MICROSCAN

VS-1 Smart Camera Guide

v4.1.3, April 2012

83-100045-02 Rev B

ISO 9001 Certified
Issued by TüV USA

Disclaimer

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

Latest Manual Version

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

Technical Support

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

Warranty

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

Microscan Systems, Inc.

United States Corporate Headquarters
+1.425.226.5700 / 800.762.1149

United States Northeast Technology Center
+1.603.598.8400 / 800.468.9503

European Headquarters
+31.172.423360

Asia Pacific Headquarters
+65.6846.1214

Contents

PREFACE

Welcome!

Purpose of This Manual

This manual contains detailed information about the VS-1 Smart Camera family.

Manual Conventions

The following typographical conventions are used throughout this manual.

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

1

CHAPTER 1 Introduction

FIGURE 1–1. VS-1 Smart Camera

Product Summary

The VS-1 Smart Camera, one of our Visionscape® family of networked Smart Cameras, combines a rugged IP67 smart camera form-factor with the broad applicability, flexibility, and proven vision toolkit of Visionscape®. Designed for use in a broad range of vision applications, the VS-1 Smart Camera provides a cost effective, easily deployed solution for manufacturers to monitor quality, control processes, or identify and trace parts on their production lines.

VS-1 Smart Cameras are configured as flexible, general-purpose smart cameras with C-mount optics and separate lighting. The VS-1 Smart Camera comes standard with built-in digital I/O, serial communications, and Ethernet networking. All vision processing is done on-board using a high performance, embedded CPU. A real-time, multitasking operating system ensures deterministic performance and facilitates integration in high-speed manufacturing lines.

The VS-1 Smart Camera offers an extensive array of built-in vision processing tools, including Data Matrix and bar code reading, optical character recognition (OCR), image processing, image analysis and feature extraction, flaw detection, object location, calibrated dimensional measurements, and various custom processing options. Developed and perfected on prior generations of our machine vision systems, these tools have already been successfully applied in thousands of production installations worldwide.

Setup of a new vision application to run on the camera is done on a host PC on the same network using the same powerful graphical application environment as the rest of the Visionscape® line. Our patented Visionscape® step program architecture allows running the same vision application program on any VS-1 Smart Camera, leveraging the end-user's investment in application development and training.

Scaleability and compatibility with the rest of the Visionscape® family set the VS-1 Smart Camera apart from other smart cameras in the market today. The same point-and-click environment can be used to configure applications deployed on smart cameras and GigE cameras.

The Smart Camera family supports 16 MB of Non-Volatile memory for Kernel and Job saving (3 MB maximum AVP size) and 64 MB of RAM for operation. The camera sensor (depending on model) supports acquisitions of:

- 1024x768 (XGA) images at a maximum rate of 30 frames/sec
- 648x494 (VGA) images at a maximum rate of 60 frames/sec

It also supports 8 opto idolated digital IO lines in addition to a dedicated light connector for controlling and powering an external light (VS-1 Smart Camera HE1610T configuration only). Partial scan at higher frame rates is supported.

There is full support for:

- Acquisition modes, including:
  - Exposure (64 usec to 58.8 msec)
  - Analog gain (0.1 to 63.0)
  - Offset (-255 to +255)
- VS-1 Smart Camera I/O:
  - 1 trigger
  - 1 strobe output
  - 3 opto isolated outputs
  - 4 additional opto isolated general purpose I/O fully assignable in the AVP

Features and Benefits

- Compact, all-in-one smart camera configuration for ease of integration
- Built-in digital I/O, serial communications and Ethernet networking for open connectivity to other equipment
- Flexibility in lighting and optics selection for broad applicability
- Comprehensive, fully-featured vision processing toolkit
- Graphical environment for fast application development
- No conventional programming required for setup of complex vision applications

Applications

- Part presence/absence
- Assembly verification
- Inspection
- Gauging
- Part location/orientation detection
- Alignment/guidance
- Automatic ID (Data Matrix, bar code, OCR)

Package Contents

Before you install Visionscape® software and mount, wire, and connect your VS-1 Smart Camera, please take a moment to confirm that the package contains the following items:

- VS-1 Smart Camera — Your package contains one of four available Smart Camera models (see Table 1–1).
- CD — Your CD contains Visionscape® software and all documentation.

VS-1 Smart Camera Models

Table 1–1 lists and describes the VS-1 Smart Camera models, including acquisition modes and resolutions.

TABLE 1–1. VS-1 Smart Camera Models and Resolutions

Number	Model	Resolution
GMV-0HT16-0CM1G	VS-1 Smart Camera (Standard Resolution)	FullScan: 648x494 PartialScan:Half: 648x227 Quarter: 648x81 Binning:324x242
GMV-1HT16-0CM1G	VS-1 Smart Camera (High Resolution)	FullScan: 1024x768 PartialScan:Half: 1024x344 Quarter: 1024x137 Binning:512x370
GMV-0HF16-0CM1G	VS-1 Smart Camera (Standard Resolution w/IntelliFind)	FullScan: 648x494 PartialScan: Half: 648x227 Quarter: 648x81 Binning:324x242
GMV-1HF16-0CM1G	VS-1 Smart Camera (High Resolution w/IntelliFind)	FullScan: 1024x768 PartialScan: Half: 1024x344 Quarter: 1024x137 Binning:512x370

The VS-1 Smart Camera supports up to a maximum of four inspections.

Effective Frame Per Second and Pipeline Operation Formulas

Use the following formula to calculate effective FrameRatePerSecond in a Visionscape® Job:

$$EFPS = \frac{1000.0}{(\text{IT IntegrationTime} + \text{VFT VendorFrameTime} + \text{VSO VS Overhead}) \text{ in msec}}$$

Where VS Overhead is typically:

Interrupt Latency + Framework Overhead = 0.5 msec max

Or

$$EFPS = \frac{1000.0}{(IT + VFT + 0.5)}$$

In Table 1–2, EFPS is given in PowerStrobe with 1msec Integration time and Job in full pipeline, meaning that the Job tools + idle processing time, i.e., less than (IT + VFT + 0.5) above.

Full Pipeline = (AVT AVP Processing Time + IDT Inspection Idle Time + LossLess Connection Overhead) < (IT + VFT + VSO) idle time is defaulted to 3% of the Inspection Time as defined in the VisionSystemStep property page.

Table 1–2 lists VS-1 Smart Camera models, modes, and frames per second.

TABLE 1–2. **VS-1 Smart Camera Modes, Ranges, and FPS**

Model	Mode	Exposure Range (usec)	EFPS (Power Strobe)	FPS	Notes
HE1610TS HE1610TIS (Standard)	Full	[64 - 50000]	55	60	648x494 Progressive Scan Camera Interface
	1/2	[64 - 50000]	102	120	648x227 Progressive Scan Camera Interface
	1/4	[64 - 50000]	176	240	648x81 Progressive Scan Camera Interface
	Pixel Add	[64 - 50000]	102	120	324x242 Progressive Scan Camera Interface
HE1610TH HE1610TIH (High Res)	Full	[90 - 50000]	28.7	30	1024x768 Progressive Scan Camera Interface
	1/2	[90 - 50000]	55	60	1024x344 Progressive Scan Camera Interface
	1/4	[90 - 50000]	102	120	1024x137 Progressive Scan Camera Interface
	Pixel Add	[90 - 50000]	55	60	512x370 Progressive Scan Camera Interface

Triggering Rules for Single Channel Devices

The VS-1 Smart Camera has a single acquisition channel. When a Job is constructed, only one Acquire can run at a time. Table 1–3 summarizes the recommended cases. Note that Visionscape® enforces this condition by honoring the first Acquire requested and generating an overrun on any other Acquire that are requested to run from the external controlling device (usually, an external trigger or a PLC).

TABLE 1–3. Triggering Rules

Job Structure	Snapshot Triggers	Behavior and Comments
1 Inspection/ Multiple Snapshots	1st triggered only or No triggers	Function without overruns, no further action from external controlling device.
1 Inspection/ Multiple Snapshots	All triggered	1st triggered externally, remaining self triggered from PicDone signal of previous Acquire.
N (max 4) inspections/ 1 Snapshot Each	All triggered/Each on a separate trigger	Works as expected in external controlling device makes sure that one Inspection is triggered at a time.

Additional Flash and System Memory

The Enhanced VS-1 Smart Camera contains twice the Flash (32MB) and twice the RAM (128MB) of the standard VS-1 Smart Camera. The firmware fully supports both models:

- The VS-1 Smart Camera (16M Flash / 64M RAM).
- The Enhanced VS-1 Smart Camera (32M Flash / 128M RAM).

  The Enhanced VS-1 Smart Camera can save an AVP of up to 16MB in size, allowing much larger AVPs to be developed for the device.

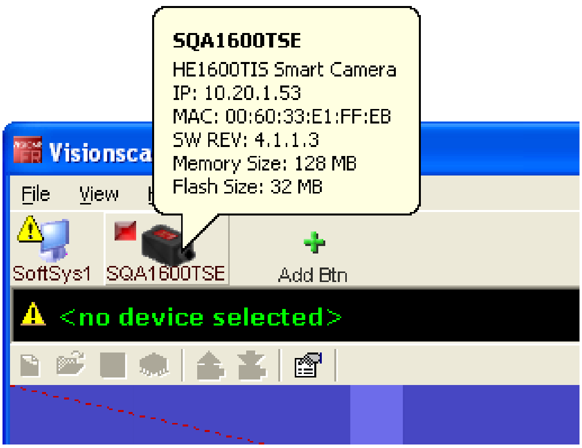
Identifying Which Smart Camera You Have

The label on the bottom of the Enhanced VS-1 Smart Camera has a part number that indicates which Smart Camera you have, and what memory is in the Smart Camera:

TABLE 1–4. How to Identify Which Smart Camera You Have

Part Number	What It Means
014-HE1610-1	VS-1 Smart Camera TS with Standard Memory
014-HE1610-2	VS-1 Smart Camera TIS with Standard Memory
014-HE1610-3	VS-1 Smart Camera TH with Standard Memory
014-HE1610-4	VS-1 Smart Camera TIH with Standard Memory
014-HE1610-5	VS-1 Smart Camera TS with Expanded Memory
014-HE1610-6	VS-1 Smart Camera TIS with Expanded Memory
014-HE1610-7	VS-1 Smart Camera TH with Expanded Memory
014-HE1610-8	VS-1 Smart Camera TIH with Expanded Memory

- In FrontRunner, the Smart Camera button tooltip lists the size of the RAM and Flash for the device:



- In the Network Viewer:
  - Enhanced VS-1 Smart Camera 16/64 — Colorized background color
  - Enhanced VS-1 Smart Camera 32/128 — No background color

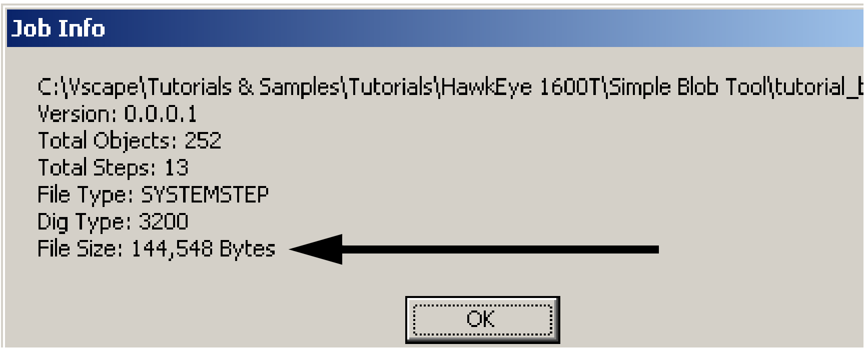
Jobs and Storage in Non-Volatile Memory

The maximum Non-Volatile Memory area for Jobs is 16MB for the Enhanced VS-1 Smart Camera (32/128), and 4.8MB for the Standard VS-1 Smart Camera (16/64). This includes the Job plus any support files if used in the AVP:

- IntelliFind models
- OCV/OCR Fonts
- Perl scripts
- Acquire Tool tiff or bmp image list (when programmed to capture from disk)

To determine the size of an AVP from FrontRunner, select File > Show Job Info... and navigate to the AVP file on the disk. FrontRunner displays a dialog that shows the File Size of the AVP and other statistics about the Job.

FIGURE 1–2. File Size of Job



Determine support file sizes by adding up the file sizes of the models, fonts, perl scripts, and tiff images used in the AVP (these are usually stored under \Vscape\Jobs Contours, Fonts subdirectories). Overall, the maximum size used (AVP + support files) cannot exceed 16MB for the enhanced VS-1 Smart Camera or 4.8MB for the standard VS-1 Smart Camera.

CHAPTER 2 System Components

This chapter contains information about system components, and information to help you connect the VS-1 Smart Camera. Specific information describes connectors, adapters, cables, pinouts, and signals.

Note: There are no user serviceable parts inside.

Basic Components

Table 2–1 lists the VS-1 Smart Camera hardware components.

TABLE 2–1. VS-1 Smart Camera Hardware Components

Number	Component	Description
Cameras		
GMV-0HT16-0CM1G	HE1610TS	VS-1 Smart Camera VGA
GMV-1HT16-0CM1G	HE1610TH	VS-1 Smart Camera XGA
GMV-0HF16-0CM1G	HE1610TIS	VS-1 Smart Camera VGA w/IntelliFind™
GMV-1HF16-0CM1G	HE1610TIH	VS-1 Smart Camera XGA w/IntelliFind™
Starter and Accessory Kits		
GMV-1HF16-0SK1G	HE1610TSK	VS-1 Smart Camera 1610T Starter Kit
Cables		

TABLE 2–1. VS-1 Smart Camera Hardware Components (Continued)

Number	Component	Description
98-HT00-0CE0	HETENET-100	Cable, Ethernet M12-4 to RJ45 Ethernet Cable Length = 10m
98-HT00-0CE2	HETENET-020	Cable, Ethernet M12-4 to RJ45 Ethernet Cable Length = 2m
98-HT00-0CP0	HETPC-100	Cable, Power and Primary I/O M12-8 to pigtail Length = 10m
98-HT00-0CS0	HETAC-100	Cable, Serial and Auxilary I/O M12-8 to pigtail Length = 10m
98-HT00-0CD2	HETSC-020	Cable, Serial and M12-8 to DB9 Length = 2m
98-HT00-0CL5	HETLC-050	Cable, Light M5-4 to pigtail Length = 5m
Adapters		
98-HT00-0LM1	010-026800	Adapter, Lighting for Doal and DF-100 NER Lights
98-HT00-0CM1	HETBMA-1	Adapter, Camera mount, Standard
98-HT00-0CM2	HETBMA-2	Adapter, Camera mount, Backward Compatible
98-0HT00-0TA0	HELTA-050	Adapter for Lens Protections Tubes (IP67) 98-0HT00-0TA1 (glass) 98-0HT00-0TA2 (no glass)
Lights		
98-HT00-0LD1	010-208700	Light, DOAL 50 V2, with HE1x00T Compatible connector
98-HT00-0LF1	010-609600	Light, DF-150, with HE1x00T Compatible connector
98-HT00-0LR1	010-609500	Light, R-60, with HE1x00T Compatible connector

VS-1 Smart Camera

TCP/IP Port Connectivity

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 and each one monitored by a special *heartbeat* connection on port 49079 (see Table 2–2).

TABLE 2–2. VS-1 Smart Camera TCP/IP Connectivity

Port Name	Protocol	Number	Note
File Transfer Port	FTP	21	Allows the Host to send and retrieve files from the VS-1 Smart Camera.
Telnet Port	Telnet	23	Console terminal to the Device, runs the vxWorks OS console target shell.
DHCP Client Port	DHCP	68	Supports the assignment of IP addresses from a DHCP server for the VS-1 Smart Camera.
Web Server	HTTP	80	Allow access to boot parameters when HyperTerminal over Serial and Bootloader menu is not convenient.
Routing Port	RIP	520	Receives and updates local routing tables from the Network.
IO Service Port	TCP	49049	Controls IO on the camera, i.e., Physical and Virtual IO and receives IO change notification events back to the client.
Pic and Live Acquire Port	TCP	49050	Takes pictures and goes to Live Video.
Camera RPC Port	TCP	49059	When in control, sends editing and runtime commands to the camera; for example, Start, Stop, Download, Flash, etc...
Connection Monitor Port	TCP	49079	This special connection is created automatically whenever any of the other connections above is made by a client (when using the vskit libraries or by FrontRunner). It monitors the connections and provides a timely mechanism to report connection drops to the client (within a few seconds).
Reports and Statistics Port	TCP	49200	Used by FrontRunner and vskit programming library for camera reports. Reports are defined by the AVP and can include any results with or without images. Connections can be programmed to be lossless, i.e., inline with the Inspection or lossy, i.e., at a particular rate per second.
PartQ Retrieval Port	TCP	49201	Retrieves the reports records stored inline in camera memory.

TABLE 2–2. VS-1 Smart Camera TCP/IP Connectivity (Continued)

Port Name	Protocol	Number	Note
Reports and Statistics Control Port	TCP	49202	Allow control over a Report Connection, in particular update rate and allows records to be added/removed from the connection.
Serial TCP Ports	TCP	49211 49212 49213 49214	Send Formatted Output Strings serially over TCP as programmed in the AVP by the Formatted Output Step
Camera Login/Command Port	UDP	49493	Gains control to the camera in order to edit and modify its network parameters. A network login to the camera is required to gain control with the VS-1 Smart Camera.
Camera Announce Port	UDP	49495	Broadcasts Smart Camera identity on the current subnet used by Network View in FrontRunner, provides general counters, camera name, IP, IP in control, camera status, and camera software version.

Serial Port Connectivity

FrontRunner always communicates over Ethernet with the VS-1 Smart Camera; however, the Serial port is fully supported by the Job in the form of allowing Serial triggers and Serial ASCII reporting output via the Acquire, Digital Input and Formatted Output Tools.

The serial string(s) are formatted using the *printf C syntax*. Special characters can be used and are summarized in Table 2–3.

TABLE 2–3. Special Characters

Sequence	Output Character
\a	Bell (alert)
\b	Backspace
\f	Formfeed
\n	New line
\r	Carriage return
\t	Horizontal tab
\v	Vertical tab
\'	Single quotation mark

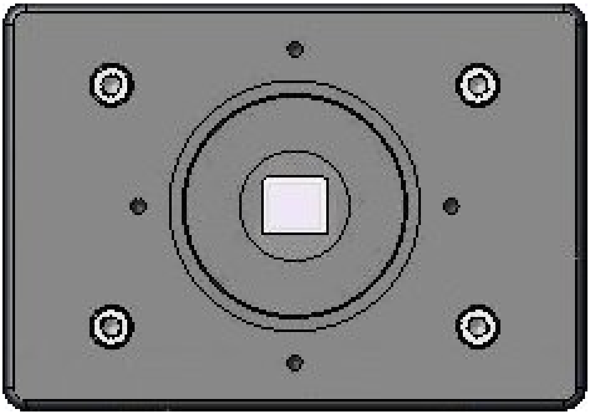
TABLE 2–3. Special Characters (Continued)

Sequence	Output Character
\"	Double quotation mark
\\	Backslash
ooo	ASCII character in octal notation
\x*hhh*	ASCII character in hexadecimal notation

Front Panel

Figure 2–1 shows the front C-Mount Lens threads for the VS-1 Smart Camera.

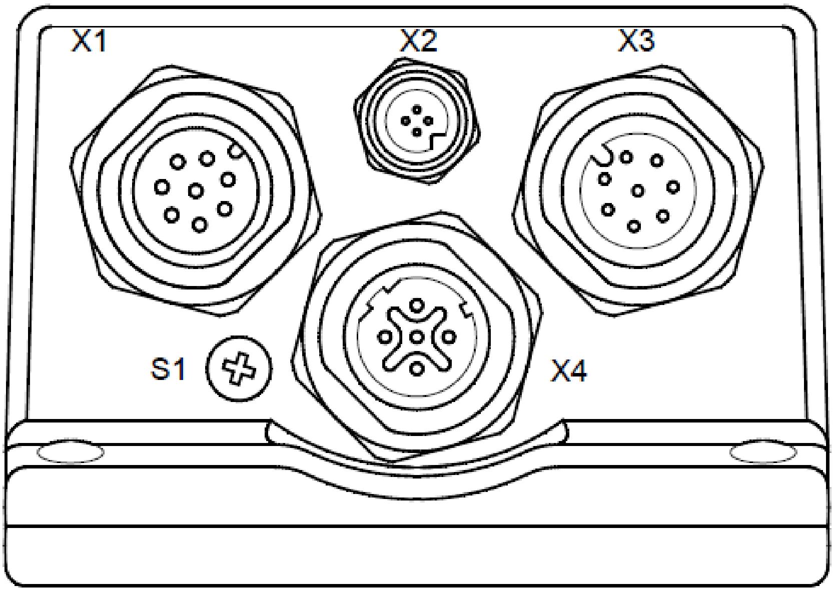
FIGURE 2–1. Front Panel



Rear Panel

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

FIGURE 2–2. **Rear Panel Layout**



- X1 — Power and Primary I/O (M12 A-coded, plug)
- X2 — External Light Power, Strobe only (M5, socket)
- X3 — Secondary I/O, Serial (M12 A-coded, socket)
- X4 — Industrial Ethernet (M12 D-Coded)
- S1 — QuicSet™ (Remove screw for access)

Note: On earlier productions units only. Function not required.

Mode/Status LEDs

Figure 2–3 shows the mode and status LEDs.

FIGURE 2–3. Mode/Status LEDs

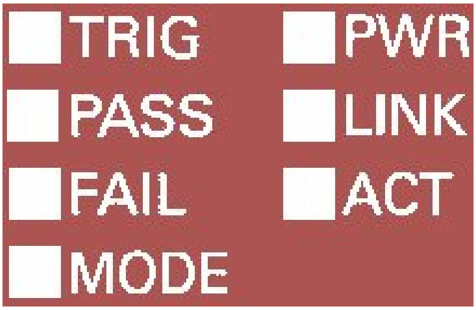


Table 2–4 describes the mode and status LEDs.

TABLE 2–4. Mode/Status LEDs

Name	Description	LED Color
TRIG	Trigger LED	Yellow
PASS	Pass LED	Green
FAIL	Fail LED	Red
MODE	Mode LED	Yellow
PWR	Power on LED	Green
LINK	Ethernet Link	Green
ACT	Ethernet ACT	Yellow

These LEDs convey visually power-on status and error codes.

Important Label Information

Each VS-1 Smart Camera has its own label, which contains important information about that Smart Camera.

- MODEL NUMBER/CATALOG NUMBER — The model number for your VS-1 Smart Camera.
- PN — The part number of your VS-1 Smart Camera
- SN — The serial number of your VS-1 Smart Camera.
- MAC ADDRESS — This unique address is important because, by default, a VS-1 Smart Camera's name on the network is "HawkEyeXXXXXX," where

XXXXXX is the last six alphanumeric characters in its MAC address. So, for the VS-1 Smart Camera with MAC address 00:60:33:E1:FF:FA, the default network name is “HawkEyeE1FFFA”.

- Optional IntelliFind License Key (applies to HE1610TIS and HE1610TIH only)

Power-on Sequence

Each stage of the power-on sequence drives the LEDs in a binary up-count fashion according to Table 2–5. The LEDs illuminate before the test is executed and remain in that pattern until the next test is run or an error condition is detected and displayed.

Error Codes

In the event of an error being detected, the beeper beeps five times and an error code (in binary form) representing the test that failed flashes on the LEDs. The LEDs continue to flash until the <Escape> character is sent on the serial port, at which point an error message is logged to the serial port and the Diagnostic Monitor is launched. The serial port terminal server must be set to the following parameters: 115200, 8, N, 1.

Table 2–5 describes the Mode/Status LED power-on sequence and error codes.

TABLE 2–5. Mode/Status LED Power-On Sequence and Error Codes

Mode	Fail	Pass	Trig	Test Performed
			•	Data Line Test
		•		Address Line Test
		•	•	Bootloader CRC Check
	•			Kernel CRC Check
	•		•	RS-232 Internal Loopback
	•	•		FPGA Load Test
	•	•	•	FPGA PCI Config Test
•				FPGA Video Buffer Test
•			•	FPGA DMA Transfer Test
•		•		FPGA Expose Done Interrupt Test
•		•	•	FPGA Read Done Interrupt Test

Beeper

The VS-1 Smart Camera's beeper is user configurable and indicates Pass/Fail conditions.

Mounting and Wiring the VS-1 Smart Camera

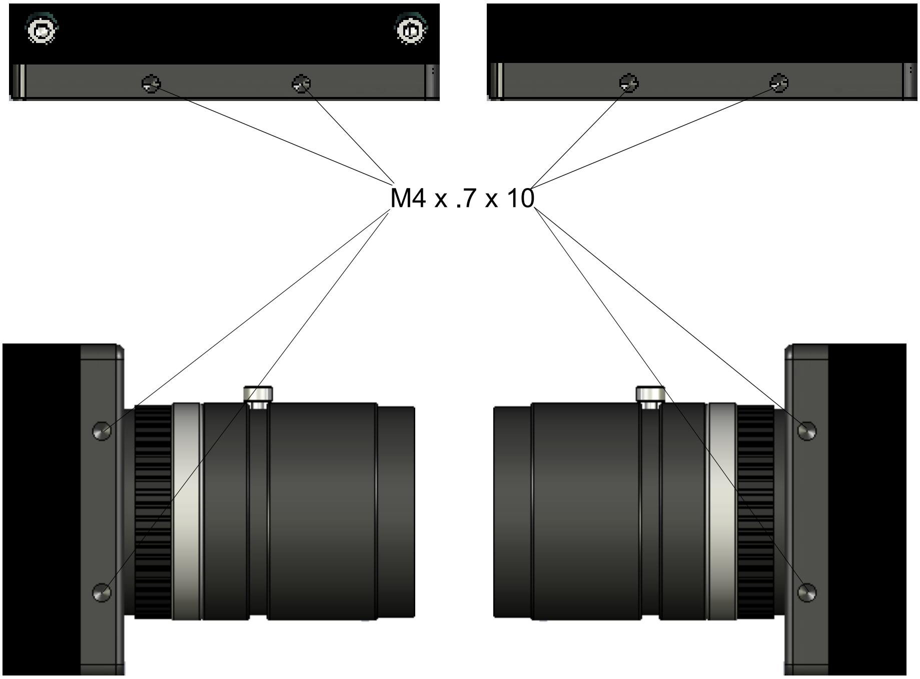
- Mount the camera securely in its camera stand (not supplied).
- Make sure the camera is mounted at the correct distance for the optics you've purchased.
- Connect the Ethernet cable (X4) and the power cable (X1) to the VS-1 Smart Camera (see "Rear Panel" on page 2-5). Connect the VS-1 Smart Camera to a 24V power supply.

Mounting Using Front Block

Note: Do not insulate the mounting block. The mounting block of the VS-1 Smart Camera is part of the heat dissipation system, and metal-to-metal contact is required for effective cooling. Refer to Appendix C, "Specifications," for mounting block dimensions.

You can mount the VS-1 Smart Camera using the M4 holes located on the top, bottom, and each side of the front block, as shown in Figure 2–4.

FIGURE 2–4. **Locations for Mounting Using Front Block**

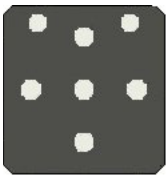


Caution: Using screws that are too long for the threaded holes may damage the VS-1 Smart Camera. The accessory mounting blocks use 10mm machine screws.

Mounting Using Standard Mounting Block

You can mount the VS-1 Smart Camera using the standard mounting bloc (see Figure 2–6).

FIGURE 2–5. **Standard Mounting Block**

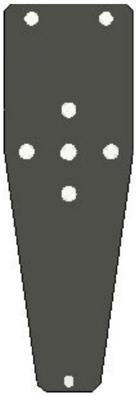


Part Number - 98-HT00-0CM1

Mounting Using Accessory Mounting Block

You can mount the VS-1 Smart Camera using the accessory mounting block (see Figure 2–6). The backward compatible mounting block positions a VS-1 Smart Camera in the same position as a VS-1 Smart Camera for optical alignment.

FIGURE 2–6. Accessory Mounting Blocks

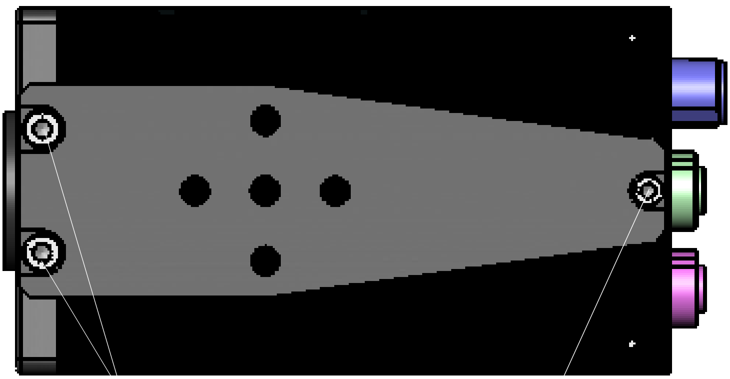


Part Number - 98-HT00-0CM2

Location for Backward Compatible Mounting Block

Figure 2–7 shows the screw hole locations for the backward compatible mounting block.

FIGURE 2–7. Location for Backward Compatible Mounting Block



Caution: Using longer screws may damage the VS-1 Smart Camera.

Field I/O Wiring Examples

Input Opto Wiring

Figure 2–8 shows the input opto wiring for isolated NPN and PNP sources.

FIGURE 2–8. **Trigger Input Opto Wiring (NPN and PNP)**

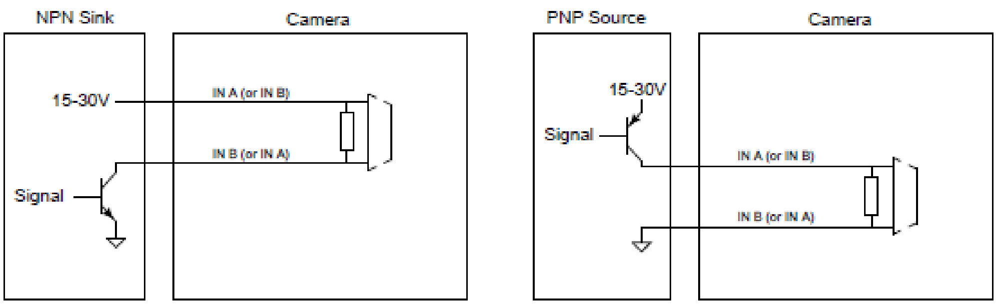
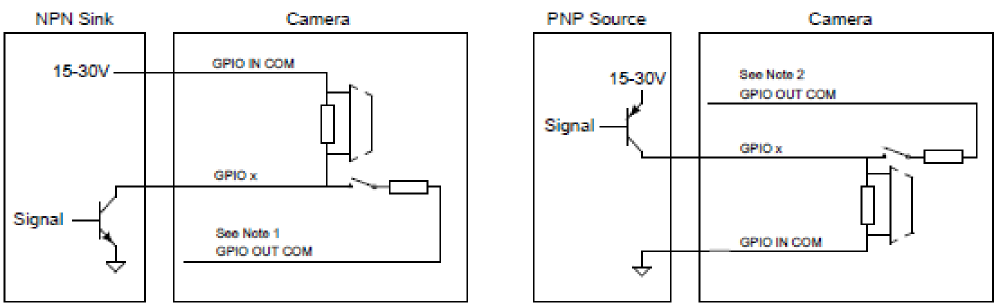


Figure 2–9 shows the input opto wiring for non-isolated NPN and PNP sources.

FIGURE 2–9. **GPIO Input Opto Wiring (NPN and PNP)**

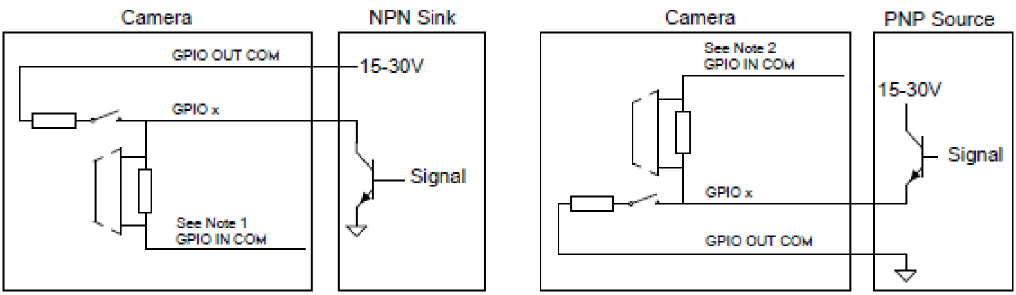


Note 1: GPIO OUT COM typically connected to ground.
Note 2: GPIO OUT COM typically connected to Voltage.

Output Opto Wiring

Figure 2–10 shows the output opto wiring for isolated input.

FIGURE 2–10. GPIO Output Opto Wiring (NPN and PNP)



Note 1: GPIO IN COM typically connected to ground.
Note 2: GPIO IN COM typically connected to Voltage.

Figure 2–11 shows the output opto wiring for isolated relay and PLC inputs.

FIGURE 2–11. Output Opto Wiring (Relay and PLC Inputs)

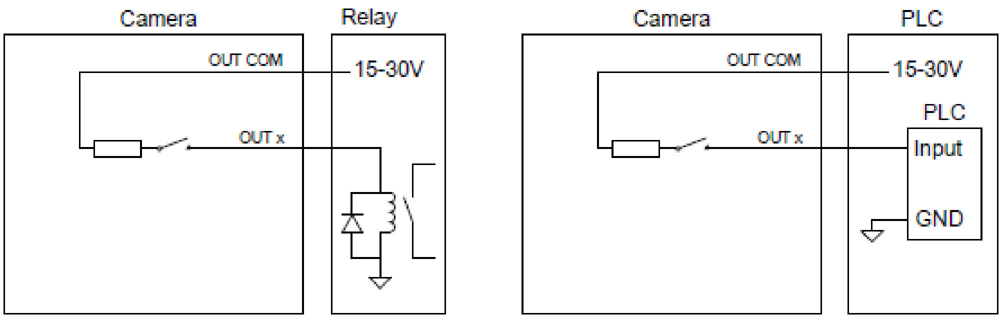
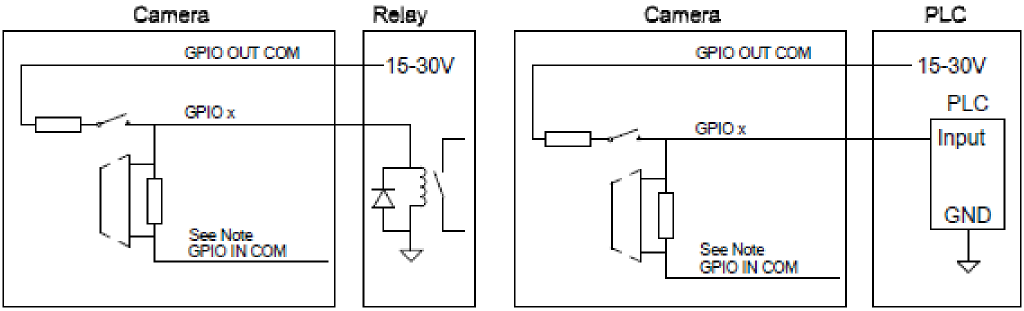


Figure 2–12 shows the output opto wiring for non-isolated inputs.

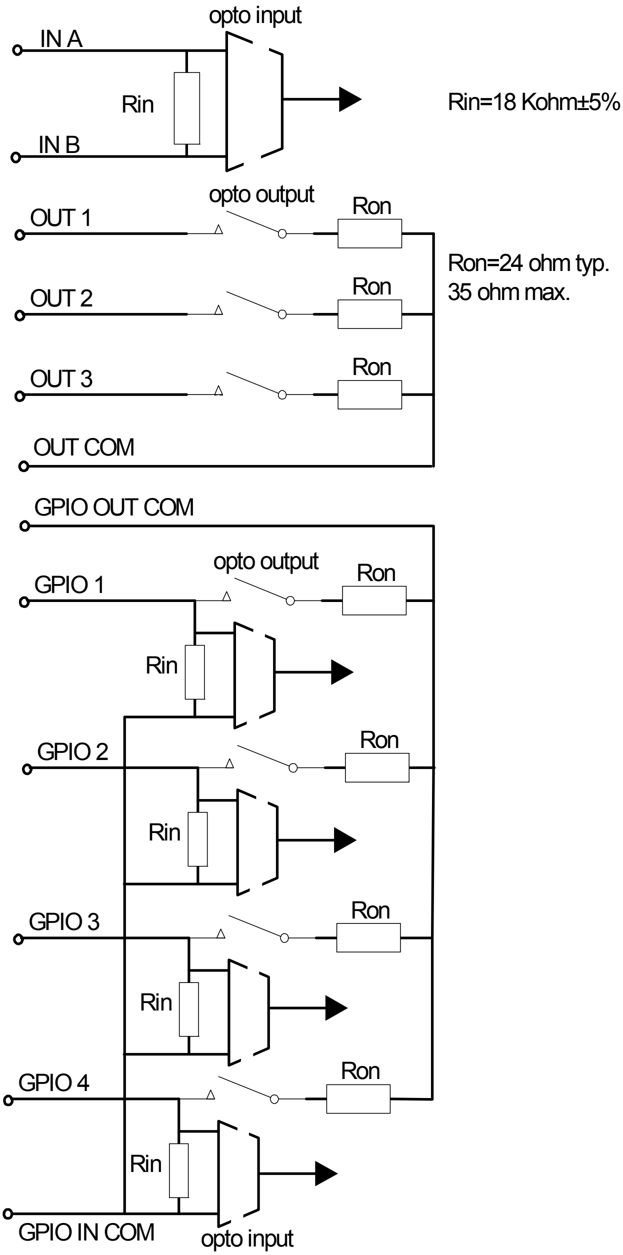
FIGURE 2–12. **GPIO Output Opto Wiring (Relay and PLC Inputs)**



1. GPIO IN COM typically connected to ground.

Caution: The maximum current that can pass through the optoisolators is 50 mA. Non-isolation setup can cause damage to the VS-1 Smart Camera if excessive voltage is applied to the optoisolators.

FIGURE 2–13. **VS-1 Smart Camera I/O Simplified Circuit Diagram**



External Strobe and 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.

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 variation in location of the part within the Field of View (FOV) may result. The FOV specified for the VS-1 Smart Camera should be large enough to cover the variation in location.

Power Requirements

Refer to Table 2–6 when determining the power supply requirements for your Smart Camera Vision System.

TABLE 2–6. Smart Camera Vision System Power Requirements

Component	24VDC
VS-1 Smart Camera	24 volts @ 350ma typical

Power Supply Wiring

For complete information about the power supply wiring, see “Power and Primary I/O Connector” on page A-1, and “HETPC-100 - Power and Primary I/O Cable” on page B-1.

3

CHAPTER 3 Ethernet/IP Communications

This chapter contains information about Visionscape® Smart Camera HE1600T support for Ethernet/IP (EIP) communications.

Ethernet/IP is an industrial protocol that provides out of the box connectivity to Allen-Bradley PLCs for the HE1610T Smart Camera. Ethernet/IP provides two mechanisms for exchanging data between a PLC and a device:

- Implicit Messaging (also called I/O) — Implicit messages are messages set up automatically by the PLC on connection to the device to refresh data at given intervals. Conceptually, a block of data (memory) is shared between the PLC and the device. This block contains binary data made up of simple scalar types: integers, floats and strings. This block of data is kept in sync on the PLC and on the device typically every 10 msec. Both the PLC and the device can write and read data from this block. This is implemented in V3.7.3.
- Explicit Messaging — Explicit messages are messages you set up in your PLC program to retrieve data at specific points in time. This is different from the implicit mechanism which exchanges a block of data at regular intervals as soon as a connection to the device is made, without the need to write any special instructions in the PLC program. This is not supported by V3.7.3.

Note: Messaging is only operational in online mode when a job is running on the Smart Camera. Only I/O Points will function when running in Setup Mode on the PC.

A/B Logix PLCs That Support Ethernet/IP I/O Messaging

With this release only Allen-Bradley PLCs that are Ethernet/IP ready (i.e., Logix series PLCs) are supported using implicit messaging. Table 3–1 details the Controllers with required Interface modules that support Ethernet/IP Implicit (i.e., I/O) messaging.

TABLE 3–1. Allen-Bradley PLCs That Are Ethernet/IP Ready

Cat. No.	Memory (max)	I/O (max)	Local I/O	Programming Software	Programming Languages	Ethernet/IP Interface
1769 CompactLogix	1.5 Mbytes	30 Compact I/O modules	1769 Compact I/O	RSLogix 5000	• relay ladder • function block diagram (FBD) • structured text (ST) • sequential function chart (SFC)	• 1769-L32E (embedded) • 1769-L35E (embedded)
1768 CompactLogix	3 Mbytes	30 Compact I/O modules	1769 Compact I/O	RSLogix 5000	• relay ladder • function block diagram (FBD) • structured text (ST) • sequential function chart (SFC)	• 1768-ENBT
FlexLogix	512 Kbytes	512 digital 128 analog	1794 FLEX I/O 1797 FLEX Ex I/O	RSLogix 5000	• relay ladder • function block diagram (FBD) • structured text (ST) • sequential function chart (SFC)	• 1788-ENBT
ControlLogix	8 Mbytes	128,000 digital 3,800 analog	1756 ControlLogix I/O	RSLogix 5000	• relay ladder • function block diagram (FBD) • sequential function chart (SFC)	• 1756-EN2T • 1756-EN2F • 1756-ENBT
SoftLogix5800	PC dependent	PC dependent	none	RSLogix 5000 C/C++ routines	• relay ladder • function block diagram (FBD) • structured text (ST) • sequential function chart (SFC)	• PC Ethernet card

Two blocks are defined that allow the manipulation of IO points, camera control and status bits and user-defined data by both the PLC and the HE1610T device. These two blocks are identical except for the size of the user-defined data, which is either 64 bytes or 400 bytes.

The PLC manipulates the data within the block through what is called assemblies. There are two assemblies defined, one for OUTPUT (writing to the Binary Block of data) and one for INPUT (reading from the Binary Block of data) as seen by the PLC. For Visionscape, the OUTPUT assembly is used to receive/read data from the PLC and the INPUT assembly is used to send/write data to the PLC.

Note: The PLC programmer needs to select the appropriate assembly/assemblies to manipulate the block of data within the PLC program. In this release, the block of data is made up of a fixed section containing 32 bit of IO points and a user-defined section.

In Visionscape®, the data is accessed by standard mechanisms and the appropriate assembly is selected automatically, these mechanisms are standard virtual IO for the fixed IO section of the block and Perl scripts for the user-defined section. The fixed IO section is dynamic, i.e., changes to IO points in the block are sent to/received from the PLC on the next implicit scan. Changes to the user section of the block are latched by the Inspection at the start of the vision cycle and sent back to the PLC at the end of the vision cycle.

Ethernet/IP IO Points

Ethernet/IP IO Points can be read and written by their corresponding virtual IO points with the existing tool set including DigitalInput and DigitalOutput steps. The state of the IO points is updated each time the camera receives an Output Assembly; the values are latched when the DigitalInput step is executed.

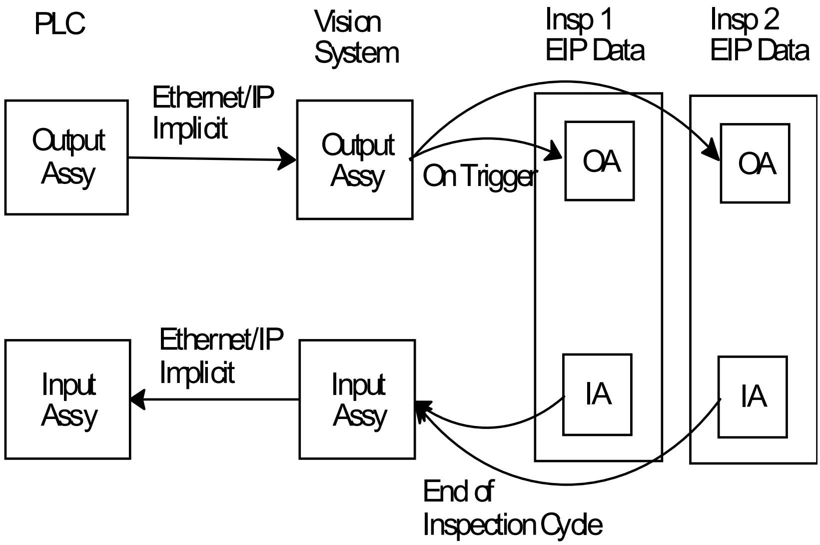
Changes to the IO Points from a DigitalOutput step or other output IOPointDm are written immediately to the Input Assembly. The values will be received by the PLC on the next implicit Input Assembly transfer.

Note: Resolution of the I/O is dependent on the PLC's Ethernet/IP Requested Packet Interval (RPI) setting. I/O transitions shorter than the duration of the RPI will not be visible to the PLC. Please refer to Rockwell documentation for further information regarding this setting. To resolve I/O from a device of width x msec, the RPI setting of the Ethernet module must be programmed to x/2 msec or lower. For example, to detect pulses generated by the device in the PLC program of width 10 msec, set the RPI to 5 msec or less.

Binary Data Block

The Inspection Step contains a BinaryBlockDm named "EIP Data". This datum contains a copy of the Input Assembly and Output Assembly data blocks. The contents are copied from the system Output Assembly at the start of processing of an inspection to the Inspection's latched copy of the Output Assembly. This is at the time of the trigger for triggered inspections or the start of an inspection cycle for non-triggered inspections.

The Inspection's instance of the Input Assembly is written back to the system's Input Assembly at the end of an inspection cycle. This is at the time when Data Valid is asserted at the end of the vision cycle. Only the bytes written by the inspection are changed in the system's Input Assembly. If no bytes are written by an inspection, the system Input Assembly will be unchanged. This allows multiple asynchronous inspections to write to separate sub-blocks within the user data block and not interfere with each other. The UserID tag register always reflects the last inspection cycle that has completed.



The contents of the User Data Block are read and written by inserting Custom Steps. Two Perl scripts (BinaryBlockRead & BinaryBlockWrite) are provided to access the latched Inspection OUTPUT assembly and INPUT assembly data, respectively. The assembly objects are described in the following two tables:

INPUT Assembly (Instance 100) (80 bytes)		OUTPUT Assembly (Instance 112) (80 bytes)	
DINT	**Meaning**	**DINT**	**Meaning**
0	32 Bits of Camera Virtual IO	0	32 Bits of Camera Virtual IO

INPUT Assembly (Instance 100) (80 bytes)		OUTPUT Assembly (Instance 112) (80 bytes)	
DINT	**Meaning**	**DINT**	**Meaning**
1	User-defined tag value	1	User-defined tag value
2	Camera Status Register	2	Camera Control Register
3	Last Error	3	Reserved
4...19	64 bytes of user data	4...19	64 bytes of user data

INPUT Assembly (Instance 101) (416 bytes)		OUTPUT Assembly (Instance 113) (416 bytes)	
DINT	**Meaning**	**DINT**	**Meaning**
0	32 Bits of Camera Virtual IO	0	32 Bits of Camera Virtual IO
1	User-defined tag value	1	User-defined tag value
2	Camera Status Register	2	Camera Control Register
3	Last Error	3	Reserved
4...103	400 bytes of user data	4...103	400 bytes of user data

The definitions of the fields are listed below.

DINT 0: 32 Bits of Camera Virtual IO

This 32 bit value maps to a subset of virtual IO points on the camera (VIO 129 through VIO 160).

For the INPUT assembly, the value of this register always reflects the state of the corresponding IO points from the camera. By using the corresponding virtual IO point on the camera, these IO points can reflect the state of any IOOutputDm such as picture done, or set by a user expression using a DigitalOutput step.

For the OUTPUT assembly, the value of this register is written to the corresponding IO points of the camera. They can be used by any IO input datum, such as triggers, or read by using a DigitalInput step.

DINT 1: User-defined Tag Value

The user-defined tag value is a single DINT that is sent to the camera through implicit messaging to Assembly Instance 2. When the camera receives a new user tag from Assembly Instance 112 (or 113), it is latched in the inspection's copy of the Assembly and will be reflected through Assembly Instance 100 (or 101) at the end of the inspection cycle. This allows the PLC to match inspection results in the Input Assembly to the Output Assembly that was latched at the start of the same inspection cycle.

DINT 2: Camera Control Register

TABLE 3–2. Control Register Bit Definitions

BIT	Meaning
0	Stop, transition 0->1 to stop inspections
1	Start, transition 0->1 to start inspections
2	Clear error, transition 0->1 to clear Last Error
3	Clear command status, transition 0->1 to clear command stat
4…31	Reserved

DINT 2: Camera Status Register

TABLE 3–3. Status Register Bit Definitions

BIT	Meaning
0	System Online 1 = System is online and a job is loaded on the camera 0 = Camera is not ready for inspection to go online; it is either loading a job or no job is loaded on the system.
1	Inspection Online 1 = Inspection is online, the job is running on the camera 0 = Inspection is offline
2	Command Complete. Cleared by issuing ‘Clear Command Status’ 1 = Last issued command has completed 0 = Last issued command is being processed
3	Command Succeeded. Updated when command is completed. Cleared by issuing ‘Clear Command Status’ 1 = Success 0 = Failure, check Last Error for more information
4…31	Reserved

Commands sent to the Control Register are ignored when the camera is offline; the camera must be online before sending a Start trigger.

DINT 3: Last Error

Last Error returns the last error code sent to the Ethernet/IP subsystem. It will return all zeros if no error occurred. The last error code can be reset to zero by transitioning the Clear Last Error bit of the control register from zero to one. Error bits are allocated below:

Mnemonic	Bit Mask	Description
EIP_ERROR_TRUNCATED	0x0001	Set when the AVP writes data in bytes 64 - 399 of the user data section, and the Ethernet/IP requests the small assembly.

DINT 4…19/4…103: User Data

This contains 64 (or 400 bytes) of user-defined data. The content of the data block is defined by how the bytes are written by the vision job using Perl tools. The data is unstructured within the Ethernet/IP transport; it is up to the PLC and Vision programmer to ensure the data structures match on each end of the transport.

The user-defined data is programmed in the AVP by using two Perl scripts described in the next section. Data allocated in the user-defined block must match equivalent PLC data types. The mapping between Data types in the AVP to PLC types is summarized below:

Datum Type	BinaryBlockIO Type	PLC Native Type[1]	Byte Count
StatusDm	BOOL	DINT	4
IntDm	INTEGER	DINT	4
DoubleDm	FLOAT	REAL	4
StringDm	STRING	STRING (DINT + STRLEN)	4 + STRLEN
PointDm	POINT	REAL x 2	8
LineDm	LINE	REAL x 3	12
DistanceDm	DIST	REAL	4
AreaDm	AREA	REAL	4
AngleDm	ANGLE	REAL	4

[1] All data in the user-defined binary block is aligned on 4-byte boundary to match the tanglement rules expected by the PLC. Padding bytes are added (for STRING) as necessary so the next type following a STRING is 4 bytes aligned.

User Data Block Accessor Perl Tools

Two Perl scripts (BinaryBlockRead & BinaryBlockWrite) are provided to access the latched Inspection OUTPUT assembly and INPUT assembly data, respectively.

- The BinaryBlockRead Perl script must be inserted in the AVP after the AcqStep has executed and before any other tool that uses the data in the binaryblock. Typically, this will be as the first tool inside the Snapshot.
- The BinaryBlockWrite Perl script must be inserted in the AVP after any other tool that calculates results to be stored in the binaryblock. Typically, this will be as the last tool in the Inspection.

BinaryBlockRead/Write Perl Scripts

Both Script tools share a similar UI interface. Additional features are available for the BinaryBlockWrite script to control when Data is sent to the Network.

BinaryBlockRead Perl Tool UI

First Datum Block Offset:	0
Add Datum Type:	BOOL
Add Datum Reference at End:	<click to execute>
Edit Datum Reference:	<click to execute>
Remove Datum Reference:	<click to execute>
Datum Reference to Edit/Remove:	1
Delete Data RefList:	<click to execute>
INT Error Code	0
Status	True

BinaryBlockWrite Perl Tool UI

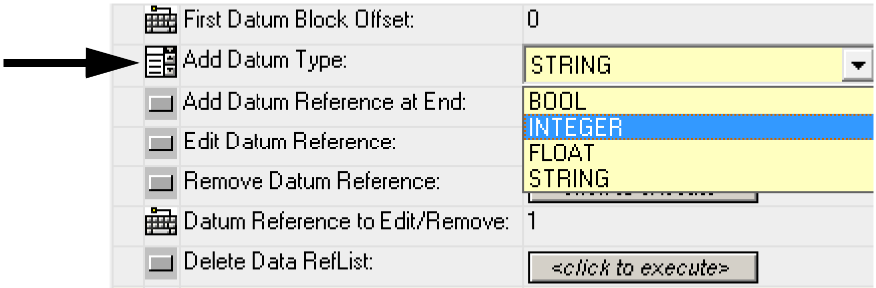
Send Data Block:	At End of Cycle
Include User Tag First:	☐
First Byte Block Offset:	0
Add Datum Type:	BOOL
Add Datum Reference At End:	<click to exec
Edit Datum Reference:	<click to exec
Remove Datum Reference:	<click to exec
Datum Reference to Edit/Remove:	1
Delete Data RefList:	<click to exec
INT Error Code	0
Status	True

The base user interface lets the user build the mapping between the binary data in the Ethernet/IP block to a set of Datums in the .

- First Datum/Byte Block Offset — Offset in bytes from the beginning of the user block where the data is read from/written to. This value is aligned on a 4 byte boundary by the tool with respect to the beginning of the user-defined binary block.

- Add Datum Type — List of Datum types for mapping bytes in the binary block to Datums and vice-versa as shown below:

– Perl Tool - BinaryBlockRead - Add Datum Type:



- BOOL (4 bytes) — Can be connected to a StatusDm (Checkbox).

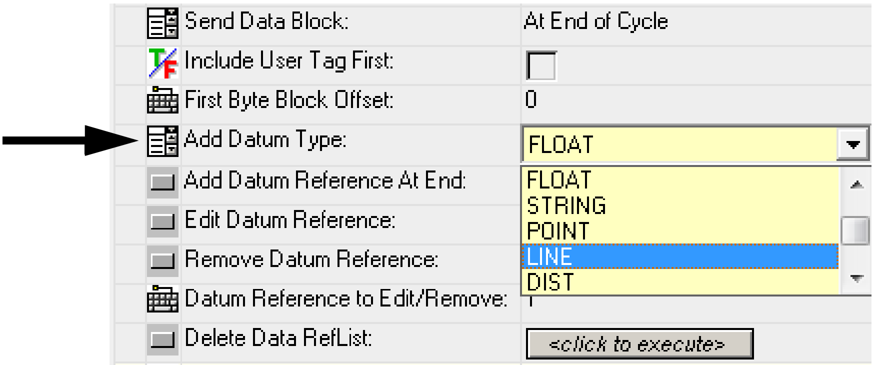
Note: The boolean value should be read from bit 0 of the DWORD on the PLC. The state of the remaining bits 1 through 31 is not guaranteed.

- INTEGER (4 bytes) — Can be connected to an IntDm (Thresholds, etc…).

- FLOAT (4 bytes) — Can be connected to a DoubleDm (Tolerances, etc…).

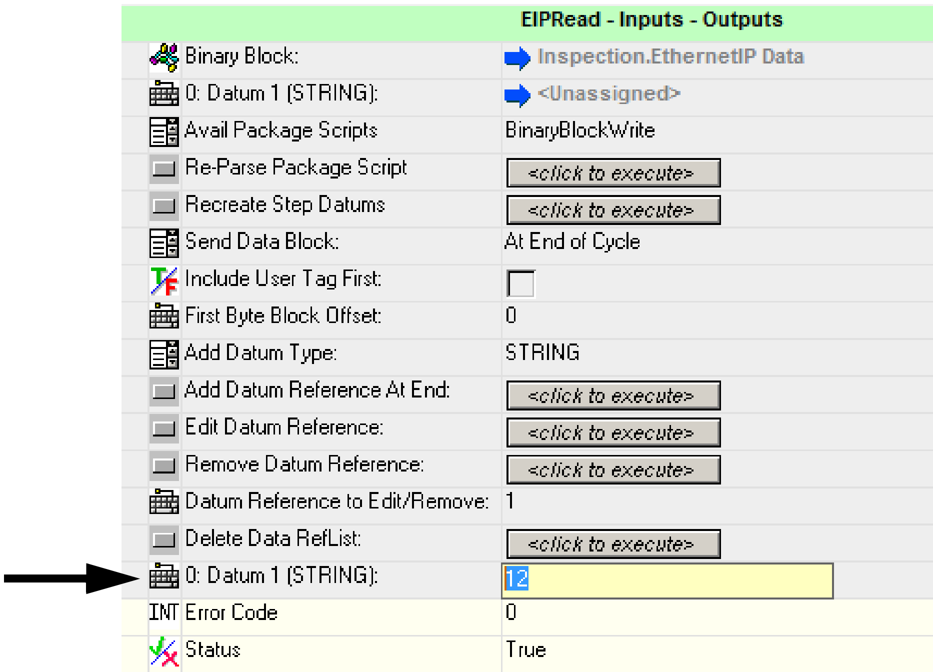
- STRING (4 + LEN Bytes) — Can be connected to a StringDm (Match String, etc…). When adding a String, its maximum expected length must also be specified to properly reserve a fixed number of bytes in the block. A 4 byte Length field (at the beginning of the STRING) is allocated automatically in the block also.

– Perl Tool - BinaryBlockWrite - Add Datum Type:



- BOOL (4 bytes) — Can be connected to a StatusDm (Pass/Fail).
- INTEGER (4 bytes) — Can be connected to an IntDm (Counters, etc…).
- FLOAT (4 bytes) — Can be connected to a DoubleDm (Score, etc…).
- STRING (LEN bytes + 4) — Can be connected to a StringDm (DMR Text, etc…). When adding a String, its maximum expected length must also be specified to properly reserve a fixed number of bytes in the block. A 4 byte Length field (at the beginning of the STRING) is allocated automatically in the block also.
- POINT (3 Floats) — Can be connected to a PointDm (Location, etc…).
- LINE (3 Floats) — Can be connected to a LineDm (FastEdge, etc…).
- DIST (1 Float) — Can be connected to a DistanceDm (Pt2Pt Distance, etc…).
- AREA (1 Float) — Can be connected to an AreaDm (Blob Area, etc…).
- ANGLE (1 Float) — Can be connected to an AngleDm (Line Angle, etc…).

The Example String of length 12 shown below actually uses 16 bytes in the block:

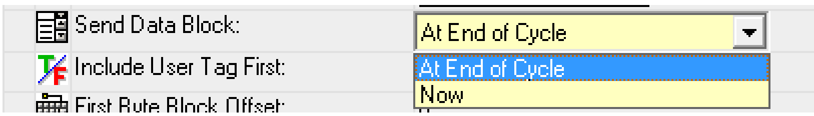


- Add Datum Reference At End — Adds a Datum Reference of type “Add Datum Type” at the end of the datum list. The reference can then be connected to point to a particular Datum in the . For Read operation, Datum must be connected to a tool that runs after the BinaryBlockRead tool. For Write operation, Datum must be connected to a tool than runs before the BinaryBlockWrite tool.

- Edit Datum Reference — Edits the Datum Reference at the given index (specified by the Datum Reference to Edit/Remove property) and changes its type to the current “Add Datum Type” selection. Note that, if the previous reference was connected to a Datum in the , this reference will be reset by this operation. The index must be a number between 1 and N where N is the number of Datum references in the block. For example, for EIPRead example above: to edit/remove “0: Datum 1 (STRING)”, enter 1 in that field.

- Remove Datum Reference — Removes the Datum Reference at the given index (specified by the Datum Reference to Edit/Remove property).

- Datum Reference to Edit/Remove — Specify the Datum reference by index that is selected for Editing or Removal (as described in the previous two bullets).
- Delete Data Reflist — Deletes the entire block datum reference list, i.e. deletes the mapping from Binary Block to Datums (BinaryBLockRead) or the mapping from Datums to Binary Block (BinaryBlockWrite), hence allowing the mapping to be reconstructed.
- Send Data Block (Write Script only) — Allow the BinaryBlock to be sent to the Network either at the end of the current Inspection cycle or immediately.

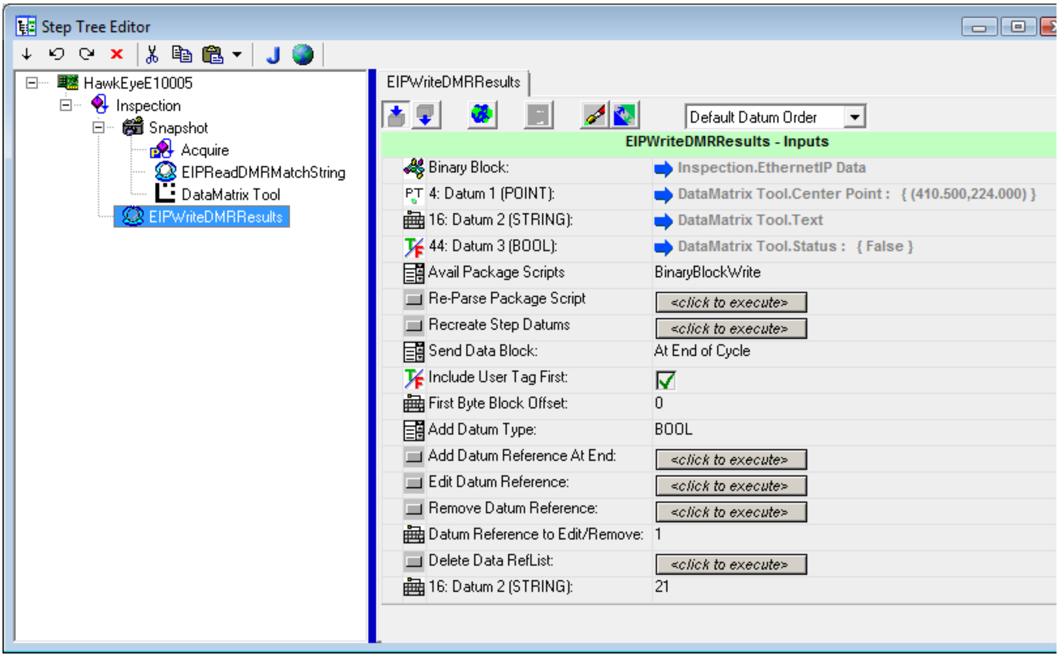


- Include User Tag First (Write Script only) — When checked, the first 4 bytes of the User Data block will be a copy of the User Tag that was sent by the PLC for this Inspection cycle.

Example Ethernet/IP Read/Write Test

This AVP reads the match string from the Network through Ethernet/IP at the beginning of each cycle and sets it in a Data Matrix tool. The decode string, position and status of the Data Matrix tool are sent at the end of each cycle to the Network through Ethernet/IP as results.

FIGURE 3–1. Ethernet/IP Read/Write Test



CHAPTER 4

Visualization HMIs

This chapter contains information about Visionscape® Smart Camera HE1600T support for Visualization HMIs.

The HE1600T features a built in runtime monitoring web page that can be viewed from any supported browser on the same network. Supported browsers include:

- Internet Explorer 5.0 or later
- Firefox 3.0 or later

The Runtime Page shows an image from the Visionscape® Smart Camera HE1600T, along with inspection counters and buttons to control certain aspects of the display. A title bar displays the camera name, IP address and the name of the Job. Options are available to change if and where the counters, buttons, and titlebar are displayed. Additionally, up to 10 values from the job can be displayed along with each image. These values can either be overlayed over the image, or shown as a tabular report underneath the image.

Note: The HMI web page will not automatically detect if the Vision Job it is connected to has been changed or edited. Therefore, in this instance, please refresh the page manually (via the web browser refresh button).

You can set all settings and options through a series of option pages that can appear over the main display. All parameters are saved as cookies in the web browser environment so that, the next time the Runtime Page is loaded for that device, the layout and settings are retained.

The default behavior is:

- Images and counters are for the first inspection in the job
- All images (pass & fail) are shown
- The display is refreshed automatically at regular intervals (auto=on)
- Graphics are overlaid on the image

Note: For performance reasons, not all graphics are available when viewing images on the web page. Only vector graphics are displayed.

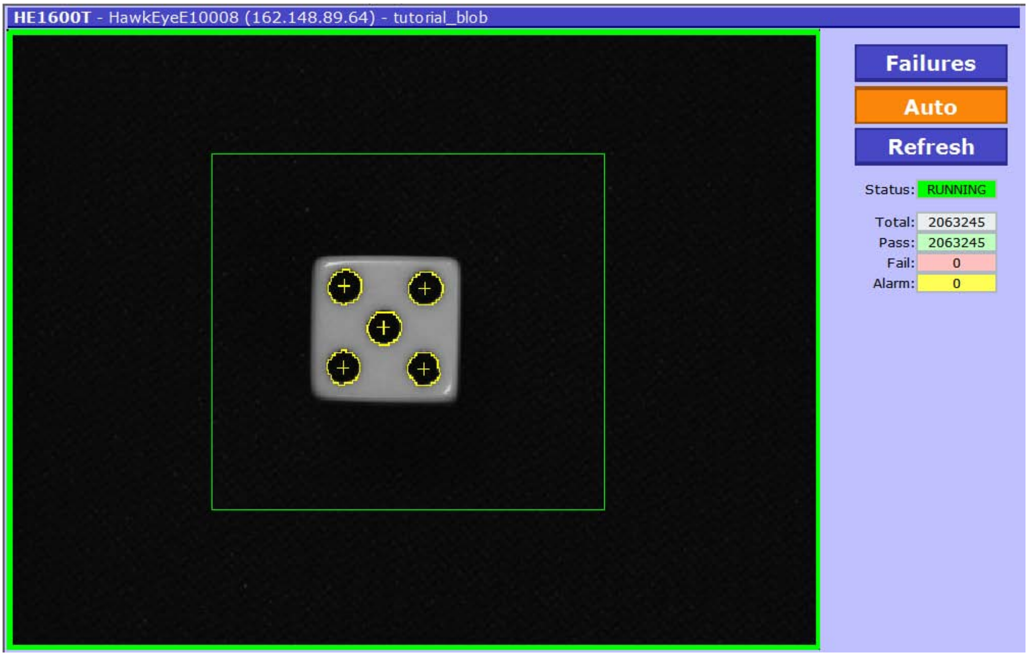
- The border around the image signifies the status of the inspection (green=pass, red=fail)

The Runtime Image Page is accessed through a URL that contains the IP address of the camera, and optional parameters. The default page is accessed by specifying the IP address of the camera in a web browser. For example:

http:// 161.218.121.58
(example only; use actual IP address of the HE1600T)

If no previous settings have been set by the user, the display will be similar to the screen in Figure 4–1

FIGURE 4–1. Main HMI Window



The web page includes the following elements:

- Title Bar — Displays the name and IP address of the camera, and the name of the job (AVP). Note that the file extension (.avp) is removed from the displayed file name.
- Failures Button — When this button is selected, only images related to failed inspections are displayed.
- Auto Button — When this button is selected, the image and counters are updated automatically. If the button is not selected, both the image and counters are frozen.
- Refresh Button — Selecting this button manually updates the image and counters.

  Note: If the Refresh Button is selected while the system is in Auto-Refresh Mode, the image disappears and statistics freeze for a period of approximately 10 seconds after which point the system returns to normal operation.

- Status — The run status of the inspection (RUNNING or STOPPED).
- Counters — Displays the Total, Pass, Fail and Alarm counters for the selected inspection.

Settings Pages

You can configure all the options and settings using the settings pages. By default, the Settings button is not displayed; to display the Settings button, see Figure 4–8, "Buttons," on page 4-10.

To display the settings pages, specify the URL with the "setopt=1" parameter:

http://ip_address/?setopt=1

This will display the Runtime Page overlayed with the Options Setup page:

FIGURE 4–2. Options Setup Page

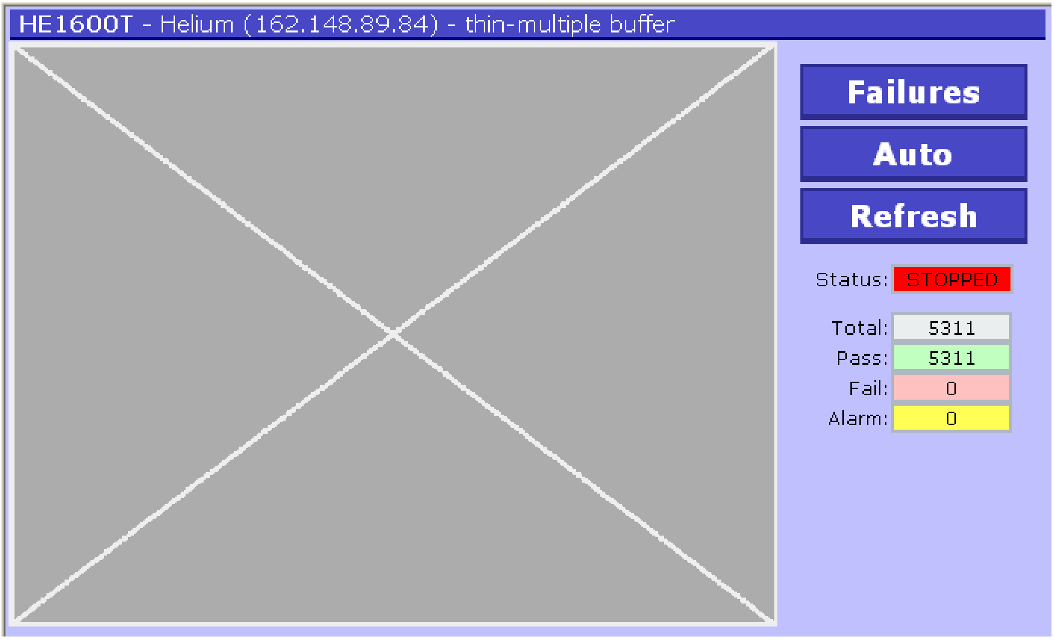
Use the tabs at the top of the screen to navigate between the several setup pages. To close the setup screens and return to the main display, use the Close button ("X") at the upper right corner.

The Layout page controls many layout features, which are organized into groups.

Layout Options

You can configure the overall layout of the Runtime Page via the Settings pages. Figure 4–3 shows a default configuration:

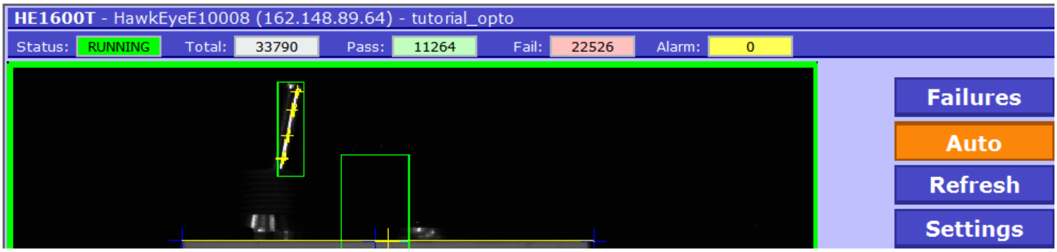
FIGURE 4–3. Default Configuration



Buttons, status, and counters appear to the right of the image area. The buttons are sized for use via a touch screen.

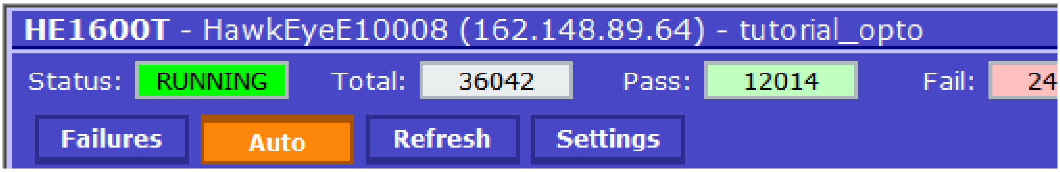
Figure 4–4 displays how the layout has been changed to position the counters at the top, shown without titles to save room. Additionally, an Options button now appears in the right side area.

FIGURE 4–4. **Buttons at Right & Counters & Status Above Image**



Another example with buttons and counters at the top:

FIGURE 4–5. **Status, Counters, & Buttons Displayed Above Image**



It is also possible to hide all elements except the image.

Modes

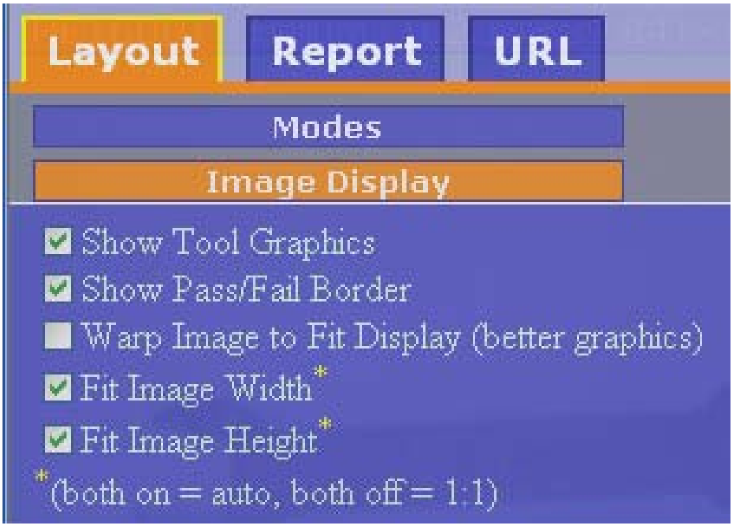
Selecting the Modes group results in the following options being displayed:

FIGURE 4–6. **Modes**

- Launch in Auto Mode — Determines whether or not the Runtime Page defaults to be in auto-refresh mode when launched. The default is On.
- Launch in Failures Mode — Determines whether or not the Runtime Page defaults to be in show failures mode when launched. The default is Off.

Image Display

FIGURE 4–7. Image Display



- Fit Image Height/Fit Image Width — These two settings determine how the image is scaled to fit the display area. If both are off, then no scaling is performed and the image is displayed 1:1. If both are on, then auto scaling is performed, fitting the width or height depending on which fits the display area better. Otherwise, the image is scaled either to fit the width or height. The default is Auto (both On).

- Show Pass/Fail Border — Displays a border around the image. Green = pass, red = fail. The default is On.

- Show Tool Graphics — Shows tool graphics overlayed on the image. Not all tool graphics are supported. The default is On.

- Warp Image to Fit Display — Scaling the image to fill the display area can have an adverse effect on the quality of the graphics displayed. As an example, lines can be missing. This setting improves the quality of the displayed graphics. Turning this off will reduce the overhead on the 1600T. The default is On.

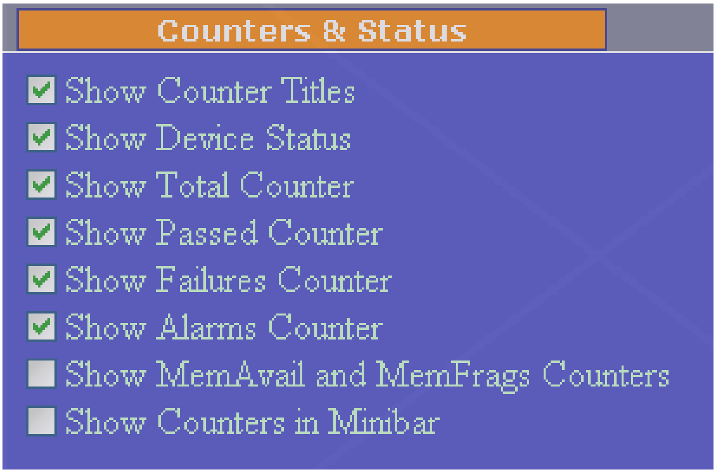
Buttons

FIGURE 4–8. Buttons

- Show Auto Button — If on (default), the Auto button is shown.
- Show Buttons in Minibar — If on, the buttons are shown in the Minibar area, which appears under the titlebar. If off (default), the buttons will be shown to the right of the image area.
- Show Failures Button — If on (default), the Failures button is displayed.
- Show Graphics Button — If on, the Graphics button is shown. This button controls if the graphics are overlayed on the image. The default is Off.
- Show Refresh Button — If on (default), the Refresh button is displayed.
- Show Settings Button — If on, the Settings button is displayed.

Counters & Status

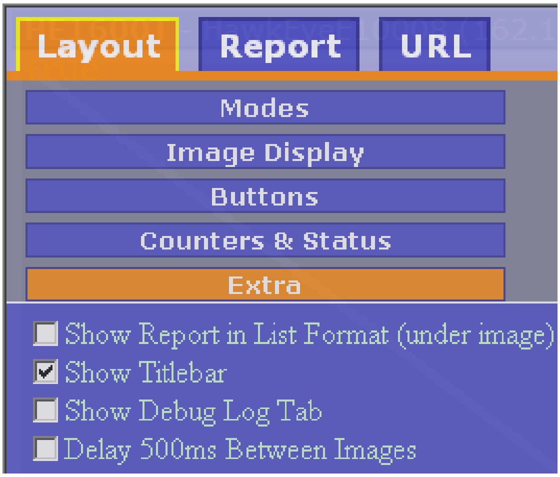
FIGURE 4–9. Counters & Status



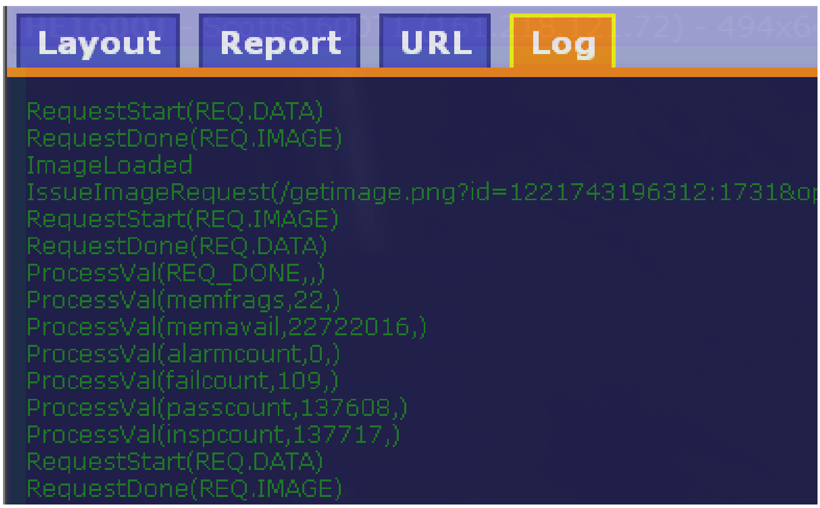
- Show Alarms Counter — If on (default), the alarms counter is shown.
- Show Counters in Minibar — If on, the counters are shown in the Minibar area, which appears under the titlebar. If off (default), the counters will be shown to the right of the image area.
- Show Counter Titles — If on (default), a title is displayed to the left of each counter.
- Show Device Status — If on (default), the device status (RUNNING, STOPPED) is displayed.
- Show Failures Counter — If on (default), the failures counter is shown.
- Show MemAvail and MemFrags Counters — If on, two counters are shown which display memory use status for the HE1600T. The default is Off.
- Show Passed Counter — If on (default), the passed counter is displayed.
- Show Total Counter — If on (default), the total counter is shown.

Extras

FIGURE 4–10. Extras



- Delay 500ms Between Images — If on, delays 500ms between image captures.
- Show Debug Log Tab — If on, the Log tab is displayed to the right of URL. Click on the Log tab to display information that will be helpful for debugging purposes:



- Show Report in List Format — If on, the report is shown in tabular form below the image. If off (default), the report is overlayed on top of the image.
- Show Titlebar — If on (default), the titlebar is shown.

As each option is checked or unchecked, the effect can be seen immediately by observing the layout of the Runtime Page shown behind the Options Setup Page.

Pressing Save saves these settings so that they become the default behavior the next time the page is launched.

Pressing Defaults resets the stored settings to the original defaults the next time the page is launched.

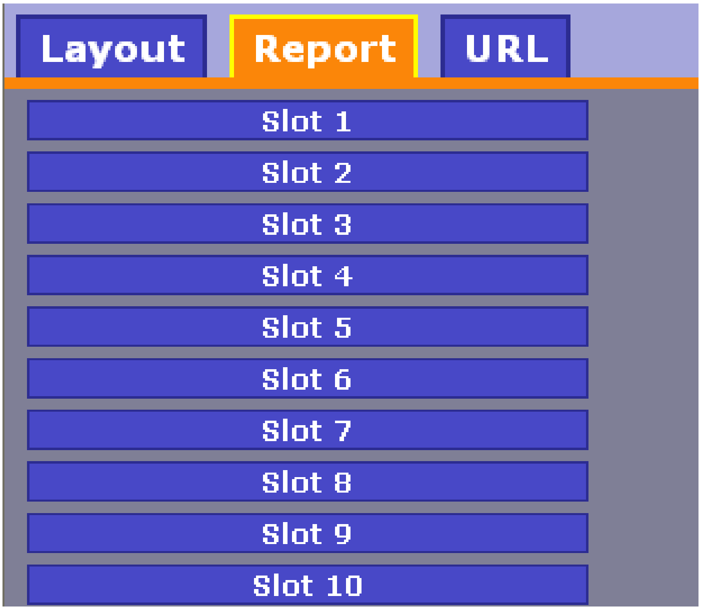
The Close button (“X” in upper right corner) will return you to the main Runtime Page.

Report Tab & Setup Screen

NOTE: *Changing the report configuration requires reloading the web page in order to take effect.*

The Report Tab displays the Report Setup screen:

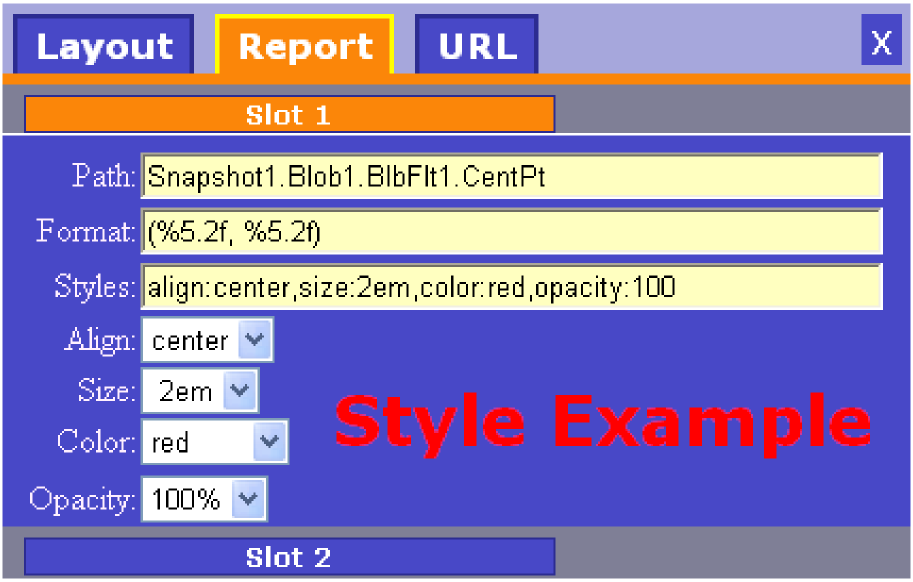
FIGURE 4–11. Report Setup Screen



Data Values from datums in the selected inspection can be formatted and overlaid on the displayed image or shown in a table below the image (see “Displaying the Output of a Datum” on page 4-19). This is specified by assigning one of 10 data report slots. If overlayed on the image, each of these slots will represent a row in the display area, which is evenly split into 10 equal sized rows. The spacing will depend on the overall size of the display area, which in turn is dependant on the dimensions of the browser window. If the report is shown in list form, each slot corresponds to one of 10 rows.

Selecting a slot to configure results in the following display:

FIGURE 4–12. Slot 1 Selected



At a minimum, you must specify the path to a datum. The inspection is implied, so it is not in the path. In the example above, the path

Snapshot1.Blob1.BlbFlt1.CentPt

is specified in the first slot.

Note: For more information, see "Copying a Symbolic Name to the Clipboard" on page 4-19.

This would display the value overlaid over the image near the top of the image display area. If Slot 5 had been used instead, it would appear closer to the center.

By default, the displayed format will be appropriate for the datum type requested. However, the format can be changed by specifying a printf style format string.

The format codes must be consistent with the expected data types. If the result is an integer, then a %d format is expected, floating point numbers require %f type formats. A boolean value is formatted as a string ("True" or "False"). Therefore, use the %s format when using a boolean. A detailed list of format codes is not documented here; please refer to printf documentation.

For array values, each element of the array will be passed in turn to the format string. For example, if a PointDm is being used, there are four expected array values corresponding to X, Y, angle, scale. (The order is the same as for variant access via VB). An example of using a format for PointDm:

(%.2f,%.2f) angle=%.1f scale=%.1f

This will display a result similar to:

(23.23,45.10) angle=3.2 scale=1.0

The later array values can be considered optional and can be omitted if desired. For example, to display just the x and y values of a PointDm, use the format string:

center = (%6.2f, %6.2f)

This will display a result similar to:

center = (134.22, 452.12)

If no format string is specified, an appropriate default format is used. For example, for a LineDm, by default the datum value will be displayed as:

A = value, B = value, C = value

Style

The default display of a report value is left justified, and uses a default font and color. If so desired, all visual aspects of the displayed report value can be modified. If the Style field is used, it has the format:

style:value,style:value,…

For example, to set the text size to 9pt, and align to the right, the following would be specified:

size:9pt,align:right

TABLE 4–1. Style Options

Option Name	Values	Default
align	Left, Right, Center	Left
color	Any named HTML color (red, blue, etc) or hexadecimal HTML color code (FF0000=red)	Yellow
CSS identifier	CSS values	
opacity	Number range 0 - 100 Setting this number to less than 100 will cause the displayed text to be translucent.	100
size	CSS text size values (examples: 3em, 9pt, 22px)	9pt

You can use the CSS identifiers to alter other display aspects. For example, the following will show a red background color for the text:

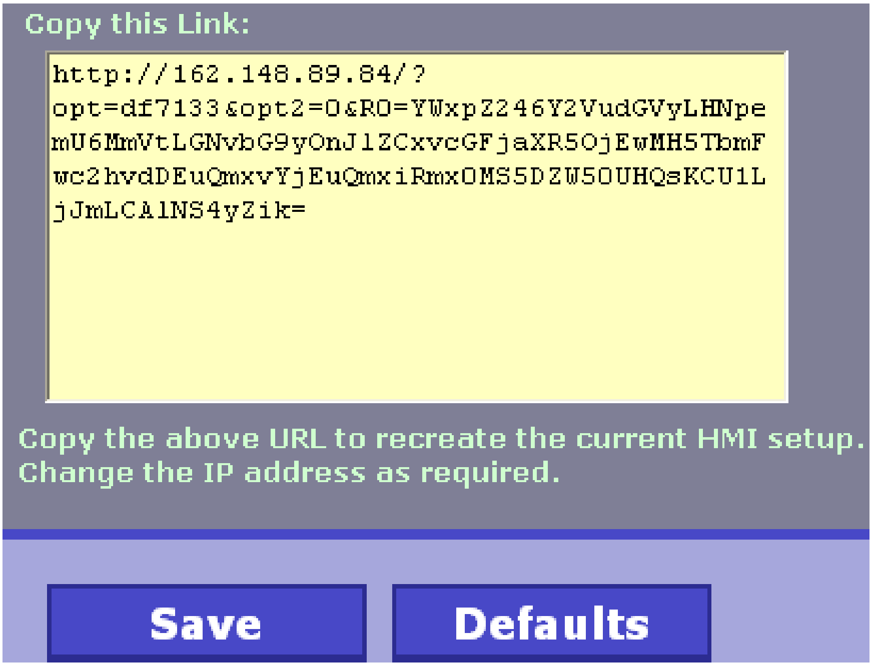
backgroundColor:red

To set some of the more common styles, the combo boxes for Style, Size, Color, and Opacity can be used. The styles field will be updated automatically.

URL Tab

Selecting the URL tab brings up the following display:

FIGURE 4–13. URL Display



The displayed URL can be copied and then used in a browser window to completely replicate the current setup.

Miscellaneous Points

- You can insert line breaks into format strings by using embedded HTML codes. To introduce a line break, use “<br />”.
- Commands and options are case sensitive. This is a limitation of javascript and CSS.

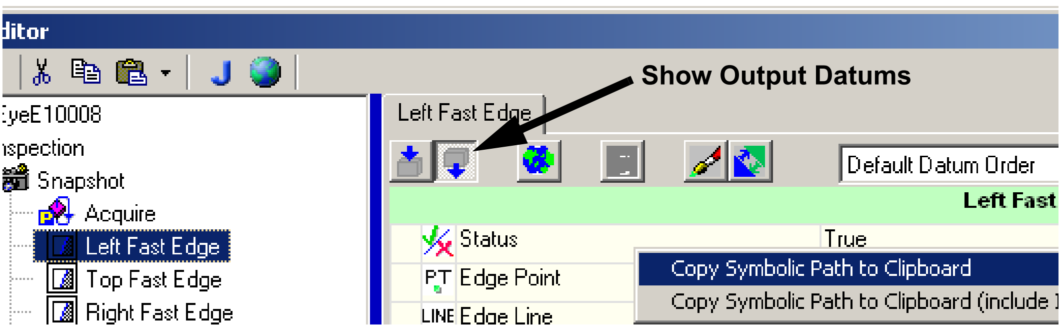
Displaying the Output of a Datum

Copying a Symbolic Name to the Clipboard

Use the following procedure to copy a symbolic name to the Clipboard:

1. In FrontRunner, stop your Job if it's still running.
2. Click Editor.
3. In the left pane, click on the tool that contains the datum(s) you want to copy to the Clipboard (Figure 4–14).
4. In the right pane, click Show Output Datums (Figure 4–14).
5. In the right pane, put the mouse pointer on the datum whose symbolic name you want to copy to the Clipboard (Figure 4–14).
6. Right click and select "Copy Symbolic Name to Clipboard":

FIGURE 4–14. Copy Symbolic Name to Clipboard



FrontRunner will display a message to the effect that "<symbol> was saved to the clipboard".

7. Click OK.
8. Close the Editor.
9. Restart your Job in FrontRunner.

Pasting a Symbolic Name into the Report

Use the following procedure to paste a symbolic name into the report:

1. Go to your web browser.
2. Click Settings. If the Settings button is hidden:
   a. Type the following:

      http://ip_address/?setopt=1

   b. Go to Layout > Buttons and click (to select) Show Settings Button.
3. Click Report.
4. Click on one of the slots.
5. Put the mouse pointer to the right of Path, right click and select Paste.
6. Press Enter.

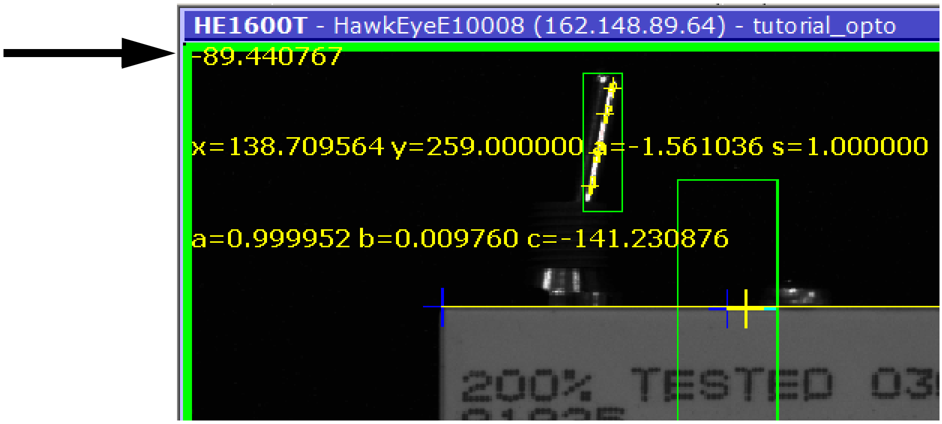
   You should see a screen similar to the one in Figure 4–15.

FIGURE 4–15. Symbolic Name of Datum Pasted into Slot 1

7. Click Close ("X") at the upper right corner.

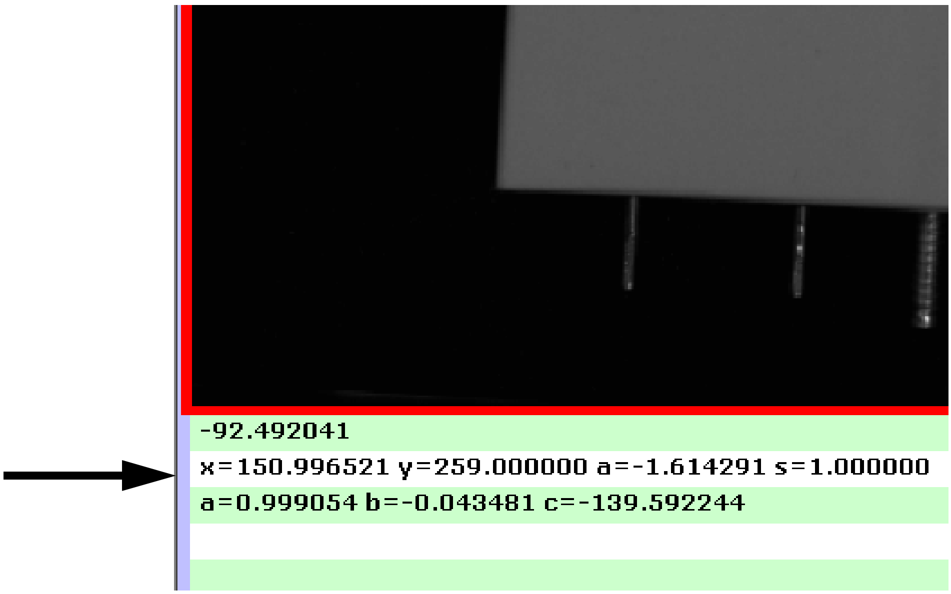
   You will see a screen similar to the following:

FIGURE 4–16. HMI Screen w/Datum Output Displayed Over Passed Image



If you've selected Show Report in List Format (under image), you will see a screen similar to the following:

FIGURE 4–17. HMI Screen w/Datum Output Displayed Below Failed Image



Adding Options to the Base URL

Specify an option by adding it to the end of the URL:

http://ip_address/?option=value

Note: Don't forget the "?" separating the URL from the optional parameter(s).

Specify additional options by separating them with the "&" character:

http://ip_address/?option1=value1&option2=value2&option3=value3

Basic Options

Note: Some basic options can be changed by specifying optional values at the end of the URL. A much richer superset of these options can be configured by using the Settings Pages. It is possible to completely control the behavior of the Runtime Page without the use of optional parameters in the URL.

You can turn on/off the graphics overlay using the "graphics" URL option. This is a setting that can have the value "on" or "off". For example, to turn the display of graphics off, launch the web page with the following URL:

http:// 161.218.121.58/?graphics=off

TABLE 4–2. Basic Options

Option	Description
graphics	On = (Default) Graphics are shown overlaid on the image. (Note: Not all graphics are supported for web page display.) Off = Graphics are not shown.
passfailborder	On = (Default) Displays a border around the image. Off = Displays no border.

5

CHAPTER 5

Optics and Lighting

This chapter contains information specific to the Optics and Lighting options for the VS-1 Smart Camera.

FIGURE 5–1. **VS-1 Smart Camera**

Optics (1610T Only)

The VS-1 Smart Camera uses C-Mount lenses. Table 5–1 contains lens sizes and Microscan part numbers.

TABLE 5–1. Lens Sizes and Microscan Part Numbers

Part Number	Size
98-92800571	Lens: 8.5mm
98-92800572	Lens: 12mm
98-92800573	Lens: 16mm
98-92800574	Lens: 25mm
98-92800575	Lens: 35mm
98-92800576	Lens: 50mm
98-92800577	Lens: 75mm

Table 5–2 contains working distances and fields of view with various lenses.

TABLE 5–2. Working Distance and Fields of View1

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)	0.0	108 203	91 248	81 305	38 234	36 328	36 495	28 582
Horiz FOV (mm) Camera Face Distance (mm)	0.5	41 94	44 140	51 197	28 201	43 297	33 460	26 561
Horiz FOV (mm) Camera Face Distance (mm)	1.0	28 74	34 107	38 157	29 193	29 282	32 439	25 546
Horiz FOV (mm) Camera Face Distance (mm)	5.0		9 56[2]	13 75	14 118	17 197	21 333	20 461
Horiz FOV (mm) Camera Face Distance (mm)	10.0			7 61	9 94	11 159	14 269	15 395
Horiz FOV (mm) Camera Face Distance (mm)	15.0				6 87	9 144	11 232	13 357

Notes:

1. Lens focus ring adjusted to closest position.
2. 20mm or less working distance in front of lens.

Lighting Connector

For complete information about the Light Port Connector, see “Light Port Connector” on page A-5.

A

APPENDIX A

Connector Pinouts

This appendix contains information about the VS-1 Smart Camera connectors:

VS-1 Smart Camera Connectors

Power and Primary I/O Connector

Figure A–1 shows the pinout of the Power and Primary I/O connector.

FIGURE A–1. **Power and Primary I/O Connector “X1” - Rear View**

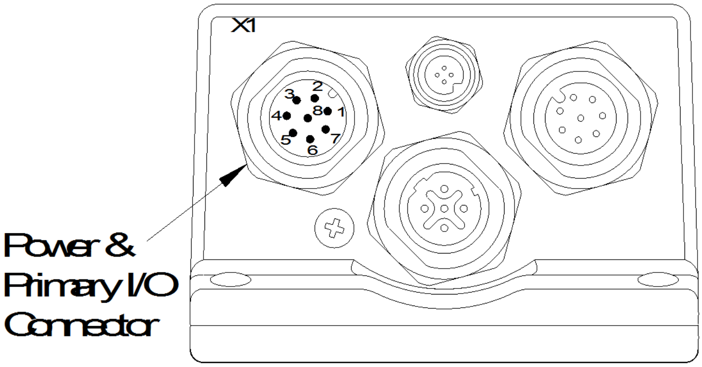


Table A–1 lists the supplier for the Power and Primary I/O connector mating connector and cable.

TABLE A–1. **Power and Primary I/O Connector Mating Cable**

Supplier	Part Number	Description
Microscan	98-HT00-0CP0	Connector Type M12 8 Pin A Coded, Female Cable 10M

Table A–2 describes the Power and Primary I/O connector signals.

TABLE A–2. **Power and Primary I/O Connector Signals**

Pin	Short Designation	Direction	Function	Notes
1	OUT1	Out	S/W Programmable	3
2	OUT2	Out	S/W Programmable	3
3	OUT3	Out	S/W Programmable	3
4	Power Input	In	Fused +24V, 1 Amp max.	1
5	Out Common		Output Common	4
6	SENSOR	In	S/W Programmable Trigger	2
7	SENSOR	In	S/W Programmable Trigger	2
8	Power Return	Common	DC Ground	1
Shell				

Notes:

1. Non-isolated utility power for sensor and/or opto current loops.
2. Bipolar isolated camera sensor trigger:
   – Logic 0: 0 to 5V
   – Logic 1: 15 to 30V
   – Rin: 18k Ohms
3. Bipolar isolated output switch, Ron = 35 ohm max., Ion < 50ma.,
   Voff < ±50VDC, 60VAC isolation (short circuit protected: 160-240 ma pk.).
4. Common wire for Out1, Out2, and Out3 Opto-Isolated outputs.
5. Total output current for all opto outputs must not exceed 150ma.

Serial and Secondary I/O Connector

Figure A–2 shows the pinout of the Serial and Secondary I/O connector.

FIGURE A–2. Serial and Secondary I/O Connector Pinout "X3" - Rear View

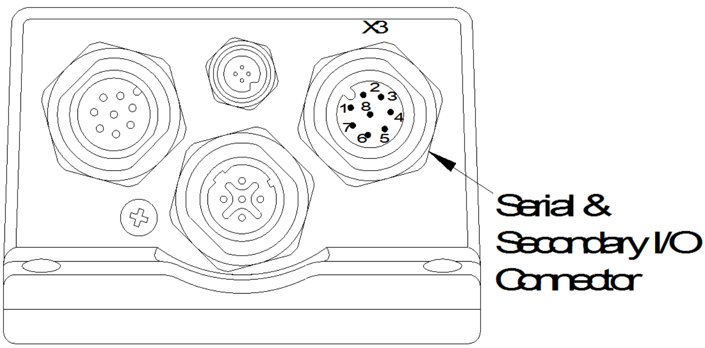


Table A–3 lists the supplier for the Serial and Secondary I/O connector mating connector and cable.

TABLE A–3. Serial and Secondary I/O Connector Mating Cable

Supplier	Part Number	Description
Microscan	98-HT00-0CS0	Connector Type M12 8 Pin A Coded, Male Cable 10M

Table A–4 describes the Serial and Secondary I/O connector signals.

TABLE A–4. Serial and Secondary I/O Connector Signals

Pin	Signal Name	Direction	H/W Description	Notes
1	GPIO 3	In/Out	S/W Programmable General Purpose I/O	1,2
2	GPIO 4	In/Out	S/W Programmable General Purpose I/O	1,2
3	Td	Out	RS-232 Transmit	3
4	Rd	In	RS-232 Receive	4
5	GPIO 1	In/Out	S/W Programmable General Purpose I/O	1,2
6	GPIO 2	In/Out	S/W Programmable General Purpose I/O	1,2
7	GPIO Out Com		GPIO Output Common	5
8	GPIO In Com		GPIO Input Common	6
Shell	Chassis Ground		RS-232 Common	

Notes:

1. Bipolar isolated current input 5-24V, 1-5ma., 60 VAC isolation:
   - Logic 0: 0 to 5V with respect to GPIO In Com
   - Logic 1: 15 to 30V with respect to GPIO In Com
   - Rin: 18k Ohms with respect to GPIO In Com
2. Bipolar isolated output switch, Ron = 35 ohm max., Ion < 20ma., Voff < µ±30VDC, 60VAC isolation (short circuit protected: 160-240 ma pk.) with respect to GPIO Out Com.
3. Serial communication Transmit signal (RS-232)
4. Serial communication Receive signal (RS-232)
5. Common wire for General Purpose I/O opto-isolated outputs.
6. Common wire for General Purpose I/O opto-isolated inputs.

Ethernet Connector

Figure A–3 shows the pinout of the Ethernet connector.

FIGURE A–3. Ethernet Connector Pinout "X4" – Rear View

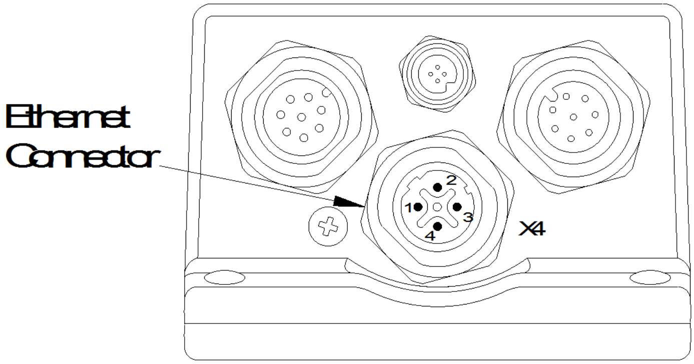


Table A–5 lists the supplier for the Ethernet connector mating connector and cable.

TABLE A–5. Ethernet Connector Mating Cable

Supplier	Part Number	Description
Microscan	98-HT00-0CE0	Connector Type M12 4 Pin D-Coded, Male Cable 10M (Unshielded)
	98-HT00-0CE2	Connector Type M12 4 Pin D-Coded, Male Cable 2M (Unshielded)

Table A–6 describes the Ethernet connector signals.

TABLE A–6. Ethernet I/O Connector Signals

Pin	Signal Name
1	Transmit+ (Td+)
2	Receive+ (Rd+)
3	Transmit- (Td-)
4	Receive- (Rd-)

Light Port Connector

Figure A–4 shows the pinout of the Light Port connector.

FIGURE A–4. **Light Port Connector Pinout "X2" – Rear View**

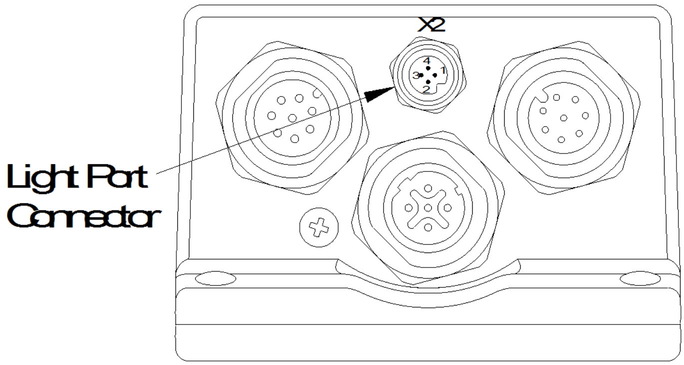


Table A–7 lists the supplier for the Light Port connector and cable.

TABLE A–7. **Light Port Connector Mating Cable**

Supplier	Part Number	Description
Microscan	98-HT00-0CL5	Connector Type M5 4 Pin, Male Cable 5M

Table A–8 describes the Light Port connector signals.

TABLE A–8. **Light Port Connector Signals**

Pin	Signal Name	State	Voltage	Duration	Notes
1	Light Output	On/Strobe	12V	Infinity/64us -> 58.8ms	1
		Power Strobe	24V	64us -> 1ms Maximum	2
2	Light Output Return				
3	Strobe Output	Strobe	5V	64us -> 58.8ms (VGA) 86us -> 58.8ms (XGA)	
4	Strobe Output Return				

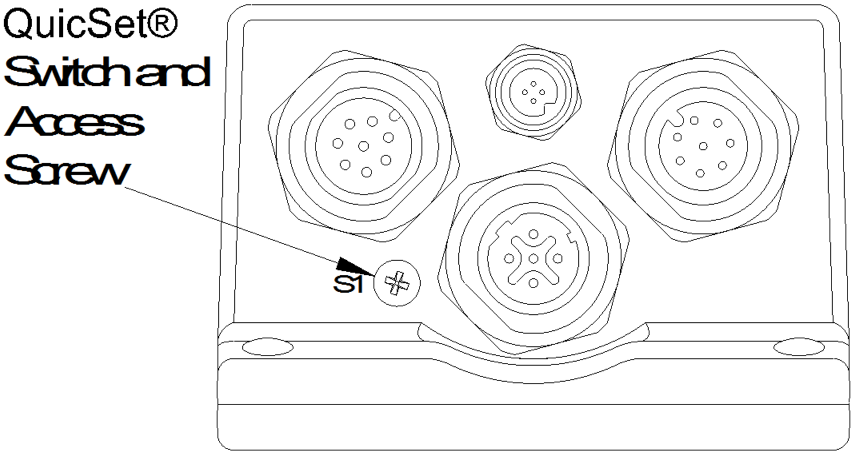
Notes:
1. 12V @ 400ma Maximum
2. 5V Current sourcing pulse, (20ma short circuit current)

QuicSet® Switch

Note: On earlier production units only. The QuicSet function is not used on the VS-1 Smart Camera.

Figure A–5 shows the location of the QuicSet switch and access screw.

FIGURE A–5. QuicSet® Switch and Access Screw "S1" – Rear View



The QuicSet switch is located beneath the access screw, as shown at location S1 in Figure A–5. To access the QuicSet Switch, remove the sealing screw with a Phillips screwdriver. To maintain the IP rating of the chassis, the sealing screw must be installed and tightened to 0.6 Nm. If IP rating is not critical to your application, hand tighten snug. Do not overtighten!

APPENDIX B

Cable Specifications

This appendix contains information about the VS-1 Smart Camera cables.

Note: Cable specifications are published for information only. We do not guarantee the performance or quality of cables provided by other suppliers.

TABLE B–1. **Cable Part Numbers and Descriptions**

Catalog Number	Part Number	Descriptions
98-HT00-0CP0	HETPC-100	Power and Primary I/O Cable 10 Meter
98-0HT00-0CS0	HETAC-100	Serial and Secondary I/O Cable 10 Meter
98-0HT00-0CE2	HETENET-020	Ethernet 2 Meter
98-HT00-0CE0	HETENET-100	Ethernet 10 Meter
98-HT00-0CL5	HETLC-050	Lighting Control Cable 5 Meter

HETPC-100 - Power and Primary I/O Cable

The HETPC-100 Power and Primary I/O cable is a single-ended shielded cable with an M12 x 8 Pin A Coded male connector on one end. Figure B–1 shows the wiring for the HETPC-100 Power and Primary I/O cable.

FIGURE B–1. Power and Primary I/O Cable

Table B–2 describes the signals for the HETPC-100 Power and Primary I/O cable.

TABLE B–2. HETPC-100 Power and Primary I/O Cable

M12 Pin	Signal Name	Wire Color	Description
1	Opto 1 Out	White	Out1
2	Opto 2 Out	Brown	Out2
3	Opto 3 Out	Green	Out3
4	Camera Power	Yellow	Fused +24V, 1 Amp max.
5	Opto Out common	Grey	Output Common
6	Opto In B	Pink	Trigger in B
7	Opto In A	Blue	Trigger in A
8	Sensor Power return	Red	DC Ground
Shell	Chassis Ground		

HETAC-100 - Serial and Secondary I/O Cable

The HETAC-100 Serial and Secondary I/O cable is a single-ended shielded cable with an M12 x 8 Pin A Coded female connector on one end. Figure B–2 shows the wiring for the HETAC-100 Serial and Secondary I/O cable.

FIGURE B–2. HETAC-100 Serial and Secondary I/O Cable

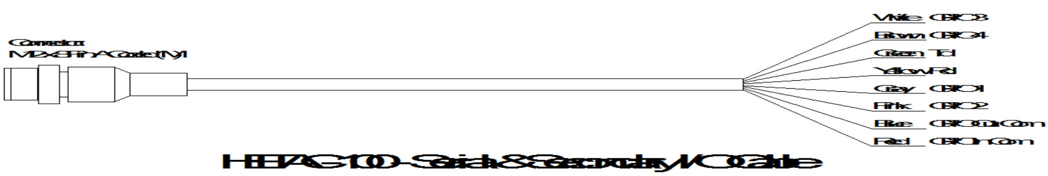


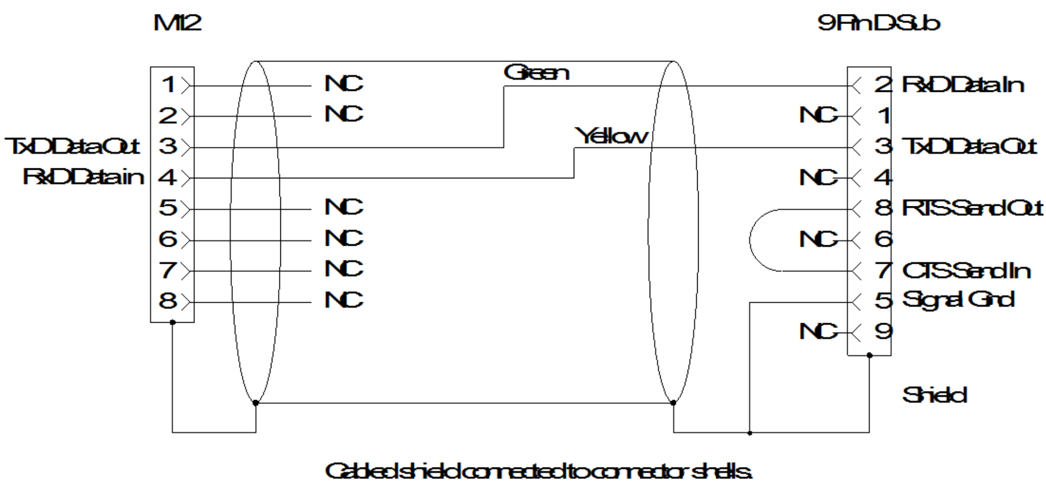
Table B–3 describes the signals for the HETAC-100 Serial and Secondary I/O cable.

TABLE B–3. HETAC-100 Serial and Secondary I/O Cable Signals

M12 Pin	Signal Name	Wire Color	Description
1	GPIO 3	White	General Purpose I/O 3
2	GPIO 4	Brown	General Purpose I/O 4
3	Td	Green	RS-232 Transmit
4	Rd	Yellow	RS-232 Receive
5	GPIO 1	Grey	General Purpose I/O 1
6	GPIO 2	Pink	General Purpose I/O 2
7	GPIO Out Com	Blue	GPIO Output Common
8	GPIO In Com	Red	GPIO Input Common
Shell	Chassis Ground		RS-232 Common

Figure B–3 shows how to connect the HETAC-100 Serial and Secondary I/O cable to a Serial Port 9 Pin D-Sub serial port connector.

FIGURE B–3. Connecting HETAC-100 Serial Cable to Serial Port Connector



HETENET-XXX - Ethernet Cable

The Ethernet cable has an M12 x 4 Pin D Coded male connector on one end, and a male RJ-45 connector on the other end. Figure B–4 shows the HETENET-XXX Ethernet cable.

FIGURE B–4. HETENET-XXX Ethernet Cable

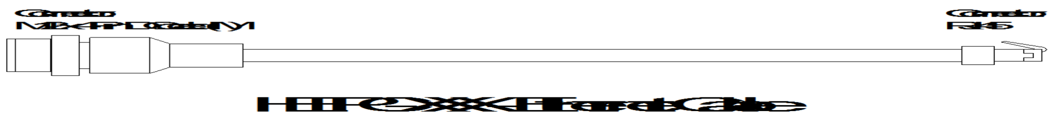


Table B–4 lists the available lengths for the HETENET-XXX Ethernet cables.

TABLE B–4. HETENET-XXX Ethernet Cable Lengths

Catalog Number	Part Number	Length
98-HT00-0CE2	HETENET-020	2 Meter
98-HT00-0CE0	HETENET-100	10 Meter

Table B–5 describes the signals for the HETENET-XXX Ethernet cables.

TABLE B–5. HETENET-XXX Ethernet Cable Signals

M12 Pin	RJ-45	Signal Name	Wire Color	Description
1	1	Td+	White/Orange	Transmit+
3	2	Td-	Orange	Transmit-
2	3	Rd+	White/Green	Receive+
4	6	Rd-	Green	Receive-

HETLC-050 - Lighting Control Cable

The Lighting Control cable is a single-ended cable with an M5 x 4 Pin A Coded male connector on one end. Figure B–5 shows the HETLC-050 Lighting Control cable.

FIGURE B–5. HETLC-050 Lighting Control Cable

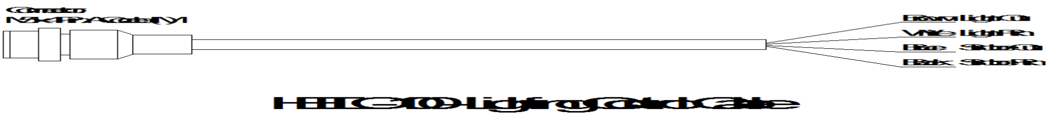


Table B–6 describes the signals for the HETLC-050 Lighting control cable.

TABLE B–6. HETLC-050 Lighting Control Cable Signals

M5 Pin	Signal Name	Wire Color	Description
1	Light_Out	Brown	Light Output
2	Light_Rtn	White	Light Output Return
3	Strobe_Out	Blue	Strobe Output
4	Strobe_Rtn	Black	Strobe Output Return

APPENDIX C Specifications

This appendix contains specifications and dimensions for the VS-1 Smart Camera and its mounting blocks.

TABLE C–1. Specifications

	HE1600TS HE1610TIS	HE1610TH HE1610TIH
Height	45.50mm (1.791")	
Width	65.00mm (2.559")	
Depth	128.40mm (5.055")	
Weight	0.82kg (1.85 lbs)	
Power	20.4 to 28.8 volts 24 volts @ 250ma Nominal (not including external lighting)	
Lighting	Support for various external lighting options	
Lens Type	C Mount	
Communications	Ethernet; RS232, Baud rates from 1200bps to 115.2 Kbps	
I/O	1 Opto-isolated input trigger 3 Opto-isolated outputs 4 Opto-isolated general purpose inputs/outputs 1 TTL level strobe output	
Humidity	10% to 95%, non-condensing Operating and Storage	
Protection	IP67 (with optional lens tube)	
Processor	High Performance RISC Processor	
Memory	16MB non-volatile memory, 64MB SDRAM 32MB non-volatile memory, 128MB SDRAM	
Acquisition	Progressive Scan, full and partial frame capture	

TABLE C–1. Specifications (Continued)

	HE1600TS HE1610TIS	HE1610TH HE1610TIH
Sensor	1/3 Inch CCD	
Cell Size	7.4μm (H) x 7.4μm (V)	4.65μm (H) x 4.65μm (V)
Image Resolution	648 x 494 pixels, 256 grey levels (8 bits per pixel)	1024 x 768 pixels, 256 grey levels (8 bits per pixel)
Electronic Shutter	64μs - 50ms	86μs - 50ms
Operating Temperature	0°C to 50°C (32°F to 122°F)	
Storage Temperature	-20°C to 70°C (-4°F to 158°F)	
Compliance	CE, FCC, RoHS	
Certifications	EN63126:1997 +A1 +A2 +A3 EN61000-6-2:2001 EN61000-6-2:2006 EN61000-6-4:2001 EN61010-1:2001-02 FCC Part 15, Subpart B, Class A	

Dimensions

FIGURE C–1. **VS-1 Smart Camera Dimensions**

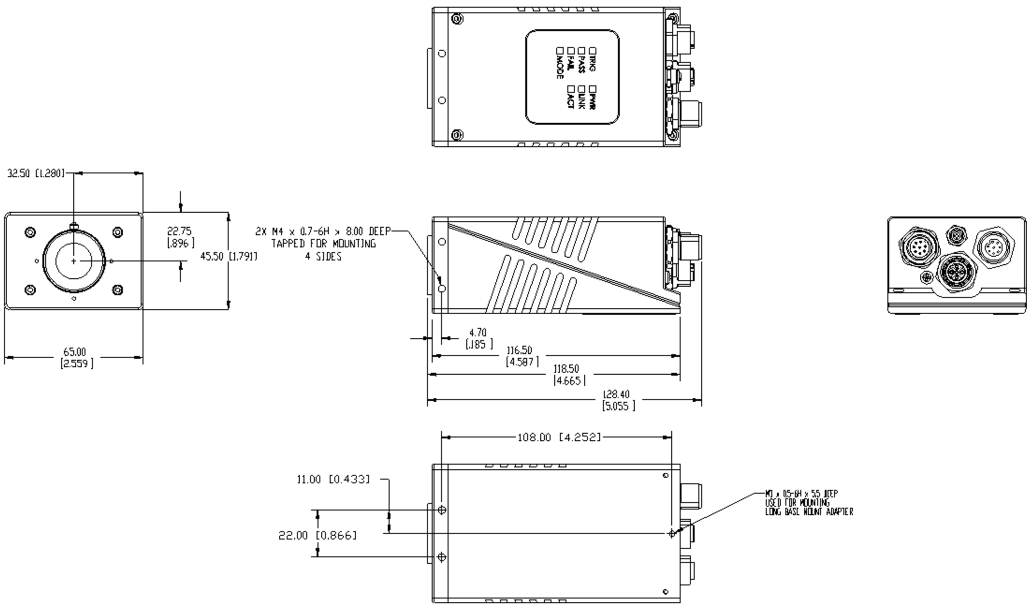


FIGURE C–2. VS-1 Smart Camera Lens Protector Dimensions

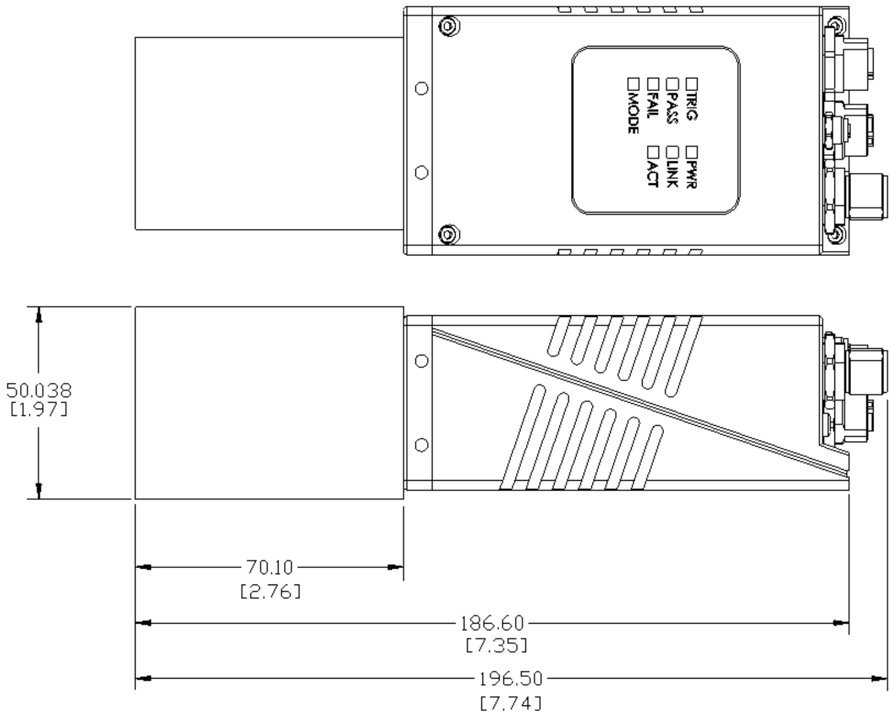


TABLE C–2. Lens Protector Part Numbers and Descriptions

Catalog #	Part #	Descriptions
98-HT00-0TA1		Lens Protection Housing with Glass screen. IP65 Rated
98-HT00-0TA2	HELTA-050	Lens Protection Housing with Plexiglas screen. IP65 Rated
98-HT00-0TA0	A1-40366-1	Lens Protection Adapter

FIGURE C–3. Standard Mounting Block Dimensions

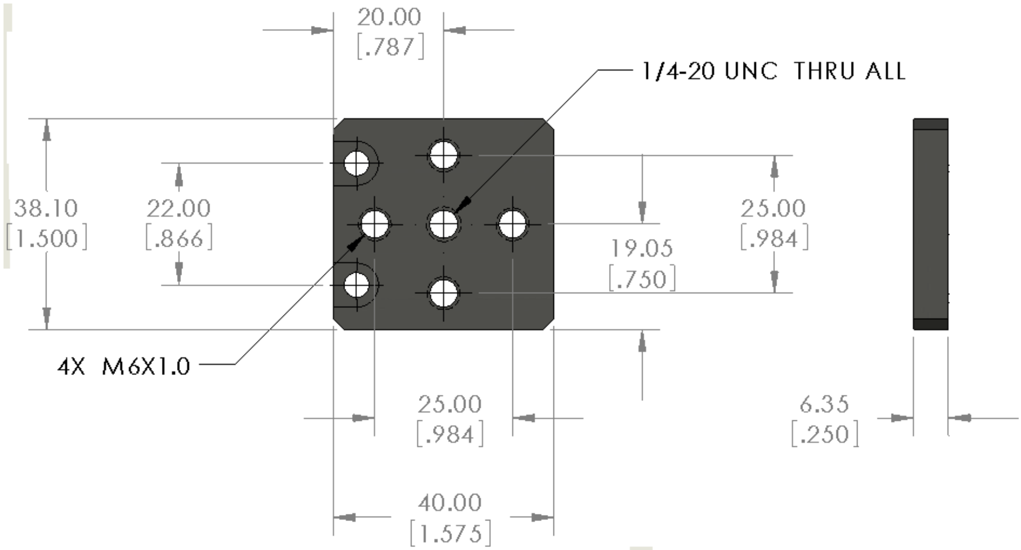


FIGURE C–4. Backward Compatible Mounting Block Dimensions

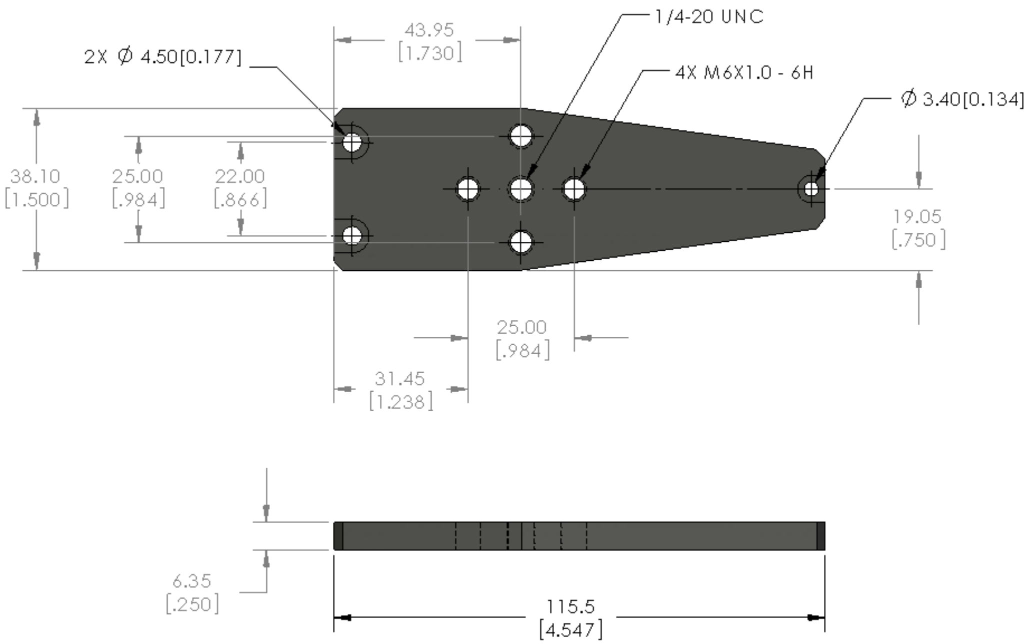
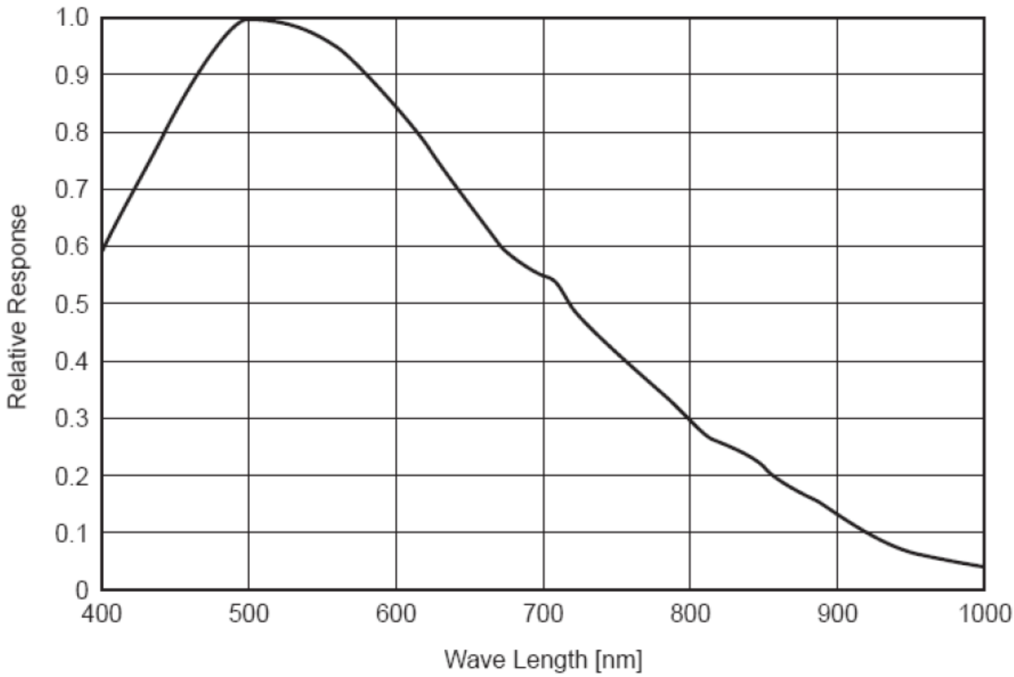
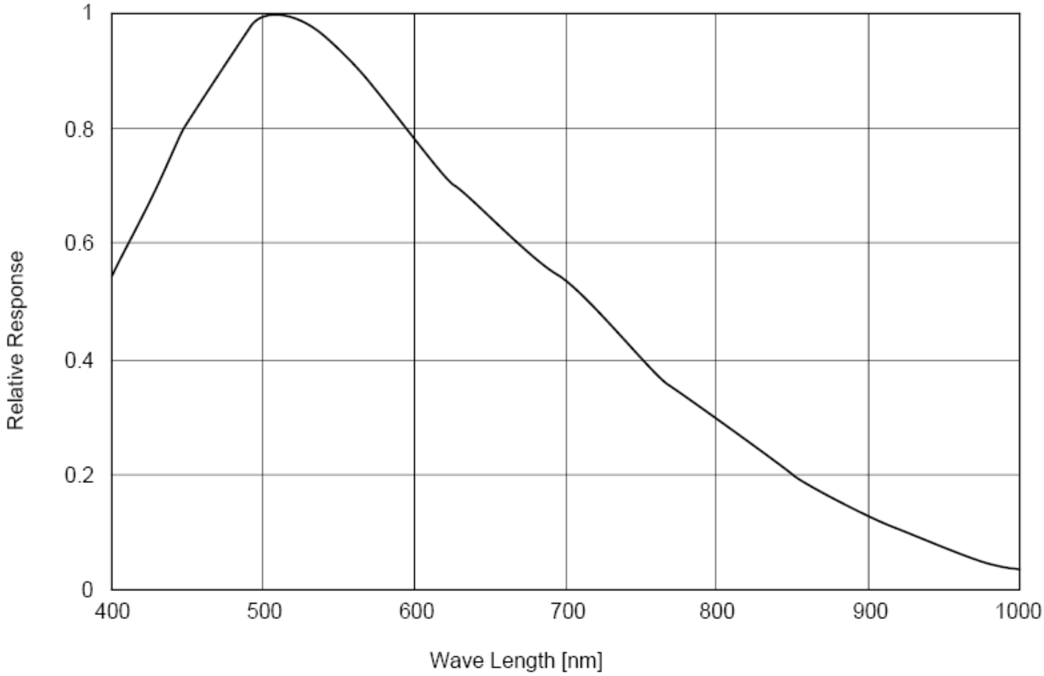


FIGURE C–5. Spectral Sensitivity Characteristics (VGA)



Note: Excludes lens characteristics and light source characteristics.

FIGURE C–6. Spectral Sensitivity Characteristics (XGA)



Note: Excludes lens characteristics and light source characteristics.

D

APPENDIX D Setting Up Network Communications

By default, the VS-1 Smart Camera utilizes Dynamic Host Configuration Protocol (RFC2131) for dynamic IP addressing. You can also configure the camera to use a "static" IP address. Please contact your MIS department to determine if your network uses DHCP addressing. If your network uses static IP addressing, contact your MIS department to obtain a unique static IP address for the camera before proceeding.

Note: Important: Although the use of DHCP is convenient when initially setting up the camera, it should not be used when deployed. In a production environment, all networked components should use static IP addresses to prevent communication issues related to DHCP renewals.

You can change network parameters using the Visionscape® Network Browser (Start > Visionscape > Tools > Visionscape Network Browser) once the camera is connected via Ethernet (on a subnet or using a crosslink cable).

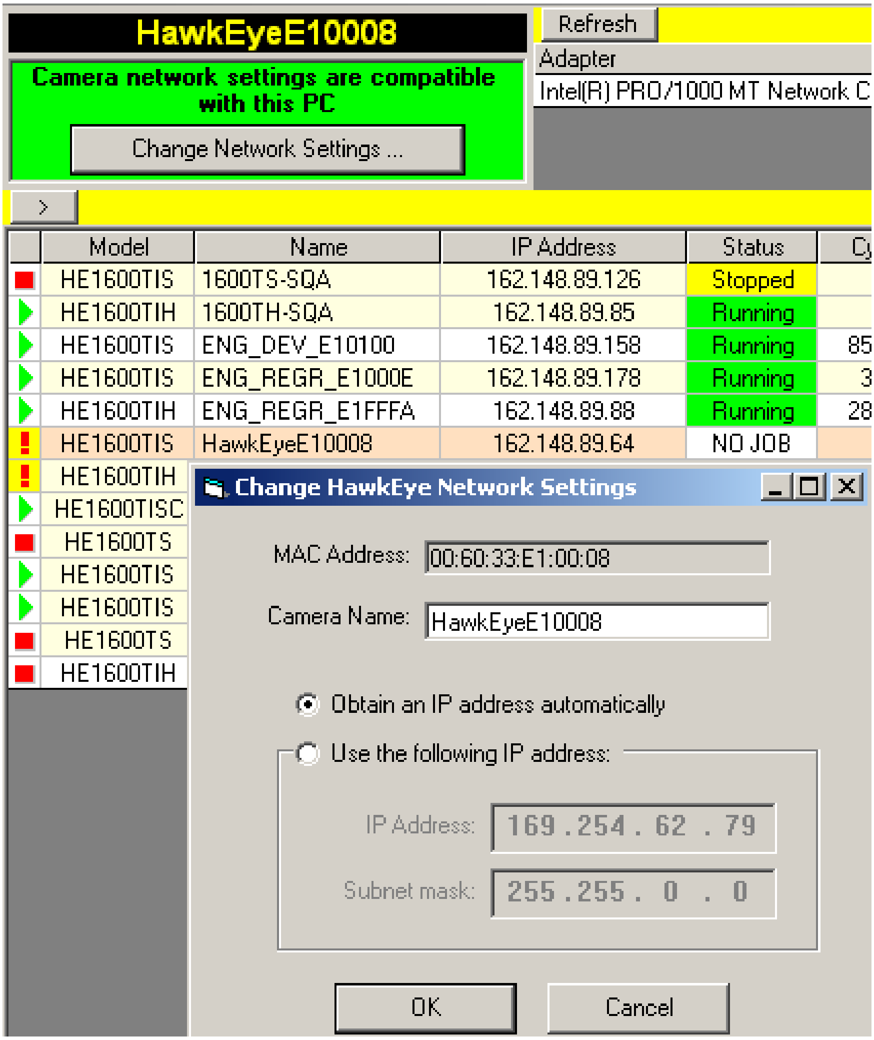
1. Select a camera in the Network view and click Change Network Settings.

   You will need to take control of the camera before proceeding to the Network Settings dialog; the default username is "hawkeye" and the default password is "vision".

   This is the standard and most straight forward method of setting the network parameters on the camera as it does not require a serial connection.

2. Select DHCP (“Obtain an IP address automatically”) or Static IP (“Use the following address”) and fill in a logical address and network mask, as shown in Figure D–1. Also, see Table D–1.

FIGURE D–1. Network Overview and Network Settings



3. When you are done, click OK. The camera will be rebooted automatically for the changes to take effect.

Note: All the Network parameters of the camera are stored in Non-Volatile memory and can also be modified from the Bootloader (a low-level diagnostics program) that can be invoked using HyperTerminal and a Serial connection. This is described in Appendix E, "Updating Firmware on VS-1 Smart Cameras," as many unique and useful parameters can only be changed using this method.

VS-1 Smart Camera Connection Matrix

Table D–1 describes the various options such that the Camera and PC can communicate fully over TCP/IP.

TABLE D–1. IP Addresses

Camera	Host PC	DHCP on LAN?	Action	Connection
DHCP	DHCP	Yes	Both get IP addresses from Server	Yes
DHCP	DHCP	No	Camera and Host may fall over to compatible IP Addresses (169.254.x.x) (255.255.0.0) (Automatic Private IP Addressing (APIPA))	Maybe. Not all PCs will default to (169.254.x.x). Some will default to (0.0.0.0) – set static IP on PC to APIPA address (169.254.x.x 255.255.0.0).
Static	Static	N/A	Camera and Host must be set to compatible static IP address. Default static IP for camera (as delivered) is: 192.168.254.3. Set PC to (for example) 192.168.254.2. Both on (255.255.255.0) subnet.	Yes if IP Addresses are compatible.

To check a camera IP address, simply select the camera in FrontRunner Network Overview.

To check a host PC IP address1:

1. At the DOS prompt, type:

   ipconfig

VS-1 Smart Camera Boot Parameters

The VS-1 Smart Camera stores network and other parameters in Non-Volatile memory. You can edit these by connecting a Serial cable to the camera and HyperTerminal. The communication parameters must be set to 115200-No Parity-8-1-No Flow Control.

Note: *The VS-1 Smart Camera parity parameter must always be set to 'No Parity'.*

Changing Network Parameters

To change network parameters:

1. Start a HyperTerminal session with the communication parameters set to 115200-No Parity-8-1-No Flow Control.

2. While you hold down the Escape key, cycle power on the VS-1 Smart Camera. Information similar to the following is displayed:

```
HawkEye Smart Camera
400MHz 16MB Flash 64MB RAM
-----------------------------
Name: HawkEyeE1000E
Static IP: 161.218.121.183
Network Mask: 255.255.255.0
Gateway: 0.0.0.0
DHCP: 1
MAC Address: 00:60:33:e1:12:34
1. Modify User Parameters
2. Reset User Account
3. Exit to Application
>>
```

3. Type *1* to begin stepping through network parameters.

4. For each parameter that is displayed, simply type in new information (to change the parameter), or press the Return key to display the next parameter. Eventually, you'll see the message "Save parameters to flash? [y/n] -

5. Type y and press the Return key. You're finished changing network parameters.

E

APPENDIX E

Updating Firmware on VS-1 Smart Cameras

This appendix contains two procedures for updating your Smart Camera:

- "Updating the Firmware with Smart Camera Update" on page E-1
- "Updating the Firmware with the Bootloader" on page E-3

Updating the Firmware with Smart Camera Update

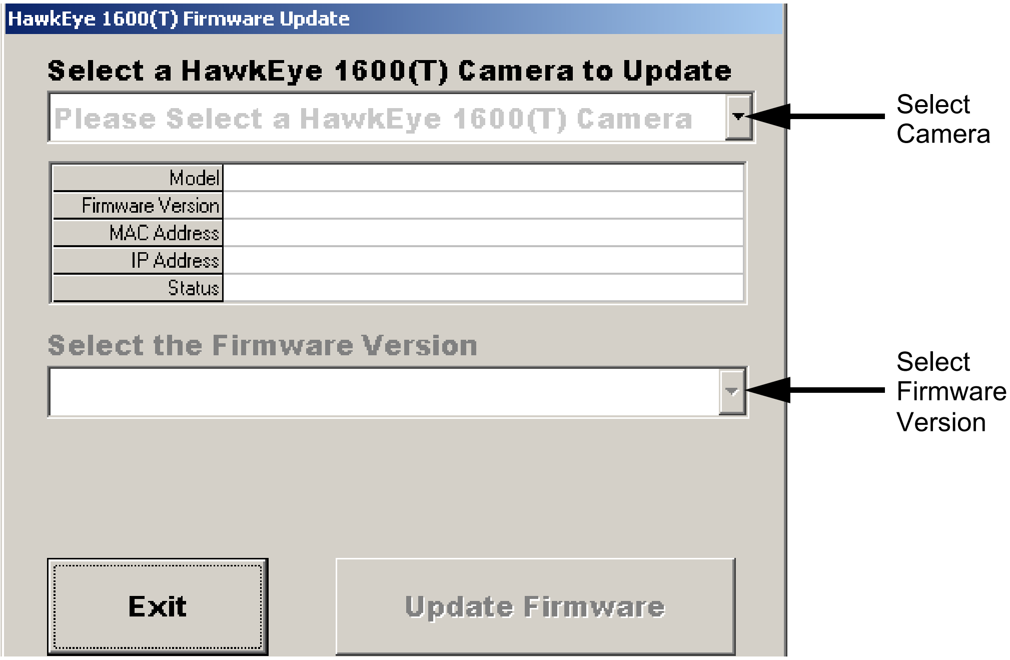
Note: Do not cycle power on the Smart Camera during this procedure.

Use the following procedure to update the firmware on your VS-1 Smart Camera with the Smart Camera Update utility:

1. Start the Smart Camera Update utility by selecting Start > Visionscape > Tools > HawkEye 1600(T) Firmware Update.

   The Smart Camera Update utility displays its main screen, as shown in Figure E–1.

FIGURE E–1. Smart Camera Update Main Screen



2. In the top pull-down menu, select the Smart Camera you want to update.
3. In the bottom pull-down menu, select the desired firmware version.
4. Enter the user name (default is hawkeye) and password (default is vision) for the Smart Camera
5. Click Update Firmware.

   You see the progress of the Update procedure as a set of stages and progress bars. After completion, either of the following is displayed:

   – A success message indicating that your Smart Camera has been successfully updated

   – A failure message with the reason for the failure and what stage of the Update failed
6. Click Exit.

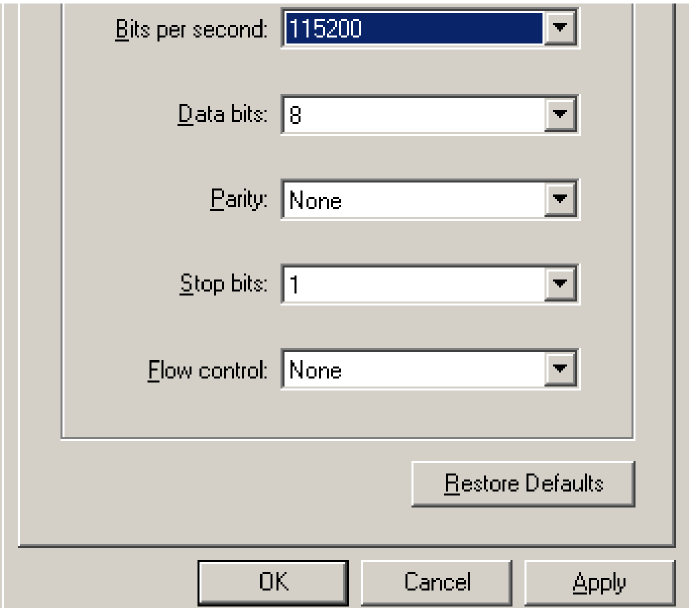
Updating the Firmware with the Bootloader

If power is lost while the Smart Camera Update is in progress, the Smart Camera will not boot when power is restored. When this occurs, you will need to use the Bootloader application in Non-Volatile memory (which is not erased by the Smart Camera Update utility) to update the firmware.

Use the following procedure to update the firmware on your VS-1 Smart Camera with the Bootloader:

1. Start a HyperTerminal session with the following settings:

FIGURE E–2. **COM1 Settings**



2. Click OK.

3. While you hold down the Escape key, cycle power on the VS-1 Smart Camera. Information similar to the following is displayed:

```
HawkEye Smart Camera
400MHz 16MB Flash 64MB RAM
-----------------------------
Name: HawkEyeE1000E
Static IP: 161.218.121.183
```

```
Network Mask: 255.255.255.0
Gateway: 0.0.0.0
DHCP: 0
MAC Address: 00:60:33:e1:12:34
1. Modify User Parameters
2. Reset User Account
3. Exit to Application
>>
```

4. Type *adv* and press the Enter key.

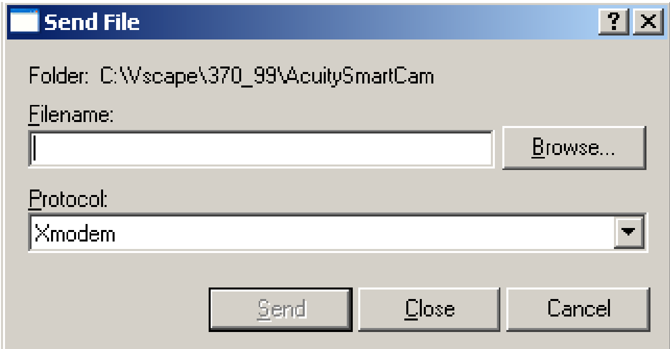
   The advanced menu is displayed:

```
Advanced Options Menu
Diag Monitor V 1.01.00
------------------------------
dm          Display Menu
.
.
.
x           File Transfer
.
.
.
help        Help
>>
```

5. Type *x* and press the Enter key.
6. From the Transfer menu, select Send File....

   The Send File dialog box is displayed, as shown in Figure E–3.

FIGURE E–3. Send File Dialog Box



7. Click Browse, and navigate to C:\Vscape\373_xx\AcuitySmartCam.

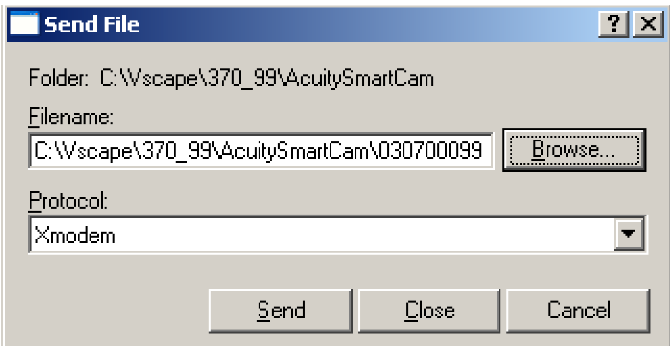
8. Highlight filename.SEC.

9. Click Open.

10. In the Protocol: drop-down menu, select Xmodem.

    After you click Open, your screen should look similar to one in Figure E–4:

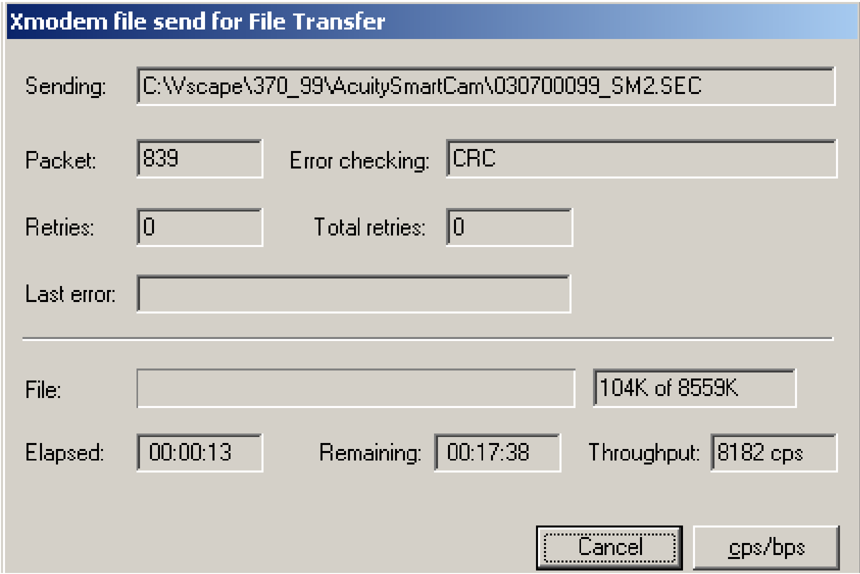
FIGURE E–4. Send File Dialog Box After Filename and Protocol Chosen



11. Click Send.

    You should see an Xmodem File Send screen similar to the one in Figure E–5:

FIGURE E–5. Xmodem File Send in Progress



When the file transfer terminates, you will see a screen similar to the one displayed in Figure E–6.

FIGURE E–6. Xmodem File Send in Progress

```
  spkr  Speaker Utility
  leds  LED Utility
  cache Cache Toggle
  cpu   CPU Clocks/Regs
  phy   Dump PHY Regs
  ftp   FTP Image
  x     File Transfer
  ver   Show Version Info
  r     Reboot
  j     Jump Program Counter
  q     Quit
  hist  History Queue
  !     Run History Queue
  !!    Run Last Command
  help  Help
>> x

Transfer file and execute...
CCCCCCCCCCC


Transferred 8764032 bytes -- running silent until execution completes!
Execution complete!
>>
```

12. Type *q* and press the Enter key to exit the advanced menu and return to the main menu.

13. Type *3* and press the Enter key to exit the main menu.

You're finished. You can exit HyperTerminal.

Main Menu Items

Modify User Parameters

This utility allows you to change network configuration values. You are prompted to enter each parameter one at a time. To enable DHCP, for example, the sequence is as follows:

```
>> 1
Default network name: HawkEye™ 1600T_Test -
Static IP: 161.218.121.183 -
Subnet mask: 255.255.255.0 -
Gateway: 0.0.0.0 -
```

```
Use DHCP: 0 - 1
Save parameters to flash? [y/n] - y

Saving boot parameters to flash……done.
```

- Network Name — This is limited to 19 characters. Any input string longer than 19 characters is truncated automatically to the first 19 characters. The string is erased by entering the '.' (period) character.
- Static IP, Subnet Mask, Gateway — These must contain an address in valid IP format. If you enter an invalid address, you are prompted to re-enter the value until a valid address is entered or the Enter key is pressed (effectively skipping the update for this field).
- DHCP — The value for DHCP value is either a 0 (zero) or 1 (one).
- MAC Address — You cannot change the MAC Address, since this is set in the factory.

Reset User Account

Each VS-1 Smart Camera is protected by a User/Password combination where the password is encrypted using the unit's MAC address. The Reset User Account utility allows you to reset the User/Password combination to the default values of:

User: hawkeye
Password: vision

Resetting the account is password protected with the string *vision*.

```
>> 2
Enter password to reset User Account to defaults: ******
```

Exit to Application

The Exit to Application selection exits the Main Menu and switches execution to the beginning of application code stored in Non-Volatile memory. If there is no valid application program in Non-Volatile memory, the Main Menu is re-displayed.

Error Codes

When the Smart Camera fails to boot (i.e., self tests fail), the beeper beeps continuously and the LEDs on the front of the unit flash the error code in binary form. Table E–1 describes the Mode/Status LED power-on sequence and error codes.

TABLE E–1. Mode/Status LED Power-On Sequence and Error Codes

Mode	Fail	Pass	Trig	Test Performed
			•	Data Line Test
		•		Address Line Test
		•	•	Bootloader CRC Check
	•			Kernel CRC Check
	•		•	RS-232 Internal Loopback
	•	•		FPGA Load Test
	•	•	•	FPGA PCI Config Test
•				FPGA Video Buffer Test
•			•	FPGA DMA Transfer Test
•		•		FPGA Expose Done Interrupt Test
•		•	•	FPGA Read Done Interrupt Test

F

APPENDIX F

Ethernet/IP Communication with ControlLogix PLCs

This appendix describes how to connect an Allen-Bradley ControlLogix PLC to the VS-1 Smart Camera. You should have a working knowledge of TCP/IP, the CIP protocol (Ethernet/IP), and Rockwell RSLogix 5000 software. For more information on the CIP protocol, visit www.odva.org. For more information on installing, configuring, and using RSLogix as well as identifying Rockwell-specific hardware, visit www.rockwell.com.

This information was written against the hardware and software setup listed in Table F–1. The screen shots and examples assume this hardware; therefore, adjust the Rockwell-specific settings in the examples to comply with your specific hardware set.

TABLE F–1. Hardware and Software Configuration

Software/Hardware	Notes
1756-A4 4-slot Chassis	
ControlLogix 5555 PLC Using	slot 0
1756-ENBT Ethernet Bridge	Static class C IP address is 10.2.1.172, using slot 1
Rockwell RSLogix 5000	v16.21
VS-1 Smart Camera	v4.1 software must be downloaded to camera, static class C IP address is 10.2.1.20

Overview of the System

In this example, the VS-1 Smart Camera will be decoding a Data Matrix. The image acquisition will be triggered by an Ethernet/IP IO point. The match string for the Data Matrix code will be sent from the PLC to the VS-1 Smart Camera over Ethernet/IP implicit messaging. Also, the resulting DMR code location, decoded string and pass/fail status of the tool are sent back to the PLC over Ethernet/IP implicit messaging.

The first step in setting up the VS-1 Smart Camera to communicate with a PLC is to define the IO mappings and structure of the data to be sent between the devices.

IO Mapping

For acquisition triggering, we will be using Virtual IO point 129. This maps to bit 0 of DINT 0 of the Ethernet/IP data blocks.

TABLE F–2. Camera I/O Point Assignment

Bit/Instance	I/O Point	Name
1	129	Acquisition Trigger Low->High
2	130	Unused in this example
…	…	Unused in this example
32	160	Unused in this example

Output Assembly Instance 112

For this example, we have low data size requirements and will use the smaller assemblies that hold 64 bytes of user data. The complete assembly is:

DINT	Meaning
0	32 Bits of Camera Virtual IO
1	User defined tag value
2	Camera Control Register
3	Reserved
4…19	64 bytes of user data

For this example application, the user data portion will contain a string of length 28 characters. Bytes 0-3 (or DINT 4) of the user data block contain the string length and bytes 4-31 (or DINTs 5-11) contain the string contents. The complete assembly is:

DINT	Contents
0	Bit 0 = Acquire trigger, bits 1-31 unused 0x0
1	0x00000001
2	0x00000000
3	0x00000000
4	String length
5-11	Match String
12…19	Unused

Input Assembly Instance 100

In this example application, we will also be using the Assembly Instance 100 that holds 64 bytes of user data. The complete assembly is:

DINT	Meaning
0	32 Bits of Camera Virtual IO
1	User defined tag value
2	Camera Status Register
3	Last Error
4...19	64 bytes of user data

In this example, we will be storing Data Matrix results into the user data block of the assembly. We will be returning the center point of the Data Matrix code that is composed of three floating point numbers (X, Y, and angle). This data will be followed by the decoded string of the Data Matrix up to a length of 28 characters. This is then followed by a Boolean containing the pass/fail status of the Data Matrix Tool. The complete assembly is:

DINT	Contents
0	Unused (Bit 0 = Acquire trigger status)
1	PLC User Tag (always 0x00000001 in this example)
2	Camera Status Register
3	Last Error
4	Echo of PLC User Tag (0x00000001)
5	Center Point X
6	Center Point Y
7	Center Point Angle
8	Decoded String Length
9-15	Decoded String
16	Data Matrix Tool Pass Status
17…19	Unused

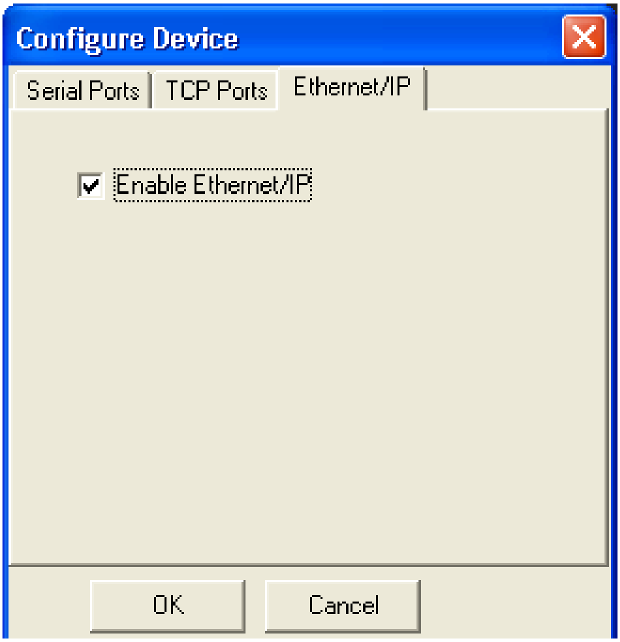
Note: Each discrete numeric data type (integers, floats, and booleans) in Visionscape® will map to a DINT in the assembly when using the provided Perl scripts for reading and writing Ethernet/IP data.

Programming the VS-1 Smart Camera

Setting Up The VS-1 Smart Camera

To communicate via Ethernet/IP, it must be activated on the camera. Activate it using the Configure Device dialog box. The camera must be restarted before Ethernet/IP activation changes will take effect. Once Ethernet/IP is enabled on the VS-1 Smart Camera, a PLC will be able to establish a connection with the camera regardless of whether a job is running on the camera that utilizes Ethernet/IP.

FIGURE F–1.



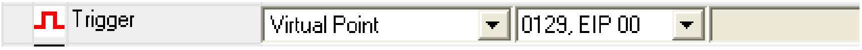
F
Ethernet/IP Communication with

Setting Up the VS-1 Smart Camera Job

Acquisition

Starting with a new job on the VS-1 Smart Camera, first set up the image acquisition. As we defined in our IO Mapping definition previously, the trigger should be set to Virtual Point 129. This IO point will be synchronized with the first bit of the first DINT of the Ethernet/IP assemblies when Ethernet/IP is enabled and running.

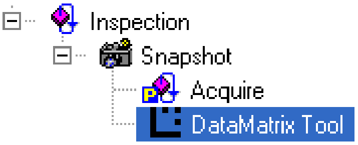
FIGURE F–2.



Data Matrix Tool

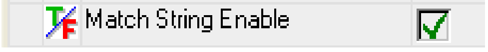
Insert a Data Matrix Tool into the snapshot to decode the expected DMR codes. Ensure that the ROI is large enough to cover the DMR locations of all the images expected at runtime.

FIGURE F–3.



On the property page of the Data Matrix Tool, select the Match String Enable option. With this option set, the tool will only Pass if the decoded string matches the expected ‘Match String’.

FIGURE F–4.



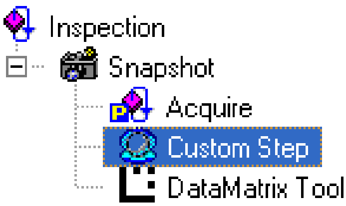
Once the job is configured to acquire images and decode them as desired, the Ethernet/IP connectivity can be added.

Connecting the PLC Output Assembly Data to VS-1 Smart Camera Datums

Now we will insert a tool to read the match string data from the Ethernet/IP output assembly and connect it to the DMR tool. Clicking on the Data Matrix Tool in the Step Tree View, insert a Custom Step above the Data Matrix Tool. The Custom Step must be inserted above the Data Matrix tool so that it is executed and the Match String property is updated before the Data Matrix Tool runs.

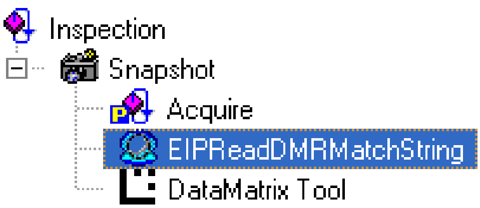
Note: Custom Step is found under the Script tab of the Insert Step dialog.

FIGURE F–5.



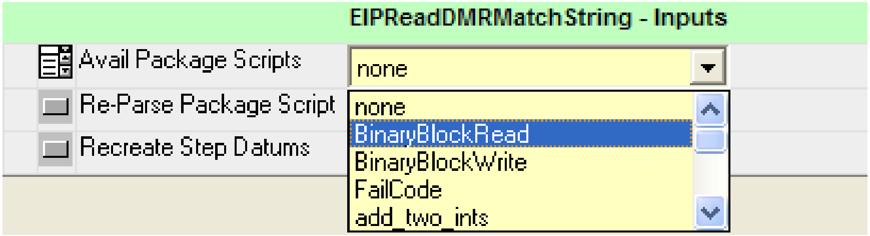
Note: You can rename the step for better readability, if so desired (Figure F–6).

FIGURE F–6.



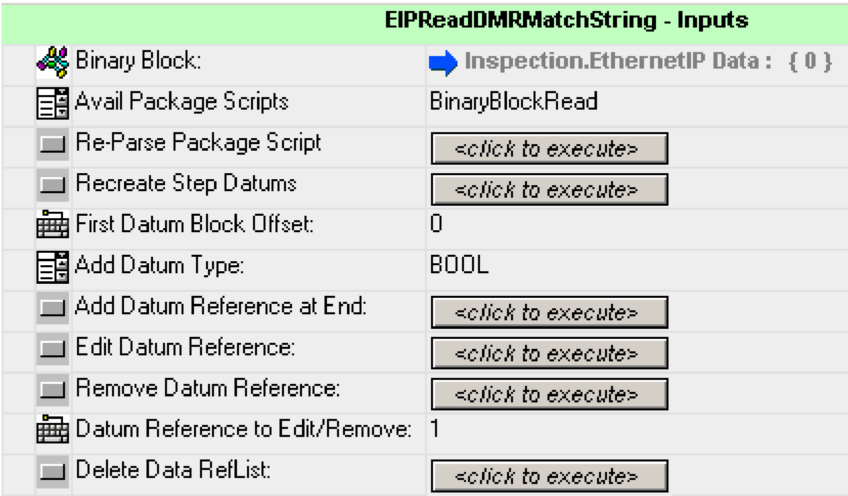
On the Custom Step property page, choose the BinaryBlockRead script from the Avail Package Scripts property (Figure F–7). This is a Perl script that takes data elements from an Ethernet/IP output assembly and connects them to Visionscape® datums in a vision job.

FIGURE F–7.



Several new properties will appear on the page (Figure F–8).

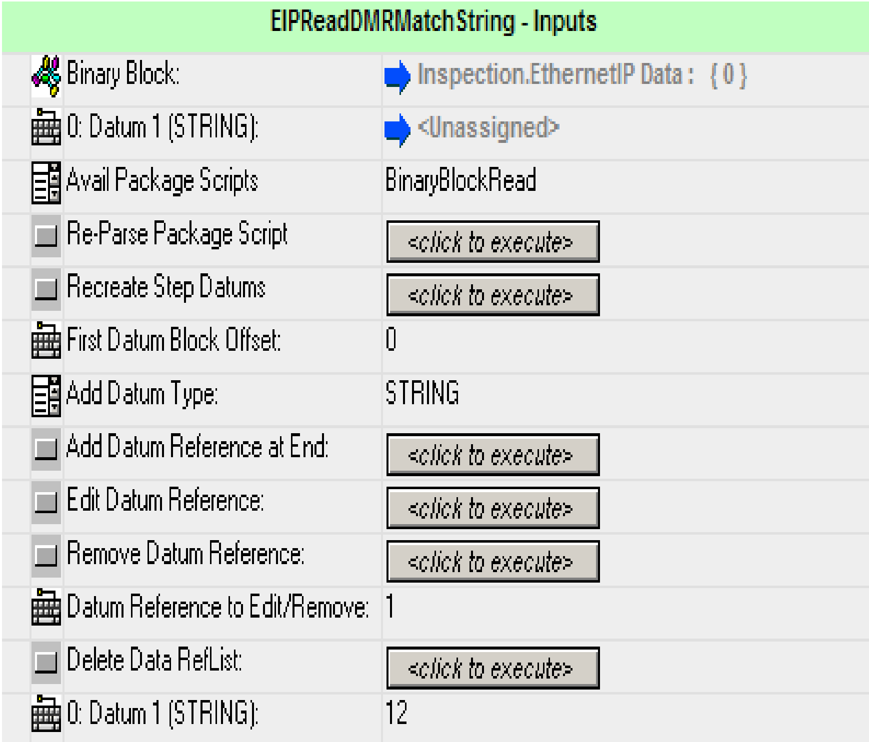
FIGURE F–8.



As previously defined, in this example there is a single string datum from bytes 0-31 of the user data portion of the output assembly. Since the string starts at byte 0, leave ‘First Datum Block Offset’ set to the default value of 0.

Next, add the ‘Match String’ data by changing ‘Add Datum Type’ to String and clicking ‘Add Datum Reference at End’. Two new properties will appear on the property page, an input datum and a reference datum both named ‘0 Datum 1 (String)’.

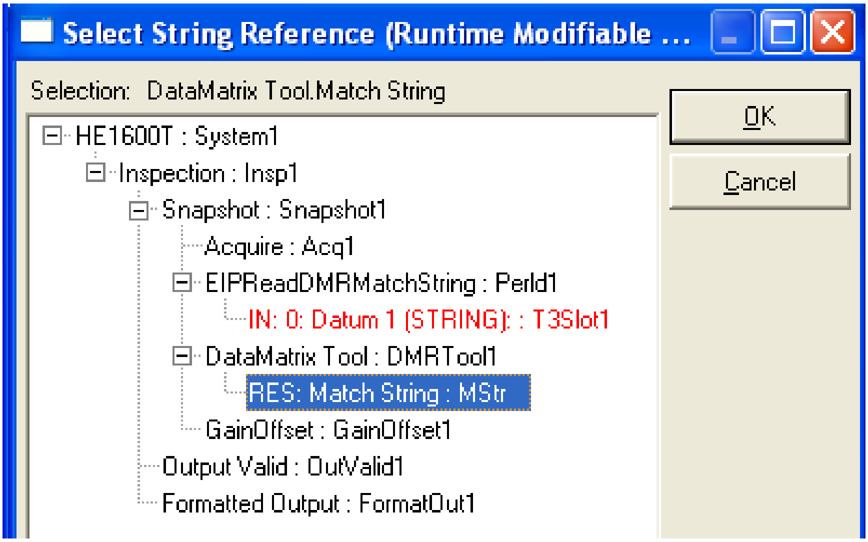
FIGURE F–9.



The 0 at the beginning of the name indicates a string starting at assembly byte location 0 of the user data block. The value 12 in the datum at the bottom of the property page indicates the default string length of 12 bytes. Since this example allows strings up to 28 bytes, change the last value to 28.

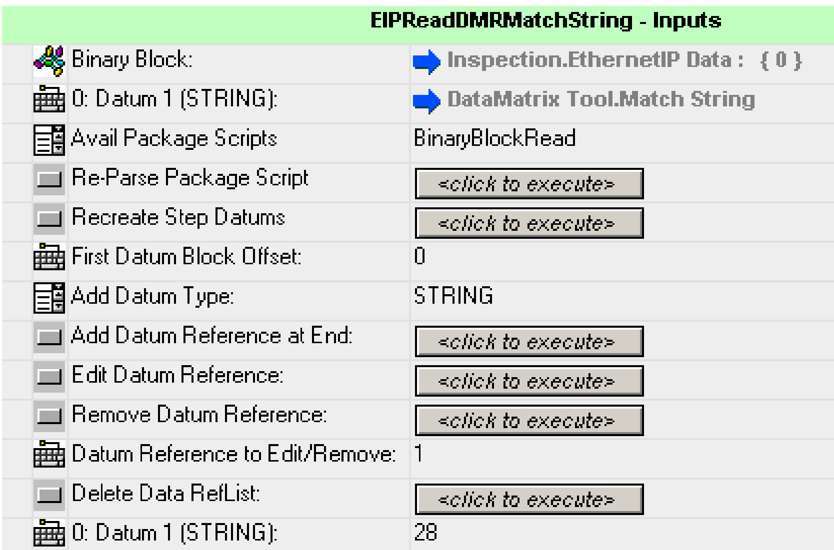
Next, select where to store the string in the vision job by selecting the input ‘0 Datum 1’ reference selector and choosing the Match String of the Data Matrix Tool.

FIGURE F–10.



The property page should now show a single string datum starting at byte 0, of length 28 bytes that will be placed into the Data Matrix Tool Match String (Figure F–11).

FIGURE F–11.



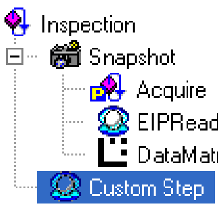
EIPReadDMRMatchString - Inputs	
Binary Block:	Inspection.EthernetIP Data : { 0 }
0: Datum 1 (STRING):	DataMatrix Tool.Match String
Avail Package Scripts	BinaryBlockRead
Re-Parse Package Script	<click to execute>
Recreate Step Datums	<click to execute>
First Datum Block Offset:	0
Add Datum Type:	STRING
Add Datum Reference at End:	<click to execute>
Edit Datum Reference:	<click to execute>
Remove Datum Reference:	<click to execute>
Datum Reference to Edit/Remove:	1
Delete Data RefList:	<click to execute>
0: Datum 1 (STRING):	28

At this point, every time this inspection runs, the match string of the Data Matrix tool will be updated to reflect the string value in the Ethernet/IP output assembly at the time the inspection was triggered.

Connecting VS-1 Smart Camera Results to the PLC Input Assembly Data

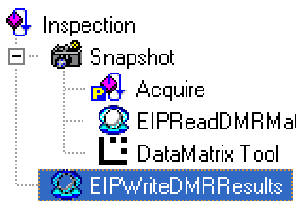
Next, we will insert a tool to take the results of the Data Matrix Tool and put them into the Input assembly that is sent back to the PLC from the VS-1 Smart Camera. To do this, insert another Custom Step into the job. Since we want to include results of the Data Matrix Tool, the Custom Step must be placed **below** the Data Matrix Tool so the results will be valid at the time the Custom Step executes. In this example, we will place it after the Snapshot step as follows:

FIGURE F–12.



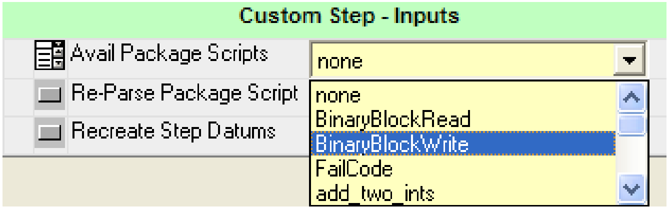
We also rename this step also for readability.

FIGURE F–13.



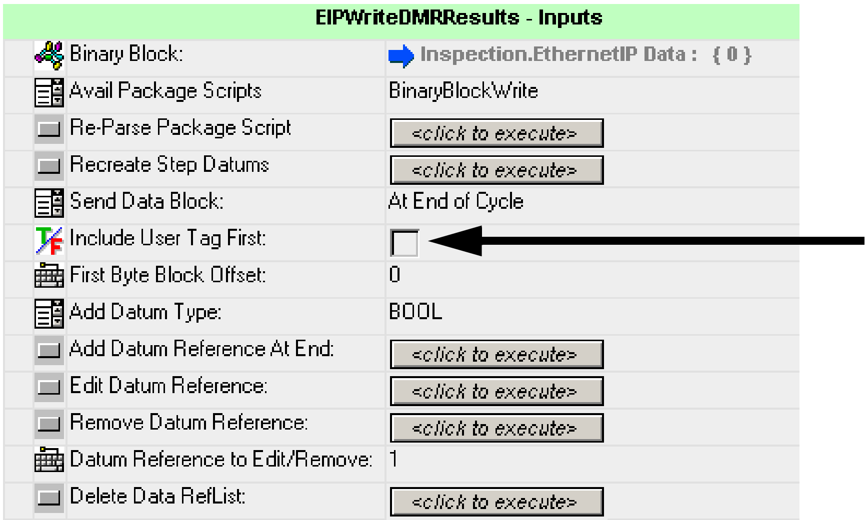
Now, since we want to write data from Visionscape® to Ethernet/IP, choose the BinaryBlockWrite script from the Avail Package Scripts property. This Perl script will take the data from Visionscape® datums and place them into the Ethernet/IP input assembly's user data block.

FIGURE F–14.



New properties will appear (Figure F–15):

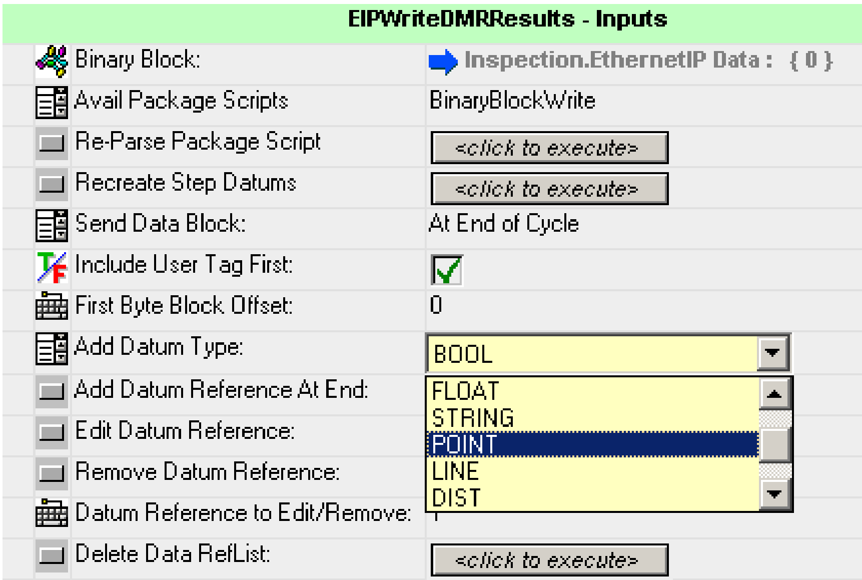
FIGURE F–15.



Since we want to echo the user tag back inside of the user data block in this example, select the box labeled Include User Tag First (Figure F–15).

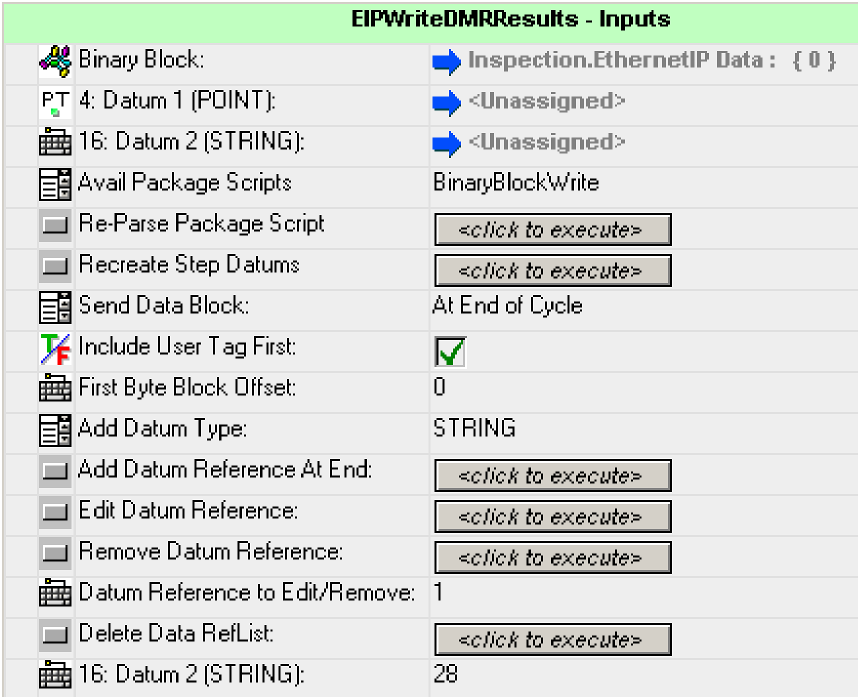
As defined earlier, the Ethernet/IP input assembly user data block will contain the Data Matrix center point, decoded DMR string and the DMR pass/fail status. Add those data types in sequence by first selecting Add Datum Type of Point and clicking ‘Add Datum Reference at End’.

FIGURE F–16.



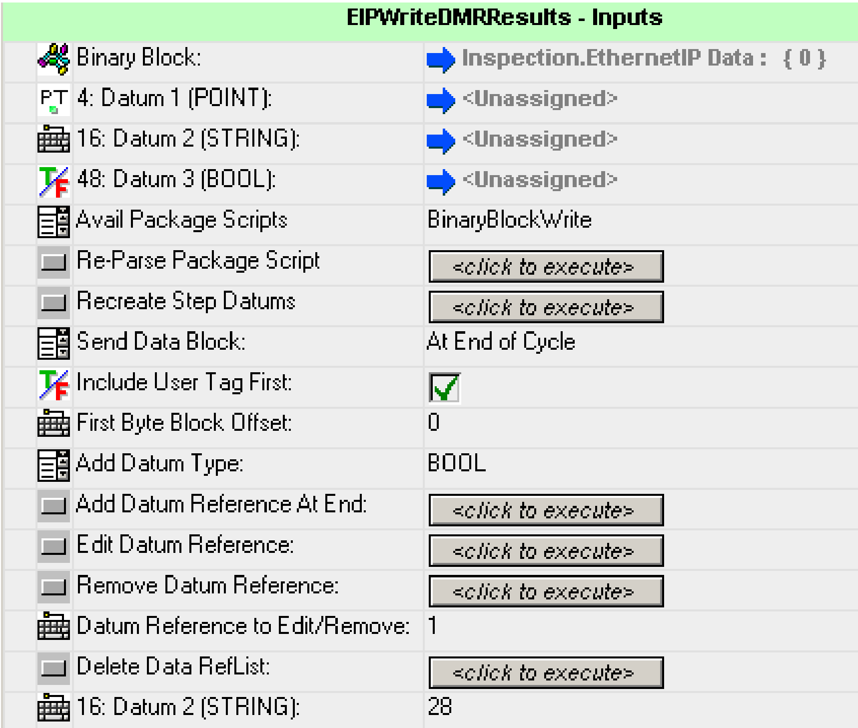
Then, choose Add Datum Type of String and click Add Datum Reference. Since we have allocated space for 28 characters in the data block, change the resource datum labeled Datum 2 (STRING) to 28 (Figure F–17).

FIGURE F–17.



Finally, add a Boolean reference by selecting Add Datum Type BOOL and clicking Add Datum Reference. The resulting property page should look like the one in Figure F–18. Showing a Point data type taking bytes 4-15, a String data type from bytes 16-47, followed by a Boolean data type in bytes 48-51. The first datum starts at byte 4 because we selected ‘Include User Tag First’ which will place the user tag in bytes 0-3, ahead of the datum contents.

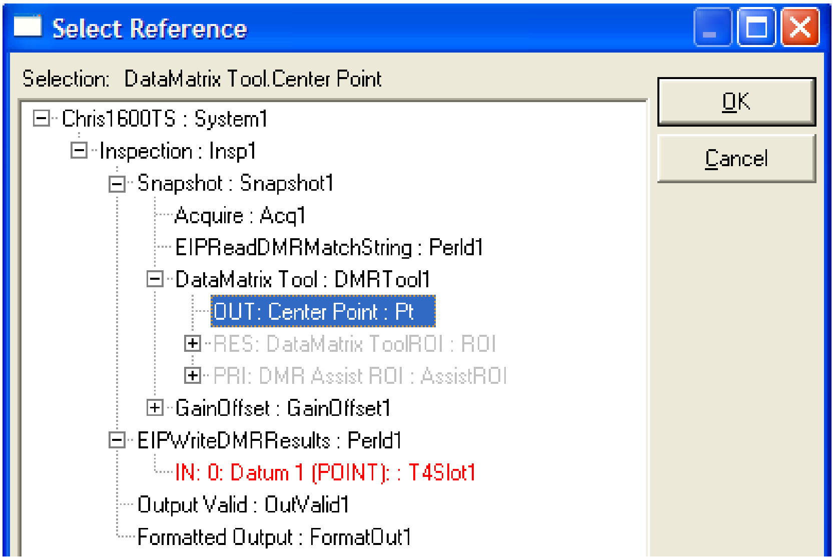
FIGURE F–18.



EIPWriteDMRResults - Inputs	
Binary Block:	Inspection.EthernetIP Data : { 0 }
4: Datum 1 (POINT):	<Unassigned>
16: Datum 2 (STRING):	<Unassigned>
48: Datum 3 (BOOL):	<Unassigned>
Avail Package Scripts	BinaryBlockWrite
Re-Parse Package Script	<click to execute>
Recreate Step Datums	<click to execute>
Send Data Block:	At End of Cycle
Include User Tag First:	✓
First Byte Block Offset:	0
Add Datum Type:	BOOL
Add Datum Reference At End:	<click to execute>
Edit Datum Reference:	<click to execute>
Remove Datum Reference:	<click to execute>
Datum Reference to Edit/Remove:	1
Delete Data RefList:	<click to execute>
16: Datum 2 (STRING):	28

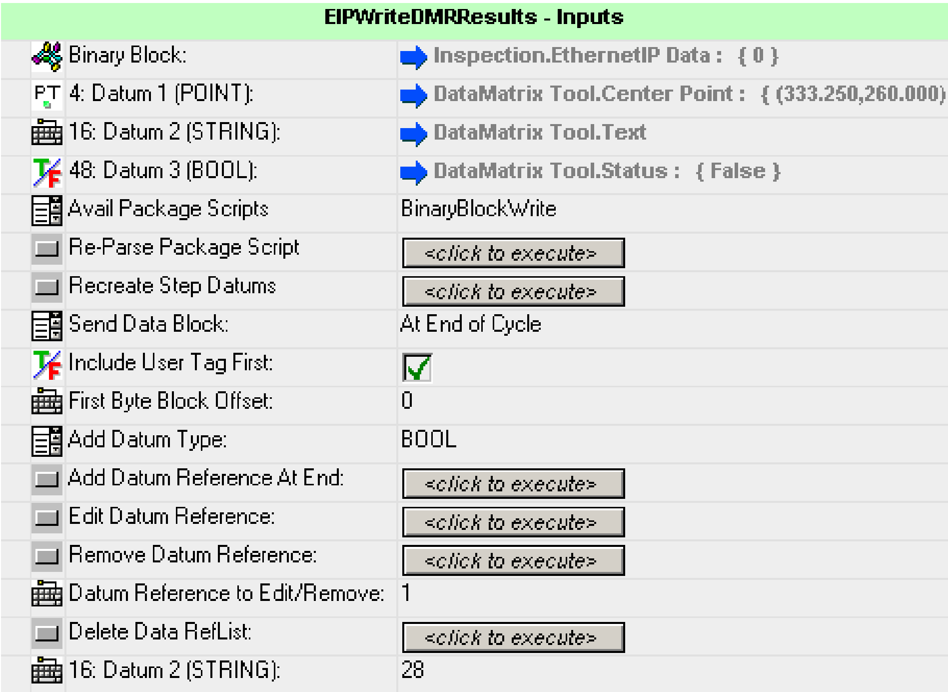
Next, connect each of the datums to the source of the result. In the case of the Point, connect it to the Data Matrix Tool Center Point (Figure F–19).

FIGURE F–19.



Similarly, connect the String to the Data Matrix Tool Text output and the Boolean to the Data Matrix Tool Status output. The resulting property page should look like the one in Figure F–20:

FIGURE F–20.



This step will now fill in the Ethernet/IP input assembly with the corresponding data at the end of each inspection cycle.

The Visionscape® job is now completed and ready to be downloaded to the camera.

Setting Up the PLC Program

The RSLogix 5000 environment uses “tags” to represent all data in the PLC program, including communications. Setting up the PLC entails the following:

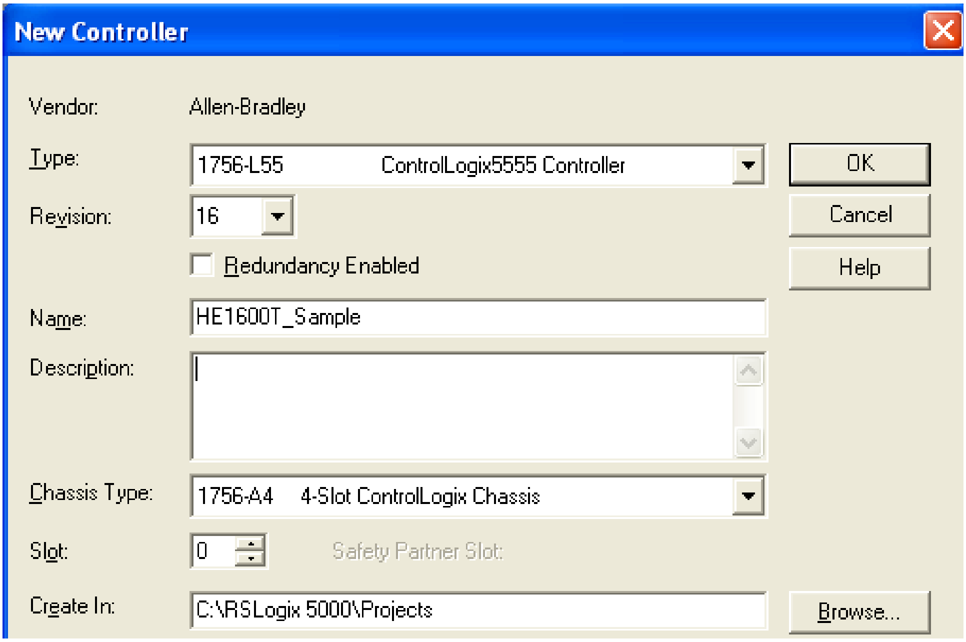
1. Creating a project appropriate to the hardware configuration.
2. Adding a generic “I/O” module to represent the VS-1 Smart Camera.
3. Defining the data types and structures to be communicated between the VS-1 Smart Camera and the PLC.
4. Downloading the program to the PLC.

Once tags have been defined to correspond to the data object of the input and output assemblies, you can use these tags in your ladder logic.

Configuring the Hardware

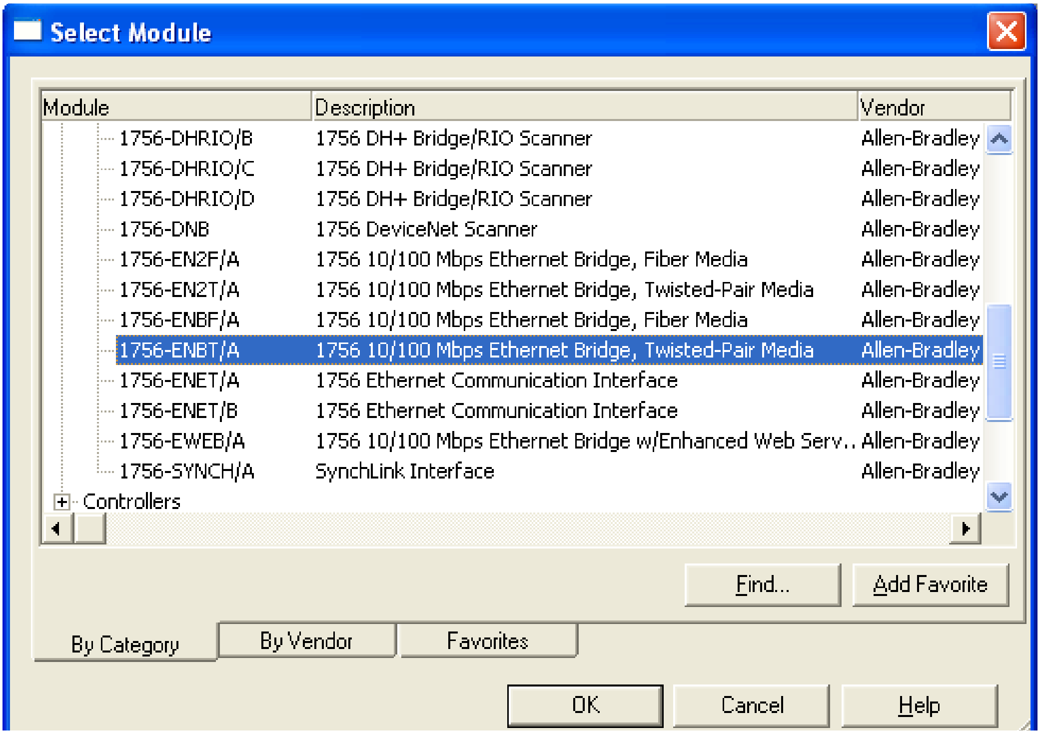
Within RSLogix 5000, use the File > New... menu to create a new project. Select the appropriate controller Type, Revision, Chassis Type, and Slot specific to your hardware configuration. Type in a name for your program in the “Name:” field, then click OK.

FIGURE F–21.



Once RSLogix 5000 creates the program, the Tree View displays different nodes for Tags, Tasks, etc. The last node in the tree is the "I/O Configuration". Right-click on this and select "New Module..." to add the Ethernet Bridge to the configuration.

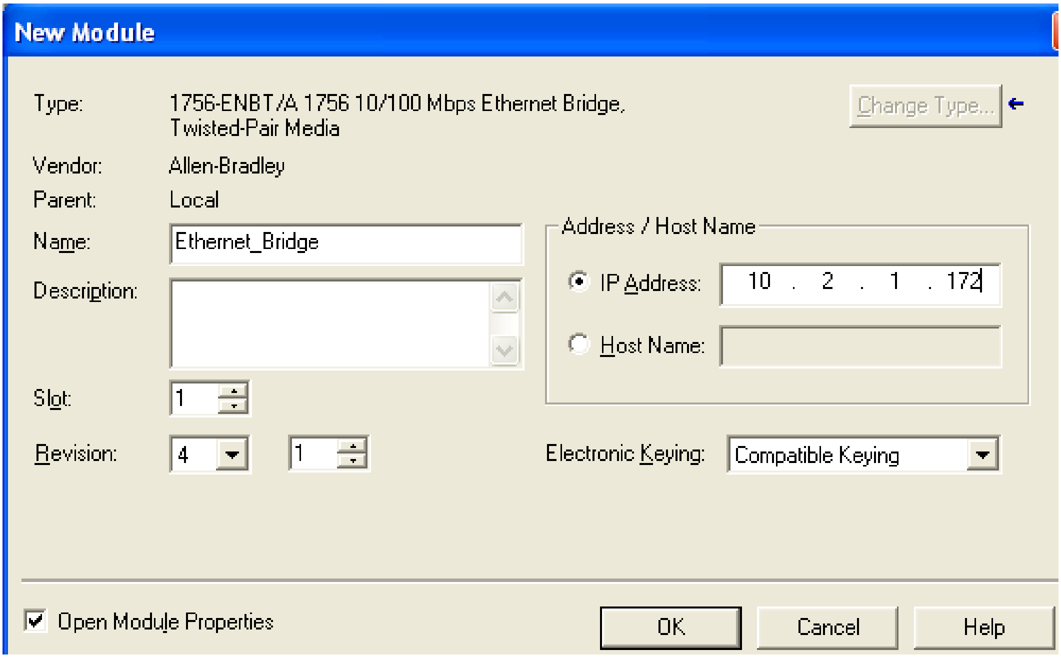
FIGURE F–22.



This is specific to your hardware setup; you need to know what bridge module is being used in your system. After clicking OK, you need to set up the IP configuration of the module.

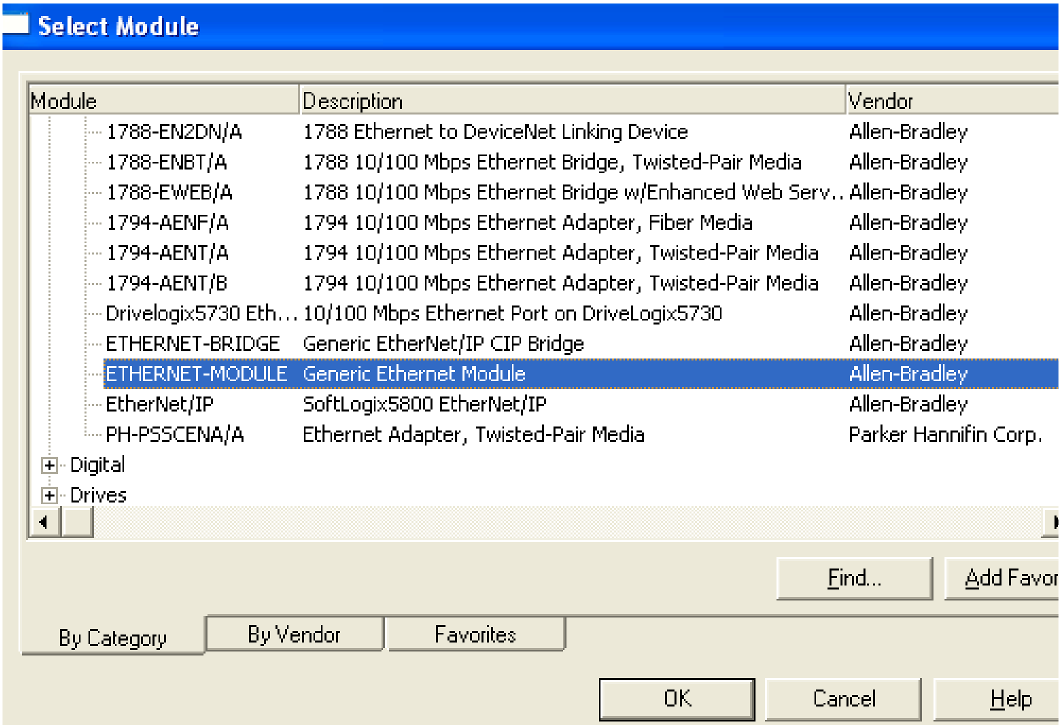
Set a useful name for the Ethernet Bridge in the “Name:” field and enter the static IP address of the bridge. In this example, the IP address is 10.2.1.172. In this configuration, the module is in slot 1 of the chassis. Click “OK” to continue.

FIGURE F–23.



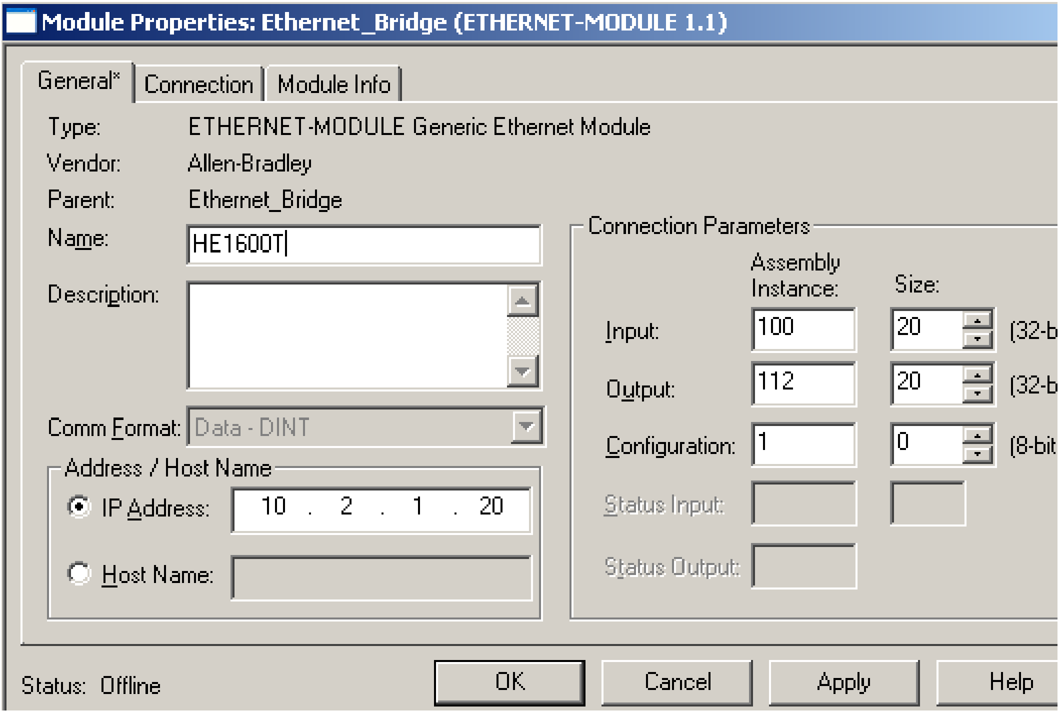
Now, you can add a module for the VS-1 Smart Camera. Go back to the Tree and select the “Ethernet_Bridge” under the I/O Configuration, right-click it and select “New Module...” again. Then, you can select a “Generic Ethernet Module”.

FIGURE F–24.



Click OK. The configuration dialog box for the generic module is displayed (Figure F–25).

FIGURE F–25.



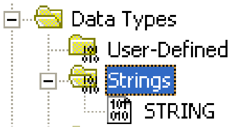
Enter “VS-1 Smart Camera” for the name. Enter the IP address of the camera. In this example, the static IP address of the camera is 10.2.1.20. For the I/O assemblies, the input assembly is instance 100, and it is 20 DINTs of data. The Output assembly is instance 112, and is also 20 DINTs of data. The “Configuration” assembly is not used by the camera, so enter in instance 1, size 0 and click OK.

At this point, the program creates a set of Controller Tags for each instance of the I/O assemblies of the new module. You can examine the tags by double-clicking “Controller Tags” in the RSLogix 5000 Tree View.

Defining Data Types

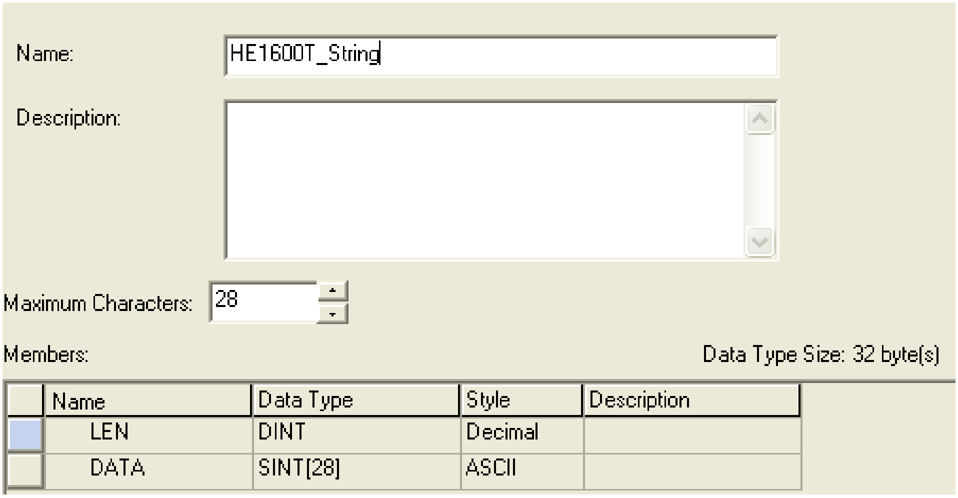
For better readability and programming, we will define data types that correspond to our assembly structures. First we will define a new String type that has space for 28 characters, the same as we defined in our assemblies. Right click on Data Types -> Strings in the tree and select New String Type....

FIGURE F–26.



Name the string and set the Maximum Characters to 28. Including the 4 bytes of string length data, this will result in a total data size of 32 bytes.

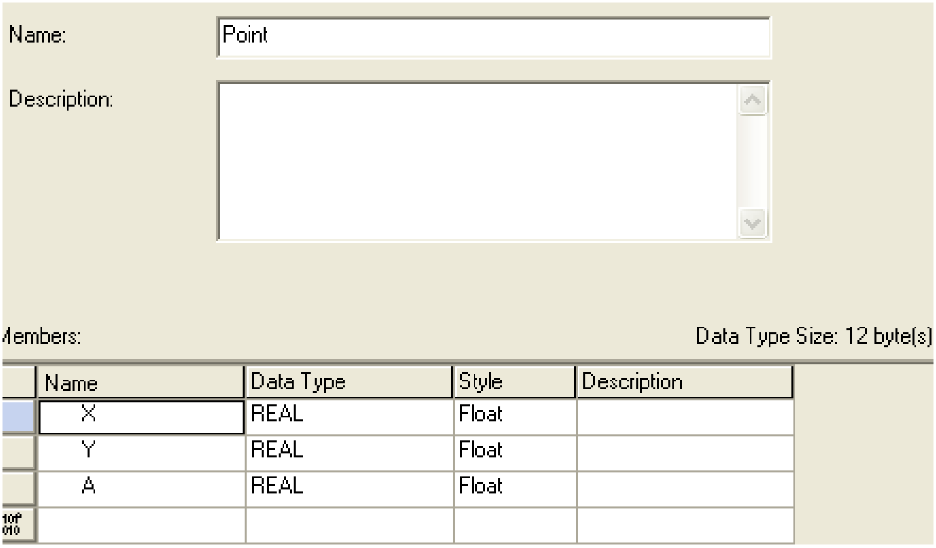
FIGURE F–27.



Next, we will create a data structure to represent the results we will be receiving from the VS-1 Smart Camera in the input assembly.

First, define the structure of a Point that is composed of 3 floating point numbers. Right click on Data Types -> User-Defined to create the Point type. Add three Members of type REAL to hold the X, Y and angle values of the point. The total structure size should be 12 bytes.

FIGURE F–28.

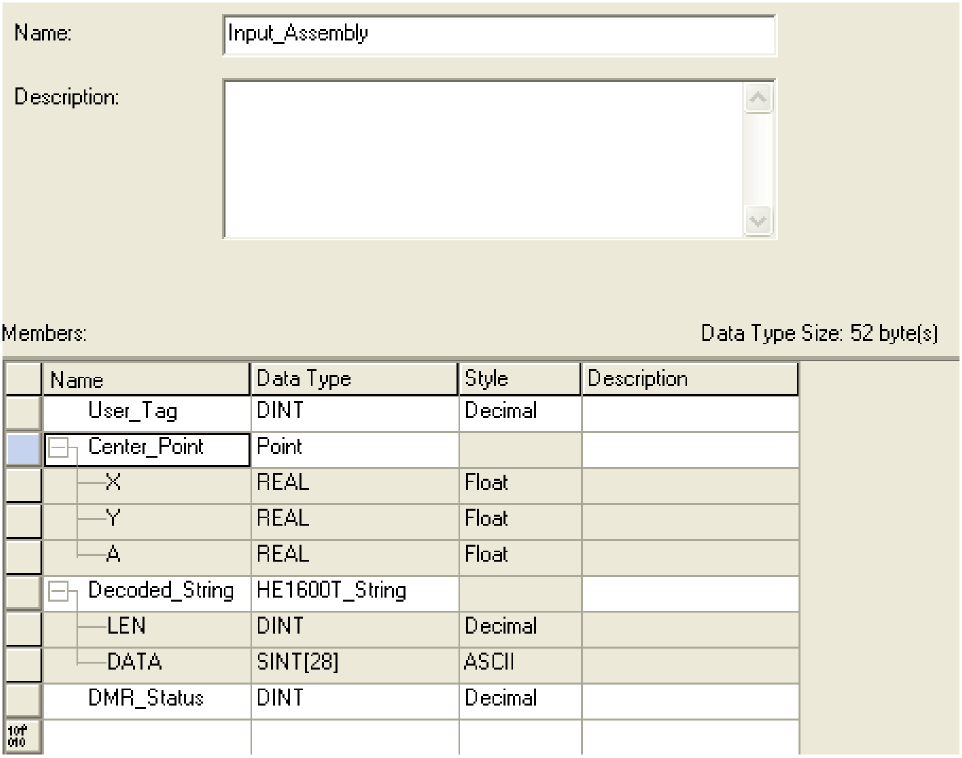


Next, create the structure representing the input assembly contents. Again, right click on Data Types -> User Defined and select New Data Type. Add the data types in the same order as we previously defined the assemblies to be.

- A User Tag of type DINT.
- The Center Point of the decoded DMR that is of our newly defined type Point, composed of three floats.
- Decoded String which is our new VS-1 Smart Camera String type of 28 characters.
- DMR Status of type DINT which will contain the pass/fail status of the Data Matrix Tool.

Once completed, the full input assembly data type should total 52 bytes and appear as shown in (Figure F–29).

FIGURE F–29.



Name	Data Type	Style	Description
User_Tag	DINT	Decimal	
Center_Point	Point		
X	REAL	Float	
Y	REAL	Float	
A	REAL	Float	
Decoded_String	HE1600T_String		
LEN	DINT	Decimal	
DATA	SINT[28]	ASCII	
DMR_Status	DINT	Decimal	

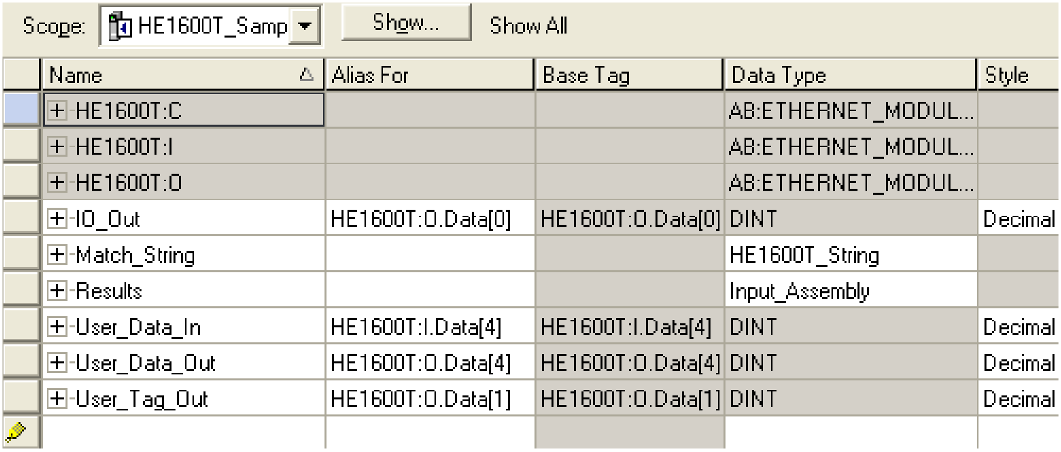
Add Controller Tags

Now that we have defined the data types we will need, we can add Controller Tags that will be placeholders to contain the data. Right click on Controller Tags in the tree and select Edit Tags.

Add a tag named Match_String of type HE1600T_String to hold our output match string. Also, add a tag named Results of type Input_Assembly to hold the results coming from the camera.

In addition, we can create aliases that map to the Ethernet/IP memory locations for better readability of the program. For this example, we will use the alias IO_Out as an alias for the first DINT of the output assembly and User_Tag_Out for the second DINT. The aliases User_Data_In and User_Data_Out will correspond to the fifth DINT of the respective input and output assemblies. The resulting set of controller tags will look as shown in Figure F–30):

FIGURE F–30.

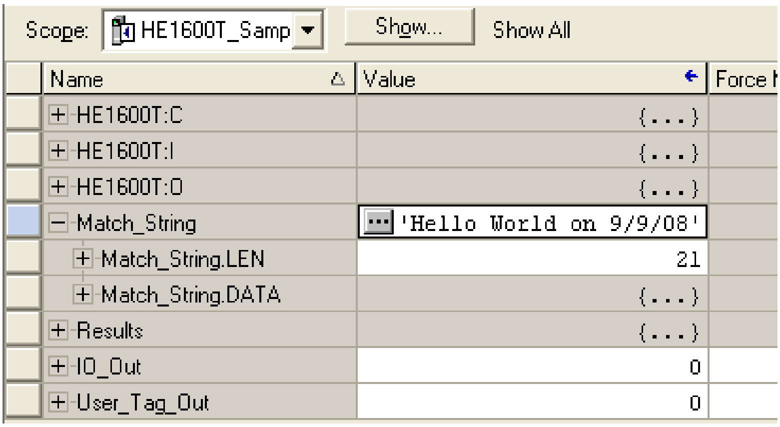


Scope: HE1600T_Samp | Show... | Show All

Name	Alias For	Base Tag	Data Type	Style
+ HE1600T:C			AB:ETHERNET_MODUL...	
+ HE1600T:I			AB:ETHERNET_MODUL...	
+ HE1600T:O			AB:ETHERNET_MODUL...	
+ IO_Out	HE1600T:O.Data[0]	HE1600T:O.Data[0]	DINT	Decimal
+ Match_String			HE1600T_String	
+ Results			Input_Assembly	
+ User_Data_In	HE1600T:I.Data[4]	HE1600T:I.Data[4]	DINT	Decimal
+ User_Data_Out	HE1600T:O.Data[4]	HE1600T:O.Data[4]	DINT	Decimal
+ User_Tag_Out	HE1600T:O.Data[1]	HE1600T:O.Data[1]	DINT	Decimal

In this example, we have a single constant match string, so we will pre-define the match string contents in the controller tag. Right click on Controller Tags and select Monitor Tags and fill in the Value of the Match_String tag. In this example the string should read ‘Hello World on 9/9/08’.

FIGURE F–31.



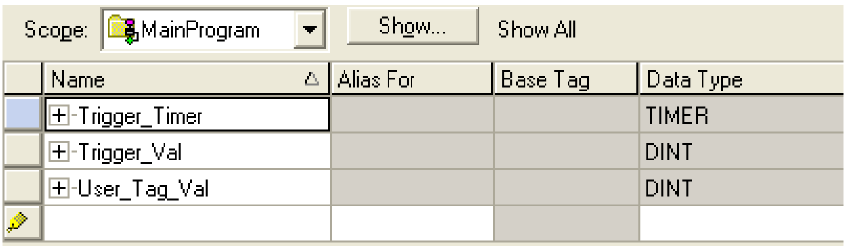
Scope: HE1600T_Samp | Show... | Show All

Name	Value	Force M
+ HE1600T:C	{...}	
+ HE1600T:I	{...}	
+ HE1600T:O	{...}	
− Match_String	'Hello World on 9/9/08'	
+ Match_String.LEN	21	
+ Match_String.DATA	{...}	
+ Results	{...}	
+ IO_Out	0	
+ User_Tag_Out	0	

Add Program Tags

In this example, we will be creating a timed trigger, so we will need a TIMER and a place to store the current trigger state. From the program tree, double click on Tasks -> MainTask -> MainProgram -> Program Tags and select the Edit Tags tab. Create a tag named Trigger_Timer of type TIMER and Trigger_Val of type DINT. We will also create a User_Tag_Val of type DINT to hold our user tag value which we will set to a value of 1 on the Monitor Tags tab.

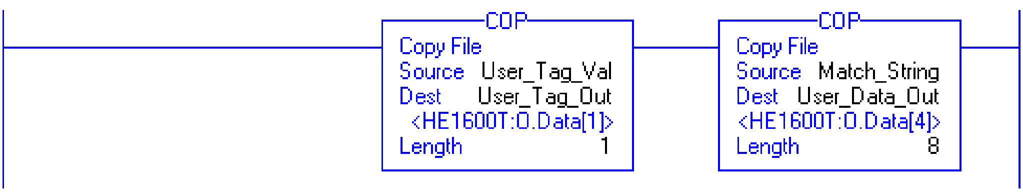
FIGURE F–32.



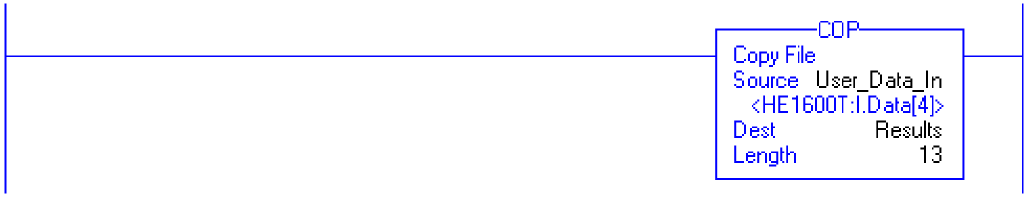
Ladder Logic

With our data types defined and tags created, we can now create the program using ladder logic. In this example, we will copy the user tag and match string from our controller tag to the Ethernet/IP output assembly on every cycle. Similarly, we will copy the results from the Ethernet/IP input assembly to our Results structure in the controller tags on every cycle.

We will use the file copy function COP to copy data between our tags and the assemblies. Our first rung will copy the user tag from User_Tag_Val to the User_Tag_Out alias and the Match_String to the User_Data_Out alias. The Length parameter corresponds to the number of DINTs to copy, in the case of our 32 byte HE1600T_String type, the length will be 8. This will write the user tag and match string to the output assembly to be received by the VS-1 Smart Camera.



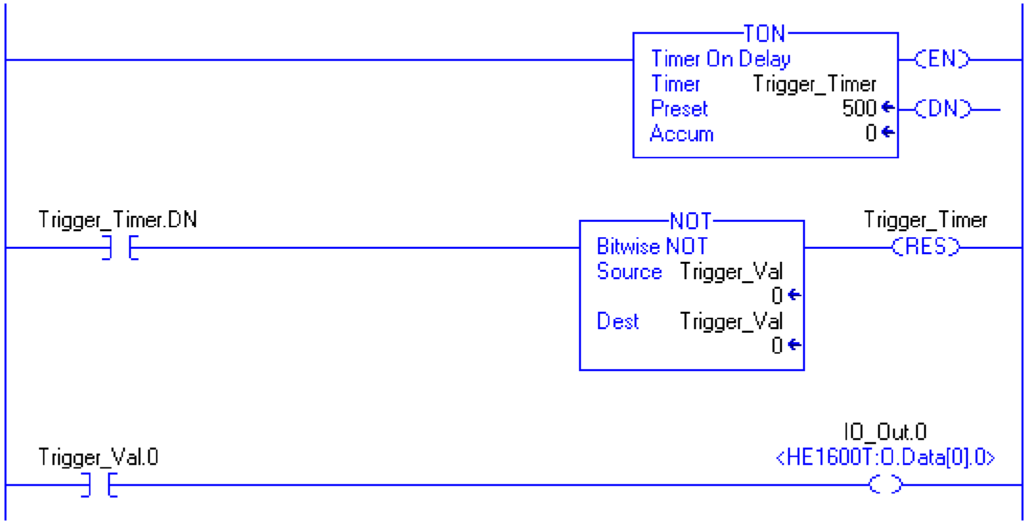
Next, we'll copy the results in the input assembly to our controller tag Results. Again, we use the COP function to copy the 52 byte Input_Assembly structure from the user data portion of the input assembly to the Results tag.



Finally, we'll create a trigger event using a timer. We will create a 1-second trigger by toggling an IO point every half second. To do this, we'll use the TON timer using our program tag name Trigger_Timer and set the Preset value to 500.

Then, on another rung, we will check for a timer done condition. If true, we then toggle the value of our Trigger and reset the timer.

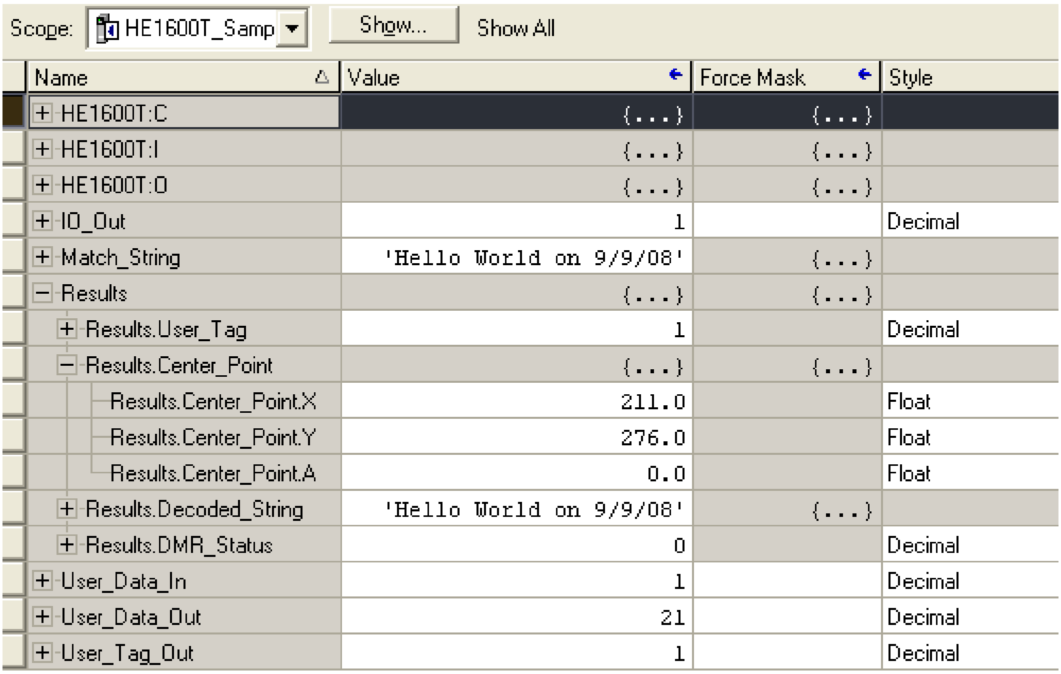
On another rung, we will copy the value of a bit from the Trigger Value and write it to the IO bit 0 of the Ethernet/IP output. This will correspond to Virtual IO 129 in the Visionscape® program.



Run the Program

Now, the program can be downloaded to the controller and put into run mode. If the camera is also running, the results from the VS-1 Smart Camera can be viewed by clicking on the Controller Tags and selecting the Monitor tab.

FIGURE F–33.



Name	Value	Force Mask	Style
+ HE1600T:C	{...}	{...}	
+ HE1600T:I	{...}	{...}	
+ HE1600T:O	{...}	{...}	
+ IO_Out	1		Decimal
+ Match_String	'Hello World on 9/9/08'	{...}	
− Results	{...}	{...}	
+ Results.User_Tag	1		Decimal
− Results.Center_Point	{...}	{...}	
Results.Center_Point.X	211.0		Float
Results.Center_Point.Y	276.0		Float
Results.Center_Point.A	0.0		Float
+ Results.Decoded_String	'Hello World on 9/9/08'	{...}	
+ Results.DMR_Status	0		Decimal
+ User_Data_In	1		Decimal
+ User_Data_Out	21		Decimal
+ User_Tag_Out	1		Decimal

Assembly Data

Introduction to an “Assembly”

Data on Ethernet/IP devices is represented in an object called an Assembly. Data is sent to or retrieved from the assemblies by using either explicit messages or implicit (Producer/Consumer) messages.

- Explicit messages are messages you set up in your program to retrieve data at specific points in time. Implicit messages are automatically set up by the program to refresh data at given intervals.
- Implicit messaging is also called the “Producer/Consumer” model, as one device produces data and another device consumes it. With implicit messaging, the camera produces output data at regular intervals. The PLC also produces data at regular intervals and updates the camera accordingly.

When you set up the Generic Ethernet Module for the VS-1 Smart Camera, you entered in specific instances of I/O assemblies as well as the data size for each. These assemblies are connected implicitly by the PLC. The camera sends updated data to the Input assembly, represented by the HE1600T:I tags, and the PLC sends updated data from the Output assembly, represented by the HE1600T:O tags.

The following sections detail the different assemblies supported by the camera.

Assembly Instance 100 (Input)

This assembly supports implicit messaging and represents 20 DINTs of data. Typically, you connect to this instance by configuring the Generic Ethernet Module according to the instructions listed in the section “Setting Up The PLC Program”. Because the connectivity is implicit, this data is refreshed at regular intervals by the network.

DINT	Meaning
0	32 Bits of Camera Virtual IO
1	User defined tag value
2	Camera Status Register
3	Last Error
4...19	64 bytes of user data

DINT 0: 32 Bits of Camera VIO

This 32 bit value maps to a subset of Virtual IO points on the camera (VIO 129 through VIO 161). The value of this register will always reflect the state of the corresponding IO points from the camera. By using the corresponding VIO point on the camera, these IO points can reflect the state of any IOOutputDm such as picture done, or set by a user expression using a DigitalOutput step.

Bit/Instance	Visionscape Virtual I/O Point
1	129
2	130
…	…
32	160

Since the camera virtual IO points are bi-directional and also map to the output assembly, the state of these inputs should be reflected back in the output assembly to keep the camera and PLC's view of the IO states synchronized.

DINT 1: User Defined Tag Value

The User defined tag value is a single DINT that is sent to the camera through implicit messaging to Assembly Instance 112. When the camera receives a new user tag from Assembly Instance 112, it is latched in the inspection's copy of the assembly and will be reflected through Assembly 100 at the end of the inspection cycle. This allows the PLC to match inspection results in the Input Assembly to the Output Assembly that was latched at the start of the same inspection cycle.

DINT 2: Camera Status Register

BIT	Meaning
0	Online: 0=offline (booting, loading job), 1=online
1	Run mode: 0=stopped, 1=running
2	Command Complete
3	Command Succeeded
4…31	Reserved

Commands sent to the Control Register are ignored when the camera is offline; the camera must be online before sending a Start trigger.

DINT 3: Last Error

Last Error returns the last error code sent to the Ethernet/IP subsystem. It will return all zeros if no error occurred. The last error code can be reset to zero by transitioning the Clear Last Error bit of the control register from zero to one.

DINT 4…19: User Data

This contains 64 bytes of user defined data. The content of the data block is defined by how the bytes are written by the vision job using Perl write methods. The data is unstructured within the Ethernet/IP transport; it is up to the programmer to ensure the data structures match on each end of the transport.

Assembly Instance 112 (Output)

This assembly supports implicit messaging and represents 20 DINTs of data (80 bytes). Because the connectivity is implicit, this data is refreshed at regular intervals by the network.

DINT	Meaning
0	32 Bits of Camera Virtual IO
1	User defined tag value
2	Camera Control Register
3	Reserved
4…19	64 bytes of user data

DINT 0: 32 Bits of Camera VIO

This 32 bit value maps to a subset of Virtual IO points on the camera (VIO 129 through VIO 161). The value of this register will be written to the corresponding IO points of the camera. They can be used by any IO input datum, such as triggers, or read by using a DigitalInput step.

Since the camera virtual IO points are bi-directional and also map to the input assembly, the states of these outputs should reflect the states from the input assembly to keep the camera and PLC's view of the IO states synchronized.

DINT 1: User defined tag value

The User defined tag value is a single DINT that is stored on the camera. This value is reflected back to the client through Assembly Instance 100.

DINT 2: Camera Control Register

BIT	Meaning
0	Stop, transition 0->1 to stop inspections
1	Start, transition 0->1 to start inspections
2	Clear error, transition 0->1 to clear Last Error
3	Clear command status, transition 0->1 to clear command stat
4…31	Reserved

DINT 3: Reserved

This double word is reserved for future use. The PLC should write all zeros.

DINT 4…19: User Data

This contains 64 bytes of user defined data. The content of the data block is defined by how the bytes are written by PLC logic. The data is unstructured within the Ethernet/IP transport; it is up to the programmer to ensure the data structure read by the Visionscape® job matches the PLC format.

G

APPENDIX G

Visionscape Debug Viewer

The VsLogView Utility can be used to view diagnostic messages from Smart Cameras and software systems. The utility is started from the Windows Start menu by selecting Microscan Visionscape->Tools->Diagnostics->Visionscape Debug Viewer.

Diagnostic messages from software systems are displayed automatically. The messages normally seen in the debug window are replicated here. The Visionscape Debug Viewer Utility also includes features to sort and filter the diagnostic messages.

To filter the displayed messages, type text into the Filter edit box. This will only show messages which contain the string specified. Preceding the search string with a minus sign “-” will only show messages which do not contain the string.

In order to display diagnostic messages from VS-1 Smart Cameras a connection must first be established by selecting the name of the camera from the drop down box and pressing the “Open” button.

FIGURE G–1. Smart Camera Selection

The Visionscape Debug Viewer Utility permits logging messages to a file. First select the desired file location by using the “Set Log File...” option in the “File” menu. This setting is saved so it need only be specified once. Once the log file location is set, logging can be enabled by either:

- Selecting the “Log to...” option from the “File” menu
- Clicking on the logging status area on the toolbar

FIGURE G–2. Logging Enable

Index