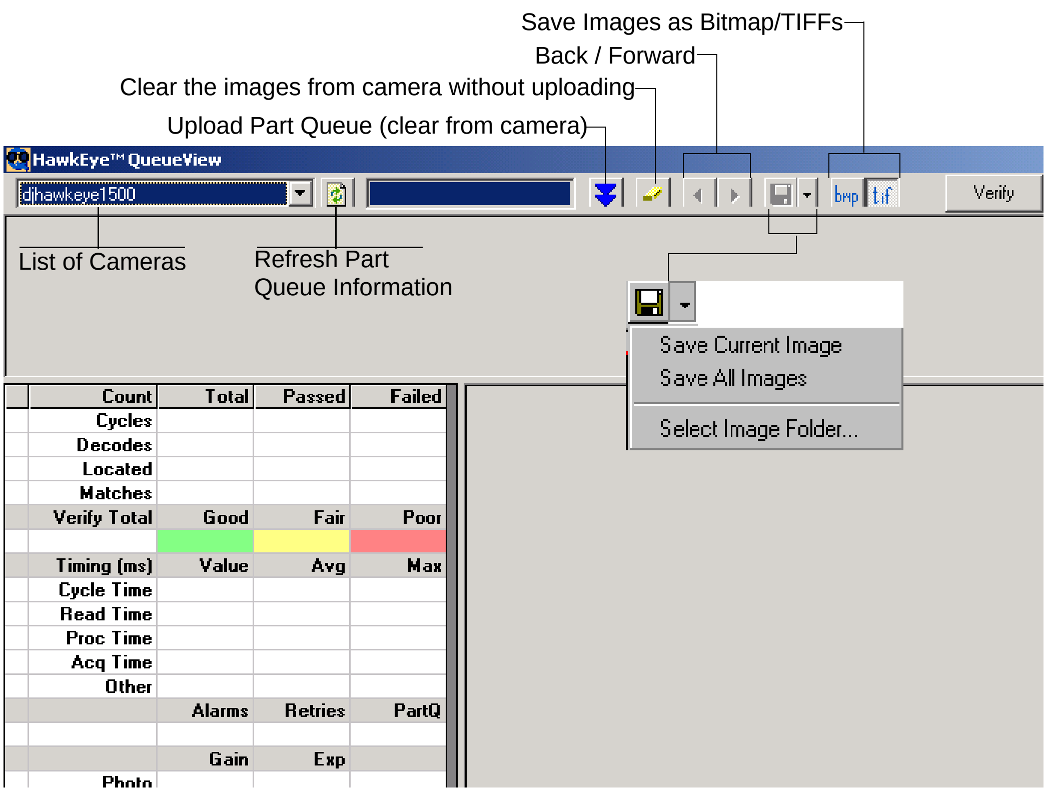
You can use ReadRunner to activate the Part Queue on the camera and save specific images inline with the running inspection.

You can save All images, Passed images, or Failed images (decode failures, locator failures, match failures, or runtime errors).

You can specify the number of records to store (the maximum is up to the amount of memory on the camera). Once images are stored on the camera, you can use QueueView to pull them off the camera, examine them, and save them to disk. To start QueueView and upload records:

1. Select **Start**, **Programs**, **ReadRunner 2.4**, **Utilities**, and finally **HawkEye QueueView**. The QueueView window is displayed. Figure 4–82 describes the QueueView options and buttons.

FIGURE 4–82. QueueView Window



– List of Cameras — Select the camera whose part queue you want to view.

```
You can edit the text in this drop down list of cameras to
enter a static IP address for a reachable camera on a
different subnet:
```

a. Highlight <Enter_Static_IP_Address>.

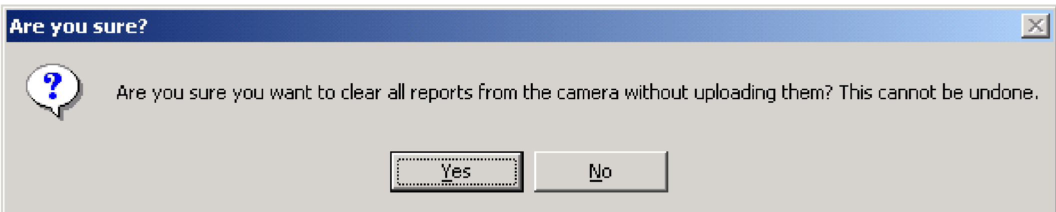
b. Enter the IP address and press Return.

– Refresh Part Queue Information — This button clears the screen and displays the most current cameras on the network.

– Upload Part Queue (clear from camera) — Uploads the images to QueueView and removes them from the camera.

– Clear images from camera without uploading — If you click this button, QueueView displays the Are you sure... dialog box, as shown in Figure 4–83.

FIGURE 4–83. Are you sure? Dialog Box



– Arrow Keys — Allows you to move through all the images uploaded from the part queue. When you click on a specific image, the report information for that image is displayed in the report panel.

– Save Images — Select one of the following:

- Save Current Image — Saves the current image.
- Save All Images — Saves all images.
- Select Image Folder — Specifies the folder where you want to save the image or images.

Note: These buttons are activated after you upload the part queue.

   – Save Images as Bitmap — Specifies that QueueView should save images onto the PC as .bmp files.

   – Save Images as TIFFs — Specifies that QueueView should save images onto the PC as .tif files. This is the default.

   – Verify — If this report contains verification results, then this button activates a verification report display.

2. From the **List of Cameras** pull down menu, select a camera.
3. Click **Upload Part Queue (clear from camera)**.

QueueView loads the entire Part Queue.

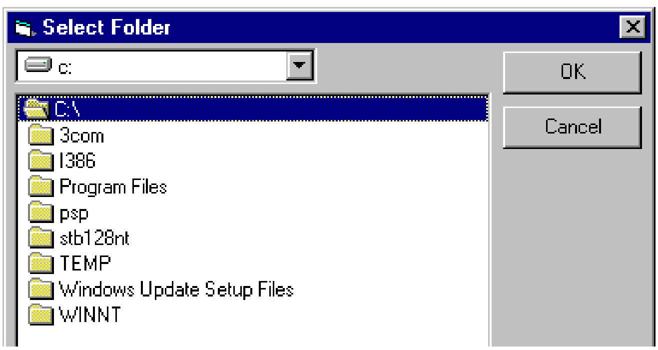
Saving Images to the PC Using QueueView

To save images to the PC using QueueView:

1. Click either **Save Images as Bitmap** or **Save Images as TIFFs**.
2. Select **Select Image Folder**.

QueueView displays a screen similar to the following:

FIGURE 4–84. Select Folder Window



3. Specify where you want to save the images. Click OK.

4. Click either Save Current Image or Save All Images.

QueueView saves the image(s) to the specified location on the PC. The file names of the saved images have the following format:

```
sqa001_000001173.tif
                  └─File Extension
          └─Cycle Number
  └─Camera Name
```

Saving the Current Image

To save the current image, press F9, or select Save Current Image from the File menu. ReadRunner saves the current image in the Images folder within the ReadRunner folder. The file name of the saved image has the following format:

```
sqa001_000001173.tif
                  └─File Extension
          └─Cycle Number
  └─Camera Name
```

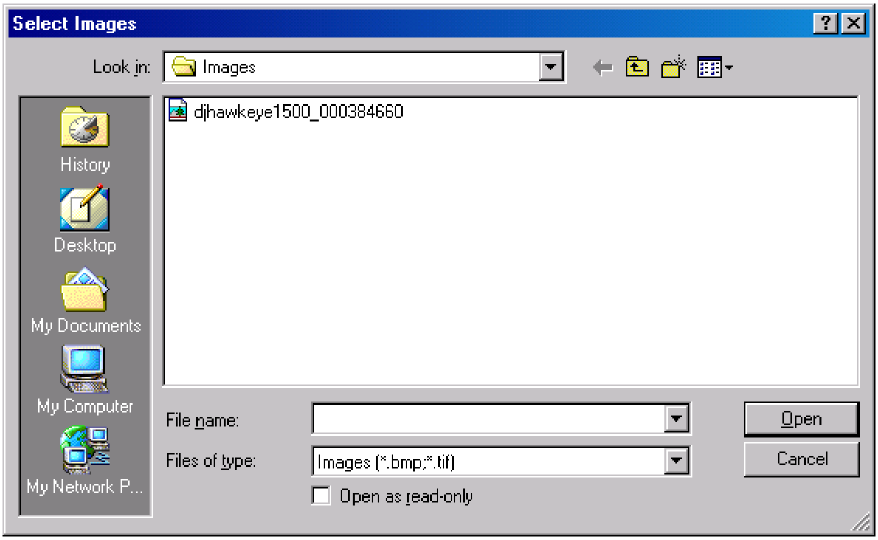
Loading Image Files to the Camera

If the camera fails to read an image, you can save that failed image onto the PC (see “Configuring the Part Queue” on page 4-120), and then load the image back to the camera for debugging purposes. To load an image file to the camera:

1. Press Ctrl+I, or select Load Image Files To Camera from the File menu.

2. Click Yes when asked if you wish to control camera X. ReadRunner displays the Select Images window, as shown in Figure 4–85.

FIGURE 4–85. Select Images Window



3. Navigate to the appropriate folder, select the file(s) you want to load onto the camera, and click Open. ReadRunner loads the file(s) onto the PC for debugging.

   – If the failure was a decode failure, you will want to modify the advanced parameters until it passes (see "Modifying Decoding Parameters" on page 4-104).

   – If the failure was a location failure, you may need to adjust the ROI (see "Defining the Region of Interest" on page 4-26).

4. Press Ctrl+Q to return the camera to acquisition (see "Returning the Camera to Acquisition" on page 4-130).

Returning the Camera to Acquisition

After loading an image file to the HawkEye™ 1500 camera for reading, you can return the camera to acquisition using this menu item. To return the camera to acquisition:

1. Press Ctrl+Q, or select Use Camera for Acquisition from the File menu.

2. Click Yes when asked if you wish to control camera X. ReadRunner resumes reading images placed in its field of view.

The camera returns to normal acquisition.

The Filmstrip Recorder

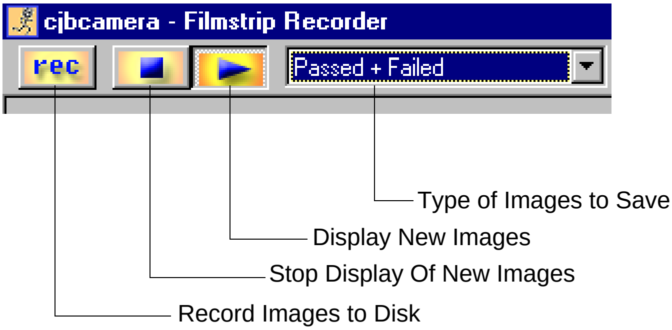
You can use the Filmstrip Recorder to save images directly to disk files for debugging purposes. It is an alternative to using the Part Queue and the QueueView application. Here are some important considerations when choosing between the two methods:

- The Part Queue is lossless in the sense that every image inspected will be stored if the qualifier criteria is met. However, there is a relatively small limit to the number of images that can be stored, so only the most recent images will be available to the QueueView application. If you are tracking down a failure that occurs infrequently, this is the preferred method of obtaining debugging images, since you are guaranteed not to miss the failure.
- The Filmstrip Recorder can only record images that are transferred to the PC at whatever rate will not slow the operation of the HawkEye™ 1500. This will not be every image, so this is not an appropriate way to track down a rare failure, since you may miss it entirely. However, the only limitation for the number of images stored is the free disk space available. This may be a more appropriate method for a statistical sampling of failures over a relatively long period of time. Also, you may find it more convenient to use the Filmstrip Recorder as a quick image storing mechanism versus setting up the Part Queue and using QueueView. To maximize the Filmstrip Recorder, see “Preferences” on page 4-44.

By default, the Filmstrip Recorder stores 25 image records in memory. This can consume a sizable amount of memory. When connecting multiple ReadRunners to monitor different cameras on the same PC, it is recommended that, at most, one Filmstrip Recorder be active for a System with 128MB of memory. To display the Filmstrip Recorder window:

1. Press Ctrl+F, or select Filmstrip Recorder from the Windows menu. ReadRunner displays the Filmstrip Recorder window, as shown in Figure 4–86.

FIGURE 4–86. **ReadRunner Filmstrip Recorder Window**



2. Specify the following:

   – Record Images to Disk — When you click this button, ReadRunner begins saving files to the ReadRunner directory on the PC.

     Note: This button will continue to flash to remind you that you are in record mode.

   – Stop Display Of New Images — Click this button to temporarily stop displaying and recording images in the Filmstrip Recorder.

   – Display New Images — Continue displaying and recording images if the record button is flashing.

   – Type of Images to Save — Specifies the type of images ReadRunner saves (Passed + Failed, Failed Only, Passed Only) to the ReadRunner folder on the PC.

5

CHAPTER 5 **Reading Difficult Symbols**

This chapter explains how to use your HawkEye™ 1500 Series Smart Camera-Based Reader to read difficult and damaged symbols. For a detailed Data Matrix description, see Appendix D, “Symbology Reference,”.

General Reading Guidelines

- Always attempt to create a balanced, properly contrasted Data Matrix image.
- Minimize the geometrical distortion of the Data Matrix with proper camera alignment.
- Choose a model with appropriate FOV such that the reported cell size after the learn is between 5 and 10 pixels per cell.
- When appropriate, preprocess the image. Erode and Dilate are helpful for poor dot peen types of images.
- If Learn is not successful, use Assisted Learn. If Assisted Learn is unsuccessful, improve the mark and image quality first.

Further Explanation

Although the algorithm with default parameters is capable of reading a variety of different images, for a typical application that involves only one type of Data Matrix, always apply a learn to narrow down the reading parameters. By constricting the parameters such as Num Columns, Num Rows, ECC Level, Polarity, Image Style, Matrix Height, and Matrix Width to a specific value, the algorithm will quickly reject any Data Matrix like objects that don't fit the profile described by the parameters and get to the Data Matrix of the interest quicker. This ensures that the reading time is more consistent. Furthermore, knowing these parameters, the algorithm will not have to compute them, but simply confirm them. This saves time and improves robustness. These parameters can be displayed in the Advanced Window - Data Matrix Parameters in ReadRunner (see “Data Matrix Parameters” on page 4-104).

Once the system is learned, you can relax the parameter constraints freely to meet the application requirements. For example, if the Data Matrix quality varies such that its size can increase or decrease from the learned value by over the maximum of 10% allowed by the algorithm, you can set both Matrix Height and Matrix Width to Auto so that the size will no longer be constricted. An alternative to setting both Matrix Height and Matrix Width to Auto is to set the Matrix Height and Matrix Width by their corresponding average values obtained by learning the largest Data Matrix and the smallest Data Matrix.

Set the Orientation parameter whenever possible. If in the application the Data Matrix is always presented to the camera in the same orientation, set it to the correct value instead of leaving it at Auto. For more information, see page 4–106.

The Learn is a powerful tool and will usually succeed when applied to a Data Matrix of acceptable quality. When learn fails, you should determine:

- If the FOV of the camera is properly chosen for the application. For more information, see Pixel Per Cell on page 4–107.
- If the mark quality is acceptable with one of our field proven industry leading Data Matrix quality verification products.

If the Learn fails due to an unsuitable FOV, and such a FOV is required, then use Assisted Learn to help correct the problem. Generally, once Assisted Learn is successful, the application will be robust.

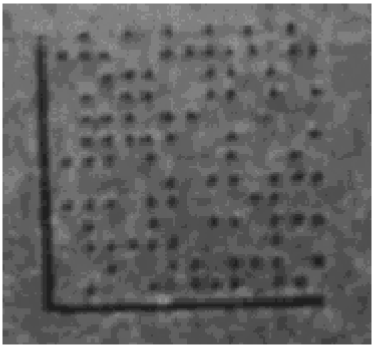
Preprocessing with Morphology

For poorly marked Data Matrices, applying morphological preprocessing operations can improve the image quality and make the reading process more robust. Proper use of the Erode or Dilate can help balance the size between the dark and light cells in a severely underprinted or overprinted Data Matrix symbol. Use Open or Close to remove minor defects on Data Matrix cells.

Erode Example

Figure 5–1 shows a poorly marked Data Matrix before ERODE is applied.

FIGURE 5–1. Example Before Applying Erode

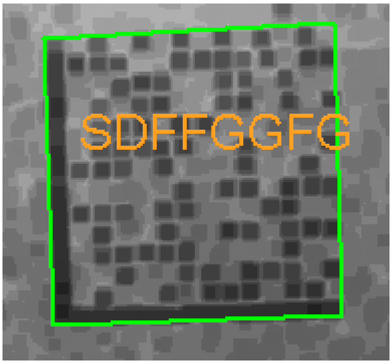


Assume that we want to apply Erosion to increase the size of the dark cells on the light background:

1. From ReadRunner, click the Add Camera button to connect a camera.
2. Highlight the camera you want to connect and click OK.
3. Click on the newly added button for the camera you just added.
4. Click on the Take Control button.
5. Click Yes when asked if you want to control the camera you just added.
6. Click the Photometry button.
7. From the Method pull down menu, select ERODE.
8. From the Iteration pull down menu, select 2.

Figure 5–2 shows the same Data Matrix after ERODE is applied with 2 Iterations.

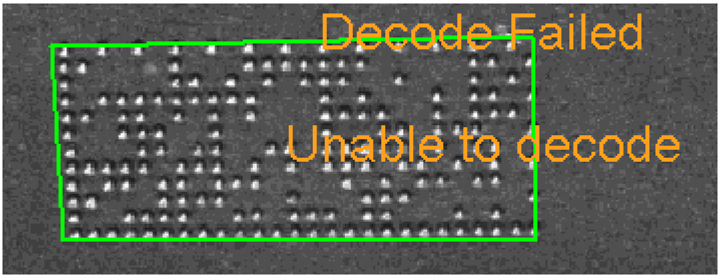
FIGURE 5–2. Example After Applying Erode



Dilate Example

Figure 5–3 shows a poorly marked Data Matrix before DILATE is applied.

FIGURE 5–3. Example Before Apply Dilate



Assume that we want to apply Dilation to increase the size of the light cells on the dark background:

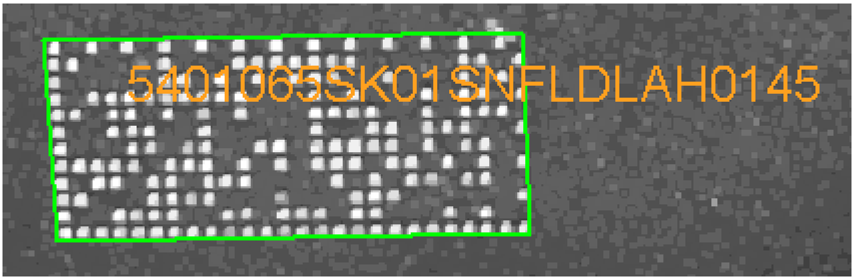
1. From ReadRunner, click the Add Camera button to connect a camera.
2. Highlight the camera you want to connect and click OK.
3. Click on the newly added button for the camera you just added.
4. Click on the Take Control button.
5. Click Yes when asked if you want to control the camera you just added.
6. Click the Photometry button.

7. From the Method pull down menu, select DILATE.

8. From the Iteration pull down menu, select 1.

Figure 5–4 shows the same Data Matrix after DILATE is applied with 1 Iteration.

FIGURE 5–4. Example After Applying Dilate



Reading Different Difficult Symbols

If an application requires reading different types of Data Matrix symbols, and the reading with the default decoder parameters does not produce satisfactory results, you can create several jobs, each containing a learned parameter for each type of the symbol. When a new image is acquired, the reader will try to decode the image with the parameter set stored in the jobs, one at a time, until a successful decode is achieved.

For details about how to use this capability, please consult the Technical Note titled “Tech Note EM-40176-1.”

CHAPTER 6

The Bootloader

This chapter describes the diagnostics and utilities for the HawkEye™ 1500.

Diagnostic Levels

HawkEye™ 1500 diagnostics can be divided into the following levels of testing:

- Level 0 diagnostics are executed as a power-on self-test from the boot loader and provide enough coverage to safely bring up the application code.
- Level 1 diagnostics are executed as part of the vxWorks application initialization scheme. Examples of tests running at this level are those that require the FPGA to be loaded or those that need system services such as timers.

In addition to these levels, an enhanced diagnostic monitor is provided for hardware debug and verification. The diagnostic monitor represents a significant enhancement of the existing terminal program and is resident within the bootloader. The monitor provides utilities and tests to aid in hardware debug and verification.

Diagnostic Monitor

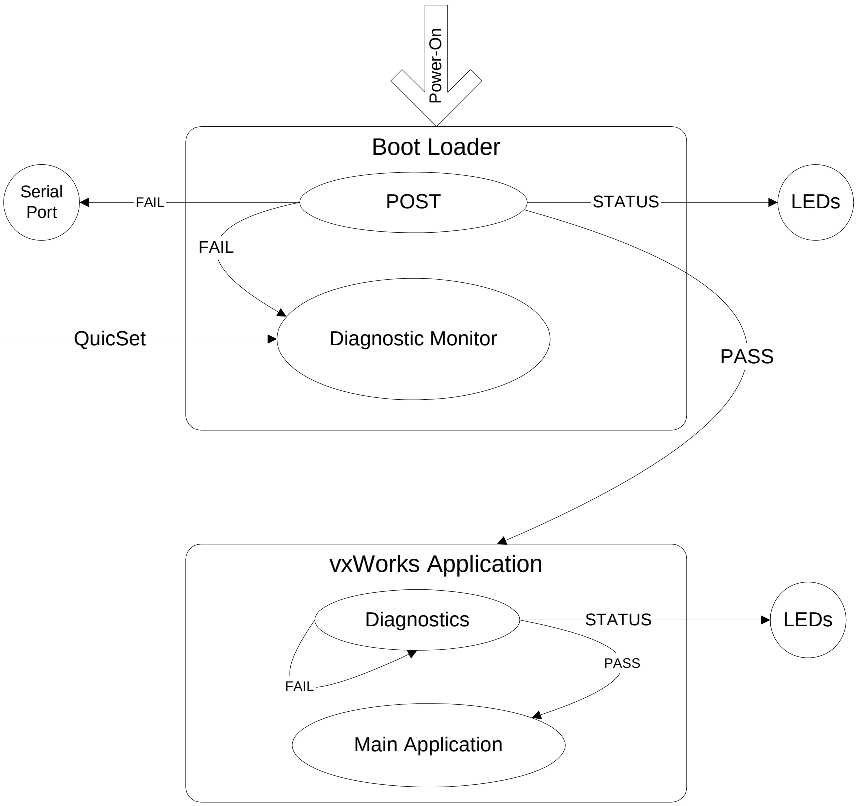
Tests, typically consisting of hardware functional blocks can be run one at a time or as part of a group. Through the use of a loop counter, the number of times a test is executed is configurable.

Additionally, utilities are provided which, as a minimum, can peek/poke memory locations.

You can access the monitor through QuicSet (press during power up ONLY). It is also accessed if there is a POST failure.

Note: A QuicSet boot does not run POST diagnostics.

FIGURE 6–1. Boot Flow



Boot Loader Power-On Self-Tests

As each power-on self-test is executed, a binary number representing the test-in-progress is displayed on the camera LEDs. Should a failure be detected, the beeper sounds five times and the LEDs flash out an error code until the QuicSet button is pressed. A diagnostic message is sent out the serial port (it's possible the message won't be seen if the failure is related to the RS-232) and the Diagnostic Monitor Menu is displayed. QuicSet boot uses the UART to report any POST error including RS-232 failure. Default communication parameters are 115,200, N, 8, 1.

Hard Error

A hard error is defined as an error that prevents the system from running properly. There are two types of hard errors:

- Hardware fault
- Invalid application block of flash

```
If a hard error is detected, the unit beeps five times and
flashes the fault code on the LEDs. It continuously flashes the
LEDs until the QuicSet button is pressed, at which time the
Diagnostic Menu is displayed.
```

Bootloader Menu

The Bootloader Menu provides a series of utilities to perform various tasks.

```
HawkEye BootLoader HE1500 Ver 5.45 (206MHz, 32MB RAM, 4MB Flash)

  d     Dump memory
  m     Modify memory
  dt    Display tests
  et    Execute test
  dbp   Dump boot params
  mbp   Modify boot params
  dm    Display menu
  dfb   Dump flash blocks
  der   Dump enet regs
  wmr   Write MAC reg
  wpr   Write PHY reg
  cpu   Dump CPU registers
  flsh  Get Flash size
```

```
  ram   Get RAM size
  cach  I-Cache control
  x     File transfer
  r     Reset unit
  j     Jump to app
  e     Display error
  h     Help
>>
```

d — Dump Memory

The “d” command dumps “units” by “width” bytes of raw data including the ASCII representation starting from address “addr” (initial default is 0).

Syntax

```
>> d [address [units [width]]]
```

```
Where:
```

- width — Specifies how wide the display field should be in bytes:
  - 1 byte
  - 2 bytes
  - 4 bytes (initial default)
- units — Represents the amount of data to be displayed (initial default is 16).

```
> d 0x00100000 256 1
00100000: 00 03 e1 52 00 09 2a 00 00 03 e1 a0 30 00 e5 91
*..R....*.....0..*
00100010: 04 93 e0 04 10 08 e2 81 38 02 e1 a0 3b 40 e2 83
*.........8..@;..*
00100020: 28 43 e1 a0 38 23 e1 a0 00 00 e1 53 ff f6 3a ff
*C(..#8....S....:*
00100030: 30 04 e5 95 04 93 e0 04 00 00 e3 54 00 04 1a 00
*.0........T.....*
00100040: 20 48 e5 9f 1f 60 e3 a0 00 3c e5 9f 10 01 e2 81   *H
..`...<.......*
00100050: fa c2 eb ff 00 04 e1 a0 0a 51 eb 04 00 0c e5 85
*........Q.......*
00100060: 00 00 e3 50 00 04 0a 00 20 04 e1 a0 10 00 e3 a0
*..P...... ......*
00100070: 0f 00 eb 04 00 00 e3 a0 a8 30 e9 1b 0c 40 e3 a0
```

```
*........0...@...*
00100080: 02 58 e2 80 a8 30 e9 1b 02 2c 00 11 ff d4 00 10
*X...0...,.......*
00100090: 02 64 00 11 c0 0d e1 a0 d8 10 e9 2d b0 04 e2 4c
*d........-...L.*
001000a0: 40 00 e1 a0 00 00 e3 54 00 04 1a 00 20 34 e5 9f
*.@....T.....4 ..*
001000b0: 1f 5a e3 a0 00 30 e5 9f 10 03 e2 81 fa a7 eb ff
*Z...0...........*
001000c0: 10 00 e3 a0 20 b0 e1 d4 00 04 e1 a0 21 82 e1 a0
*..... .......!..*
001000d0: 20 18 e2 82 0e e7 eb 04 00 04 e1 a0 0a 38 eb 04   *.
..........8...*
001000e0: 00 00 e3 a0 a8 10 e9 1b 00 f0 00 11 ff d4 00 10
*................*
001000f0: c0 0d e1 a0 d9 f0 e9 2d b0 04 e2 4c 70 01 e1 a0
*......-...L..p..*
```

m — Modify Memory

The “m” command allows you to poke register values. The contents of the specified address are displayed waiting for user input in hex format. The next register location is automatically displayed. Type “q” to quit.

Syntax

```
>> m [address [width]]
```

```
> m 0x100000
00100000:  0003-3
00100002:  e152-q
```

dt — Display Test Menu

The “dt” command displays the Diagnostic Test Menu.

Syntax

```
>> dt
```

et — Execute Test

The “et” command executes a specified test, allowing you to optionally enter a count for the number of times the test is to be executed.

Syntax

```
>> et test [loopCnt]
```

dbp — Display Boot Parameters

The “dbp” command displays the current boot parameter settings.

Syntax

```
>> dbp

TCP/IP Connectivity Settings
Default network name:        slot_009
Use DHCP:                    1
DHCP APIPA IP:               169.254.0.0
DHCP APIPA Netmask:          255.255.0.0
DHCP Retry Timer:            300
Static IP:                   192.168.254.3
Subnet mask:                 255.255.255.0
Gateway:                     0.0.0.0
Domain Name:                 acut.com
Enable UDP Announce          0
MAC Address:                 00:60:33:e0:01:01

Serial Connectivity Settings
Console TTY:                 2
Baud Rate:                   115200
Parity:                      0
Data Bits:                   8
Stop Bits:                   1
Flow Ctrl:                   0

System Parameters
EIP Enable:                  0
Load Job:                    1
Run Job:                     1
POST Flags:                  0x0200
Speaker:                     1
```

```
Advanced TCP/IP Settings
TCP Command:                49095
TCP Lossy:                  49096
TCP Lossless:               49097
TCP PLC:                    49098
TCP Heartbeat:              49094
TCP Heartbeat query:        49093
SNTP Intval:                0
SNTP Server:
```

mbp — Modify Boot Parameters

The “mbp” command modifies boot parameters. The command can be invoked in either Standalone or Manufacturing Mode.

Standalone Mode

Entering the “mbp” command by itself represents Standalone Mode and causes the bootloader to prompt you for each and every parameter. Parameters are displayed in groups of functionality. A new value may be entered or skipped altogether by pressing the Enter key. At any point, you may terminate the session by pressing the ESC key and then the Enter key. The bootloader automatically determines if a change has been made and, if so, prompts you as to whether or not changes should be saved to flash.

Syntax

```
>> mbp [field value]

>> mbp

TCP Connectivity Settings
Network Name: hawkeye -
Use DHCP: 1 -
DHCP APIPA IP: 169.254.0.0 -
DHCP APIPA Netmask: 255.255.0.0 -
DHCP Retry Timer: 300 -
IP Address: 168.192.254.3 -
Subnet Mask: 255.255.255.0 -
Gateway : 0.0.0.0 -
Domain Name : acut.com -
Enable UDP Announce -

Serial Connectivity Settings
```

```
ConsoleTTY: 0 -
Baud Rate: 115200 -
Parity: 0 -
Data Bits: 8 -
Stop Bits: 1 -
Flow Ctrl: 0 -

System Settings
EIP Enable: 0 -
Load Job: 1 -
Run Job: 1 -
POST Flags: 0x0000 -
Speaker: 0 -

Advanced TCP/IP Settings
TCP Command: 49095 -
TCP Lossy: 49096 -
TCP Lossless: 49097 -
TCP PLC: 49098 -
Heartbeat: 49094 -
Heartbeat Q: 49093 -
SNTP Intval: 0 -
SNTP Server:  -
```

Manufacturing Mode

You may modify a single field directly via the Manufacturing Mode:

```
>> mbp tty 0

Save parameters to flash? [y/n] N -
```

Table 6-1 shows the parameter names for Manufacturing Mode.

TABLE 6–1. Parameter Name for Manufacturing Mode

Description	Mfg Name	Parameters
Network Name	network ASCII string	
Use DHCP	dhcp	0 or 1
DHCP APIPA IP	dhcpip	Valid IP address
DHCP APIPA Netmask	dhcpmask	Valid IP address
DHCP Retry Timer	dhcpretry	Number of seconds before retrying DHCP
IP Address	ip	Valid IP address

TABLE 6–1. Parameter Name for Manufacturing Mode (Continued)

Description	Mfg Name	Parameters
Subnet Mask	subnet	Valid IP address
Gateway	gateway	Valid IP address
Domain Name	domain	ASCII string
Enable UDP Announce	eua	0 or 1
ConsoleTTY	tty	0, 1, or 2
Baud Rate	baud	115200, 57600, 38400, 19200, 9600, 4800, 2400
Parity	parity	0, 1, or 2
Data Bits	data	7 or 8
Stop Bits	stop	0, 1, or 2
Flow Ctrl	flow	0 or 1
EIP Enable	eip	0 or 1
Load Job	load	0 or 1
Run Job	run	0 or 1
Speaker	spkr	0 or 1
TCP Command	tcpcmd	
TCP Lossy	tcploss	
TCP Lossless	tcpless	
TCP PLC	tcpplc	
Heartbeat	tcphb	
Heartbeat Q	tcphbq	
SNTP Intval	tcpint	
SNTP Server	tcpsvr	ASCII string

dm — Display Menu

The “dm” command displays the Bootloader Menu.

Syntax

```
>> dm
```

dfb — Display Flash Blocks

The “dfb” command displays the contents of the flash block headers.

Syntax

```
>> dfb

The output is as follows:

Block   CRC   Size        Erase Count     Name
----------------------------------------------
0       F85F  00011A4C    000000ab        bootloader
1       BAF5  0000DE22    00000026        registry
2       D417  0000E74F    00000007        fpga
3       1D4B  0001E00C    00000001        settings_sec
4       C560  00000238    0000000c        bootparams_sec
5       8B43  001638FC    00000015        vxWorks
```

der — Display Ethernet Registers

The “der” command displays the contents of the Ethernet MAC/PHY registers.

Syntax

```
>> der

Bank 0            Bank 1            Bank 2            Bank 3
------            ------            ------            ------
TXCTRL: 0504     CFG:    a0b1      MMU:    3332      MCAST0: 0000
EPH:    0200     BAR:    1801      PKTNUM: 8000      MCAST2: 0000
RXCTRL: 0000     IAR1:   0000      FIFO:   8080      MCAST4: 0000
COUNT:  0000     IAR2:   0000      PTR:    0000      MCAST6: 0000
MIR:    0404     IAR3:   0000      DATA1:  01a4      MMI:    3332
RXPHY:  0000     GPR:    0000      DATA2:  01a4      REV:    3391
CTRL:   1210      ISR:    0004      ERCV:   001f

PHY Registers
0:  3400        16:  0022
1:  7809        17:  ff00
2:  0016        18:  4080
3:  f840        19:  ffc0
4:  01e1        20:  00a0
5:  0000
```

wmr — Write MAC Register

The “wmr” command writes “value” to an Ethernet MAC “register” offset from “bank”.

Syntax

```
>> wmr bank register value
>> wmr 0 4 8000      ‡ sets the software reset bit
```

wpr — Write PHY Register

The “wpr” command writes “value” to an Ethernet PHY “register”.

Syntax

```
>> wpr register value
>> wpr 0 400         ‡ puts the PHY in internal loopback
```

cpu — Display CPU Registers

The “cpu” command displays the contents of CPU registers.

Syntax

```
>> cpu

Memory Registers: (0xa0000000)
MDCNFG:  0x0491b255
MDCAS00: 0xaaaaaa9f
MDCAS01: 0xaaaaaaaa
MDCAS02: 0xaaaaaaaa
MSC0:    0xfffc4774
MSC1:    0x4115fffc
MECR:    0x0d280d28
MDREFR:  0x02300151
MDCAS20: 0x1861861f
MDCAS21: 0xfff86186
MDCAS22: 0xffffffff
MSC2:    0xfffc4111
SMCNFG:  0x00000000

Coprocessor 15 Registers:
ID:   0x6901b119
CTRL: 0x00000270
```

```
TLB:  0x3f6d8000
DAC:  0x76cdda44
FSR:  0x0000009b
FAR:  0x55d37dbc

CPSR: 0x600000d3
```

flsh — Display System Flash Size

The “flsh” command dynamically sizes system flash by loading a piece of code into memory and vectoring to it so that it can query the flash ID register.

Syntax

```
>> flsh
Flash size: 0x00400000 --> 4194304 bytes
```

ram — Display System RAM Size

The “ram” command displays the size of system RAM by writing unique data patterns to all address boundaries and checking where word-wrap is detected.

Syntax

```
>> ram
RAM size:   0x02000000 --> 33554432 bytes.
```

cach — I-Cache Control

The “cach” command controls the “state” (1=on, 0=off) of the system instruction cache.

Syntax

```
>> cach state
```

x — File Transfer and Execute

The file transfer and execute utility downloads a program over the serial link, places it in RAM, and jumps it. This utility represents a combination of the old “download” and “execute” commands.

r — Reset Unit

The “r” command resets the unit via software reset.

Syntax

```
>> r
```

j — Jump to Application

The “j” command jumps to the application code.

Syntax

```
>> j
```

e — Display Last Logged Error

The “e” command displays the last error logged by the bootloader/diagnostics.

Syntax

```
>> e
```

h — Display Command Help

The “h” command displays help for each of the commands in the Bootloader Menu.

Syntax

```
>> h
```

Diagnostic Test Menu

The Diagnostic Test Menu displays a list of tests that are currently supported for HawkEye™ 1500 diagnostic debug and verification development. Each test can be executed individually or as part of a group that is determined by attributes associated with the test.

Note: The attributes are set internal to the software and cannot be modified at runtime.

Test suites are grouped in increments of 10 starting with test 100. For example, bus tests are part of the suite numbered 100-109, flash related tests are 110-119, and so forth.

```
 10 All Tests            20 All Mfg Tests        30 All Internal Tests
 40 All Power-on Tests
100 All Bus Tests       101 Data Line           102 Address Line
103 Memory Cell         104 Inverted Cell
110 All Flash Tests     112 Flash Bootloader    113 Flash Blocks
120 All RS232 Tests     121 RS232 Internal      122 RS232 Interrupt
130 All Ethernet Tests  131 Ethernet Reset      132 Ethernet MAC Lpbk
133 Ethernet PHY Lpbk   134 Ethernet EXT Lpbk   135 Ethernet Interrupt
```

Table 6–2 enumerates which tests are executed for groups 10 ("All Tests"), 20 ("All Mfg Tests"), 30 ("All Internal Tests"), and 40 ("All Power-on Tests").

TABLE 6–2. The Tests That Are Executed

Test Name	All Tests	All Mfg Tests	All Internal Tests	All Power-on Tests
Data Line	X	X	X	X
Address Line	X	X	X	X
Memory Cell	X	X	X	
Inverted Cell	X	X	X	
Flash Bootloader	X			X
Flash Blocks	X	X	X	X
RS232 Internal Loopback	X	X	X	X
Ethernet Reset	X	X	X	X
Ethernet MAC Loopback	X	X	X	X
Ethernet PHY Loopback	X	X	X	X
Ethernet EXT Loopback	X	X		

LEDS

The system LEDs visually convey power-on status and error codes.

Power-on Sequence

Each stage of the power-on sequence drives the LEDs in a binary up-count fashion according to the table below. The LEDs are lit before the test is executed and remain in that pattern until the next test is run or an error condition is detected and displayed.

TABLE 6–3. Power-on Sequence

Mode	Fail	Pass	Trig	Test
			X	Data Line Test
		X		Address Line Test
		X	X	Flash Bootloader Test
	X			Flash Block Test
	X		X	RS232 Internal Loopback Test
	X	X		Ethernet Reset Test
	X	X	X	Ethernet MAC Loopback Test
X				Ethernet PHY Loopback Test

Error Codes

In the event of an error being detected, the system beeper will sound five times and an error code representing the test that failed will flash on the LEDs. The LEDs continue to flash until the QuicSet button is pressed, at which point an error message is logged to the serial port and the Diagnostic Monitor is launched.

APPENDIX A

HawkEye™ 1510

This appendix contains information specific to the HawkEye™ 1510 Smart Camera-Based Reader.

FIGURE A–1. HawkEye™ 1510

Optics

The HawkEye™ 1510 uses the following lenses.

- CS Mount Lenses
- C Mount Lenses (requires 5 mm extension (Microscan part number 928-0047-1)):

  CS mount + 5mm = C mount

TABLE A–1. Lens Sizes and Microscan Part Numbers

Part Number	Size
C-Mount Lenses (Requires 5mm extension: Microscan P/N: 928-0047-1)	
928-0057-1	Lens: 8mm, F/1.4-16, Filter Thread: 25.5mm P 0.5mm
928-0057-2	Lens: 12mm, F/1.8-16, Filter Thread: 25.5mm P 0.5mm
928-0057-3	Lens: 16mm, F/1.4-16, Filter Thread: 25.5mm P 0.5mm
928-0057-4	Lens: 25mm, F/1.6-16, Filter Thread: 25.5mm P 0.5mm
928-0057-5	Lens: 35mm, F/2.1-22, Filter Thread: 25.5mm P 0.5mm
928-0057-6	Lens: 50mm, F/2.8-22, Filter Thread: 25.5mm P 0.5mm
928-0057-7	Lens: 75mm, F/3.9-32, Filter Thread: 25.5mm P 0.5mm
CS-Mount Lenses	
928-0049-1	Lens: 2.8mm, F/1.3-C
928-0049-2	Lens: 4mm, F/1.2-C
928-0049-3	Lens: 8mm, F/1.2-C
928-0049-4	Lens: 16mm, F/1.4-C

Table A–2 contains working distances and fields of view with various extensions and lenses.

TABLE A–2. Working Distance and Fields of View[1]

Lens Focal Length	Extension (mm)	8 mm	12 mm	16 mm	25 mm	35 mm	50 mm	75 mm
Horiz. FOV (mm) Camera Face Distance (mm)	5.0	108 203	91 248	81 305	38 234	36 328	36 495	28 582
Horiz. FOV (mm) Camera Face Distance (mm)	5.5	41 94	44 140	51 197	28 201	43 297	33 460	26 561
Horiz. FOV (mm) Camera Face Distance (mm)	6.0	28 74	34 107	38 157	29 193	29 282	32 439	25 546
Horiz. FOV (mm) Camera Face Distance (mm)	10.0		9 56[2]	13 75	14 118	17 197	21 333	20 461
Horiz. FOV (mm) Camera Face Distance (mm)	15.0			7 61	9 94	11 159	14 269	15 395
Horiz. FOV (mm) Camera Face Distance (mm)	20.0				6 87	9 144	11 232	13 357

Notes: [1] Lens focus ring adjusted to close position.
[2] Lens focus ring at mid point position.

External Lighting Mounting Options

- HawkEye™ Light Mount Assembly (HELTMA) Brackets
  - HELTMA-1S thru HELTMA-4S (Short - 6.5 inches (165.1mm))
  - HELTMA-1L thru HELTMA-4L (Long - 8.5 inches (215.9mm))
  - HELTMA-5 (30 Degree outrigger bracket)
  - HELTMA-6 (30 Degree outrigger assembly)
- Adjustable brackets simplify mounting for Microscan NER lights
  - DOAL 25
  - DOAL 50
  - DF-150-3…

Figure A–2 shows light mount bracket HELTMA-1L with NER Doal 50 attached.

FIGURE A–2. Light Mount Bracket HELTMA-1L with NER Doal 50

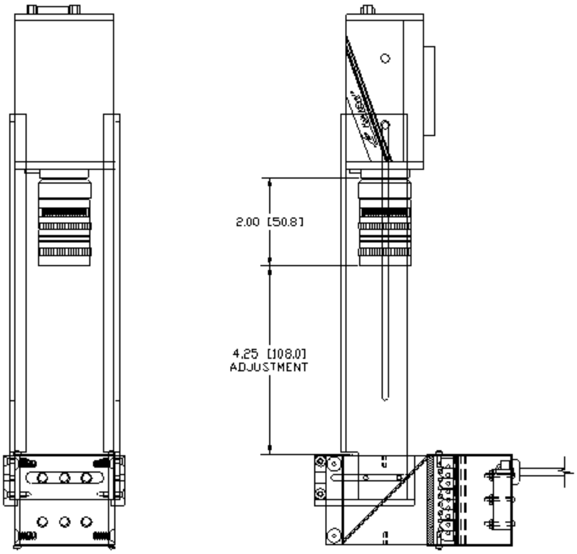


Figure A–3 shows light mount bracket HELTMA-1L with NER DF-150-3 dark field illuminator and mounting bracket attached.

FIGURE A–3. **Light Mount Bracket HELTMA-1L with NER DF-150-3**

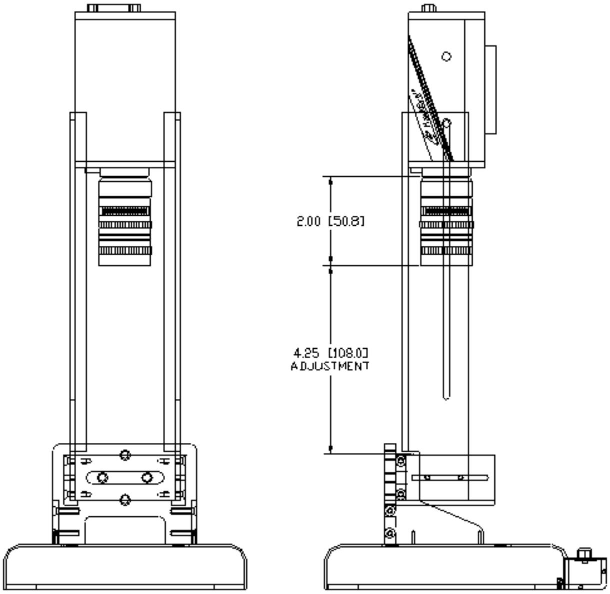


TABLE A–3. **HawkEye™ Light Mount Assembly Brackets**

Part Numbers	Description
External Lighting Brackets for On-Axis Lights	
HELTMA-1L HELTMA-1S	Adjustment range: 156.7mm [6.17”] Adjustment range: 100.1mm [3.94”] Compatible with NER DF-100 and DF-150 series Dark-Field Illuminators. Note: DF-150-3 requires NER mounting bracket part number 090-000100
HELTMA-2L HELTMA-2S	Adjustment range: 156.7mm [6.17”] Adjustment range: 100.1mm [3.94”] Compatible with NER R-100 “V2” series Ring Illuminators.
HELTMA-3L HELTMA-3S	Adjustment range: 156.7mm [6.17”] Adjustment range: 100.1mm [3.94”] Compatible with NER R-60 “V2” series Ring Illuminators.

TABLE A–3. HawkEye™ Light Mount Assembly Brackets (Continued)

Part Numbers	Description
HELTMA-4L HELTMA-4S	Adjustment range: 156.7mm [6.17”] Adjustment range: 100.1mm [3.94”] Compatible with NER Doal-50 “V2” series Diffused On-Axis Lights.
External Lighting Brackets for Off-Axis Lights	
HELTMA-5	30 Degree outrigger bracket Compatible with NER AR-50 9 Pin-D series Area Array Illuminators and BL-50 9 Pin-D series Backlights.
HELTMA-6	30 Degree outrigger assembly Includes HE1515 Bright Field Light. Requires external 12 volt power supply, compatible with NER 010-502400 (US cord), 010-502500 (EU cord) power supplies.

Power for Lights

Options

- External Independent Power
- Internal Power Connector
  - Continuous – 12v, 0.5A
    - DOAL 25, DOAL 50, DF100-1 ONLY
  - Strobe – Use external power and strobe modules
    - When LIGHTING:EXTERNAL is selected TTL O/P 1 (Pin 5) becomes Strobe Trigger

Do not exceed 0.5 amp continuous current draw or you may damage the HawkEye™ 1500.

Lighting Connector

For complete information about the Light Port Connector, see “Light Port Connector” on page 2-45.

B

APPENDIX B

Troubleshooting & Frequently Asked Questions

This appendix contains troubleshooting tips and answers to frequently asked questions for the HawkEye™ 1500 Series Smart Camera-Based Reader.

Frequently Asked Questions

My camera is connected to the network and serial port, but I have no idea what the current communication settings are. How do I figure it out?

The HawkEye™ 1500 can communicate via the serial port or over Ethernet to ReadRunner. First, try using ReadRunner and add a button for either COM1 or COM2, depending on the serial port on your PC you have connected to the camera. Once connected, ReadRunner hooks reports and/or images from the camera on a timed basis. This data is transferred via Xmodem protocol. If you can connect via the serial port, click Settings>Serial/TCP Settings… and examine the network and serial port configuration.

If ReadRunner does not connect with the serial port, try using HyperTerminal (provided by the operating system). Run HyperTerminal and create a connection on your serial port with the following settings:

- Baud rate — 115200

- Parity — N
- Data bits — 8
- Stop bits — 1
- Flow control — None

These are the default camera settings. Connect to the camera and see if you get any text. If nothing comes out, press Ctrl+T. This control character causes the camera to dump an information string. If you don't get a string, then your serial port parameters do not match the camera. You must try different settings until they match and you get characters from the camera. If you get a string that looks like "->", type in:

cmd "console 0"

and reboot the unit. Reconnect to the camera and verify that result strings are properly received and displayed.

Once you get a set of serial parameters that work, you can use a series of commands to determine the camera configuration. In HyperTerminal, press Ctrl+S to take control of the unit and take it offline. Then, press Ctrl+P to get a prompt and activate terminal echo over the serial port. The following commands tell you how the camera is configured:

IPCONFIG ? — Displays the name of the camera and its current IP address

DHCP ? — Returns Y (if DHCP is on) or N (if DHCP is off)

You can use this information to verify the network configuration. Use the IP command to change the name of the camera and the address you want to use when DHCP addressing is disabled.

I have DHCP activated, but the camera reports a 169.254.x.x address. What's happening?

If your HawkEye™ 1500 is programmed for DHCP addressing and there is no DHCP available, the HawkEye™ 1500 defaults to a class B address in the 169.254.x.x subnet. This is similar to how Windows 2000 and XP function. It also allows you to connect a Windows 2000 PC and HawkEye™ 1500 with a cross-link Ethernet cable and leave DHCP addressing activated, but still communicate. Both sides of the connection pick a different 169.254.x.x address and should communicate normally.

How can I tell if the IP configuration of my PC and my camera are valid?

Your PC and camera must be operating on the same subnet. The "Subnet Mask" determines the base address of the subnet. Typically, the subnet is either 255.255.0.0 (Class B) or 255.255.255.0 (Class C). For Class B, this means the first two numbers of the address define the subnet and the last two can be any numbers from 1 to 255 each. For Class C, this means the first 3 numbers define the subnet and the last number can be any number from 1 to 255.

You can determine your PC's network configuration from a Command Prompt by running "ipconfig /all". This utility displays each network entity and all its details. Ensure that the PC and the camera have identical subnets and valid IP addresses.

How do I restore the camera to factory defaults?

ReadRunner allows you to restore the decoder and/or application mode settings to factory defaults. Some settings are persistent still (speaker on/off, communication settings, etc.), so, to truly restore all settings to factory defaults, use the RESET ALL command and reboot the camera. This restores all decoder settings, application mode settings, and communication settings to the defaults. The camera's name changes back to "HawkEyexxyyzz" (where xxyyzz is the last three octets of the camera's MAC address), and DHCP is activated.

When should I use DHCP?

DHCP is a useful protocol for dynamically addressing client computers using TCP/IP. However, there are some warnings as well. DHCP usually requires clients to "renew" addresses over a period of time. This can cause client addresses to change at any time, depending on how your network is configured. The problem is, if you are receiving data from your HawkEye™ 1500 over TCP/IP and the Ethernet address of either the PC or the camera changes, you will lose connectivity to the camera and need to re-establish it. This can mean loss of data to the client.

If your application requires that you cannot lose data, do not use DHCP. Work with your IT department to statically assign addresses to both the PC and the HawkEye™ 1500 so they do not change addresses.

I have no idea what the current settings are for the camera. What do I do?

Should you encounter a situation where you have no idea what the current settings on the camera are, you can reset the camera to its factory defaults and resume from there.

To restore the factory default settings to the camera, press Ctrl+Z, or select Restore Factory Defaults from the ReadRunner File menu. Click Yes when asked if you wish to control camera X. ReadRunner displays the message:

> Camera settings are restored to defaults

The HawkEye™ 1500 camera also beeps once.

What if Learn succeeds but read fails?

When Learn or Assisted Learn is successful, the Pixels Per Cell should ideally be set to the average cell size in pixels. This value can be estimated by dividing the Matrix Height by the Number of Matrix Rows. If this value is far greater than the default value of 6, then it can increase the decode time significantly. An attempt to optimize the HawkEye™ 1500 for speed for a variety of applications is to always set the Pixels Per Cell to 6 regardless of the actual value. In rare situations, this optimization can have negative impact on the robustness. Therefore, if an image fails to read when Learn is successful, try to increase or decrease the Pixels Per Cell gradually until the image is decoded.

My decode data is very long. Is there a way to disable the sending of this data on the serial port?

If your data is very long, it may take a long time to report this data on the serial port. If you are not using the serial port at all, you can gain this time back for processing. You can disable the serial port connectivity by using the CONSOLE command. The CONSOLE command directs the camera to connect its operating system debugging console to a particular serial port. The serial port surfaced on the camera is port 2, so use the command “CONSOLE 2” to redirect the debugging console accordingly. Once the debugging console is set, result data will no longer be sent to the serial port.

Note: Once this option is set, your camera is unusable over the serial port. To restore the camera's normal serial port connectivity, issue the command "CONSOLE 0" from the Terminal window in ReadRunner.

A connection has taken control of my camera and I can't regain control. Is there a way to break this control so I can get it back?

Yes. You have three options:

1. Reboot the camera.
2. Connect your serial cable to the camera, open HyperTerminal, and connect to the camera. Press Ctrl+S to forcibly take control of the camera and take it offline. You should see an information string with details about the name of the camera, current counts, software version, etc., when this happens. Press Ctrl+R to release control and put it back online.
3. If the camera is under control via an Ethernet connection, disconnect the Ethernet cable from the camera. Wait a few seconds and plug it back in. Any Ethernet connections are cleared and you should be able to reconnect to the camera and regain control.

When I disconnect the network cable from the PC while the camera is under control by ReadRunner, reconnecting the network cable will not allow ReadRunner to take control again. What do I do?

ReadRunner cannot regain control of the camera because control was not properly released when the network cable was disconnected from the PC. Use option 3 in the previous FAQ to regain control of the camera.

I had control of the camera over the serial port and left the machine for a few minutes. When I came back, the camera was no longer under control. What happened?

When you have control of the camera over the serial port, control is maintained as long as there is input activity on the serial port. If there is no activity for 2 minutes, then the camera is released and put back online (if it was offline). To override this timeout, take control of the camera and type Ctrl+S a second time. The timeout is now disabled until you release control again. You can use the ONLINE command (see the HawkEye™ 1500 Series Reference and Programmer Manual) to put the unit back online and maintain control.

I’m using the Part Queue to record images on the camera but, after a while, the camera runs much slower. What’s going on?

The Part Queue mechanism of the camera can store, typically, up to 30 sets of images and results in a ring buffer. This can consume most of the camera memory and can fragment memory. Memory will be fragmented whenever the Part Queue is cleared, and the queue is cleared either when the images are uploaded or cleared in QueueView. From QueueView, you can also upload the records without clearing the Queue. To avoid this memory fragmentation, upload the Queue without clearing it; the Queue will continue to reuse its records and performance will not be impacted.

Do the version numbers have to match?

Yes. For proper operation, the version of the software displayed in the Network Overview Window must match the ReadRunner revision used to connect to the camera. Starting with V2.0, ReadRunner is now capable of communicating and controlling V1.0 and V1.1 camera software.

What’s the timing for normal strobe and power strobe? I’m assuming that both strobe modes would go off immediately after the trigger (or the configured delay) and then stay on for some fixed duration? Is that correct? What's the duration? Is the duration different for each strobe mode?

The timing is identical for the Strobe and Power Strobe. The difference is the light intensity output. In Power Strobe, the LEDs are overdriven, and therefore brighter. The duration is controlled by the Exposure setting and is limited to a maximum of 1 msec for Power Strobe as opposed to 20 msec for Strobe.

Is the “Exposure” in the Photometry dialog and command the same as the “Shutter” on the HawkEye™ 15?

Yes, it is an electronic shutter internal to the CCD. It can be set between 30 usec and 20000 usec. It is the time during which light is integrated by the CCD sensor and also the time the light remains ON when in Strobe or Power Strobe.

When using an external fast Xenon strobe, then the duration of the strobe pulse is no longer controlled by the camera and the exposure acts as an electronic shutter as opposed to also controlling the duration of the light remaining on. This allows for shorter light pulses than 30 usec for very fast motion applications.

What exactly happens with auto photometry when using a sensor as a trigger? The HawkEye™ has only one chance to get an image, so I can only imagine that the settings are adjusted after each image, hoping that the adjustment will be appropriate for the next part. Am I correct?

In Motion mode, the Auto Photometry is turned off, as this requires up to 5 “small” pictures, which take about 10 msec each. By the time the Exposure and Gain are figured out, the part would have moved out of the field of view. In this mode, Manual Photometry should be used. In Stop and Scan mode, Auto Photometry can be used.

I’m trying to re-install all my computer software after it was attacked by a virus. When I run the ReadRunner install, neither the “Repair” option nor the “Remove” option seems to do anything. How can I re-install ReadRunner once this happens?

ReadRunner uses InstallShield as its install engine. All InstallShield installation projects use an identifier (or GUID) to identify whether or not the software is installed on the system. When the installation is run and the GUID is detected, you get the “Repair/Remove” dialog that lets you re-run the install with everything that was selected, or remove the software. However, if a virus has wiped out some of this information, you may not be able to re-install. To forcibly re-install, find the hidden folder “Installshield Installation Information” under C:\Program Files and delete the folder:

{2D1FFA0F-F587-11D6-8110-00C04F606F20}

The next time you run the install, it will operate as if the software is installing for the first time.

Sometimes, when using a Logitech mouse and scrolling with the wheel, I see crashes in ReadRunner especially in the Network Overview form. What can I do to fix this behavior?

This occurs with older Logitech mouse drivers when trying to scroll the Microsoft FlexGrid control with the wheel. Updating to the latest Logitech drivers will fix this problem available on the Logitech website.

Trouble Reading

Setting the HawkEye™ 1500 to Factory Default Settings

If you have problems reading or decoding a symbol, it could be because the system is set up with a “learned” Data Matrix. The system may have been configured to read a specific type of symbol; one that does not match the characteristics of the provided symbols (or the one you may have created yourself). If you do not have access to ReadRunner, you can “open up” the decoder to read almost any symbol by depressing the QuicSet button on the back of the camera for 3 to 4 seconds. For more information, see the section titled “QuicSet Decoder Factory Defaults” in the HawkEye™ 1500 Series Quick Start Guide.

Samples of Reader Programming Data Matrices

Setting Serial Communications

TTY 2400 N 8 1 N   TTY 9600 N 8 1 N   TTY 115200 N 8 1 N

Setting Triggers

TRIG T

TRIG C

Resetting

RESET DECODER

RESET APPMODE

RESET FACTORY

Setting Targeting

TARGET OFF

TARGET ON

Setting Beeper

BEEP N

BEEP Y

Setting Illumination

ILLUM OFF

ILLUM ON

ILLUM STROBE

Resetting ROI

ROI RESET

Setting Learn/Unlearn

LEARN

UNLEARN

Setting Photometry

PHOTO MAN

PHOTO AUTO

Saving

SAVE

Setting DHCP

DHCP Y

DHCP N

C

APPENDIX C

Upgrading Camera Software

This appendix describes how to upgrade the HawkEye™ 1500 Series Smart Camera-Based Reader's software using either HawkEye™ Flasher or the HawkEye™ bootloader.

Note: In this chapter, we use specific camera and software version numbers as an example. Please use the latest HawkEye™ software available and substitute in your version numbers in the procedures.

Overview

The HawkEye™ 1500 uses Flash memory to store a series of binary images that are executed when the camera starts up. These images include:

- Bootloader — Simple menu-driven interface that examines Flash memory and downloads and executes .SEC files (via the serial port).
- FPGA — Binary image that drives the image acquisition hardware.
- BootParams — Binary image that sets up networking and other persistent parameters.
- Registry — Binary image used for all decoder, acquisition, and serial communication parameters.
- vxWorks Kernel — The camera operating system itself.

Caution: While the versions of vxWorks kernel and Bootloader should be kept the same by downloading both, downloading the registry will wipe out customer configurations, and should be used only when this is desired.

Typically, when you upgrade the camera's software, you upgrade only the Bootloader, Registry and vxWorks Kernel, and you use the HawkEye™ Flasher to do so. You can use the Flasher to upgrade your camera over the network or by using the serial port. These files are installed on your hard drive in the "Camera Binaries" folder under ReadRunner (typically, "C:\Program Files\ReadRunner\Camera Binaries").

Use the HawkEye™ Bootloader only when you have no network connection to the camera and your camera fails to start up.

Using HawkEye™ Bootloader

The HawkEye™ Bootloader is a simple, menu-driven interface embedded on the camera that examines and updates the camera's flash memory. When using the bootloader, you transfer .SEC files to the camera via the serial port and execute the files. These .SEC files automatically update the appropriate section of flash memory.

Note: All .SEC files were written to flash memory at the Microscan factory when your camera was assembled. There is no specific need to re-write any portions of this flash memory with the bootloader. Use the bootloader to update your camera only when instructed to do so.

The .SEC files that created your camera's software image are installed on your hard drive in the "ReadRunner/Camera Binaries" folder.

The .SEC files on the CD include:

- bootloader_Reader.sec — Bootloader image
- registry_Reader.sec — Registry image
- vxWorks_Reader.sec — vxWorks kernel image

To run the bootloader:

1. Power up your camera while pressing in the Quicset button.
2. Connect the serial cable to the camera and your PC.
3. Run HyperTerminal by selecting Start>Programs>Accessories.
4. Connect to the appropriate serial port using the following settings:
   - Baud Rate — 115200
   - Data Bits — 8
   - Parity — N
   - Stop Bits — 1
   - Flow Control — None
5. Click OK.

   The following information is displayed:

```
HawkEye BootLoader HE1500 Ver 5.45 (206MHz, 32MB RAM, 4MB
Flash)

  d     Dump memory
  m     Modify memory
  dt    Display tests
  et    Execute test
  dbp   Dump boot params
  mbp   Modify boot params
  dm    Display menu
  dfb   Dump flash blocks
  der   Dump enet regs
  wmr   Write MAC reg
  wpr   Write PHY reg
  cpu   Dump CPU registers
  flsh  Get Flash size
  ram   Get RAM size
  cach  I-Cache control
  x     File transfer
  r     Reset unit
  j     Jump to app
  e     Display error
  h     Help
>>
```

C

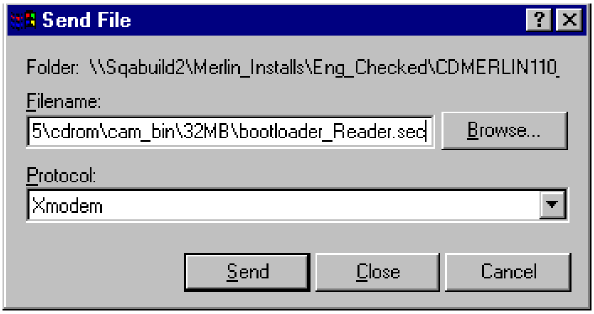
Upgrading Camera Software

6. Press X to download a .SEC file.

7. Select Send File from the Transfer menu.

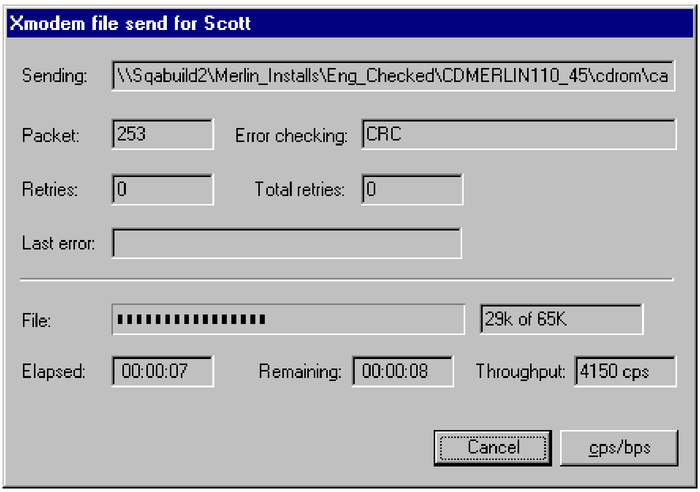
8. Browse for bootloader_Reader.sec on the CD, select Xmodem as the protocol, and then click Send, as shown in Figure C–1.

FIGURE C–1. Send File Window



Xmodem transfers the file to the camera’s memory, as shown in Figure C–2.

FIGURE C–2. Xmodem File Send Window



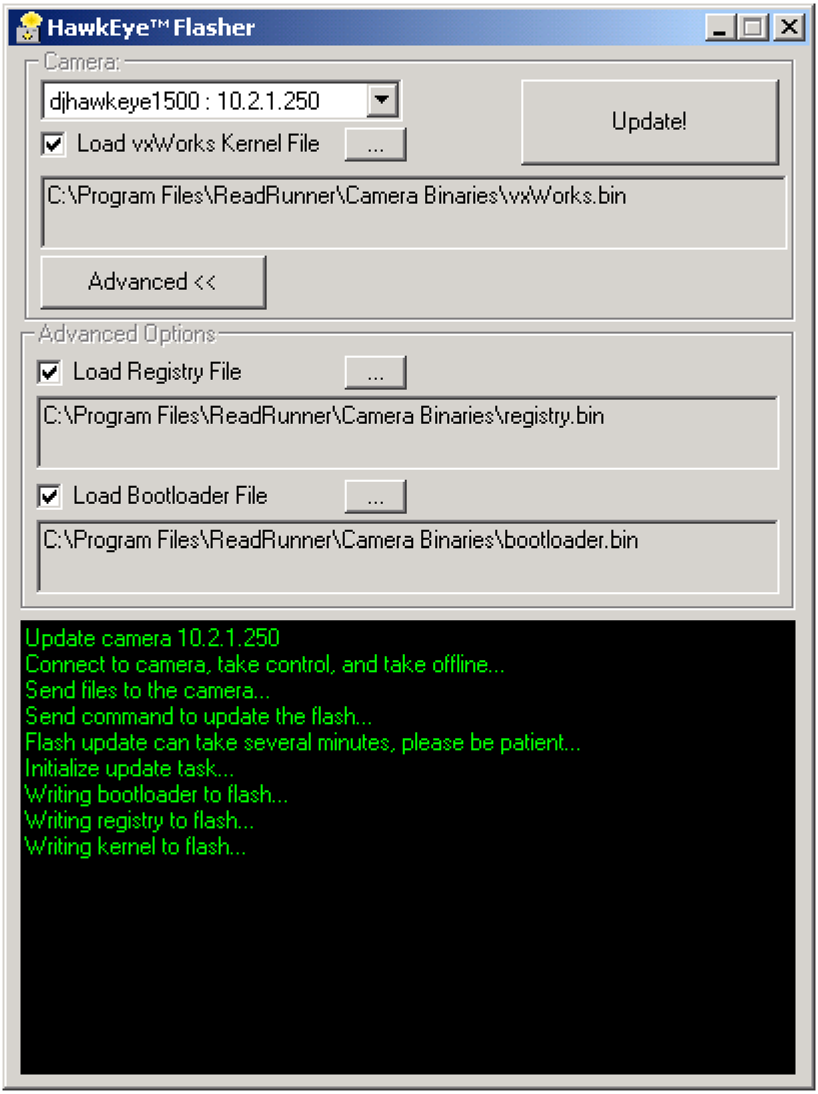
When complete, the bootloader menu is re-displayed.

9. When complete, press J to jump to the camera application.

Using HawkEye™ Flasher

The HawkEye™ Flasher is a simple application you use to upgrade your camera’s software. It allows you to select a camera, a Registry file, a Bootloader file, and/or a vxWorks kernel file, then update the camera accordingly.

FIGURE C–3. Flasher Window



The cameras that are detected on the local network are listed in the “Camera:” box, along with “COM1:” and “COM2:”. You can flash cameras on a separate subnet by selecting <Static IP Address> in the Camera drop down window and typing in the IP address of the camera. Camera devices are automatically added to the list as they are detected on the network. The last camera you selected is automatically selected when detected on the network. All file selections and settings are also restored each time you run Flasher, except for those cameras defined with <Static IP Address>.

The files you select with the Flasher are .BIN files; that is, these files are binary executable images readable by the camera. The files are transferred to the camera’s memory and are then moved to the Flash memory.

Caution: Before updating your camera’s Registry, be sure to save your camera’s configuration to the hard drive using ReadRunner. Otherwise, the configuration will be lost.

The Flasher monitors this entire process. When the process finishes, the Flasher automatically reboots the camera and reports the new software version.

The Flasher also supports several command line options:

- “-quit” — When this option is given, the Flasher automatically updates the last camera you selected according to the previous settings.
- “-doreg” — This option sets the “Load Registry File” option.
- “-noreg” — This option clears the “Load Registry File” option.
- “-dokernel” — This option sets the “Load vxWorks Kernel File” option.
- “-nokernel” — This option clears the “Load vxWorks Kernel File” option.
- “-dobtl” — This option sets the “Load Bootloader” option.
- “-nobtl” — This option clears the “Load Bootloader” option.
- “-cam name” — This option allows you choose the specific camera to update, by name.

Use the following procedure to upgrade the camera using Flasher:

1. Select Start, Programs, ReadRunner #.#, Utilities, and HawkEye Flasher. The Flasher window is displayed, as shown in Figure C–3.

2. From the Camera pull down menu, select the camera whose software you want to upgrade.

Note: You can also select <Static IP Address>. If you do so, the “Enter Camera Static IP Address” dialog box is displayed. Enter a valid IP address and click OK.

---

3. Select Load Registry File. This is the camera’s system Registry, and is similar to the Window’s Registry.
4. Click ... and navigate to the Camera Binary folder in the ReadRunner folder. Highlight registry.bin.
5. Select Load vxWorks Kernel File. This is the camera’s software kernel.
6. Click ... and navigate to the Camera Binary folder in the ReadRunner folder. Highlight vxWorks.bin.
7. Select Load Bootloader File. This is the software that boots the camera. Click ... and navigate to the Camera Binary folder in the ReadRunner folder. Highlight bootloader.bin.

Note: Always load the version of the Bootloader that corresponds to the version of vxWorks.bin to assure consistency, unless explicitly told not to in the HawkEye™ 1500 Release Notes.

---

8. Click Update!

Flasher stores the files in the flash memory of the camera. This process takes about 40 seconds.

D

APPENDIX D Symbology Reference

This appendix contains a description of Data Matrix, information about programming through the reader, and symbol samples.

Data Matrix

Data Matrix Certification

Data Matrix has been certified by AIM, Inc. (Automatic Identification Manufacturers) and ISO as a fully public-domain symbology. It is located on the Internet at **http://www.iso.org**, which provides the complete specification.

- ISO/IEC 16022:2006 Information Technology — Automatic identification and data capture techniques — Data Matrix bar code symbology specification

Data Matrix has also been selected as a standard for 2D marking by a majority of industry organizations such as Department of Defense (DoD) Unique Identification (UID) program, Semiconductor Equipment & Materials International (SEMI), Electronics Industries Association (EIA), Air Transport Association (ATA), Automotive Industry Action Group (AIAG), and Health Industry Business Communication Council (HIBCC).

What Is Data Matrix?

The Data Matrix is a unique machine-readable symbol capable of storing a large amount of information within a small physical size.

The Data Matrix symbology allows for two-dimensional encoding and decoding. For the first time, users are no longer constrained by the limitations of a printed symbol. Now, with the Data Matrix symbology, users have the ability to:

- Select the output size ranging from 0.001 square inches to 14 square inches, regardless of the amount of data encoded.
- Encode up to 3116 numeric characters or 2335 alphanumeric characters in a single symbol.
- Encode and read many supported international languages.
- Easily integrate with existing computer systems.

Data Matrix symbols are capable of carrying 25 to 100 times more information than the typical barcode. This range is directly related to the image quality that the printer is capable of producing.

The following summarizes characteristics of Data Matrix:

- Both height and width are used to encode data.
- Works with contrast as low as 20%.
- Readable through 360^{o} of rotation.
- Designed to survive harsh industrial environments.
- Codes can be marked on the surface of a part, without using a paper label.
- Several error correction schemes are available to optimize symbol damage recovery.
- Advanced image processing is used for higher read rates of cluttered and/or damaged symbols.

Figure D–1 illustrates a Data Matrix on an actual part.

FIGURE D–1. Data Matrix Example

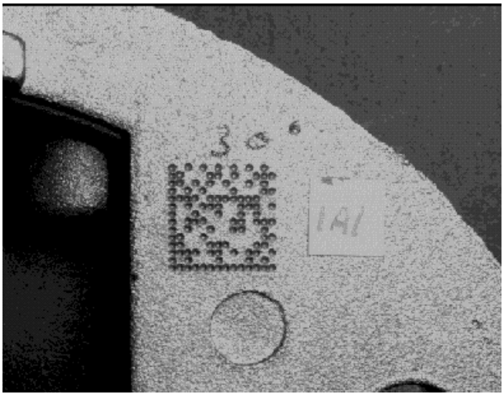
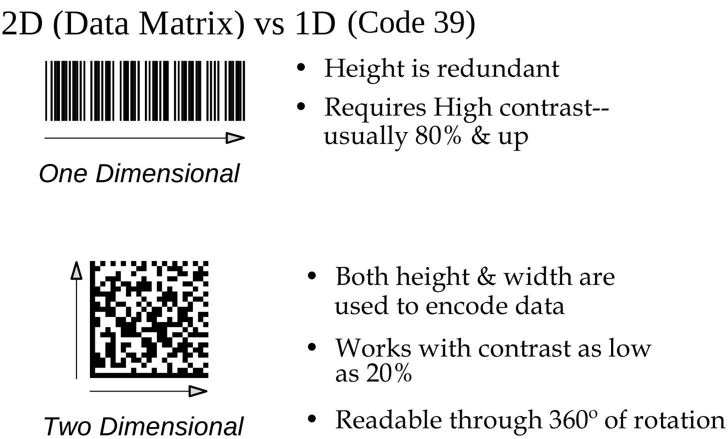


Figure D–2 illustrates the differences between 2D and 1D symbols.

FIGURE D–2. 2D (Data Matrix) vs. 1D (Code 39)

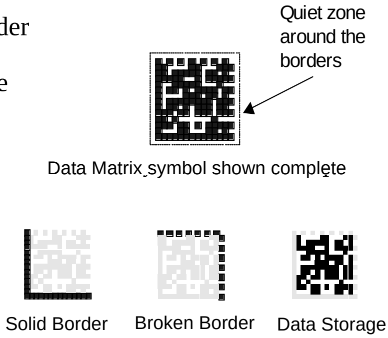


Data Matrix Components

Data Matrix consists of a solid border, a broken border, a data storage area, and a quiet zone, as shown in Figure D–3.

FIGURE D–3. Data Matrix Components

- Solid Border
- Broken Border
- Data Storage
- Quiet Zone

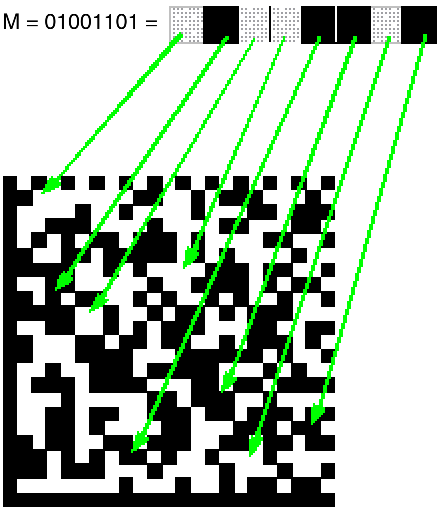


The HawkEye™ 1500 uses the solid border to calculate the rotation of a Data Matrix symbol. The HawkEye™ 1500 uses the broken border to identify the number of rows and columns in a symbol. The data storage area contains the binary information that was encoded during the construction of the Data Matrix.

The data storage area is illustrated in Figure D–4, showing how data is stored in 8-bit format. Each encoded data character can be represented by an 8-bit binary code. The character M, for example, is 01001101:

- 1 — Dark cell
- 0 — Light cell
- M — Stored in the Data Matrix, as shown in Figure D–4

FIGURE D–4. Data Matrix Storage



The Quiet Zone is a clear white space surrounding the Data Matrix, without clutter or text of any kind. The width of the quiet zone must be at least the size of one cell. The quiet zone should be at least 10% of the width or height of the symbol in order to achieve an optimum read rate.

Data Matrix Error Correction

Error correction provides safeguards and additional capabilities for handling poorly printed or damaged symbols. You select Error Correction Codes (ECC) when you encode the Data Matrix symbol. Data Matrix is the only symbology which offers both Reed-Solomon (ECC 200) and Convolutional error correction codes (ECC 000-140). Reed-Solomon is a byte-correcting scheme preferred for block damage recovery. Convolutional code is a bit correcting scheme preferred for random damage recovery. With the proper choice of Error Correction Code for a given environment, a high rate of decoding can be achieved.

In Convolutional code, as the ECC level increases, data redundancy is added to the Data Matrix symbol, as well as increased overhead (OVHD) in ECC 000-140. Additionally, as the ECC levels increase, the physical marked area increases proportionally. The redundant data is randomly placed (encoded) inside the data storage area to increase the symbol's ability to recover from damage.

In Reed-Solomon code, the OVHD varies and is based on the size of matrix. See Table D–4, "ECC 200 Symbol Attributes," on page D-8 for more information.

The examples in Figure D–5 illustrate Data Matrices with different ECC levels and how ECC affects symbol density.

FIGURE D–5. Data Matrix Examples with Different ECC Levels

ECC000 ECC050 ECC080 ECC100 ECC140 ECC200

Data = 123456789
Format = 1

Table D–1 shows the correlation between ECC levels and required overhead.

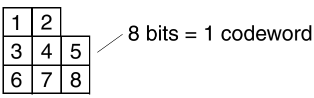
TABLE D–1. ECC Levels vs. Required Overhead

Selected ECC Levels	Required Overhead (%)
00	0
50	25
80	33
100	50
140	75
200	Various

Bit Versus Codeword

The ECC200 Data Matrix uses a byte-oriented (one Codeword) error detection and correction algorithm. The non-ECC200 uses a bit-oriented algorithm. Each ECC200 Data Matrix consists of multiple codewords. Each codeword contains 8 bits (also called elements or cells), as shown in Figure D–6.

FIGURE D–6. Shows 8 Bits as 1 Codeword



For ECC200, if any cell within the codeword is incorrectly identified, the codeword is subsequently affected by that bit and thus requires error correction. Since it is the codeword that matters in the error correction algorithm, one single bit can cause the same amount of damage as the eight wrong bits in the same codeword. For example, if one bit within the codeword is in error, the entire codeword is in error. Likewise, if multiple bits within the same codeword (e.g., 2-8 bits) are in error, this still counts as **one** codeword in error. The HawkEye™ 1500 is designed to cover both ECC200 and non-ECC200 Data Matrices.

Data Matrix Encoding Schemes

This section describes ECC 000-140 and ECC 200.

ECC 000-140

The data is encoded using one of six encoding schemes (format strings). The encoding scheme is fixed for the entire symbol. As a result, the selection of the most appropriate encoding scheme allows you to compress the symbol to its smallest form without wasting space on characters not used. Table D–2 lists the six encoding schemes and their outputs.

TABLE D–2. Encoding Schemes — Format Strings

Encoding Scheme	Format Outputs and Description	Bits per Data Character
Format 1	500 numeric characters (0-9) with spaces	3.5
Format 2	500 uppercase alphabetic characters with spaces and punctuation	4.8
Format 3	500 uppercase alphanumeric characters with spaces, commas, periods, slashes, and minuses	5.25
Format 4	500 uppercase alphanumeric characters with spaces	5.5
Format 5	500 characters with full 7-bit ASCII keyboard	7
Format 6	500 characters with full 8-bit ISO for international languages	8

ECC 200

The data may be encoded using any combination of six encoding schemes listed in Table D–3.

TABLE D–3. Encoding Schemes — File Formats

Encoding Scheme	Characters	Bits per Data Character
ASCII	Double-digit numeric ASCII values 0-127 Extended ASCII values 128-255	4 8 16
C40	Primary upper-case alphanumeric	5.33
Text	Primary lower-case alphanumeric	5.33
X12	ANSI X12 EDI data set	5.33
EDIFACT	ASCII values 32-94	6
Base 256	All byte values 0-255	8

Table D–4 lists ECC symbol attributes.

TABLE D–4. ECC 200 Symbol Attributes

Symbol Size:[1]		Data Region:		Inter-leaved Blocks	Data Capacity:			Error Correction Overhead %	Max. Correctable Percentage %
Row	Col.	Size	No.		Num. Cap	Alphanum. Cap	Byte Cap		
10	10	8x8	1	1	6	3	1	62.5	25
12	12	10x10	1	1	10	6	3	58.3	25
14	14	12x12	1	1	16	10	6	55.6	28/39**
16	16	14x14	1	1	24	16	10	50.0	25/38
18	18	16x16	1	1	36	25	16	43.8	22/34
20	20	18x18	1	1	44	31	20	45.0	23/38
22	22	20x20	1	1	60	43	28	40.0	20/34
24	24	22x22	1	1	72	52	34	40.0	20/35
26	26	24x24	1	1	88	64	42	38.9	19/35
32	32	14x14	4	1	124	91	60	36.7	18/34
36	36	16x16	4	1	172	127	84	32.8	16/30
40	40	18x18	4	1	228	169	112	29.6	15/28
44	44	20x20	4	1	288	214	142	28.0	14/27

TABLE D–4. ECC 200 Symbol Attributes (Continued)

48	48	22x22	4	1	348	259	172	28.1	14/27
52	52	24x24	4	2	408	304	202	29.2	15/27
64	64	14x14	16	2	560	418	277	28.6	14/27
72	72	16x16	16	4	736	550	365	28.1	14/26
80	80	18x18	16	4	912	682	453	29.6	15/28
88	88	20x20	16	4	1152	862	573	28.0	14/27
96	96	22x22	16	4	1392	1042	693	28.1	14/27
104	104	24x24	16	6	1632	1222	813	29.2	15/28
120	120	18x18	36	6	2100	1573	1047	28.0	14/27
132	132	20x20	36	8	2608	1954	1301	27.6	14/26
144	144	22x22	36	8*	3116	2335	1555	28.5	14/27
				2*					
Rectangular Symbols:									
8	18	6x16	1	1	10	6	3	58.3	25
8	32	6x14	2	1	20	13	8	52.4	24
12	26	10x24	1	1	32	22	14	46.7	23/37
12	36	10x16	2	1	44	31	20	45.0	23/38
16	36	14x16	2	1	64	46	30	42.9	21/38
16	48	14x22	2	1	98	72	47	36.4	18/32

Note 1: Symbol size does not include a quiet zone.
Note 2: The “Data Capacity” value for alphanumeric mode can be larger when some numeric numbers are together and smaller when the upper and lowercase alphabet are mixed.
Note*: In the largest symbol (144x144), the first eight Reed-Solomon blocks are 218 codewords long encoding 156 data codewords. The last two blocks encode 217 codewords (155 data codewords). All the blocks have 62 error correction codewords.
Note**:The maximum correctable percentage can range between 28% and 39%.

Data Matrix Specification Details

This section provides excerpts from the AIM and ISO Specification.

For new applications, ECC 200 is recommended. ECC 000-140 should only be used in closed applications where a single party controls both the production and reading of the symbols and is responsible for overall system performance.

The characteristics of the code consist of:

1. Encodable character set:

   a. Values 0-127 in accordance with ANSI X3.4, i.e., all 128 ASCII characters (equivalent to the US national version of ISO 646).

   b. Values 128-255 in accordance with ISO 8859-1; Latin Alphabet No. 1. These are referred to as extended ASCII.

2. Representation of data — A dark module is a binary 1 and a light module is a binary 0. In a reversed image (reverse polarity), a light module is a binary 1 and a dark module is a binary 0.

3. Symbol size in modules (not including the quiet zone):

   ECC 000-140
   9 by 9
   to 49 by 49
   Odd only

   ECC 200
   10 by 10
   to 144 by 144
   Even only

   Note: For more information, see Table D–4, “ECC 200 Symbol Attributes,” on page D-8.

4. Data characters per symbol (refers to maximum symbol size in ECC 200):

Alphanumeric data:	up to 2335 characters
8-bit byte data:	1555 characters
Numeric data:	3116 digits

5. Selectable error correction:

   ECC 000-140 — Four levels of convolutional error correction, plus the option to apply only error detection.
   ECC 200 — Reed-Solomon error correction.

6. Code type: Matrix

7. Orientation independence: Yes

Summary of Additional Features

The following summarizes additional Data Matrix features, including whether it is inherent to the Data Matrix, or optional:

- Reflectance reversal: (Inherent) Symbols are intended to be read when marked such that the image is either dark on light or light on dark.
- Extended Channel Interpretations: (ECC 200 only, optional). This mechanism enables characters from other character sets (e.g., Arabic, Cyrillic, Greek, Hebrew) and other data interpretations or industry-specific requirements to be represented.
- Rectangular symbols: (ECC 200 only, optional). Six symbol formats are specified in a rectangular form.
- Structured append: (ECC 200 only, optional). This allows files of data to be represented in up to 16 Data Matrix symbols. The original data can be correctly reconstructed regardless of the order in which the symbols are scanned.

Symbol Structure

Each Data Matrix symbol consists of data regions, which contain nominally square modules, set out in a regular array. In larger ECC 200 symbols, data regions are separated by alignment patterns. The data region is surrounded by a finder pattern, and as a result is surrounded on all four sides by a quiet zone border.

The Finder Pattern is a perimeter to the data region and is one module wide. Two adjacent sides (left and lower sides), forming the L-shaped boundary, are solid dark lines. These sides primarily determine physical size, orientation and symbol distortion. The two opposite sides are made up of alternating dark and light modules. Primarily, these define the cell structure of the symbol, but can also assist in determining physical size and distortion.

With regard to Symbol Sizes and Capacities:

- ECC 000-140 symbols consist of an odd number of rows and columns. Symbols are square with sizes from 9x9 to 49x49 (modules), not including quiet zones. These symbols can be recognized by the upper-right corner module being dark. For an image of reversed polarity, it would be light. Complete attributes for ECC 000-140 symbols are provided in the AIM and ISO Specification.

- ECC 200 symbols consist of an even number of rows and columns. Some symbols may be square with sizes from 10x10 to 144x144 not including the quiet zone. Other symbols may be rectangular with sizes from 8x8 to 16x48 not including quiet zone. All ECC 200 symbols can be recognized by the upper right corner being light dark. For an image of reversed polarity, it will be dark. Complete attributes of ECC 200 symbols are provided in the AIM and ISO.

Creating a Data Matrix

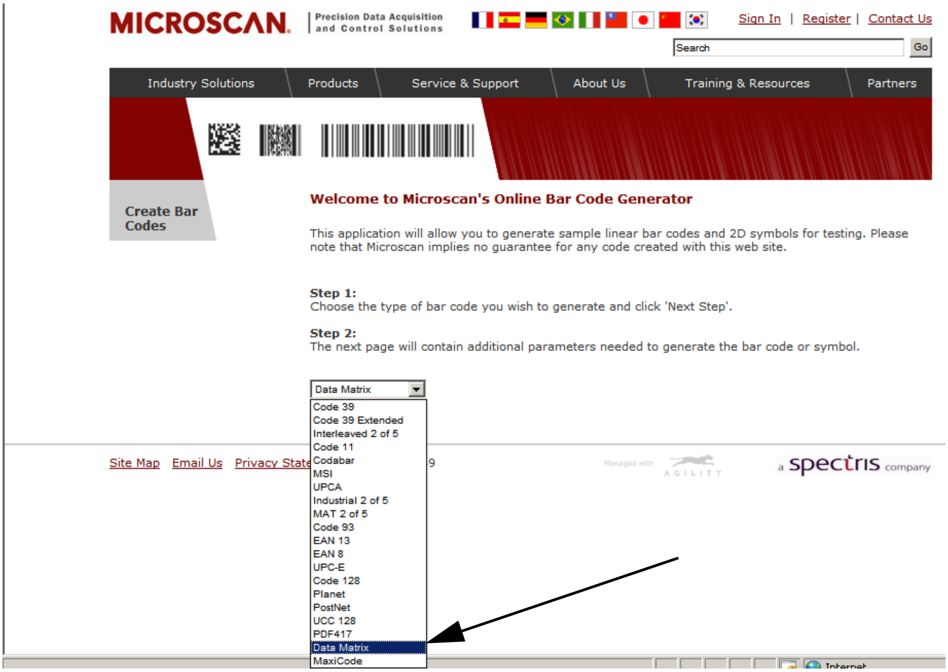
For demonstration purposes, you can create your own Data Matrix from the Microscan website:

1. From your web browser, connect to:

   http://www.microscan.com/CreateBarCodes.aspx

   You will see a screen similar to the one displayed in Figure D–7.

FIGURE D–7. Creating a Data Matrix



2. From the pull-down menu under Step 2, select Data Matrix.

3. Click Next Step.

   You will see a screen similar to the one displayed in Figure D–8.

FIGURE D–8. Creating a Data Matrix

4. Enter the data to encode.

5. Click Generate Bar Code.

For more information about label printing software packages, direct part marking technologies that apply Data Matrix marks on actual parts, and/or other solutions to your specific Data Matrix encoding/marking needs, please contact your Microscan distribution partner, account manager, or applications engineer.

Reader Programming Data Matrix

By placing a Reader Programming Data Matrix under the field of view, and depressing the QuicSet button once, the code will be read once, then the QuicSet® mode will automatically exit and return to the operational mode, with the changes applied by your Reader Programming Data Matrix.

Symbol Samples

2-D Symbols

Data Matrix (Data “123456789”)

ECC000 ECC050 ECC080 ECC100 ECC140 ECC200

PDF417 (Data “PDF417 sample”)

1-D Symbols

Code 128 (Data “This is Code 128”)

1234567890

Code 93 (Data “1234 CODE 93”)

1234567890

Code 39 (Data “ABCD CODE 39”)

1234567890

Interleaved 2 of 5 (Data “25251234567890”)

1234567890

Codabar (Data “1234567890”)

A1234567890B

EAN 13 (Data=“9876543210999”

9 876543 210999

EAN 8 (Data “76543210”)

1234 5670

UPC A (Data "98765432109")

9 87654 32109 8

5-Digit Postnet with Check Character (Data "020215")

SEMI BC412 with Both Start/Stop & Checksum

Data "AQQ45670"

Pharmacode (Data "399")

E

APPENDIX E Specifications

This appendix contains specifications and dimensions for the HawkEye™ 1500 camera, and for the mounting block and adapter.

TABLE E–1. Specifications

	HawkEye™ 1510	HawkEye™ 1515	HawkEye™ 1525
Height	44.45mm (1.750”)	44.45mm (1.750”)	59.94mm (2.360”)
Width	57.15mm (2.250”)	57.15mm (2.250”)	72.14mm (2.840”)
Depth	93.98mm (3.700”)	111.76mm (4.400”)	111.76mm (4.400”)
Weight	0.25 Kg (0.55 lbs)	0.4 Kg (0.9 lbs)	0.4 Kg (0.9 lbs)
Power	24 volts @ 350ma Typical (Optional Power Supply P/N: HEPS-1500)		
Minimum Contrast	20% @ 630nm		
Lighting	Optional 3.5” Light Port cable	Integrated high output bright-field LED with strobe operation	Integrated high output dark-field LED with strobe operation
Working Distance	Dependent on lens selection	Dependent on model selection	Dependent on model selection
Min. Element Size	Dependent on lens selection	Dependent on model selection	Dependent on model selection
FOV	Dependent on lens selection	Dependent on model selection	Dependent on model selection

TABLE E–1. Specifications (Continued)

Decode Capability	2-D: Data Matrix, PDF417 1-D: Code 39, Codabar, Code 93, 12of5, UPC/EAN, UPC-E, UPC Supplementals, Postnet, Pharmacode, Code 128
Verification	1. ISO/IEC 16022:2000 (Data Matrix) 2. ISO/IEC 15415:2004 (Data Matrix) 3. ISO/IEC 15416:2000 (one dimensional barcodes) 4. AS9132 Rev. A, 2005 (Data Matrix, formerly IAQG) 5. AIM DPM-1-2006
Communications	Ethernet; RS232, Baud rates from 1200 bps to 115.2 Kbps
I/O	1 Opto-isolated input trigger 3 Opto-isolated outputs 4 TTL level I/O Optional TTL level strobe output
Speed	Up to 30 parts per second
Operating Temperature	32° F to 104° F (0° C to 40° C)
Storage Temperature	-4° F to 149° F (-20° C to 65° C)
Humidity	Up to 95%, non-condensing

HawkEye™ 1500 Dimensions

Note: Mechanical drawings in various formats can be found on the CD.

FIGURE E–1. **HawkEye™ 1515**

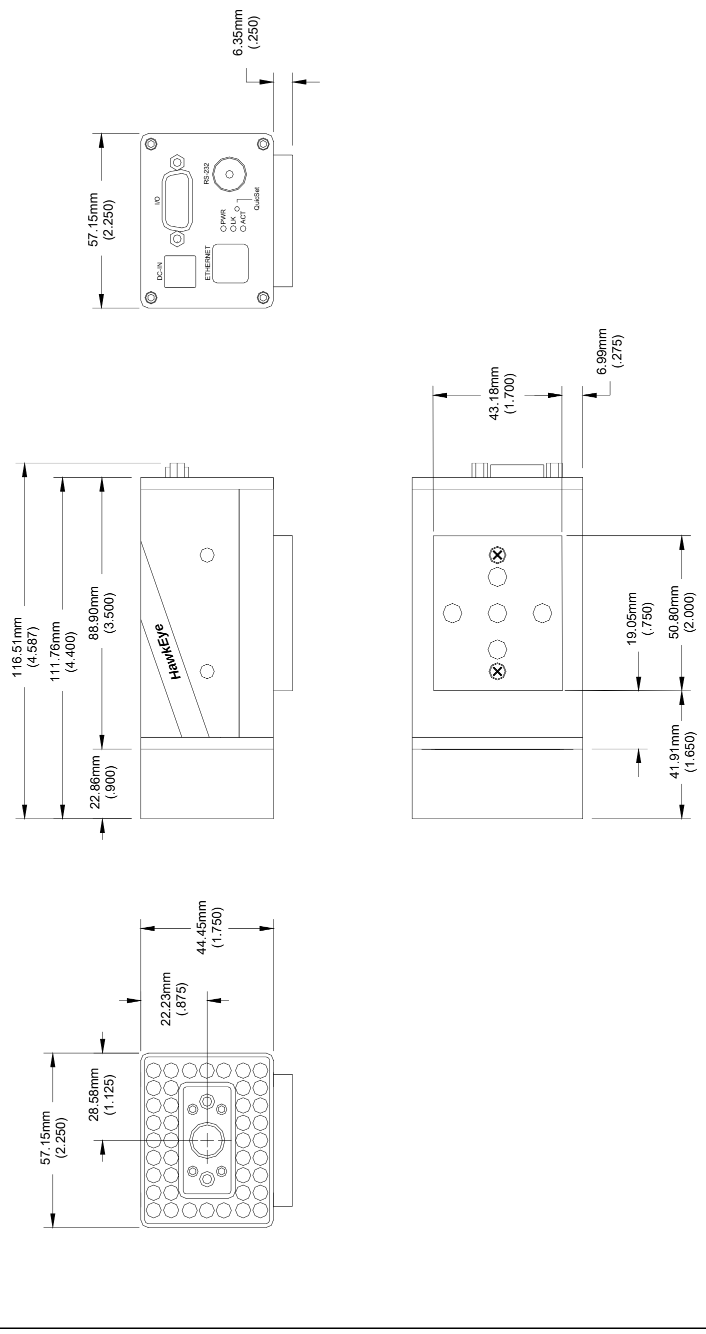


FIGURE E–2. HawkEye™ 1525

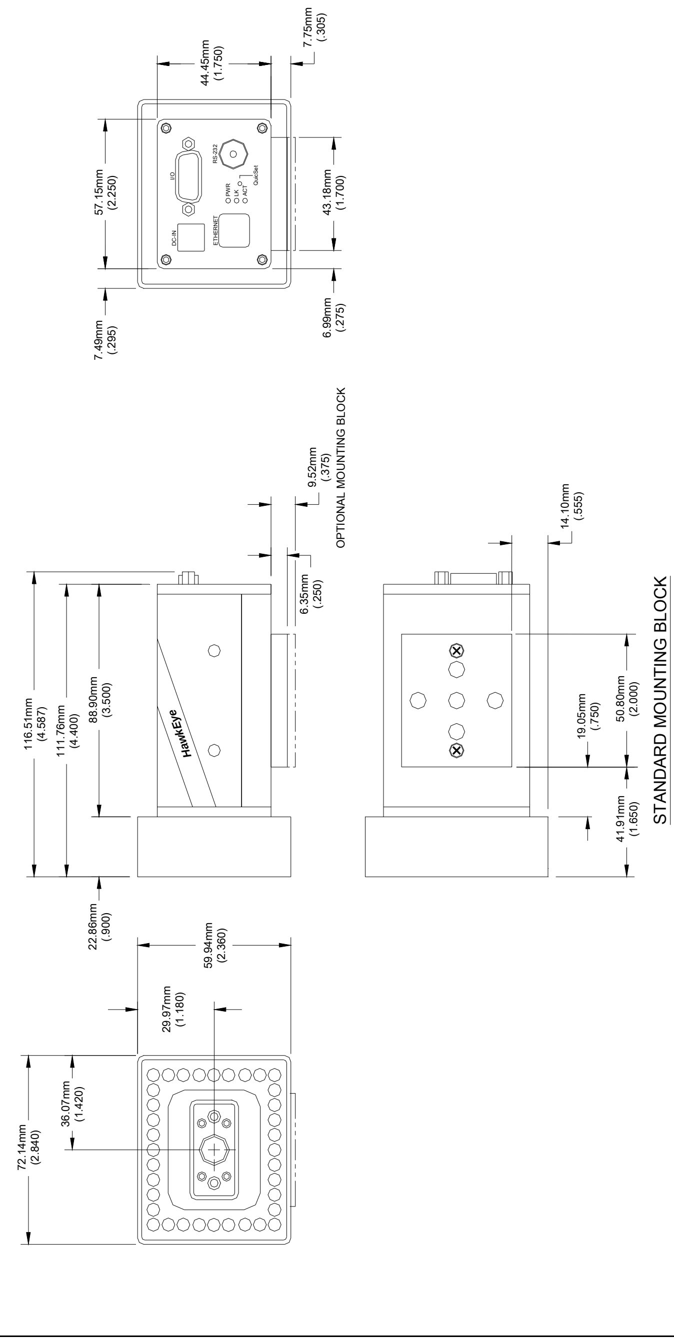
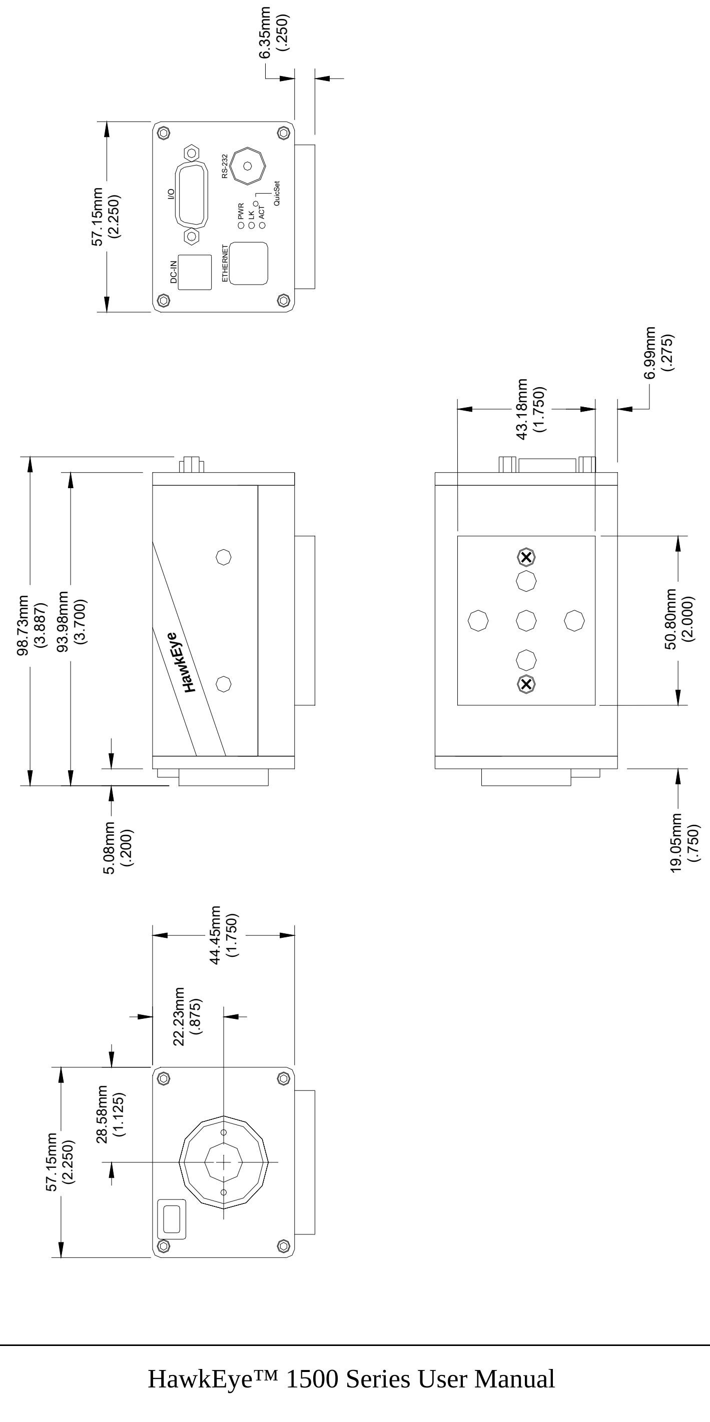


FIGURE E–3. **HawkEye™ 1510**



E

Specifications

FIGURE E–4. HEBMA-5— Standard Mounting Block (replaces HEBMA-1)

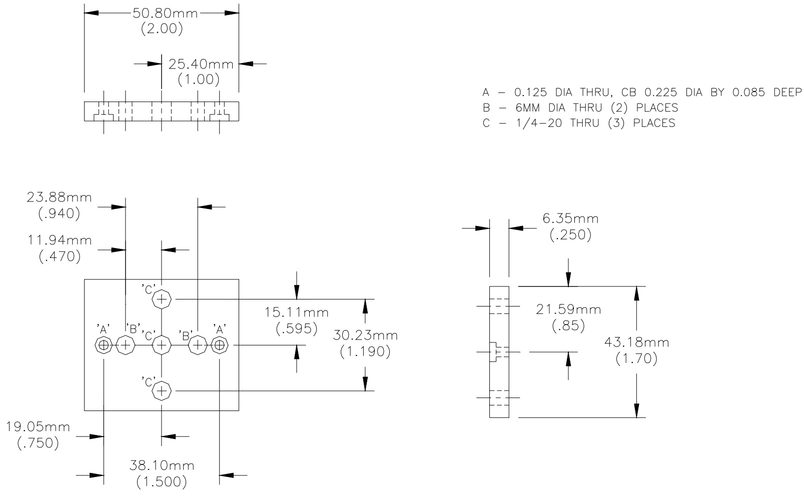


FIGURE E–5. HEBMA-2 — Optional Mounting Block for HawkEye™ 1525

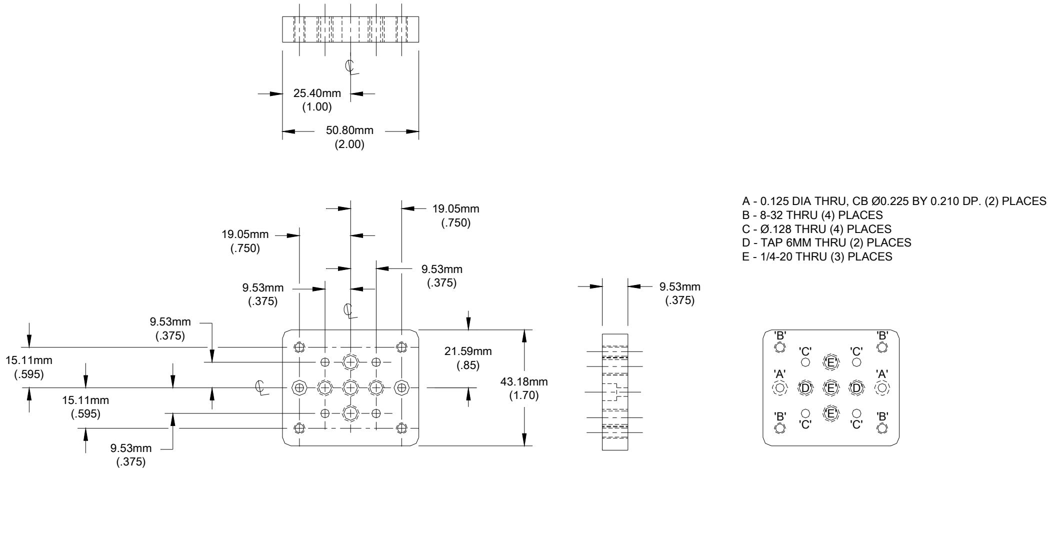


FIGURE E–6. HEBMA-3 — Mounting Block Adapter (HE 10 to HE 1510)

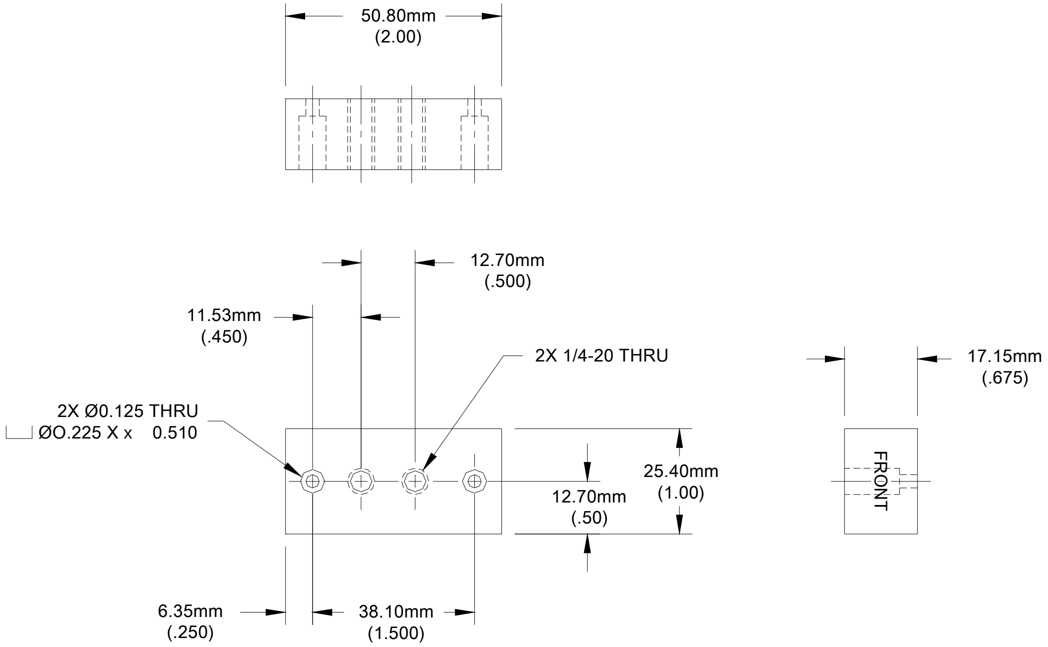


FIGURE E–7. **HEBMA-4 — Mounting Block Adapter (HE 15 to HE 1515)**

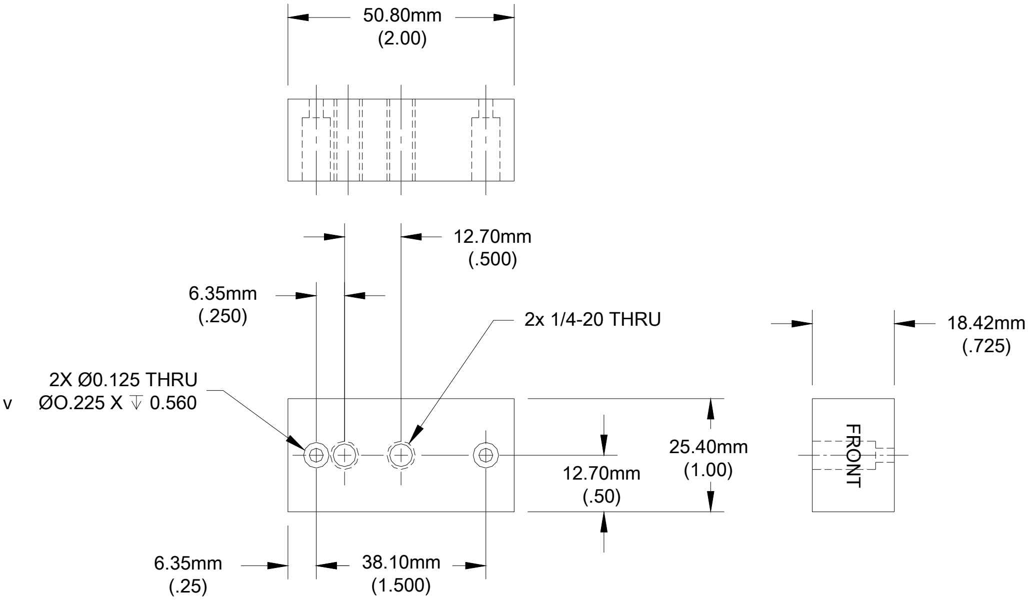


FIGURE E–8. **External I/O terminal Block Dimensions**

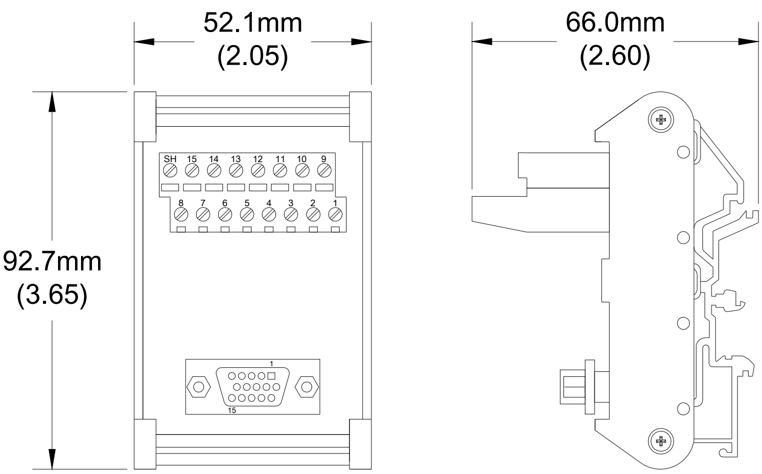
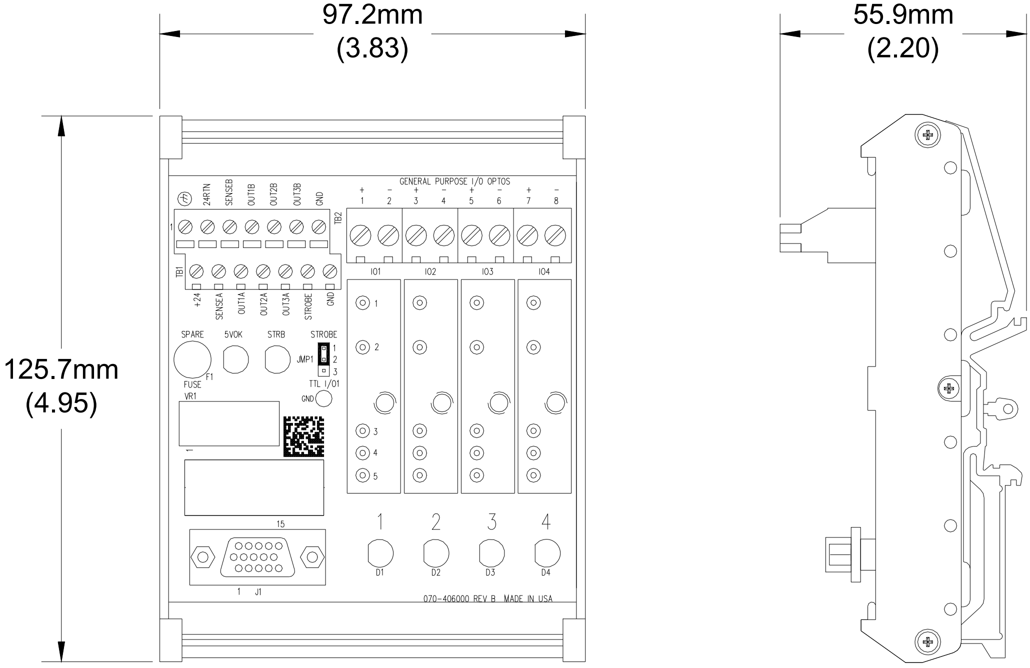


FIGURE E–9. I/O Expansion Module Dimensions



APPENDIX F

Custom Programming Using a Serial Connection

This appendix describes the special considerations you need to know when connecting via a serial port.

Here are some of the major considerations:

- When using a network connection, the device status is broadcast via UDP every 5 seconds. This means that the Device and Connection objects “know” the current status of the device at all times and can send events when the situation changes. This is not the case when using a serial port, and, if you want to emulate this behavior, the DeviceInfo must be manually updated using a timer.
- In a network connection, you can make a ReportConnection or ImageConnection and get events whenever new data is available. With a serial connection, you must manually initiate a report upload using the “REPORTREQ” command. This sample illustrates how to use a timer to “hook” images and/or reports.
- When a REPORTREQ has been issued, you must either wait for the result or cancel the report. Issuing a command with a REPORTREQ outstanding will automatically cancel the report.

The sample code presents a series of buttons at the top allowing connection to the various COM ports. The buttons are disabled if the corresponding port is not available. A tabstrip control is used to switch between three views: 1) Terminal,

2) Image, and 3) Report. The colored labels below the buttons show total cycles and passed and failed counters as obtained by querying the DeviceInfo.

When first started, only the DeviceInfo is hooked, and, therefore, the colored counters will be updated. By selecting various choices in the “Options” menu, you can hook Images or Results, as well as turn off the DeviceInfo hook. You can also select different image compression amounts. A progress bar appears at the bottom of the window during upload of Reports or Images.

Code Walkthrough

Project References:

- HawkEye Reader Object Type Library

Project Components:

- Microsoft Windows Common Controls 6.0
- HawkEye Reader ActiveX Library
- ReadRunner ActiveX Library

The code consists of a single form called frmMain. On this form, the following items are placed:

- The COM “buttons” are a control array of checkboxes whose Style property is set to Graphical so that they appear as buttons. This is an easy way of making buttons that stay depressed when you press them. In the sample, they are named btnConnect(index).
- Three labels called m_countTotal, m_countPassed, and m_countFailed. These will be updated using the Device Info.
- A TabStrip control named m_tabstrip, with the Placement property set to tabPlacementBottom.
- On top of the TabStrip are placed a Terminal control (m_terminal), an MiImageView control (m_image) and a ReportGrid (m_report)
- A timer control called SerialHook
- A progress bar called m_progress

The two main objects used are declared at the top of frmMain. A MiCoordinator (m_coord) is used to make connection easier by using its DeviceFocusSet method. A ReportConnection object is used to represent the connection to the device. When DeviceFocusSet is called, the ReportConnection will automatically be connected (or disconnected) and fire the OnConnectionReset event.

The sample code is not very long and should be easy to follow. It uses the SerialHook timer to do most of the work. When a COM button is pressed (btnConnect_Click(Index)), m_coord.DeviceFocusSet is called with the button caption as the device name. This will cause the m_connection_OnConnectionReset event to occur. If connected, the SerialHook timer is enabled.

The SerialHook_Timer event first gets the current Device from the MiCoordinator using the DeviceFocusGet method. This is so that the dev.RefreshInfo method can be used to query the camera for the DeviceInfo.

There are three mode switches used by the SerialHook: m_bHookDevInfo, m_bHookReports and m_bHookImages. If m_bHookDevInfo is TRUE, then dev.RefreshInfo is called to get the DeviceInfo, and the counts are written to the colored labels.

Hooking Images is really just a special case of Hooking Reports. If m_bHookReports is TRUE, then a “REPORTREQ” command string is built up using the currently selected image compression ratio. The final parameter is a “1” if m_bHookImages is TRUE. This command is then sent to the camera using m_conn.CommandSend()

Note that the variable m_bGettingReports is set to TRUE when a REPORTREQ is issued. This prevents issuing another request if the timer fires while a REPORTREQ is still outstanding. This is likely to happen if the application is triggered.

Please note the use of the progress events from m_connection. These are used to update the progress bar. It is also very important to handle the OnDataTransferAborted event: if any command is sent to the camera while a report is being uploaded, the transfer will be automatically aborted.

When used in this manner, the m_connection object can be used almost as if the connection were via a network connection. In particular, the m_connection_OnNewRecord event will be received with a MiCycleReport object that can be directly used to update the m_image and m_report views.

Complete Source Code

```
Option Explicit
Const VIEW_TERMINAL = 1
Const VIEW_IMAGE = 2
Const VIEW_REPORT = 3

Dim WithEvents m_coord As MiCoordinator
Dim WithEvents m_connection As ReportConnection

Dim m_bGettingReport As Boolean
Dim m_bInUpdateButtons As Boolean
Dim m_bHookReports As Boolean
Dim m_bHookImages As Boolean
Dim m_bHookDevInfo As Boolean
Dim m_nCompression As Long


Private Sub Form_Load()
    Set m_coord = New MiCoordinator
    Set m_connection = New ReportConnection

    m_bInUpdateButtons = False
    m_bGettingReport = False
    m_nCompression = 4

    ' set the SerialHook timer interval
    SerialHook.Interval = 2000
    SerialHook.Enabled = False

    ' init the menus
    Compression(m_nCompression).Checked = True
    m_bHookDevInfo = True
    HookDevInfo.Checked = m_bHookDevInfo
    m_bHookImages = False
    HookImages.Checked = m_bHookImages
    m_bHookReports = False
    HookReports.Checked = m_bHookReports

    ' init the tabstrip
    m_tabstrip.Tabs.Clear
    m_tabstrip.Tabs.Add , "terminal", "Terminal"
    m_tabstrip.Tabs.Add , "image", "Image"
    m_tabstrip.Tabs.Add , "report", "Report"
    ShowView VIEW_TERMINAL
```

```
    ' disable buttons if port is not available
    Dim i As Long
    For i = 1 To 4
        btnConnect(i).Enabled = CheckSerialPort("COM" & i & ":")
    Next i

    Disconnect
End Sub

Private Sub Form_Unload(Cancel As Integer)
    Disconnect
    Set m_coord = Nothing
End Sub

Private Function CheckSerialPort(port As String) As Boolean
    ' this function uses MiAsciiConnection to check if
    ' the COM port can be opened at all - will throw exception
    ' on conect if in use by another application
    On Error GoTo err:
    Dim serial As New MiASCIIConnection
    serial.Connect port
    If serial.Connected Then

    End If
    serial.Disconnect
    CheckSerialPort = True
    Set serial = Nothing
    Exit Function
err:
    Set serial = Nothing
    CheckSerialPort = False
End Function

Private Sub Connect(targ As String)
    m_coord.DeviceFocusSet targ
End Sub

Private Sub m_connection_OnConnectionReset()
    Dim i As Long
    m_bGettingReport = False
    If m_connection.IsConnected Then
        SerialHook.Enabled = True
    Else
        SerialHook.Enabled = False
    End If
```

```
End Sub

Private Sub UpdateButtons()
    Dim i As Long
    m_bInUpdateButtons = True
    If m_connection.IsConnected Then
        For i = 1 To btnConnect.Count
            If m_connection.Device.Name <> btnConnect(i).Caption Then
                btnConnect(i).Value = vbUnchecked
            Else
                btnConnect(i).Value = vbChecked
            End If
        Next i
    Else
        For i = 1 To btnConnect.Count
            btnConnect(i).Value = vbUnchecked
        Next i
    End If
    m_bInUpdateButtons = False
End Sub

Private Sub Disconnect()
    m_coord.DeviceFocusSet ""
End Sub

Private Sub btnConnect_Click(Index As Integer)
    If m_bInUpdateButtons Then Exit Sub

    If m_connection.IsConnected Then
        Dim i As Long
        For i = 1 To btnConnect.Count
            If btnConnect(i).Caption = m_connection.Device.Name Then
                m_bInUpdateButtons = True
                btnConnect(i).Value = vbUnchecked
                btnConnect(i).Refresh
                m_bInUpdateButtons = False
            End If
        Next i
        Disconnect
    End If
    If btnConnect(Index).Value = vbChecked Then
        Connect btnConnect(Index).Caption
    End If
    UpdateButtons
End Sub
```

```
Private Sub ShowView(v As Long)
    Select Case v
        Case VIEW_TERMINAL
            m_terminal.Visible = True
            m_image.Visible = False
            m_report.Visible = False
        Case VIEW_IMAGE
            m_terminal.Visible = False
            m_image.Visible = True
            m_report.Visible = False
        Case VIEW_REPORT
            m_terminal.Visible = False
            m_image.Visible = False
            m_report.Visible = True
    End Select

End Sub

Private Sub Form_Resize()
    On Error Resume Next
    Dim y As Long
    y = 0
    y = y + btnConnect(1).Height

    Dim w As Long
    w = ScaleWidth / 3
    m_countTotal.Move 0, y, w
    m_countPassed.Move w, y, w
    m_countFailed.Move 2 * w, y, w
    y = y + m_countTotal.Height

    m_progress.Move 0, ScaleHeight - m_progress.Height, ScaleWidth
    m_tabstrip.Move 0, y, ScaleWidth, ScaleHeight - y - m_progress.Height

    With m_tabstrip
        m_terminal.Move .ClientLeft, .ClientTop, .ClientWidth,
.ClientHeight
        m_image.Move .ClientLeft, .ClientTop, .ClientWidth, .ClientHeight
        m_report.Move .ClientLeft, .ClientTop, .ClientWidth, .ClientHeight
    End With
End Sub

Private Sub HookDevInfo_Click()
    m_bHookDevInfo = Not m_bHookDevInfo
    HookDevInfo.Checked = m_bHookDevInfo
End Sub
```

```
Private Sub HookImages_Click()
    m_bHookImages = Not m_bHookImages
    If m_bHookImages Then m_bHookReports = True
    HookReports.Checked = m_bHookReports
    HookImages.Checked = m_bHookImages
End Sub

Private Sub HookReports_Click()
    m_bHookReports = Not m_bHookReports
    If Not m_bHookReports Then m_bHookImages = False
    HookImages.Checked = m_bHookImages
    HookReports.Checked = m_bHookReports
End Sub

Private Sub Compression_Click(Index As Integer)
    m_nCompression = Index
    Dim i As Long
    For i = Compression.LBound To Compression.UBound
        Compression(i).Checked = IIf(m_nCompression = i, vbChecked,
vbUnchecked)
    Next i
End Sub


Private Sub m_connection_OnDataTransferAborted()
    m_progress.Value = 0
    m_progress.Visible = False
    m_bGettingReport = False
End Sub

Private Sub m_connection_OnDataTransferPercentComplete(ByVal nPercent As
Integer)
    m_progress.Value = nPercent
End Sub

Private Sub m_connection_OnDataTransferStarted()
    m_progress.Visible = True
    m_progress.Min = 0
    m_progress.Max = 100
    m_progress.Value = 0
End Sub

Private Sub m_connection_OnNewRecord(ByVal objReport As
MIOBJLib.IMiCycleReport)
    If m_bGettingReport Then
```

```
        m_image.NewRecord objReport
        m_report.NewRecord objReport
        m_bGettingReport = False
        m_progress.Value = 100
        m_progress.Visible = False
    End If
End Sub

Private Sub m_tabstrip_Click()
    Select Case m_tabstrip.SelectedItem.Key
        Case "terminal"
            ShowView VIEW_TERMINAL
        Case "image"
            ShowView VIEW_IMAGE
        Case "report"
            ShowView VIEW_REPORT
    End Select
End Sub

Private Sub SerialHook_Timer()
    If Not m_connection.IsConnected Then Exit Sub

    If Not m_bGettingReport Then
        If m_bHookDevInfo Then
            ' update the device info manually
            Dim dev As MiDevice
            Set dev = m_coord.DeviceFocusGet

            dev.RefreshInfo
            m_countTotal.Caption = dev.CountCycle
            m_countPassed.Caption = dev.CountPassed
            m_countFailed.Caption = dev.CountCycle - dev.CountPassed
        Else
            m_countTotal.Caption = "???"
            m_countPassed.Caption = "???"
            m_countFailed.Caption = "???"
        End If

        If m_bHookReports Then
            ' create a REPORTREQ command string
            m_bGettingReport = True

            Dim cmd As String
            cmd = "REPORTREQ "

            Select Case m_nCompression
```

```
            Case 1
                cmd = cmd & " 255"
            Case 2
                cmd = cmd & " 1"
            Case 3
                cmd = cmd & " 4"
            Case 4
                cmd = cmd & " 26"
            Case 5
                cmd = cmd & " 27"
            Case Else
                MsgBox "Invalid Compression Ratio"
            End Select

            If m_bHookImages Then
                cmd = cmd & " 1"
            Else
                cmd = cmd & " 0"
            End If

            ' send the REPORTREQ
            m_connection.CommandSend cmd
        End If

    End If
End Sub
```

Index