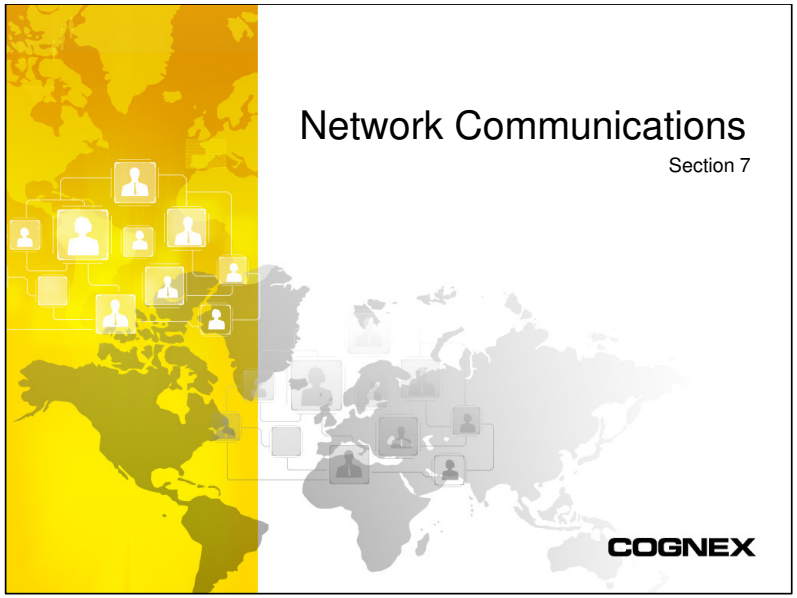
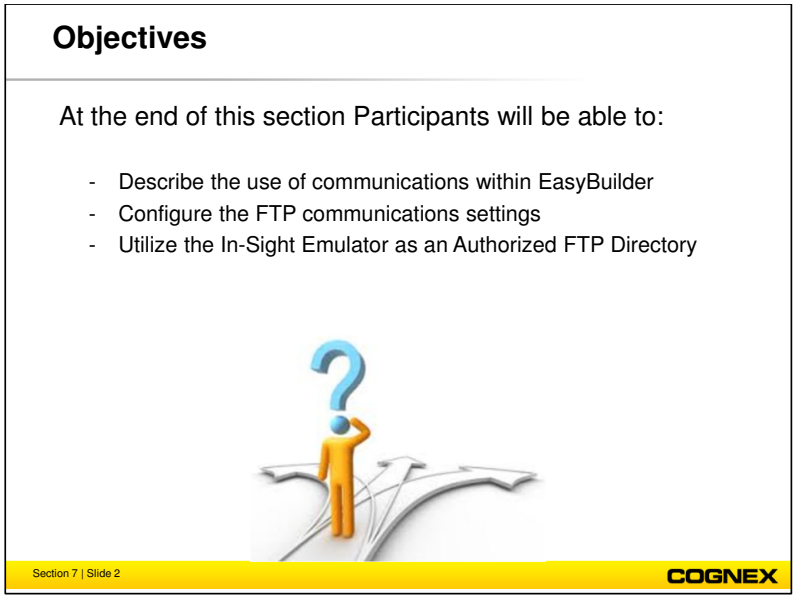


The seventh section of the In-Sight EasyBuilder Standard training will focus on **Communications**.

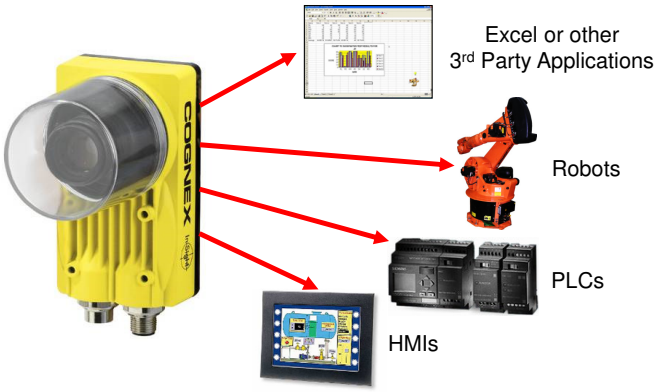
The Communication step is used to define inspection results for open data access and exchange (OPC), to configure an EasyView for the VisionView™ Operator Interface Panel, exporting images via FTP or to the SD card, and to define the network and serial communication settings of In-Sight Vision Systems, enabling them to communicate data to and from other devices, such as a robot or a Programmable Logic Controller (PLC) / Motion Controller (MC).



At the end of this section, Participants will be able to:

- Describe the use of communications within EasyBuilder
- Configure the FTP communications settings
- Utilize the In-Sight Emulator as an Authorized FTP Directory

Communications

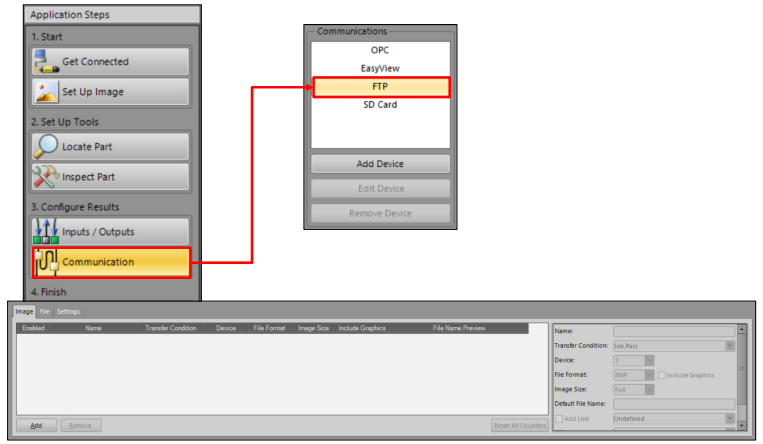


Section 7 | Slide 3 COGNEX

The **Communication** step is used to define inspection results for open data access and exchange (OPC), to configure an EasyView for the VisionView® Operator Interface Panel, exporting images via FTP, and to define the network and serial communication settings of In-Sight vision systems, enabling them to communicate data to and from other devices, such as a robot or Programmable Logic Controller (PLC) / Motion Controller (MC).

NOTE: *The In-Sight camera must be **Offline** to enable the Communication step.*
The In-Sight 8405 vision system does not support serial communication or connection to an I/O module.

FTP

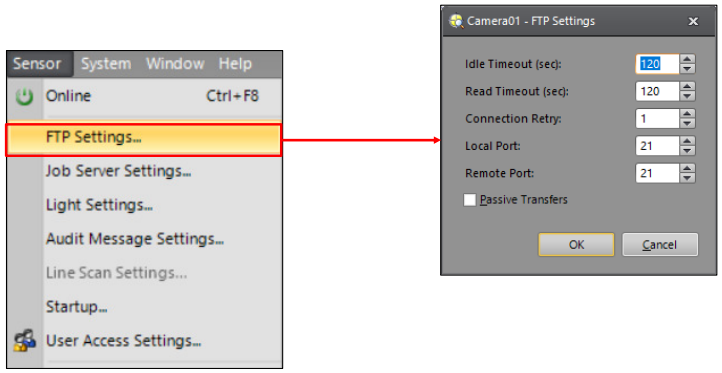


Section 7 | Slide 4 COGNEX

In-Sight vision systems have an integrated FTP client/server that allows them to share files with other vision systems on the network using the FTP protocol. The Save and Open dialogs act as FTP clients and can establish connections with FTP servers on the network to transfer job, image and firmware update files.

From the **Communication** Application Step select **FTP** from the Communications window.
The FTP window displays.

FTP Settings Dialog

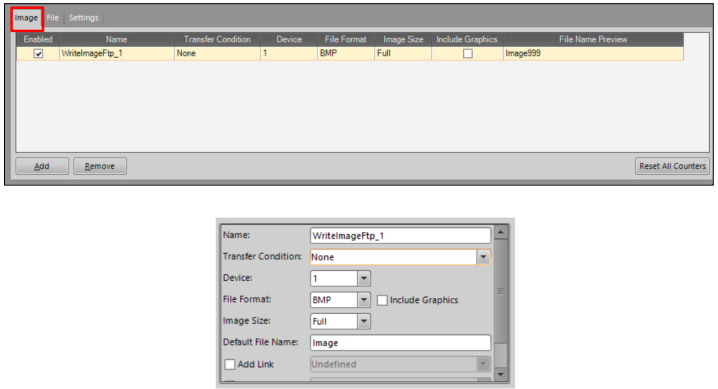


Section 7 | Slide 5 COGNEX

The **FTP Settings Dialog** configures an In-Sight sensor to share jobs, images and settings with other network devices that use the File Transfer Protocol. Users typically do not need to modify the default settings on dedicated, local In-Sight sensors where all FTP connections will occur between In-Sight sensors and other network servers, or between In-Sight sensors on different networks, then the default FTP settings may need to be changed.

To display the FTP Settings dialog, select FTP Settings from the Sensor menu.

WriteImageFTP

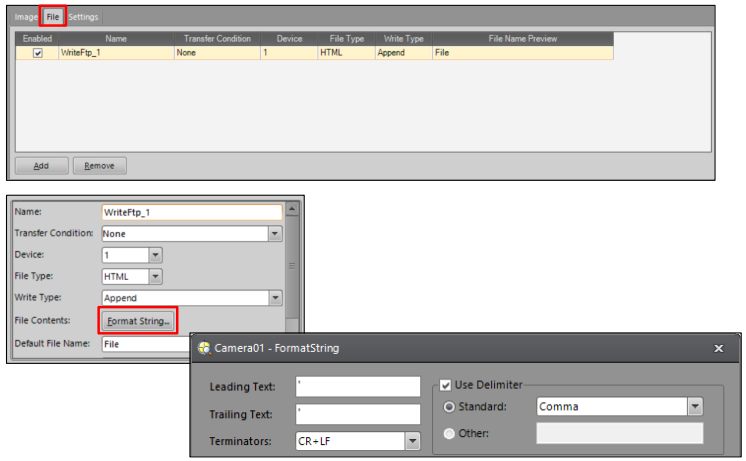


Section 7 | Slide 6 COGNEX

On the Image tab, click the Add button.
The **WriteImageFTP** tool displays.

The tool parameters will display to the right of the Image tab.
Transfer Condition defines when the data should be sent to the FTP server, for example, when a particular tool passes or fails, or an input is received. By default, none is selected, which ensures that the data will be exported every time an image is acquired when the Enabled checkbox is checked. If the Enabled checkbox is disabled, or the tool referenced is disabled, the export operation will not execute.
File Format defines the file format of the exported image: BMP (Windows bitmap format) or JPG (standard encoded JPEG format). By default, BMP is selected.
Default File Name defines the name of the image format that is exported, the default name is Image. Enter the Authorized FTP directory followed by a \ before the Image if using In-Sight Explorer as your FTP directory.

WriteFTP

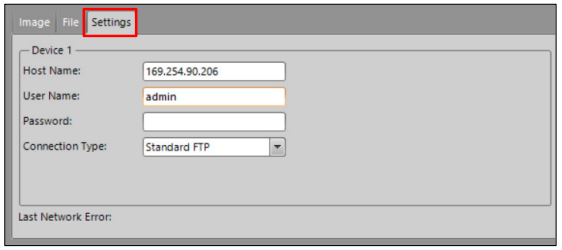


Section 7 | Slide 7 COGNEX

On the File tab, click the Add button.
The **WriteFTP** tool displays.

The tool parameters will display to the right of the File tab.
Transfer Condition defines when the data should be sent to the FTP server, for example, when a particular tool passes or fails, or an input is received. By default, none is selected, which ensures that the data will be exported every time an image is acquired when the Enabled checkbox is checked. If the Enabled checkbox is disabled, or the tool referenced is disabled, the export operation will not execute.
File Contents: Format String defines the data written to the file using the FormatString dialog.
Default File Name defines the name of the image format that is exported, the default name is File. Enter the Authorized FTP directory followed by a \ before the File if using In-Sight Explorer as your FTP directory.

FTP



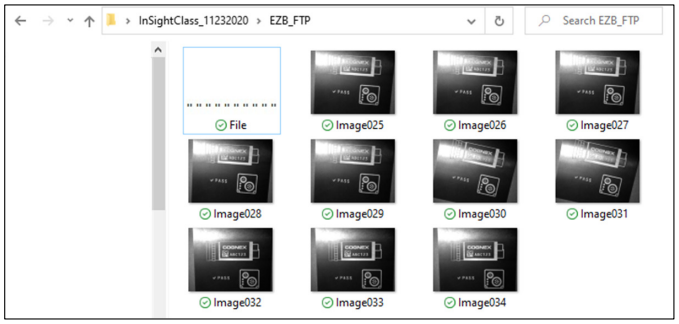
Section 7 | Slide 8 COGNEX

The Settings tab is where you will configure the In-Sight Emulator as the target device.

Host Name defines the host name (or IP Address) for the target device's FTP server.
User Name defines the valid user name for the target device's FTP server. This name does not have to exist on the In-Sight vision system that is exporting the image file or data.
Password defines the valid password for the target device's FTP server. The password is case-sensitive, and its length cannot exceed 20 characters.

Since we will be using the In-Sight Emulator as the target device, the user name is admin, and the password is left blank.

FTP

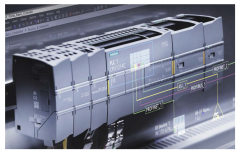


Section 7 | Slide 9 COGNEX

Go Online, trigger the camera a few times to capture images.
Go Offline and check the FTP folder.

There will be BMP files for each image captured in the folder.

Communicating with PLCs



Section 7 | Slide 10 COGNEX

This section gives an overview of how an In-Sight can send or receive values with a PLC in EasyBuilder Mode. It assumes familiarity with the PLC end of the communications.

Implicit vs Explicit Messaging

Implicit Message

- Factory Floor Protocol
- Automatic Send/Receive/Reply
- Prebuilt Functions

Explicit Messaging

- Native Mode Commands
- Only sent/reply when told
- Must send each command

Instance 12: In-Sight Firmware Version 4.10.x

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
12	0	Online	Offline Reason			Missed Acq	Acquiring	Trigger Ack	Trigger Ready
	1	Reserved	Command Failed	Command Completed	Command Executing	Results Valid	Results Buffer Overrun	Inspection Completed	Inspecting
	2	Reserved							
	3	Reserved		Test Run Ready	Job Pass	Exposure Complete	Reserved		
	4	Current Job ID (16-bit integer)							
	5								
	6	Acquisition ID (16-bit integer)							
	7								
	8	Inspection ID (16-bit integer)							
	9								
	10	Inspection Result Code (16-bit integer)							
	11								
	12	Inspection Results 0							
	...								
	499	Inspection Results 487							

Name	Data Type	Description
⊞ InSight_SetJobMsg	MESSAGE	The MSG instruction data
⊞ InSight_JobName	STRING	The new job name
⊞ InSight_SetJobData	SINT[32]	The data sent via the MSG instruction
InSight_TriggerSetJob	BOOL	0 -> 1 = Send the SetJobMsg

There are two kinds of messaging between In-Sight and the PLC: Implicit and Explicit.

Implicit Messaging is when a server sends information from predefined memory locations to a client at a given interval. For PLCs, this given interval is the RPI cycle time. Implicit Messaging uses UDP and is faster than Explicit Messaging and is thus intended for time-critical applications.

Example Results/Output Assembly

Sent from Camera to PLC

Pre-Determined

Instance 12: In-Sight Firmware Version 4.10.x

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
12	0	Online	Offline Reason			Missed Acq	Acquiring	Trigger Ack	Trigger Ready
	1	Reserved	Command Failed	Command Completed	Command Executing	Results Valid	Results Buffer Overrun	Inspection Completed	Inspecting
	2	Reserved							
	3	Reserved		Test Run Ready	Job Pass	Exposure Complete	Reserved		
	4	Current Job ID (16-bit integer)							
	5								
	6	Acquisition ID (16-bit integer)							
	7								
	8	Inspection ID (16-bit integer)							
	9								
	10	Inspection Result Code (16-bit integer)							
	11								
	12	Inspection Results 0							
	...								
	499	Inspection Results 487							

Formatted using FormatOutputBuffer and WriteResultsBuffer

With **Implicit Messaging**, you also need to set up the Assembly Objects on the PLC.

Assembly Objects represent the structure of the packet sent between the PLC and the Camera. Each bit, or group of bits, is labeled according to its function and position in the packet.

When doing There are always two Assembly Objects: (1) one for sending from the camera to the PLC (Input Assembly) , and (2) one for sending from the PLC to the camera (Output Assembly) . Each Industrial Protocol has its own packet description, but many of the Assembly Objects tend to have the same bits with very similar functionality.

Assembly Objects come in handy when you are unable to use the Add-On Profile (AOP) or Copy Rung Instruction. If you are not using the AOP, even if you are using the EDS (Electronic Data Sheet) file, your bits will not be labeled; all you will have is a large space in memory filled with bits. Using the position of the bit on this map, it is possible to determine functionality without having them labeled. This can also come in handy for troubleshooting, specially if the memory space on the PLC is not properly aligned with the bits.

Formats vary with the version of In-Sight. Complete formats and explanations are found in the In-Sight Explorer Help menu under:

"EtherNet/IP Object Model - In-Sight 4.x.x Firmware"
"EtherNet/IP Object Model - In-Sight 5.x.x Firmware"

Example Control/Input Assembly

From PLC to Camera

Pre-Determined

Instance 21: In-Sight Firmware Version 4.10.x

Instance	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
21	0	Force Offline	Reserved		Execute Command	Inspection Results Ack	Buffer Results Enable	Trigger	Trigger Enable
	1	Soft Event 7	Soft Event 6	Soft Event 5	Soft Event 4	Soft Event 3	Soft Event 2	Soft Event 1	Soft Event 0
	2	Command							
	3								
	4	User Data 0							
	...								
	495	User Data 491							

Read using FormatInputBuffers and WriteUserDataBuffer

With **Implicit Messaging**, you also need to set up the Assembly Objects on the PLC.

Assembly Objects represent the structure of the packet sent between the PLC and the Camera. Each bit, or group of bits, is labeled according to its function and position in the packet.

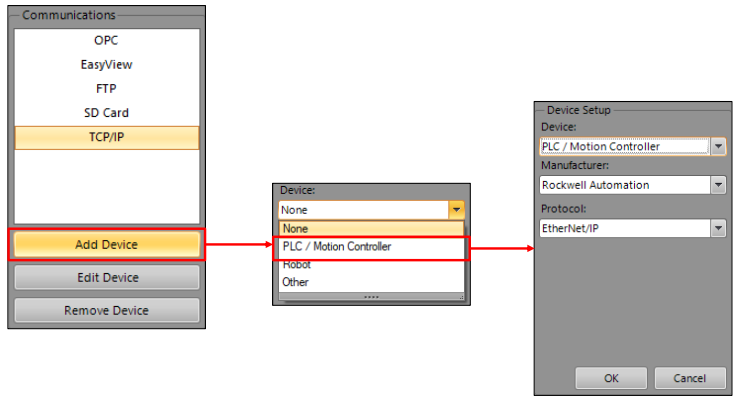
When doing There are always two Assembly Objects: (1) one for sending from the camera to the PLC (Input Assembly) , and (2) one for sending from the PLC to the camera (Output Assembly) . Each Industrial Protocol has its own packet description, but many of the Assembly Objects tend to have the same bits with very similar functionality.

Assembly Objects come in handy when you are unable to use the Add-On Profile (AOP) or Copy Rung Instruction. If you are not using the AOP, even if you are using the EDS (Electronic Data Sheet) file, your bits will not be labeled; all you will have is a large space in memory filled with bits. Using the position of the bit on this map, it is possible to determine functionality without having them labeled. This can also come in handy for troubleshooting, specially if the memory space on the PLC is not properly aligned with the bits.

Formats vary with the version of In-Sight. Complete formats and explanations are found in the In-Sight Explorer Help menu under:

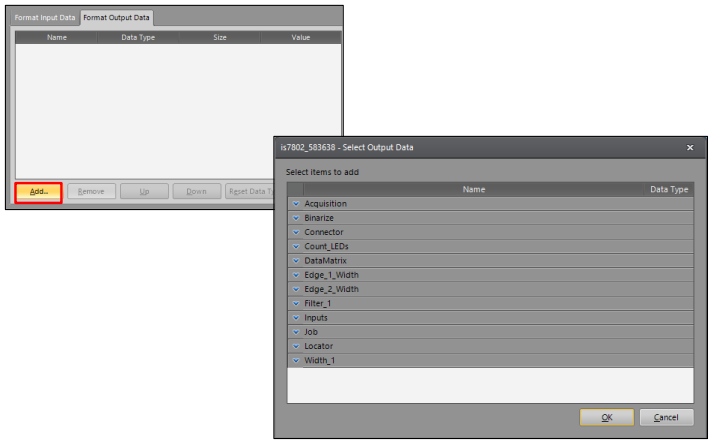
"EtherNet/IP Object Model - In-Sight 4.x.x Firmware"
"EtherNet/IP Object Model - In-Sight 5.x.x Firmware"

Rockwell ControlLogix: Communication



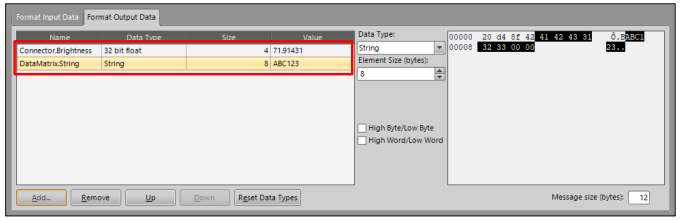
To send or receive values with a PLC, go to the Communication step in EasyBuilder. Navigate through the pull-down menus until to get to PLC/Motion Controller – Rockwell Automation – Ethernet/IP. Then click the OK button.

Rockwell ControlLogix: Communication



You will see at the bottom of the EasyBuilder window two tabs: **Format Input Data** and **Format Output Data**. Each allows you to select parameters from the vision tools in your job.

Rockwell ControlLogix: Communication



In this slide, we are showing two variables that have been selected to be output: the **Brightness** value from a tool named “Connector” and the **String** read by an IDCode tool. Default data types of 32 bit float and string are shown, buy these can be changed by using the **Data Type** pull-down menu. At the very right, the actual data that will be sent to the PLC is displayed.

Explicit Messaging

- Better suited for operations that occur less frequently
- No need for Factory Floor Protocols
- PLC issues MSG instructions to In-Sight, usually **Native Mode** commands, via telnet protocol

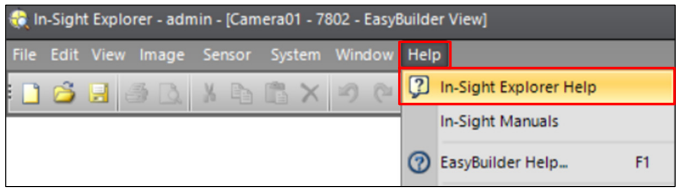
Example: change job on camera

Controller Tags

Name	Data Type	Description
⊞ InSight_SetJobMsg	MESSAGE	The MSG instruction data
⊞ InSight_JobName	STRING	The new job name
⊞ InSight_SetJobData	SINT[32]	The data sent via the MSG instruction
InSight_TriggerSetJob	BOOL	0 -> 1 = Send the SetJobMsg

Unlike Implicit Messaging, there is no RPI. Usually, the PLC sends an MSG to In-Sight Explicit, set up for a Native Mode command.

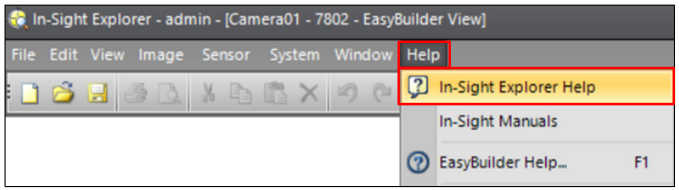
Rockwell ControlLogix: Help Topics



Get Data from an In-Sight Vision System
Send Data to an In-Sight Vision System
EtherNet/IP PLC Setup
Install the EDS Files
Industrial Communication Overview
EtherNet/IP I/O Assembly Data Format Assemblies

Use the In-Sight Explorer Help file to answer your Rockwell PLC questions. Above are a few topics to get you started.

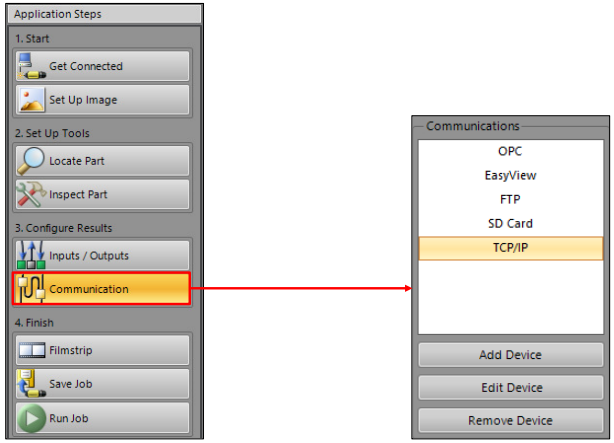
Other PLCs: Help Topics



PROFINET PLC Setup
Configuring the PLC Hardware
Setting up the Siemens S7 PLC
Triggering the Camera from the PLC
Reading Data from the PLC
Writing Data to the PLC

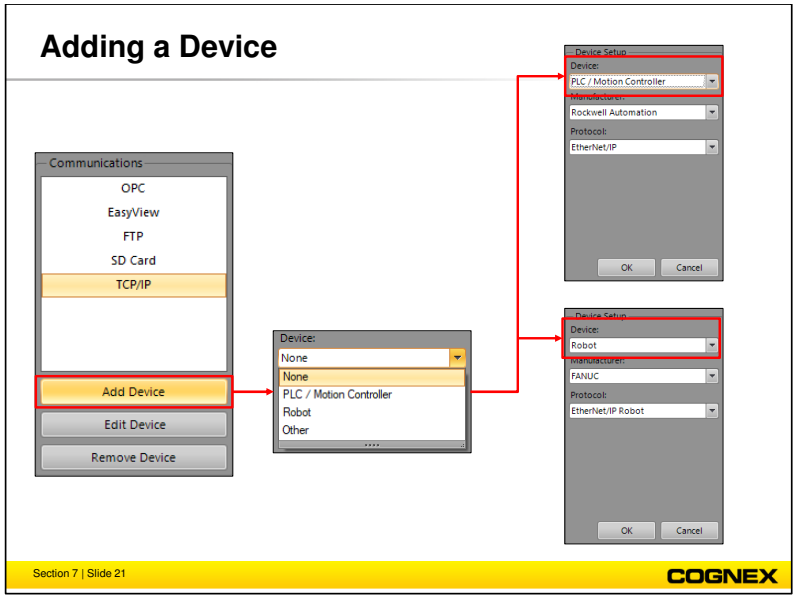
Use the In-Sight Explorer Help file to answer your PLC questions. Above are a few topics to get you started.

Quick Setup of Communications



Select a communication type to either configure OPC tags to the In-Sight OPC Server or an EasyView for the VisionView Operator Interface Panel.

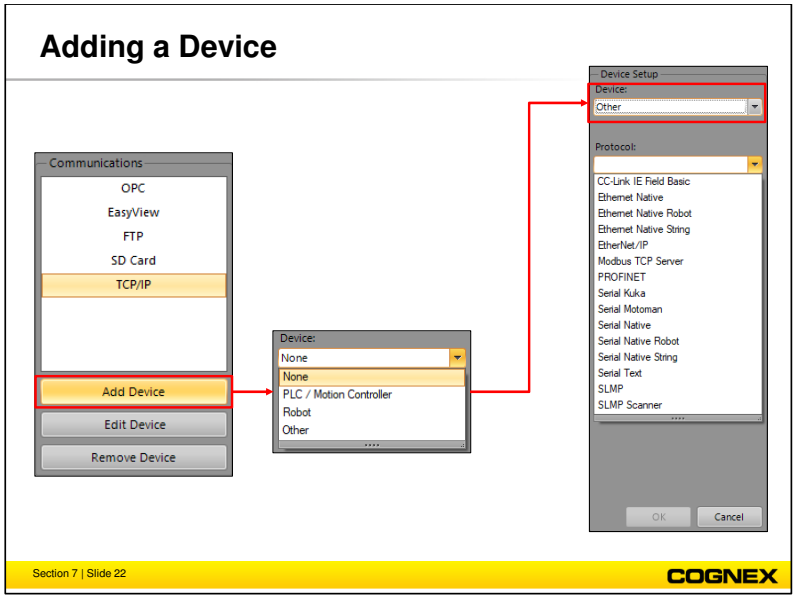
Or, click the **Add Device** button to configure communications to an external device.



Choose a communication protocol by selecting a Device, Manufacturer (if applicable), and Protocol. When selecting, the drop-down lists are contingent upon what is selected in the list above. If you do not see your desired selections, set Device to Other and Manufacturer to Other to enable all of the Protocol selections. Click **OK** to confirm.

Device Settings:

- **None** – Disables communication via Ethernet and Serial.
- **PLC / Motion Controller** – Prepares communications protocol settings for PLCs.
- **Robot** – Prepares communications protocol settings for robots.
- **Other** – Allows custom communications configurations.



An In-Sight camera can be configured to act as a TCP/IP client device, TCP/IP is a standard Ethernet protocol (Transmission Control Protocol / Internet Protocol) opening a connection between the In-Sight camera and another TCP/IP device for sharing data over the network.

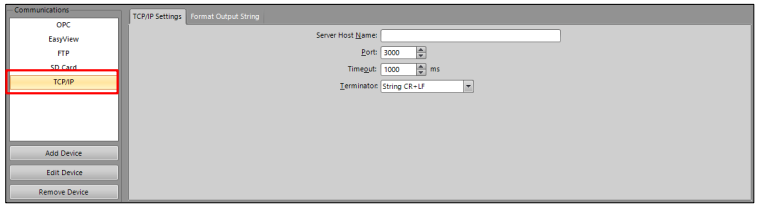
Once the TCP/IP connection has been established, EasyBuilder will initiate the communication with another TCP/IP device on the network.

NOTE: *Selecting **Other** allows for custom configuration of communications settings.*

To configure an In-Sight camera for client device TCP/IP service:

1. In the Communications group box, press the **Add Device** button.
2. From the Device Setup group box, select **Other** from the Device drop-down list.
3. Select **TCP/IP** from the Protocol drop-down box.
4. Press the **OK** button.

Example – Adding TCP/IP



Section 7 | Slide 23 COGNEX

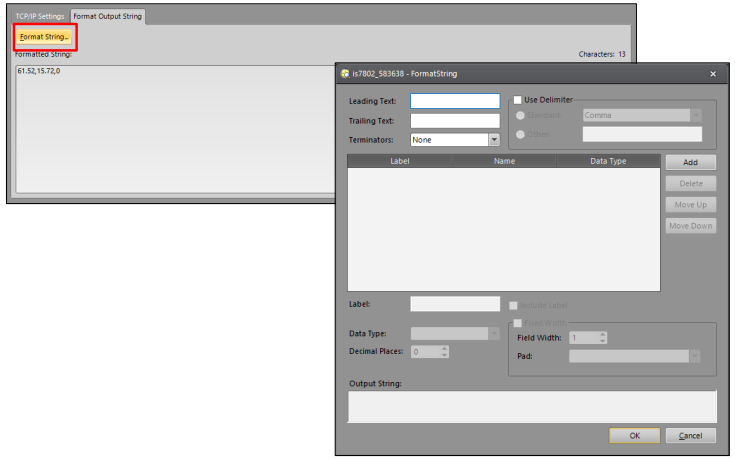
5. Click the **TCP/IP Settings** tab and enter the name of the TCP/IP service device with which to establish connection in the Server host Name field.
6. Enter the **Port** Number of the TCP/IP server at which the connection will be established between TCP/IP devices in the Port field. The Port number entered must also be assigned on the server.

NOTE: *A valid Port assignment is any unused number between 1 and 65,535, except for ports used for In-Sight communications. In-Sight Emulator users should always assign Port numbers **3000** (the default) and higher to prevent conflicts with ports reserved by services on the PC.*

7. Enter the milliseconds (100 to 15,000; default = 1000) to wait for the TCP/IP connection to be established before aborting the connection in the **Timeout** field.
8. From the **Terminator** drop-down list, select a String with Customer Terminator to specify an ASCII character (0 to 255).

NOTE: *Can be set to Client (opens port) and Server (listens for data).*
If In-Sight is Client, Server Host Name is the IP Address or Name of the device being connected to. If Server, then the Server Host Name is left blank. The Port and Terminator need to match on the Client and Server

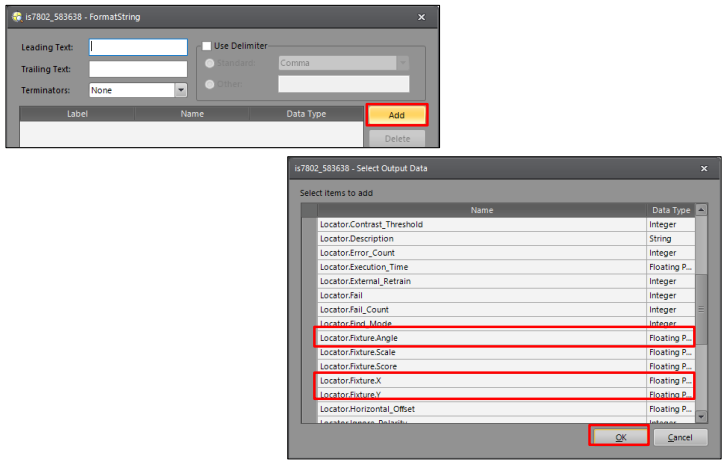
Selecting Values



Section 7 | Slide 24 COGNEX

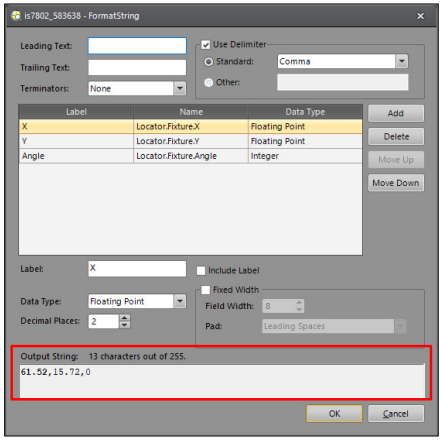
9. Once the TCP/IP settings have been established, click the **Format Output String** tab to select and format the data that will be transmitted.
10. Click the **Format String** button, which will launch the FormatString dialog where you will be able to add, customize and modify the order of the data sent.

FormatString Settings



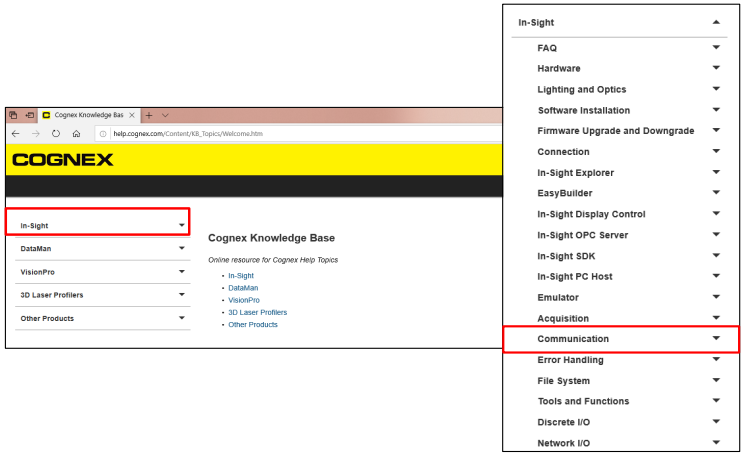
11. Within the FormatString dialog, click the **Add** button, which will launch the Select Output Data dialog.
12. The Select Output Data dialogs contain the data from any Location or Inspection Tools that were added to your job and the overall job results. From the dialog, select the appropriate data that you want sent from the In-Sight camera, and press the **OK** button.

FormatString Settings



13. After you have selected your data in the Select Output Data dialog the **FormatString** dialog will reappear.
14. You can modify the strings and/or rearrange the order of the data that will be sent by selecting the data from the list and clicking either the **Up** or **Down** buttons to set your desired order. Once you have finalized your string modifications, press the **OK** button.
15. Now that the string has been formatted, the output data will be sent after the camera completes its job execution.

Cognex Knowledge Base



Section 7 | Slide 27 COGNEX

Cognex Knowledge Base is another resource available to help you find answers to your Communication related questions.
The site to access the Knowledge Base is help.cognex.com.

Summary

- In-Sight is able to communicate with many 3rd party applications
- An In-Sight Emulator can be utilized as an Authorized FTP Directory to capture BMP images
- PLC communications may be Implicit or Explicit, and may require an AOP or EDS file, as well as assembly objects

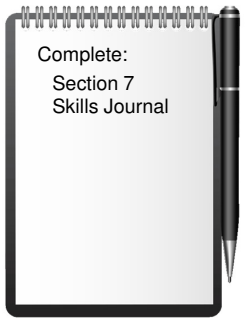
Section 7 | Slide 28 COGNEX

In this section we covered the following topics:

- In-Sight is able to communicate with many 3rd party applications
- An In-Sight Emulator can be utilized as an Authorized FTP Directory to capture BMP images
- PLC communications may be Implicit or Explicit, and may require an AOP or EDS file, as well as assembly objects

Skills Journal

Lab Exercise



Section 7 | Slide 29

COGNEX

Complete:

Skills Journal (image designed by pngtree)

Lab Exercise