COGNEX

PROFINET COMMUNICATIONS

From Siemens S7 to In-Sight, DataMan and Checker Using Portal

The information contained in this document has been developed solely for the purpose of providing general guidance to Cognex customers who need to configure communications between Cognex products and a SIMATIC S7-series PLC via PROFINET protocol using Portal software. The data contained in this document serves informational purposes only.

Information in this document does not represent a commitment on the part of Cognex and in especially is not intended to be binding upon Cognex to any particular course of business. Cognex assumes no responsibility for errors or omissions in this document. Cognex does not make any express or implied representation or warranty as to the accuracy or completeness of the information for a particular purpose. Cognex shall have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of this document.

COGNEX

Table of Contents

Revision History

Date	Description	Initials
17-APR-2013	Initial Release	KRC

Introduction

This document gives a general overview of how to connect various Cognex products to a Siemens S7 processor using Portal software.

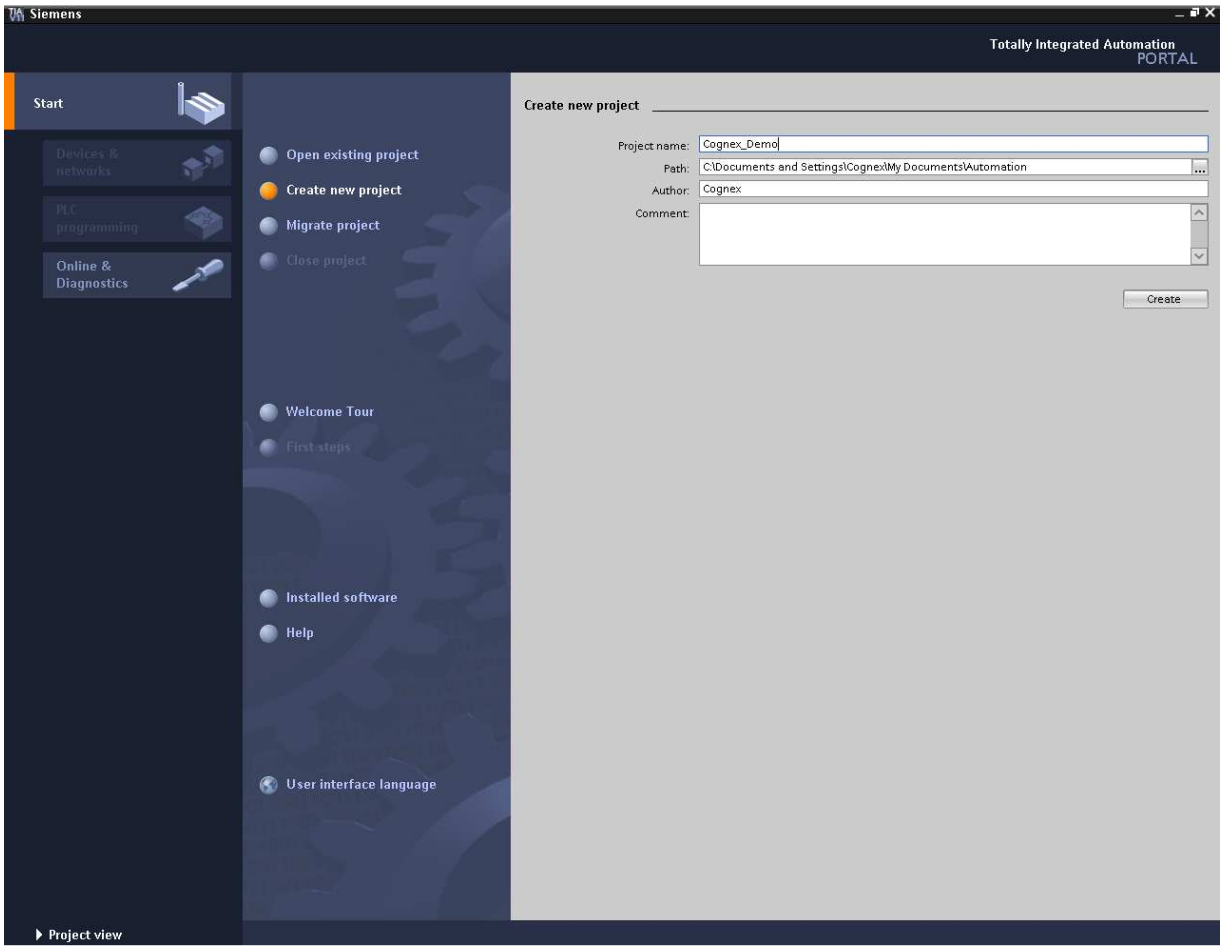
Also, please refer to the DataMan Control Command Reference document installed with the DataMan Setup Tool. This is located under the Setup Tool->Documentation directory.

Included with this document are two sample programs. The first is the PLC program named Cognex_Demo. This program was developed using Portal V11 SP2 Update 5. Also included is an In-Sight job file: Profinet_Demo.job. This was developed with version 4.7.3 of In-Sight Explorer.

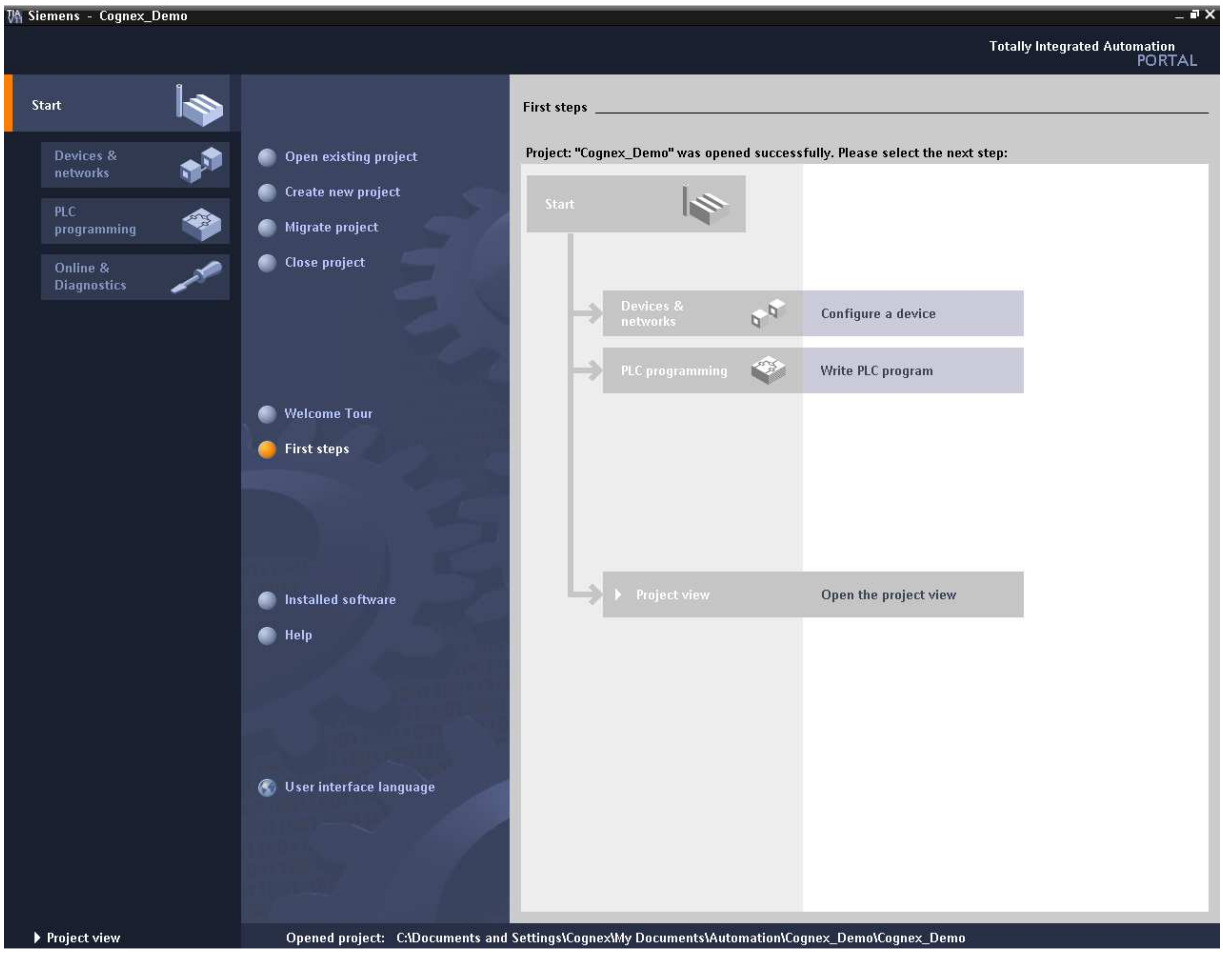
It is assumed that the Cognex library files for In-Sight, DataMan and Checker have been installed in Portal. If not, refer to the Siemens documentation on how to install library files.

Create a New PLC Program

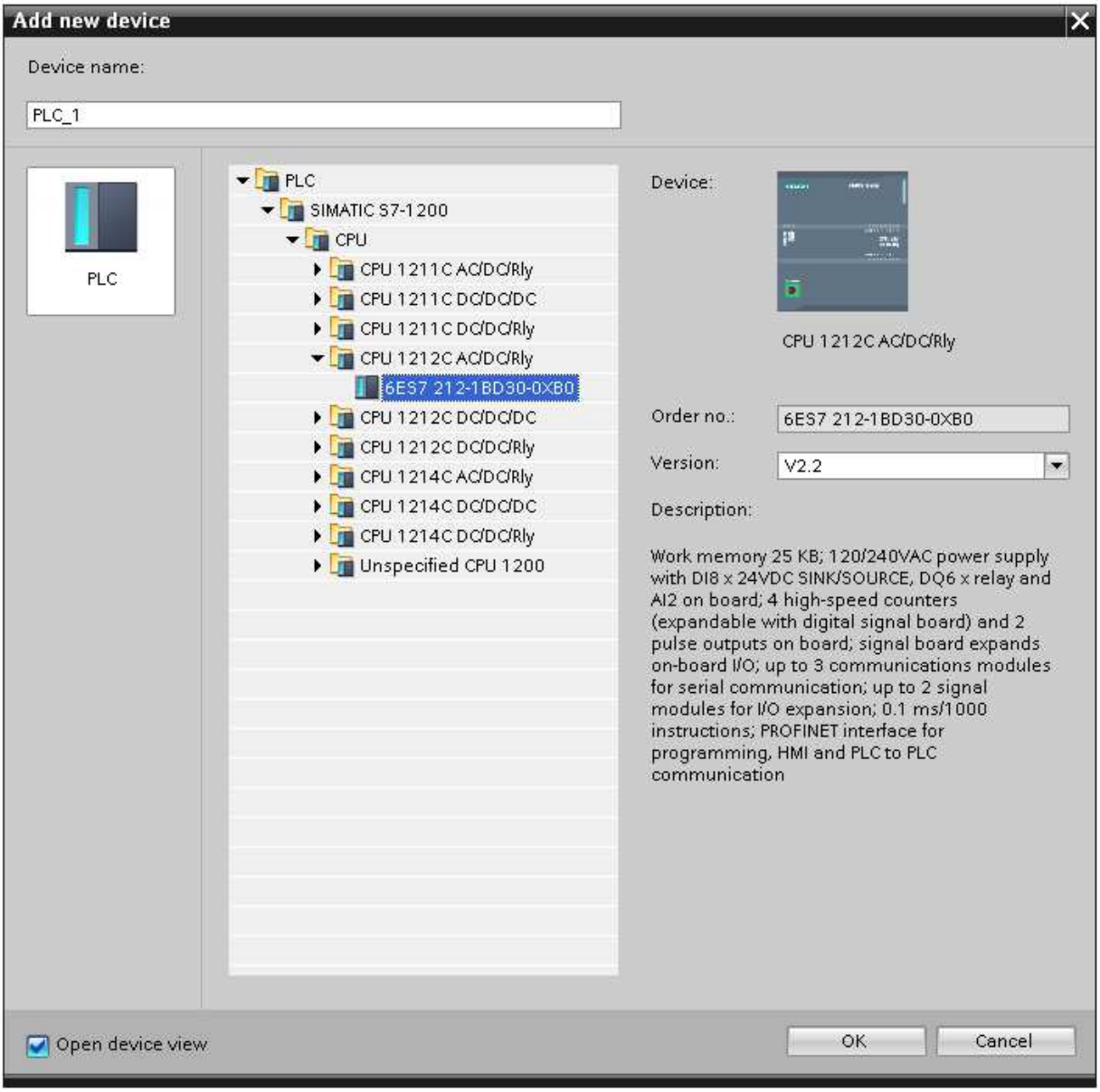
Start the Portal software and create a new project called Cognex_Demo.



The ‘First Steps’ screen appears. Select ‘Open the Project View.’

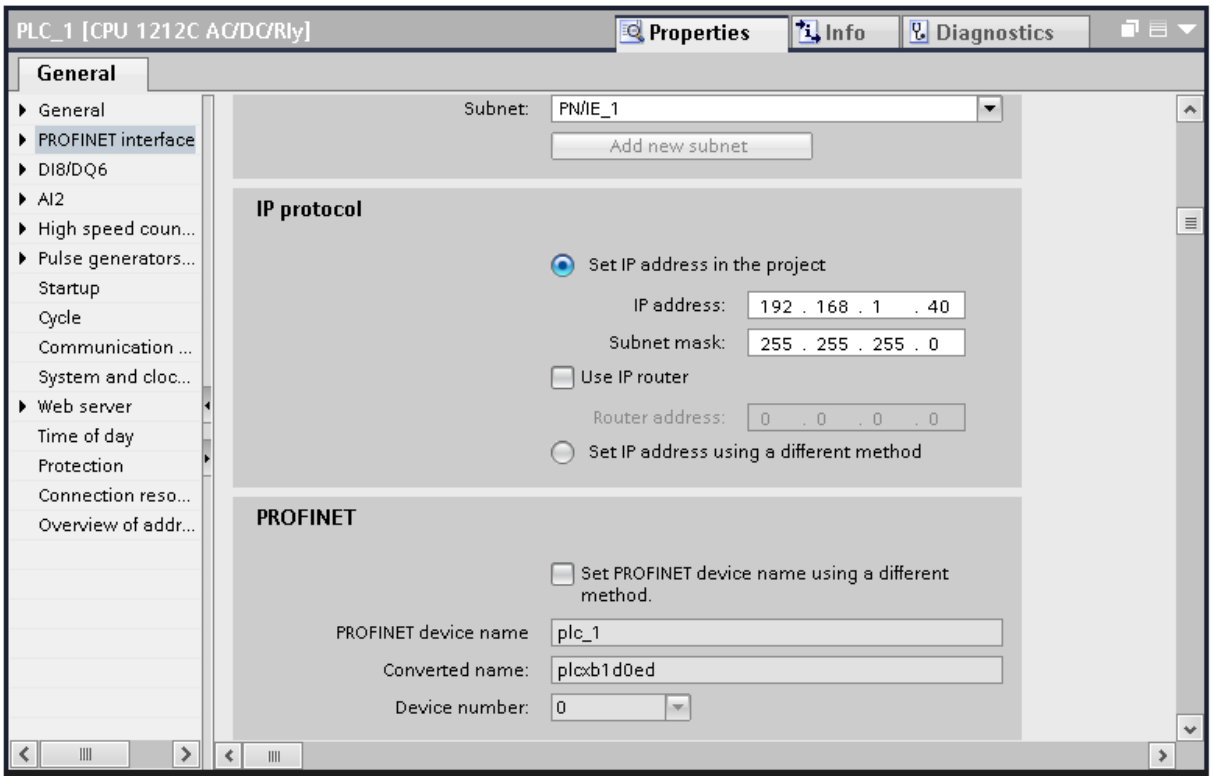


Add a new device. For this example, add CPU 1212C AC/DC/RLY.



Press ‘OK.’

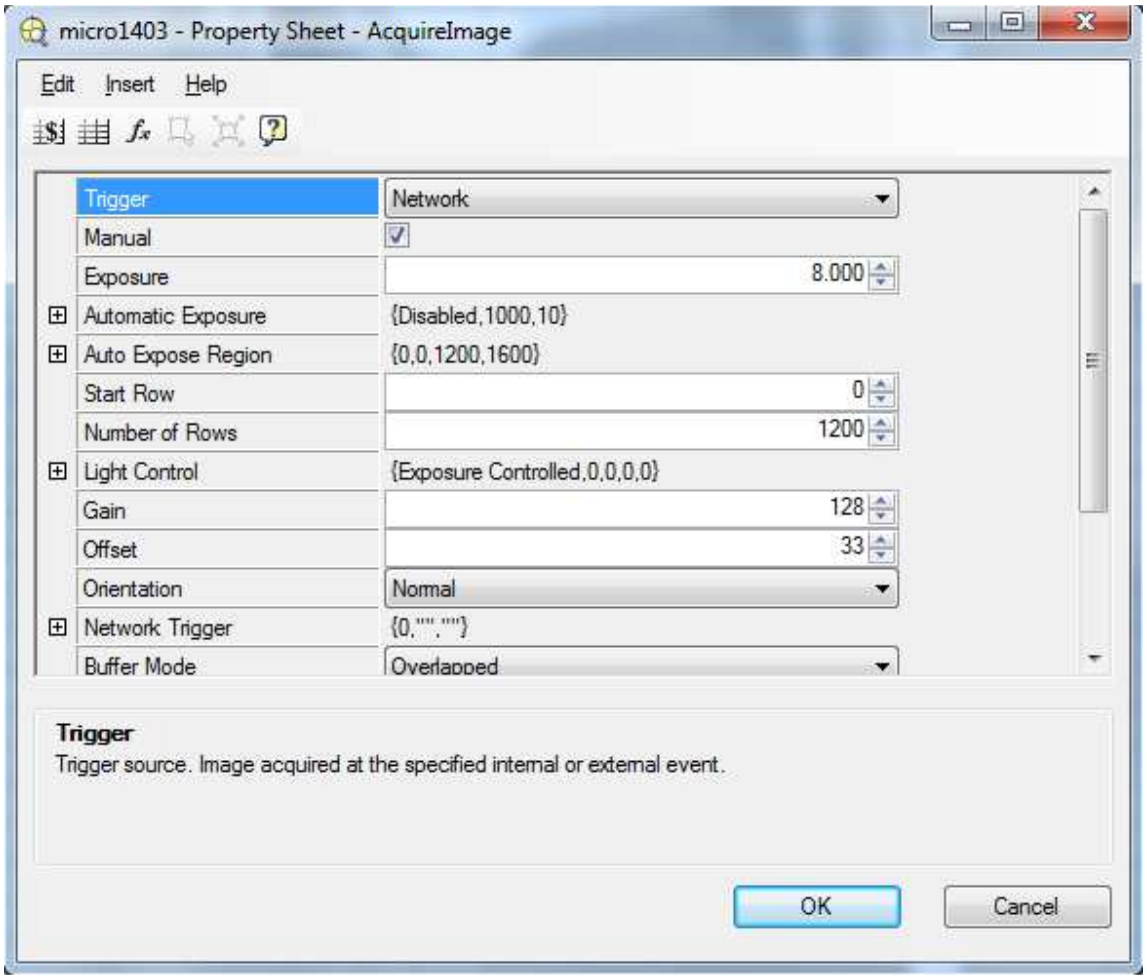
Under ‘Devices & Networks,’ select ‘Network View.’ Select the PLC. Under the ‘General Properties’ tab, select ‘PROFINET Interface.’ Change the IP Address to 192.168.1.40.



Connecting to In-Sight Cameras

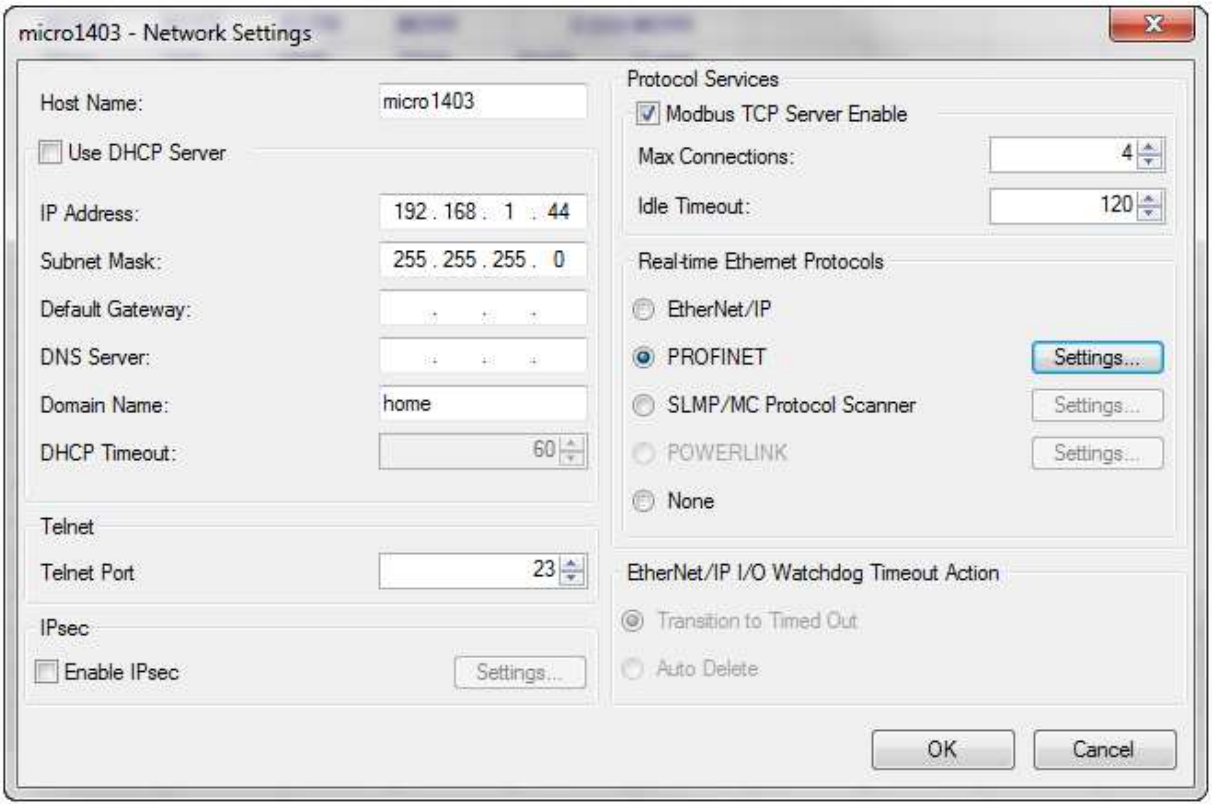
In-Sight Configuration

For this example, an In-Sight 1403 was used. The only device configuration necessary, except network settings, is to set the Trigger to 'Network':

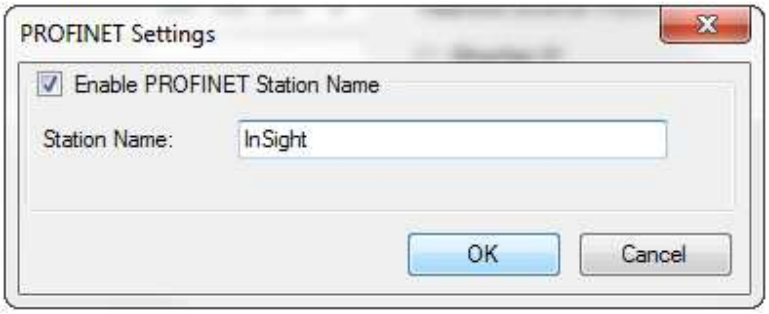


It is assumed that the user is currently connected to the camera. If not, add the camera to your network by going to System->Add Sensor/Device to Network. Select the camera by MAC address and set the IP address.

In In-Sight Explorer, go to Sensor->Network Settings. This will display a popup, allowing the user to configure a static IP address. For this example, set the IP address to 192.168.1.44:



Select ‘PROFINET’ on the ‘Network Settings’ popup to enable PROFINET communications. Then, press the ‘Settings’ button.



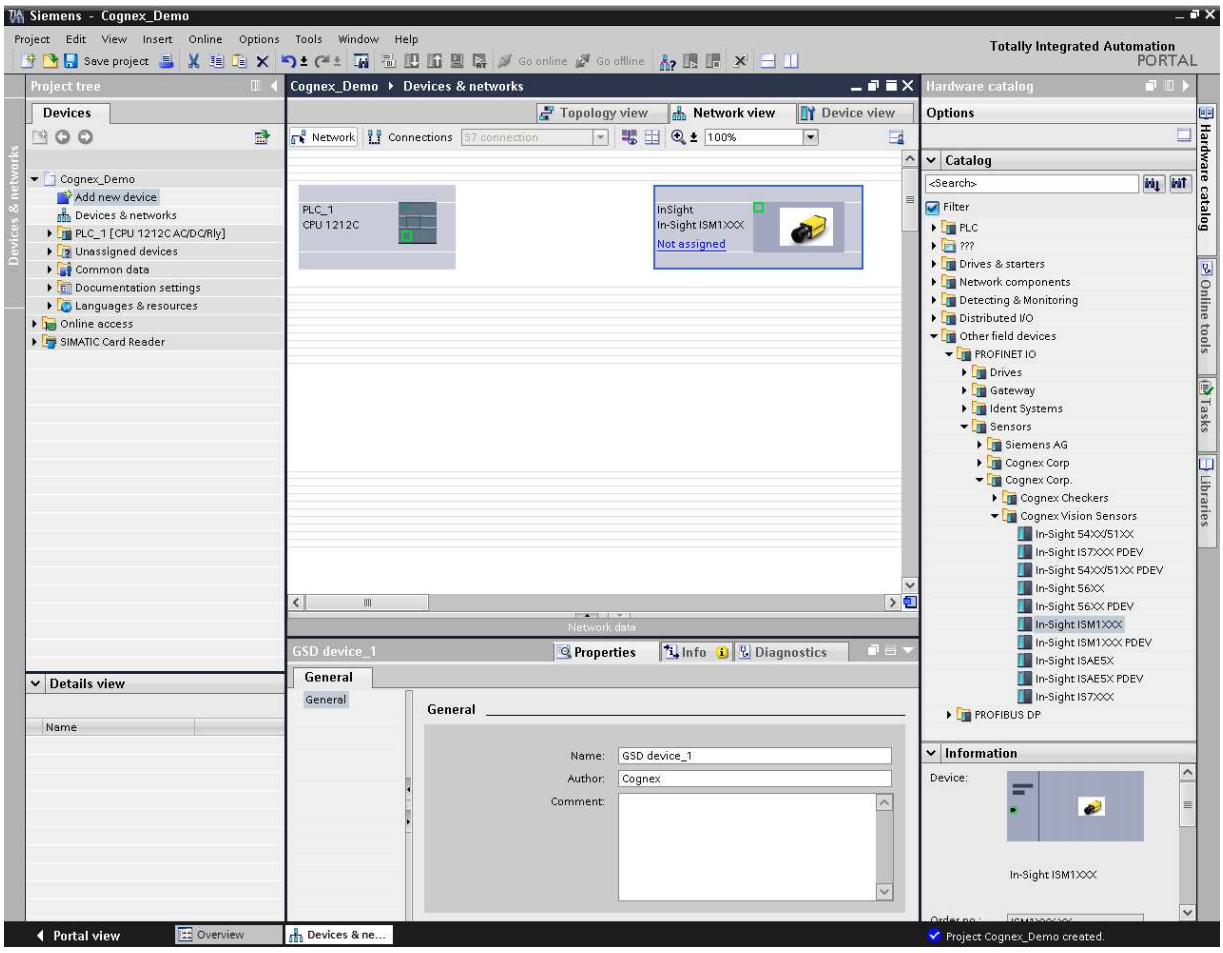
Select the ‘Enable PROFINET Station Name’ checkbox and type ‘InSight’ as the Station Name.

Press ‘OK’ on the ‘PROFINET Settings’ popup, as well as the ‘Network Settings’ popup. A message will appear stating that you must restart the system. Press ‘Yes’ to restart the system.

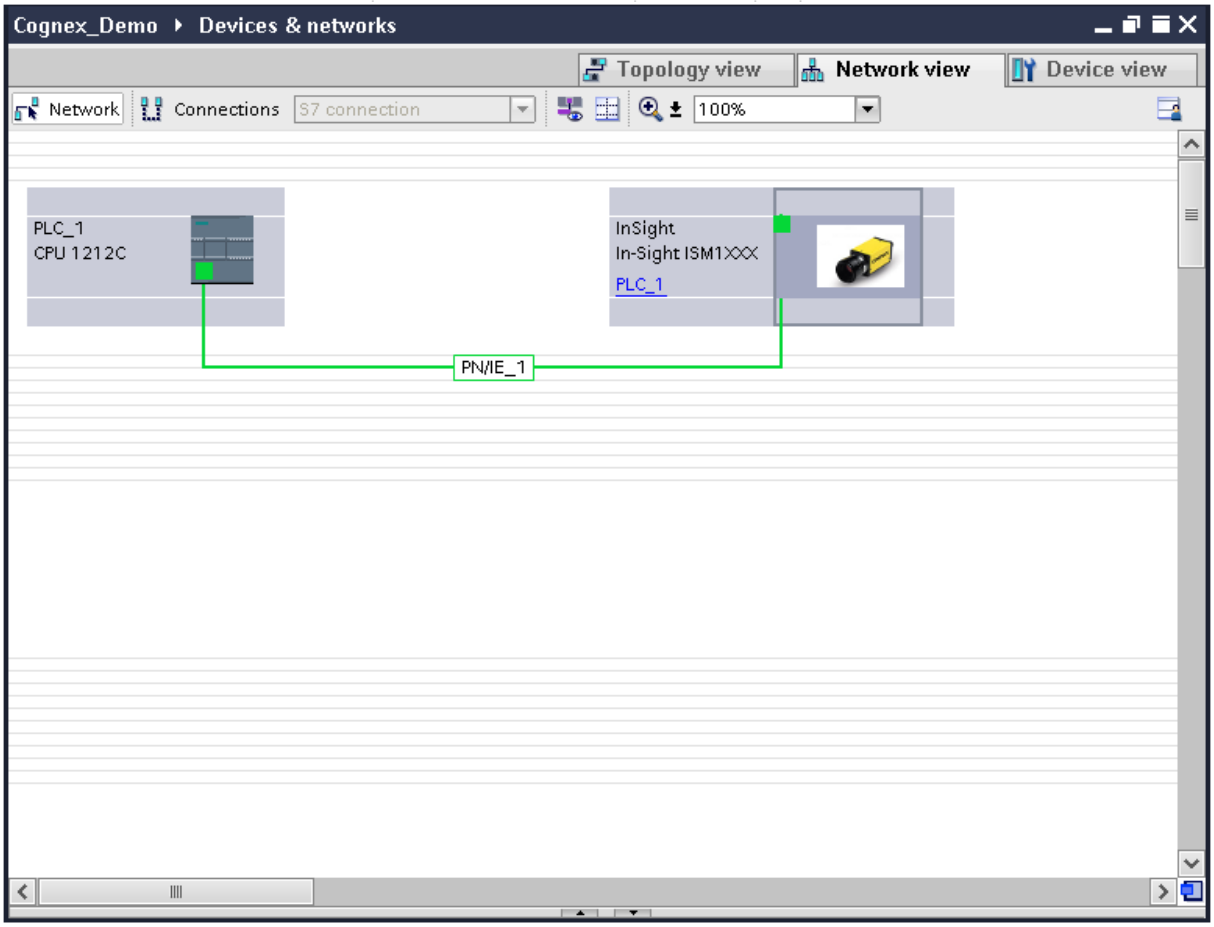
PLC and Spreadsheet Programming

Add In-Sight Camera to Network

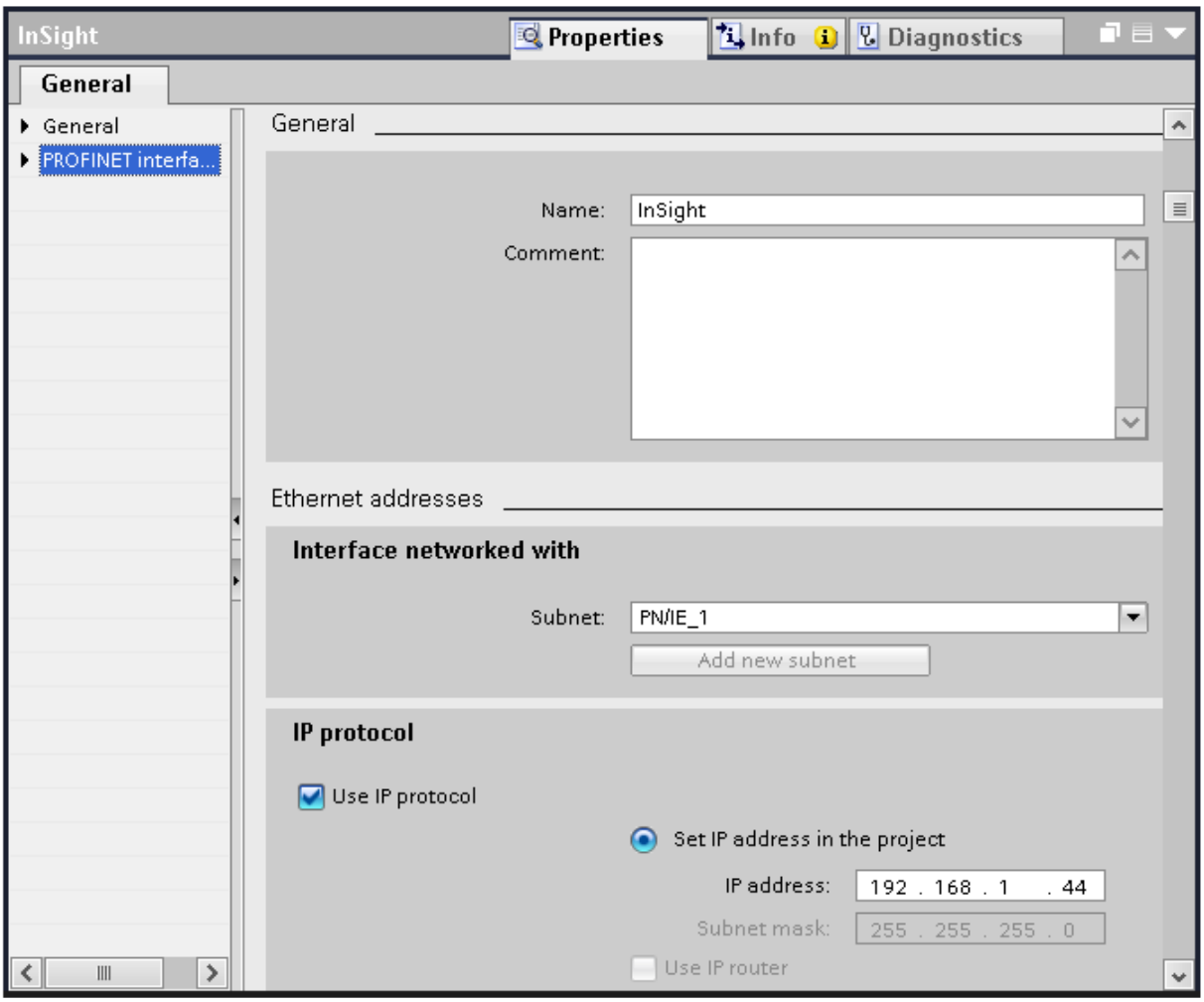
Under 'Devices & Networks,' select 'Network View.' Add an In-Sight Micro (ISM) camera. This is listed under 'Other Field Devices'➔PROFINET IO➔Sensors➔Cognex Corp➔Cognex Vision Sensors. A device can be added by dragging the library object into the 'Network View.' The ISM device will be listed as 'Not Assigned.'



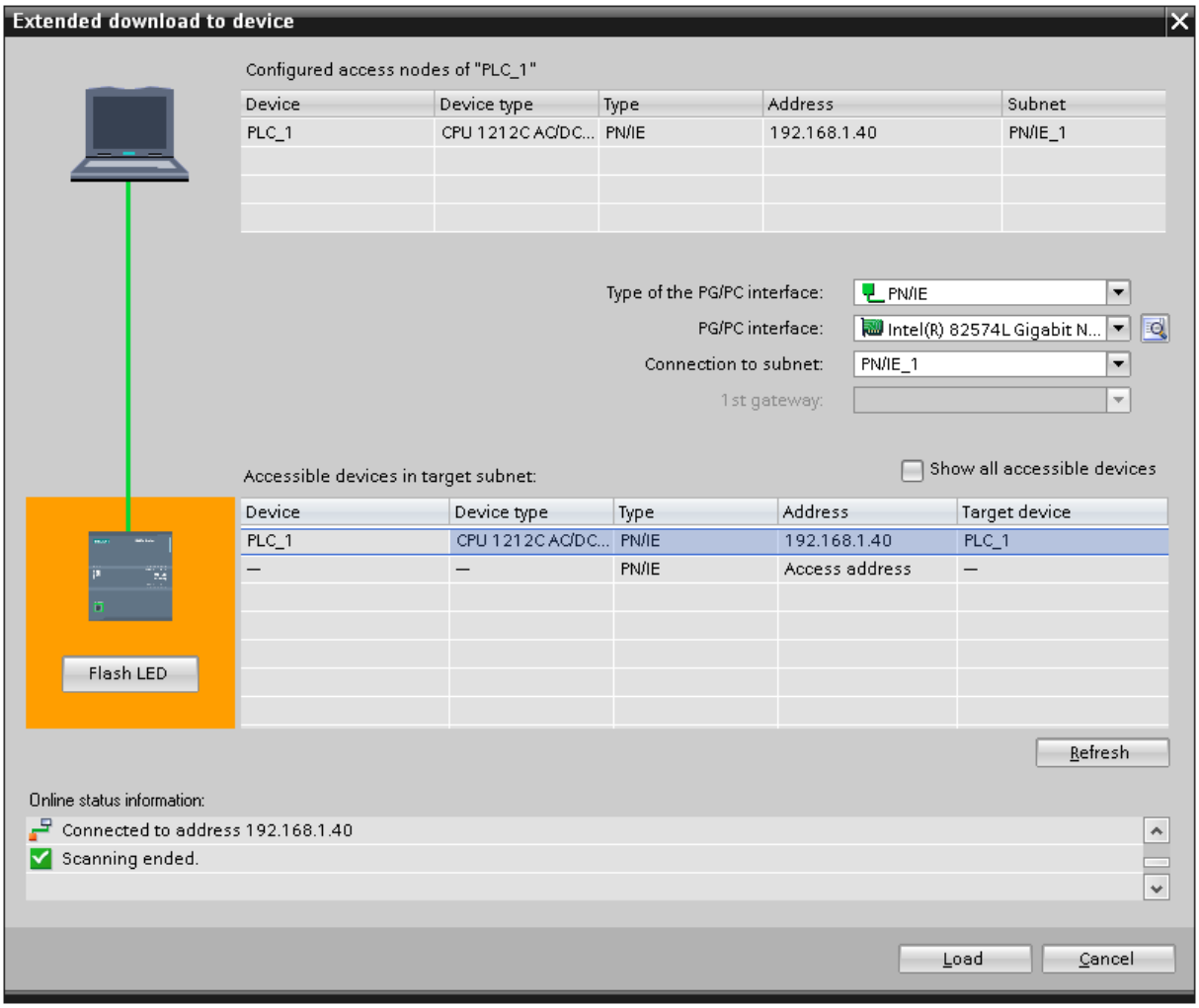
To assign the device, press and hold the left mouse button on the green box in the ISM object, and drag your mouse to the green box in the PLC. Release the mouse button. This establishes the connection between the PLC and the In-Sight camera.



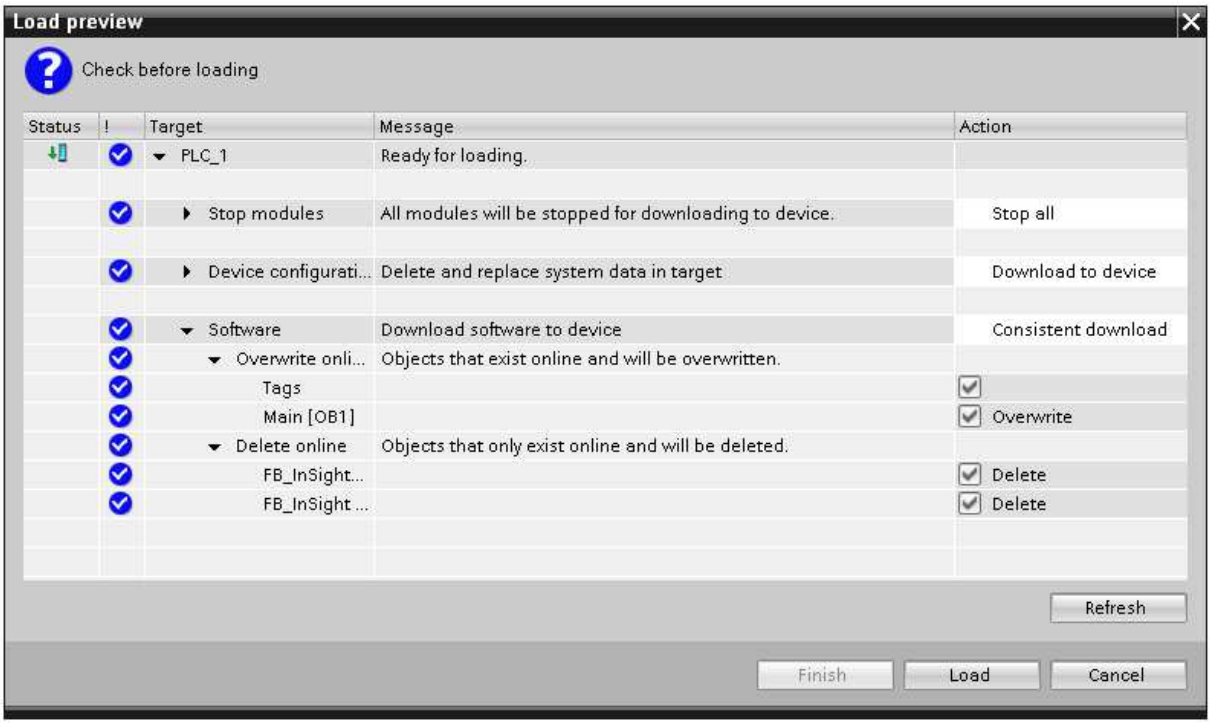
Select the camera. Under the ‘General Properties’ tab, select ‘PROFINET Interface.’ Change the name to InSight and change the IP Address to 192.168.1.44.



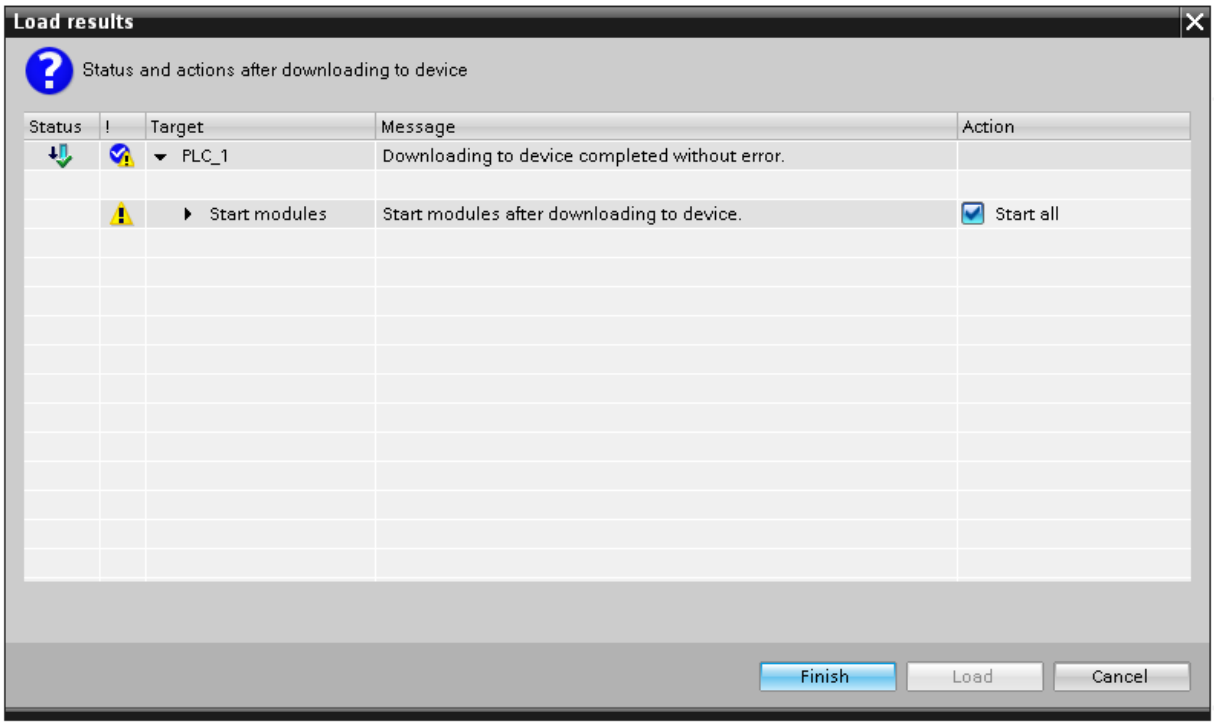
The program is ready to download to the PLC to check communication status. Select the PLC in the Project Tree. Under the ‘Online’ menu, select ‘Download to Device.’ A popup will appear.



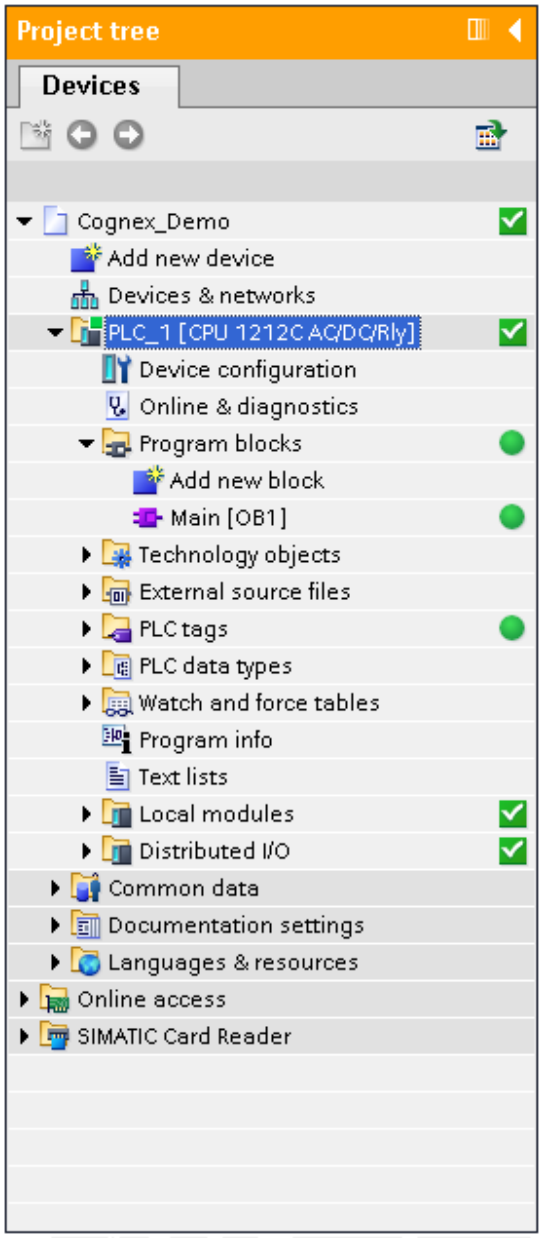
Press ‘Load.’ A new popup will appear.



Press ‘Load.’ A new popup will appear.

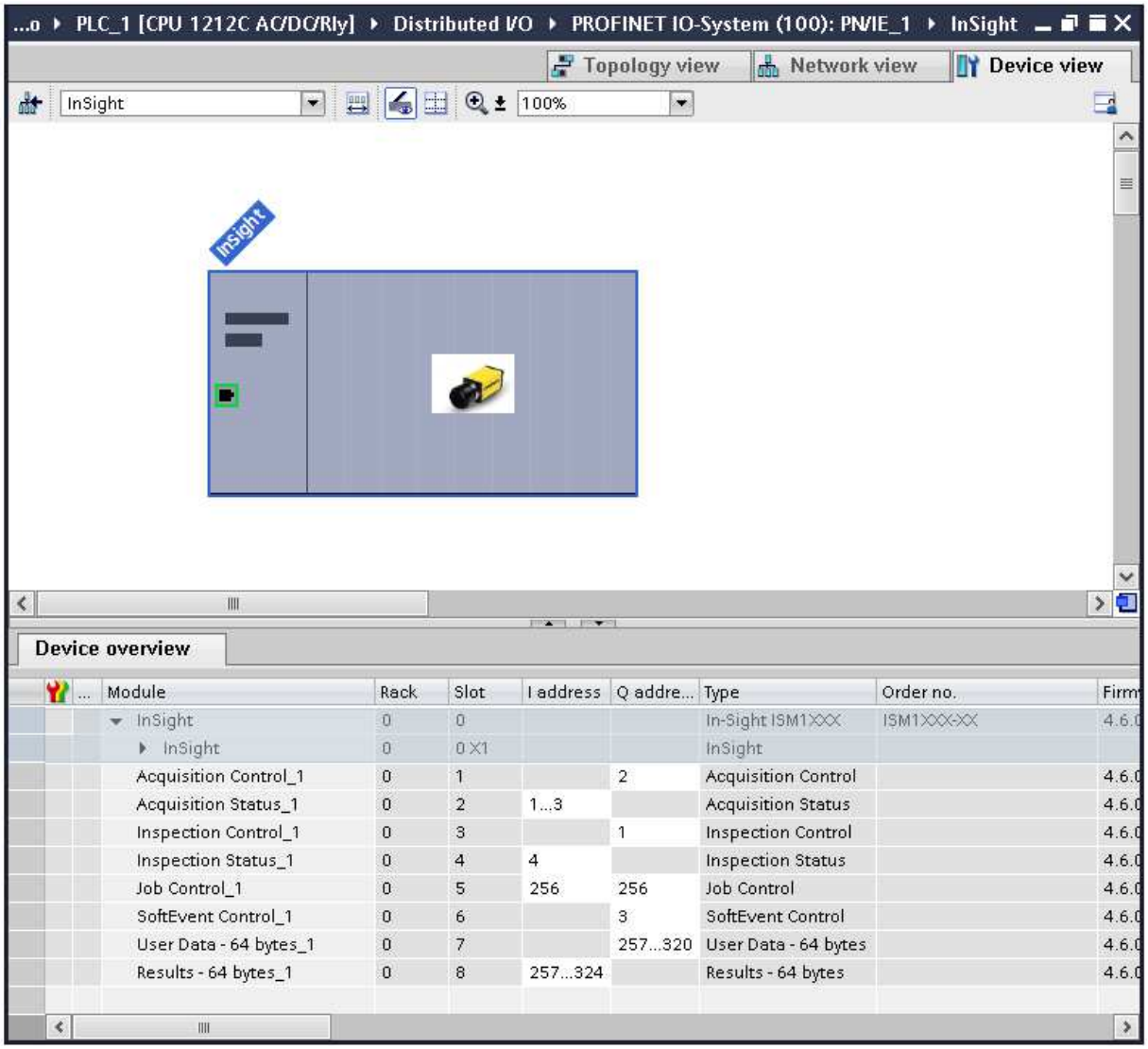


Press ‘Finish’ to finish the download process. Go online with the processor by selecting the PLC in the Project Tree, and select ‘Go Online’ under the ‘Online’ drop-down menu. If communications are correct, green check marks will appear next to all devices.



Device I/O

When the In-Sight object was added to the system, I/O was assigned for the control elements of the camera.



...	Module	Rack	Slot	I address	Q addre...	Type	Order no.	Firm
	InSight	0	0			In-Sight ISM1XXX	ISM1XXX-XX	4.6.0
	InSight	0	0 X1			InSight		
	Acquisition Control_1	0	1		2	Acquisition Control		4.6.0
	Acquisition Status_1	0	2	1...3		Acquisition Status		4.6.0
	Inspection Control_1	0	3		1	Inspection Control		4.6.0
	Inspection Status_1	0	4	4		Inspection Status		4.6.0
	Job Control_1	0	5	256	256	Job Control		4.6.0
	SoftEvent Control_1	0	6		3	SoftEvent Control		4.6.0
	User Data - 64 bytes_1	0	7		257...320	User Data - 64 bytes		4.6.0
	Results - 64 bytes_1	0	8	257...324		Results - 64 bytes		4.6.0

The I/O assignments break down as follows:

Acquisition Control Module		
Bit	**Name**	**Description**
0	Trigger Enable	This field is set to enable triggering via the Trigger bit. Clear this bit to reset the triggering mechanism.
1	Trigger	Setting this bit triggers an acquisition when the following conditions are met: • The camera is Online • The Trigger Enable bit is set • The AcquireImage function's Trigger parameter is set to Network or External
2-6	Reserved	Unused.
7	Set Offline	When this bit is set, the camera is taken Offline. The camera is put Online when this bit is cleared.

Acquisition Status Module		
Bit	**Name**	**Description**
0	Trigger Ready	Indicates when a camera can accept a new trigger. This field is true when the sensor is Online, the Trigger Enable bit is set and the camera is not currently acquiring an image.
1	Trigger Acknowledge	Indicates when a camera has been triggered by the Trigger bit being set. This bit will stay set until the Trigger bit is cleared.
2	Acquiring	Set when a camera is currently acquiring an image. It is set by either the Trigger bit being set or by an external trigger.
3	Missing Acquisition	Set when a camera misses an acquisition trigger. It is cleared when an acquisition is successfully triggered.
4-6	Offline Reason	This field is a three-bit field used to identify the cause of why a camera is offline:
7	Online	This bit is set when the camera is Online and cleared when the sensor is Offline.
8-23	Acquisition ID	This ID increments on the completion of every acquisition, regardless of the trigger source. It can be used to synchronize an acquisition with its Inspection Results.

Reason	Name	Description
0	Online	The camera is Online.
1	Programming	The camera's job is being modified.
2	Discrete Offline	A discrete signal is holding the sensor Offline.
3	Comm. Offline	A communications protocol is holding the sensor Offline.

Inspection Control Module		
Bit	**Name**	**Description**
0	Buffer Results Enable	When this bit is set, the Inspection ID, Inspection Result and Inspection Results fields are held constant until the Inspection Results Acknowledge field has been set. Up to eight inspections are held in the sensor's buffer. The camera will respond to the acknowledgement by clearing the Results Valid bit. Once the Inspection Results Acknowledge field is cleared and there is a new set of rules sent to the PLC, the Results Valid bit will no longer be cleared. If the Inspection Results Acknowledge bit is cleared and there are no more results in the sensor's buffer that are to be sent to the PLC, the Results Valid bit remains cleared.
1	Inspection Results Acknowledge	When the Buffer Results Enable bit is set, the Inspection Results Acknowledge bit acknowledges that the PLC has received the Inspection ID, Inspection Result and Inspection Results data. The next set of inspection results is then sent to the PLC. Clearing the Inspection Results Acknowledge bit causes the camera to set the Results Valid bit if the buffer is not empty.
2-7	Reserved	Unused

Inspection Status Module		
Bit	**Name**	**Description**
0	Inspecting	This bit is set when a camera is running a job.
1	Inspection Completed	This bit is toggled upon the completion of an inspection.
2	Results Buffer Overrun	This field is set when the Buffer Results Enable bit is set and the camera has discarded a set of inspection results because the PLC has not acknowledged the results, and in turn set the Inspection Results Acknowledge bit. Up to eight inspections are held in the camera's buffer. Therefore, this bit is set when the ninth inspection is added to the buffer and will overwrite the eighth inspection in the buffer. The bit is not cleared until a valid inspection occurs and a previous inspection is not overwritten.
3	Results Valid	Set when the Inspection ID, Inspection Result and Inspection Results fields are valid.
4	Job Loading	This bit is set when loading a new job. This bit only functions when the job load was initiated by the PLC using PROFINET.
5	Job Load Complete	This bit is toggled upon completion of a job load operation. This bit only functions when the job load was initiated by the PLC using PROFINET.

6	Job Load Failed	This bit is set when the last job load attempt failed. It is cleared the next time a job is successfully loaded. This bit only functions when the job load was initiated by the PLC using PROFINET.
7	Reserved	Unused.

Other Functions:

Job Control: The input byte of this module indicates the current Job ID of the camera, or 255 if the current job has no ID. Setting the output byte while the camera is Offline will cause the job with the same ID to be loaded. Valid job ID numbers are 1 through 254. The default Job ID prefix is 0, indicating that the job will not be changed. Do not use a Job ID prefix of 255. When a job without a Job ID prefix is loaded via In-Sight Explorer, an input byte of 255 will be returned to the PLC. Therefore, it would be unclear if a job with a Job ID prefix of 255 is loaded, or a job without a Job ID prefix is loaded.

Soft Event Control: Allows spreadsheet soft events to be triggered. Setting any of these bits causes the associated soft event in the In-Sight Explorer spreadsheet to be triggered.

User Data: This data can be read from the In-Sight spreadsheet, using the ReadProfinetBuffer function.

Inspection Results: This is the data that is written from the In-Sight spreadsheet, using the WriteProfinetBuffer function. If the Buffer Results Enable bit of the Inspection Control Module is set, then the inspection results will remain unchanged until acknowledged by pulsing the Inspection Results Acknowledge bit.

Inspection Results Module		
Byte	**Name**	**Description**
0-1	Inspection ID	The Acquisition ID associated with this set of results.
2-3	Inspection Result Code	Currently unused. Always 0.
4-259	Inspection Results	Inspection result data written from the spreadsheet, using the WriteProfinetBuffer function.

Device Operation

Acquisition Sequence

The camera can be triggered to acquire images by several methods. It can be done explicitly by manipulating the Trigger bit of the Acquisition Control Module, or it can be triggered by external hard wired input. Manipulating the Acquisition Control Module bits will be discussed here.

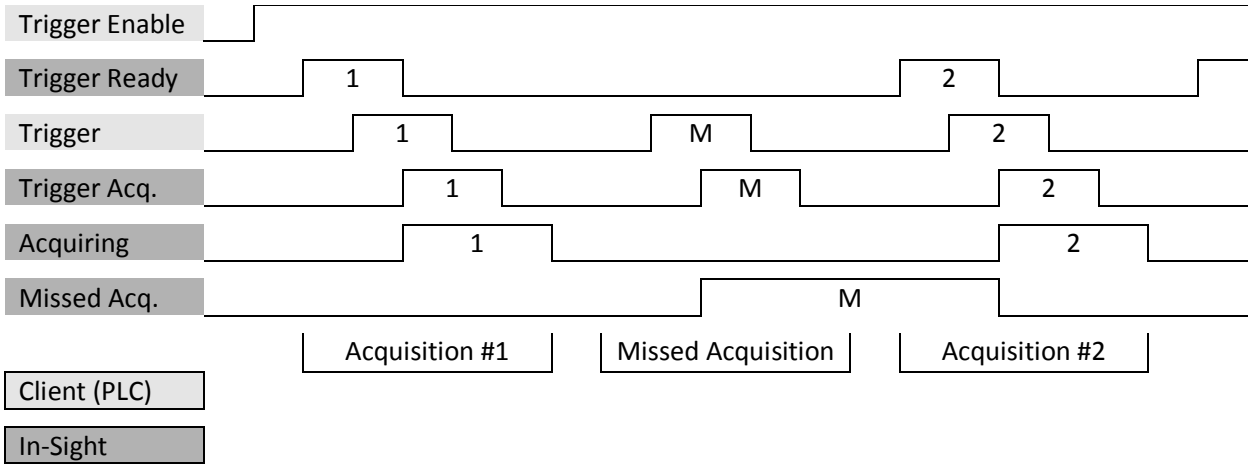
On startup, the 'Trigger Enable' bit will be false. It must be set to true to enable triggering. When the device is ready to accept triggers, the 'Trigger Ready' bit will be set to true.

While the 'Trigger Ready' bit is true, each time the camera sees the 'Trigger' bit change from 0 to 1, it will initiate an image acquisition. The client (PLC) should hold the bit in the new state until that same state value is seen back in the 'Trigger Acknowledge' bit.

During an acquisition, the 'Trigger Ready' bit will be cleared and the 'Acquiring' bit will be set to true. When the acquisition is completed, the 'Acquiring' bit will be cleared. The 'Trigger Ready' bit will again be set true once the device is ready to begin a new image acquisition.

If results buffering is enabled, the device will allow overlapped acquisition and inspecting operations. 'Trigger Ready' will be set high after acquisition is complete, but while inspecting is still in process. This can be used to achieve faster overall trigger rates. If result buffering is not enabled, the 'Trigger Ready' bit will remain low until both the acquisition and inspecting operations have completed.

To force a reset of the trigger mechanism, set the 'Trigger Enable' bit to false until the 'Trigger Ready' bit is 0. Then, 'Trigger Enable' can be set to true to re-enable acquisition.

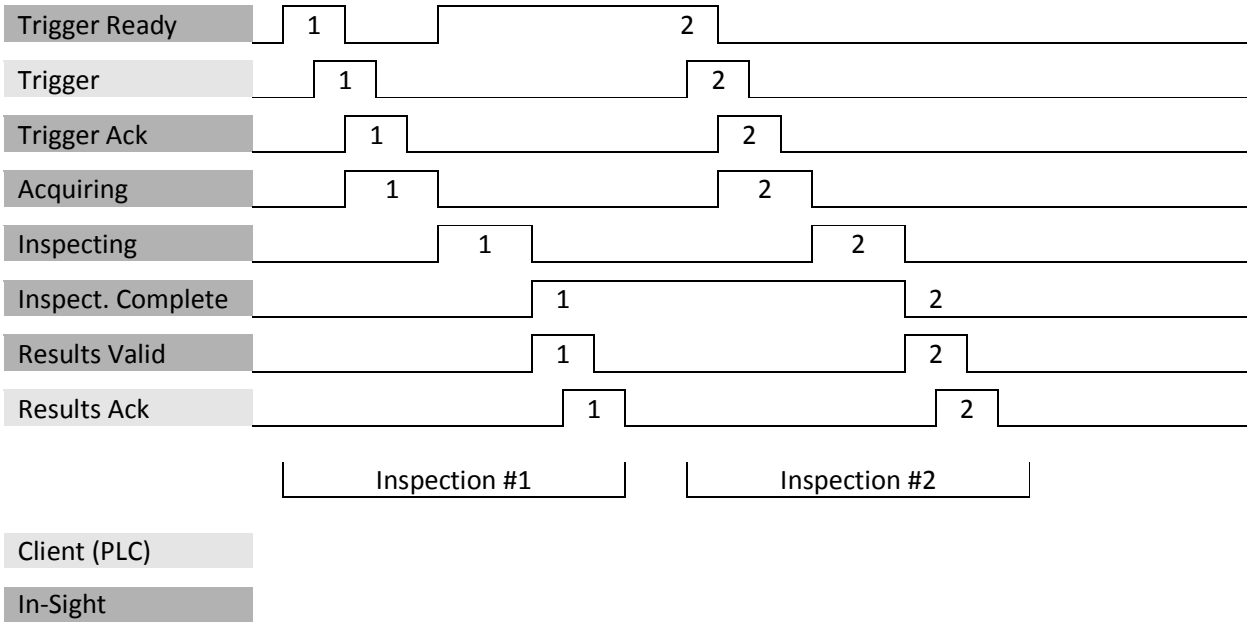


Inspection/Result Sequence

After an image is acquired, it is inspected. While being inspected, the 'Inspecting' bit of the Inspection Status Module is set. When inspection is complete, the 'Inspecting' bit is cleared and the 'Inspection Complete' bit is toggled.

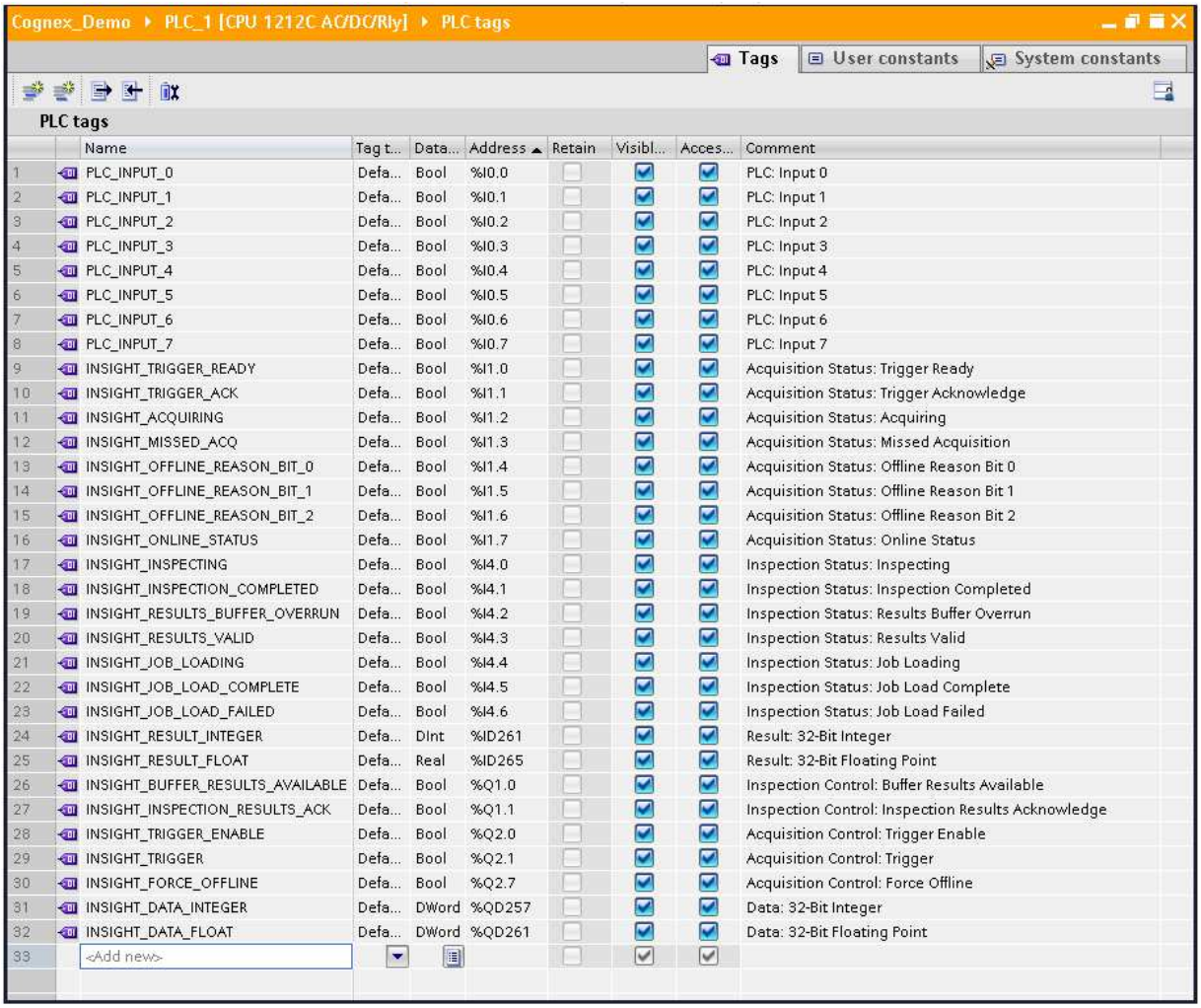
The 'Results Buffer Enable' bit determines how inspection results are handled by the camera. If the 'Results Buffer Enable' bit is set to false, then the inspection results are immediately placed into the Inspection Results Module and 'Results Valid' is set to true.

If the 'Results Buffer Enable' bit is set to true, the new results are queued. The earlier inspection results remain in the Inspection Results Module until they are acknowledged by the client by setting the 'Inspection Results Acknowledge' bit to true. After the 'Results Valid' bit is cleared, the client should set the 'Inspection Results Acknowledge' bit back to false to allow the next queued results to be placed in to the Inspection Results Module. This is a necessary handshake to ensure the results are received by the In-Sight client (PLC).



Add PLC Tags

Under the Project Tree, a folder called 'PLC Tags' is present. Expand the folder and double-click 'Show All Tags.' Add the following tags:



Cognex_Demo ▸ PLC_1 [CPU 1212C AC/DC/Rly] ▸ PLC tags

Tags | User constants | System constants

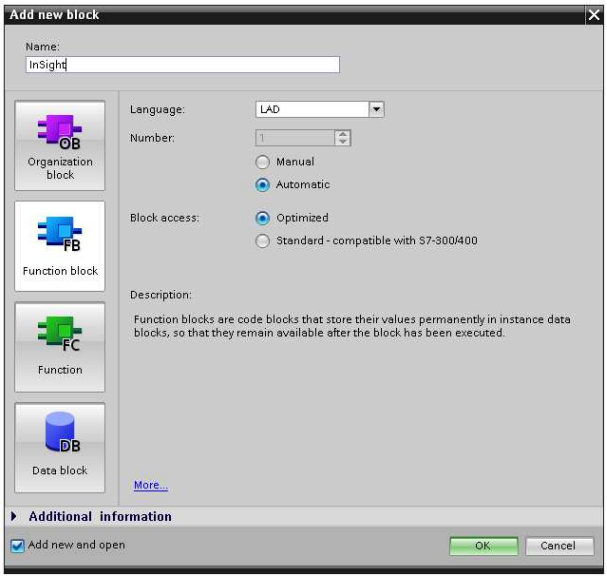
PLC tags

	Name	Tag t...	Data...	Address	Retain	Visibl...	Acces...	Comment
1	PLC_INPUT_0	Defa...	Bool	%I0.0		☑	☑	PLC: Input 0
2	PLC_INPUT_1	Defa...	Bool	%I0.1		☑	☑	PLC: Input 1
3	PLC_INPUT_2	Defa...	Bool	%I0.2		☑	☑	PLC: Input 2
4	PLC_INPUT_3	Defa...	Bool	%I0.3		☑	☑	PLC: Input 3
5	PLC_INPUT_4	Defa...	Bool	%I0.4		☑	☑	PLC: Input 4
6	PLC_INPUT_5	Defa...	Bool	%I0.5		☑	☑	PLC: Input 5
7	PLC_INPUT_6	Defa...	Bool	%I0.6		☑	☑	PLC: Input 6
8	PLC_INPUT_7	Defa...	Bool	%I0.7		☑	☑	PLC: Input 7
9	INSIGHT_TRIGGER_READY	Defa...	Bool	%I1.0		☑	☑	Acquisition Status: Trigger Ready
10	INSIGHT_TRIGGER_ACK	Defa...	Bool	%I1.1		☑	☑	Acquisition Status: Trigger Acknowledge
11	INSIGHT_ACQUIRING	Defa...	Bool	%I1.2		☑	☑	Acquisition Status: Acquiring
12	INSIGHT_MISSED_ACQ	Defa...	Bool	%I1.3		☑	☑	Acquisition Status: Missed Acquisition
13	INSIGHT_OFFLINE_REASON_BIT_0	Defa...	Bool	%I1.4		☑	☑	Acquisition Status: Offline Reason Bit 0
14	INSIGHT_OFFLINE_REASON_BIT_1	Defa...	Bool	%I1.5		☑	☑	Acquisition Status: Offline Reason Bit 1
15	INSIGHT_OFFLINE_REASON_BIT_2	Defa...	Bool	%I1.6		☑	☑	Acquisition Status: Offline Reason Bit 2
16	INSIGHT_ONLINE_STATUS	Defa...	Bool	%I1.7		☑	☑	Acquisition Status: Online Status
17	INSIGHT_INSPECTING	Defa...	Bool	%I4.0		☑	☑	Inspection Status: Inspecting
18	INSIGHT_INSPECTION_COMPLETED	Defa...	Bool	%I4.1		☑	☑	Inspection Status: Inspection Completed
19	INSIGHT_RESULTS_BUFFER_OVERRUN	Defa...	Bool	%I4.2		☑	☑	Inspection Status: Results Buffer Overrun
20	INSIGHT_RESULTS_VALID	Defa...	Bool	%I4.3		☑	☑	Inspection Status: Results Valid
21	INSIGHT_JOB_LOADING	Defa...	Bool	%I4.4		☑	☑	Inspection Status: Job Loading
22	INSIGHT_JOB_LOAD_COMPLETE	Defa...	Bool	%I4.5		☑	☑	Inspection Status: Job Load Complete
23	INSIGHT_JOB_LOAD_FAILED	Defa...	Bool	%I4.6		☑	☑	Inspection Status: Job Load Failed
24	INSIGHT_RESULT_INTEGER	Defa...	DInt	%ID261		☑	☑	Result: 32-Bit Integer
25	INSIGHT_RESULT_FLOAT	Defa...	Real	%ID265		☑	☑	Result: 32-Bit Floating Point
26	INSIGHT_BUFFER_RESULTS_AVAILABLE	Defa...	Bool	%Q1.0		☑	☑	Inspection Control: Buffer Results Available
27	INSIGHT_INSPECTION_RESULTS_ACK	Defa...	Bool	%Q1.1		☑	☑	Inspection Control: Inspection Results Acknowledge
28	INSIGHT_TRIGGER_ENABLE	Defa...	Bool	%Q2.0		☑	☑	Acquisition Control: Trigger Enable
29	INSIGHT_TRIGGER	Defa...	Bool	%Q2.1		☑	☑	Acquisition Control: Trigger
30	INSIGHT_FORCE_OFFLINE	Defa...	Bool	%Q2.7		☑	☑	Acquisition Control: Force Offline
31	INSIGHT_DATA_INTEGER	Defa...	DWord	%QD257		☑	☑	Data: 32-Bit Integer
32	INSIGHT_DATA_FLOAT	Defa...	DWord	%QD261		☑	☑	Data: 32-Bit Floating Point
33	<Add new>					☑	☑	

These tags will be used for control of the In-Sight camera.

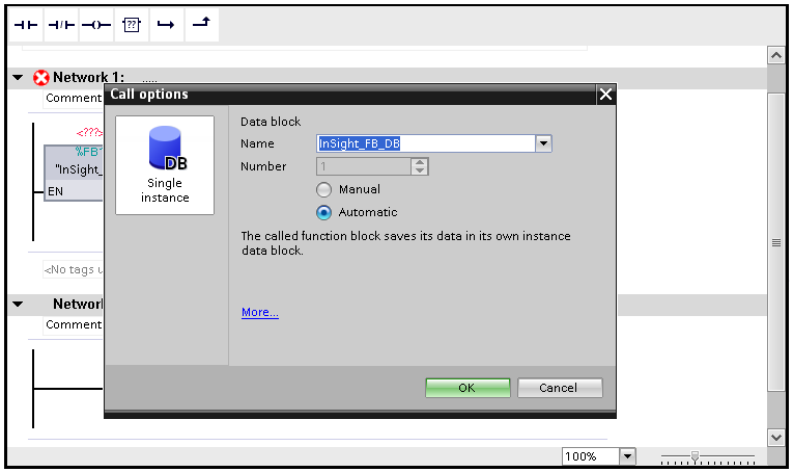
Add In-Sight Function Block

Under ‘Program Blocks,’ double-click ‘Add New Block.’ This will display a popup. Select ‘Function Block’ in the popup.



Call the Function Block InSight_FB and press ‘OK.’ This block will be FB1.

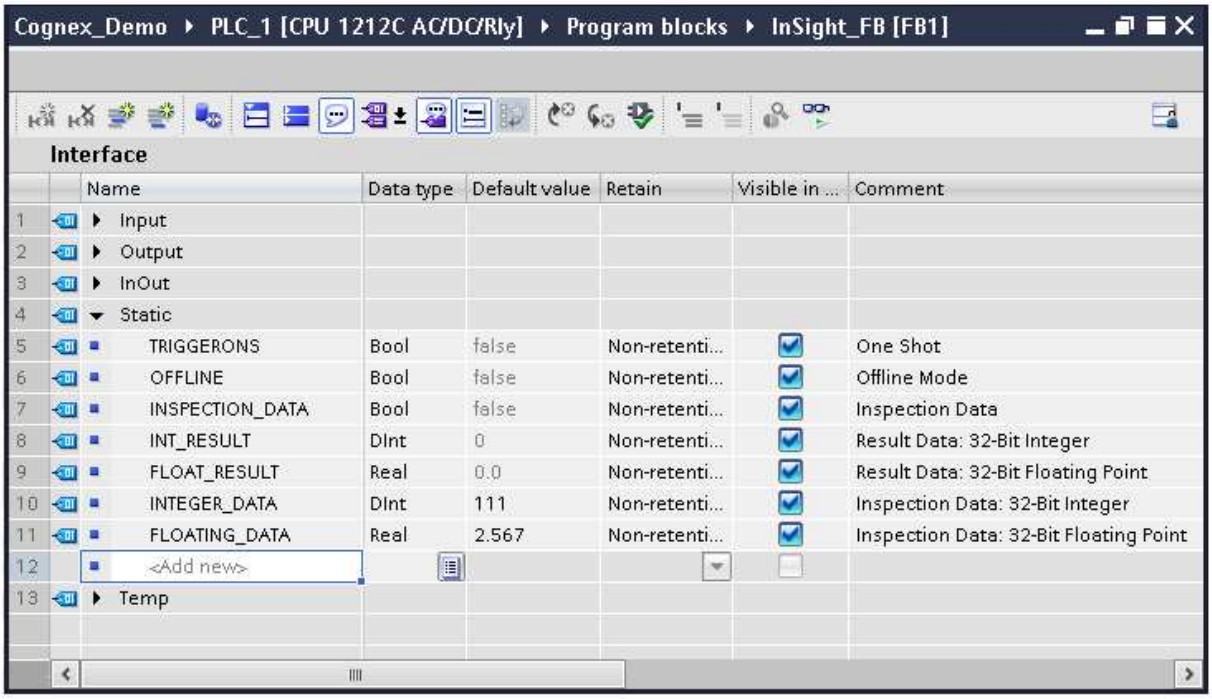
Now, the function block needs to be called. Double-Click on ‘Main [OB1]’ in the Project Tree. Add an empty box and type FB1 and press the ‘Enter’ key. This will display a popup to create a Data Block for the Function Block.



Press ‘OK’ and the function block call is complete.

Add Function Block Tags

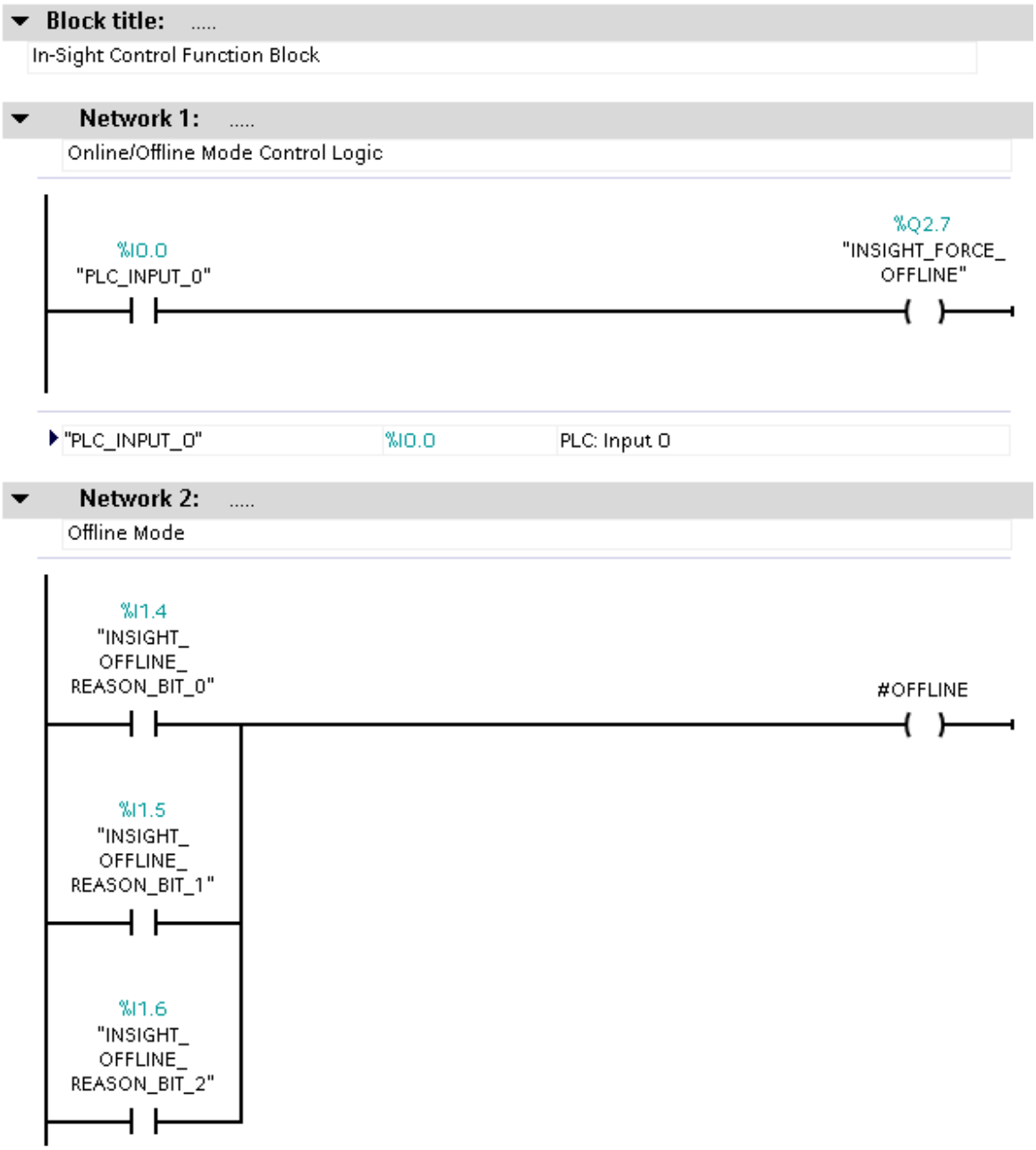
Double-click the 'InSight_FB[FB1]' function block. Add the following static tags to the Function Block:



These tags will be used in the function block to send and receive data to/from the In-Sight camera.

Add Camera Online/Offline Control

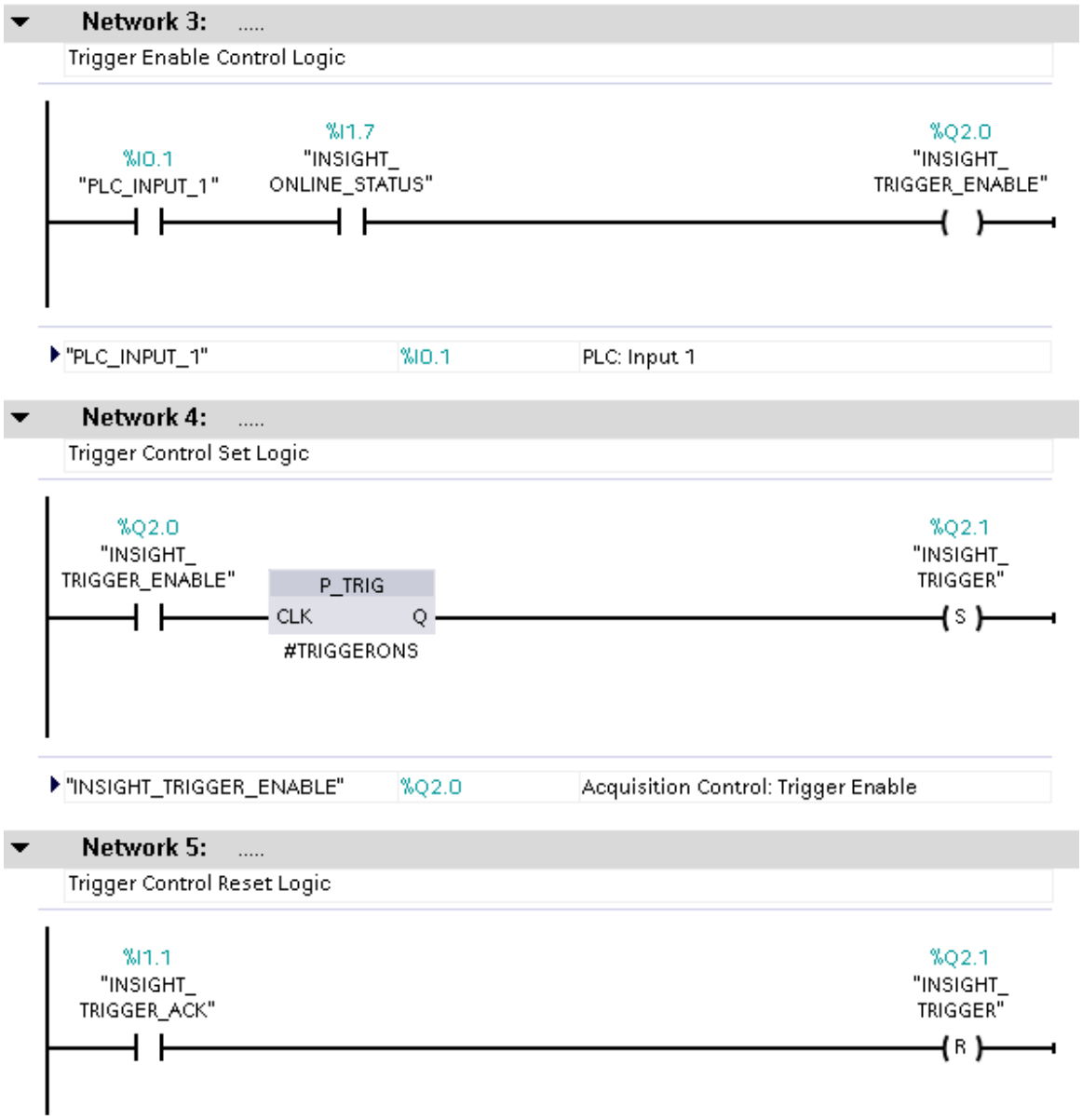
The camera can be forced offline by using the 'Force Offline' bit in the device library.



Set the 'Force Offline' bit to force the camera offline. When the bit goes low, the camera returns to Online mode. If any of the 'Offline Reason' bits are high, the camera is offline. PLC code can be written to break out each of the Offline Reasons. However, that is not shown in this guide.

Add Camera Trigger

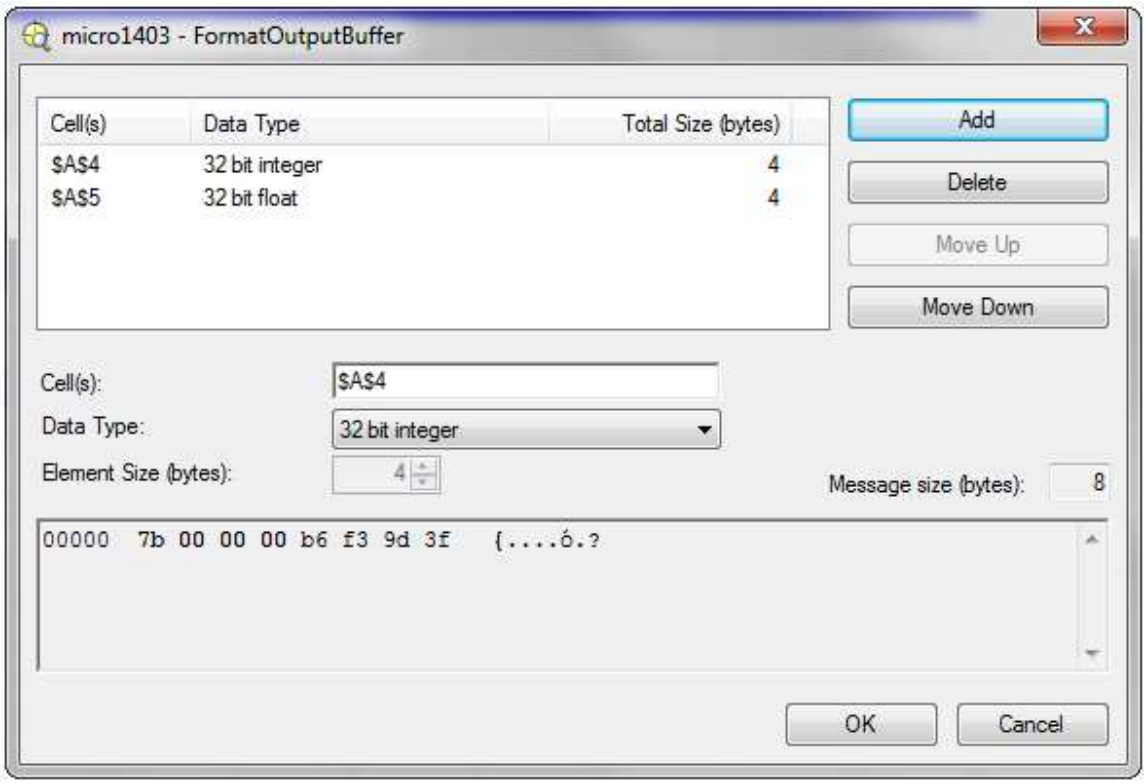
The following code shows how to add a network trigger to the PLC program:



Four tags, defined by the In-Sight device library, are used to control the trigger. In the first network (Network 3), the 'Trigger Enable' bit is set. When the PLC input I0.1 is high, and the In-Sight Camera is Online, the 'Trigger Enable' bit goes high. This allows a network trigger to be sent to the camera. The rising edge of the 'Trigger Enable' bit is used to set the Trigger output (Network 4). The Trigger output is reset when the camera sends a 'Trigger Acknowledge' signal back to the PLC (Network 5). To test the trigger to the camera, toggle PLC input I0.1.

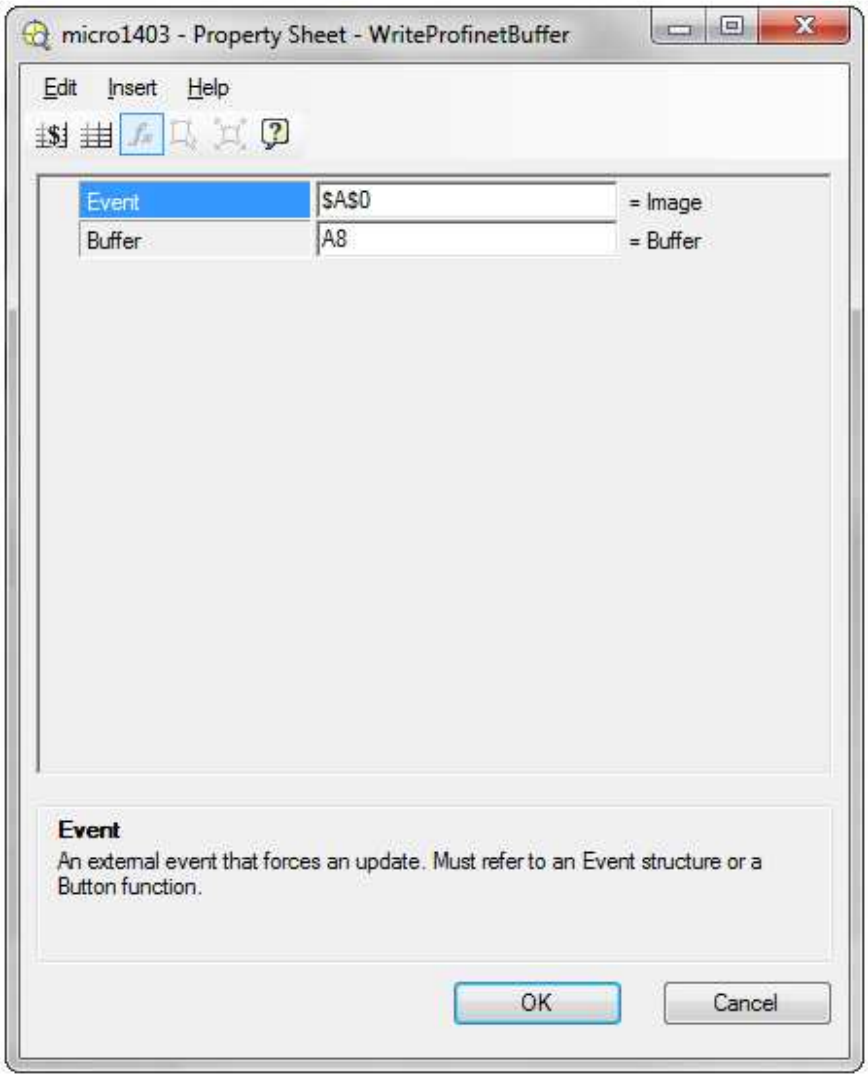
Add Camera Results

To receive data from the camera, code must be added to both the PLC program and the In-Sight spreadsheet. Add two functions to the spreadsheet: EditInt() and EditFloat(). Then, add a FormatOutputBuffer() function to the spreadsheet.

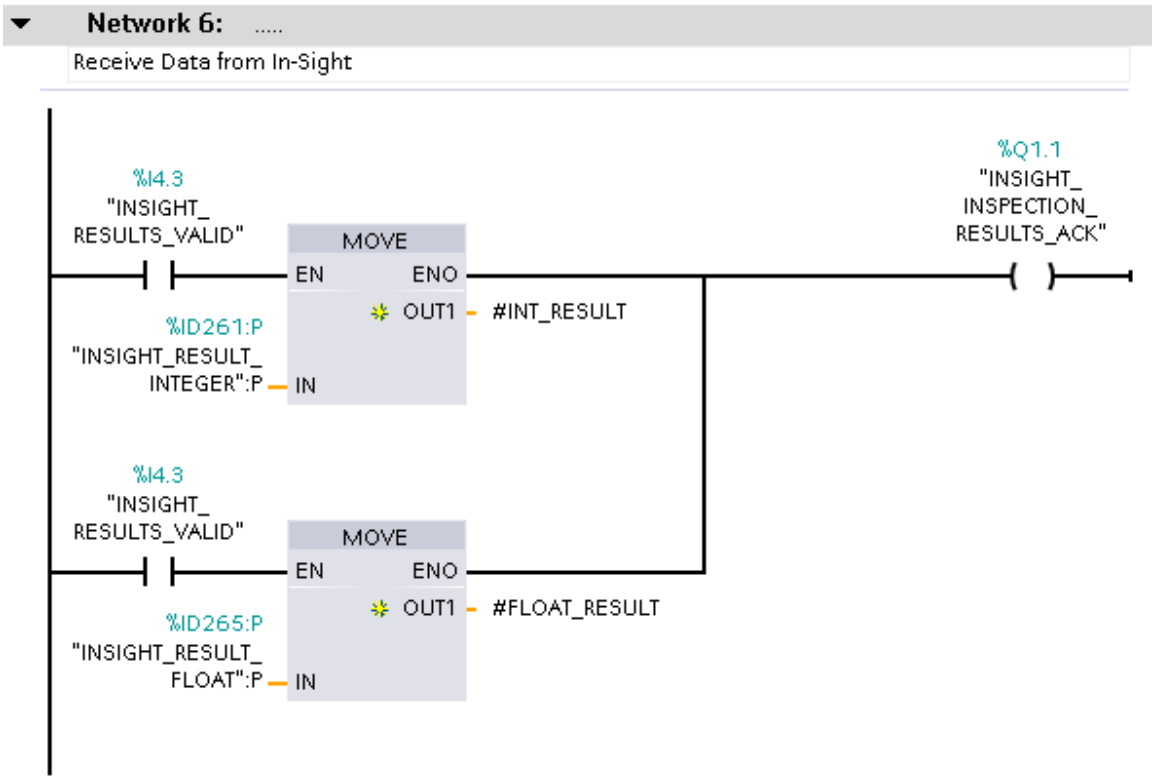


Add the integer and floating-point fields to the buffer.

Then, add a WriteProfinetOutput() function to the spreadsheet and refer to the FormatOutputBuffer() cell. This will send the integer and floating-point values when the camera is triggered.



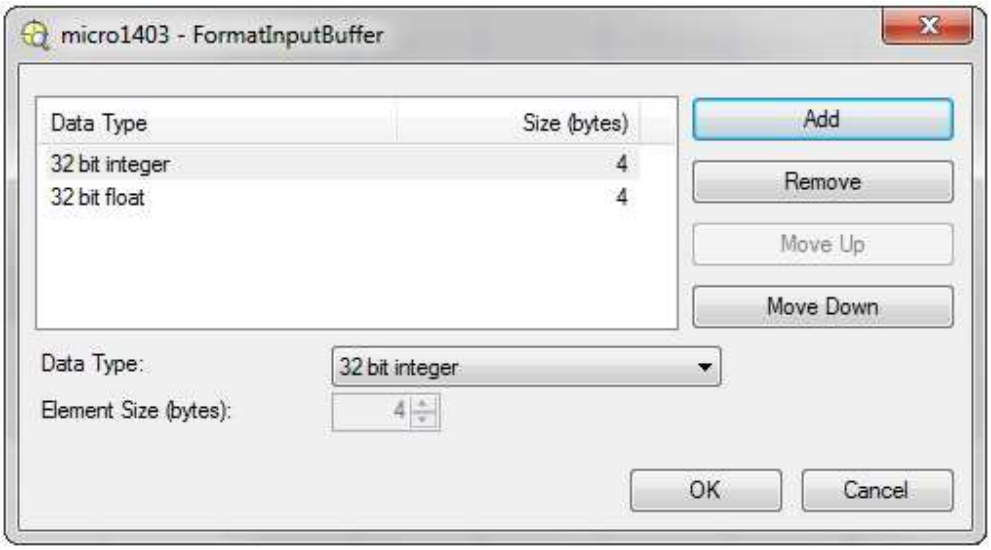
Finally, add the following PLC code:



This will move the data from the device input registers to the tags created in the function block, when the camera sends an 'Inspection Results Valid' signal.

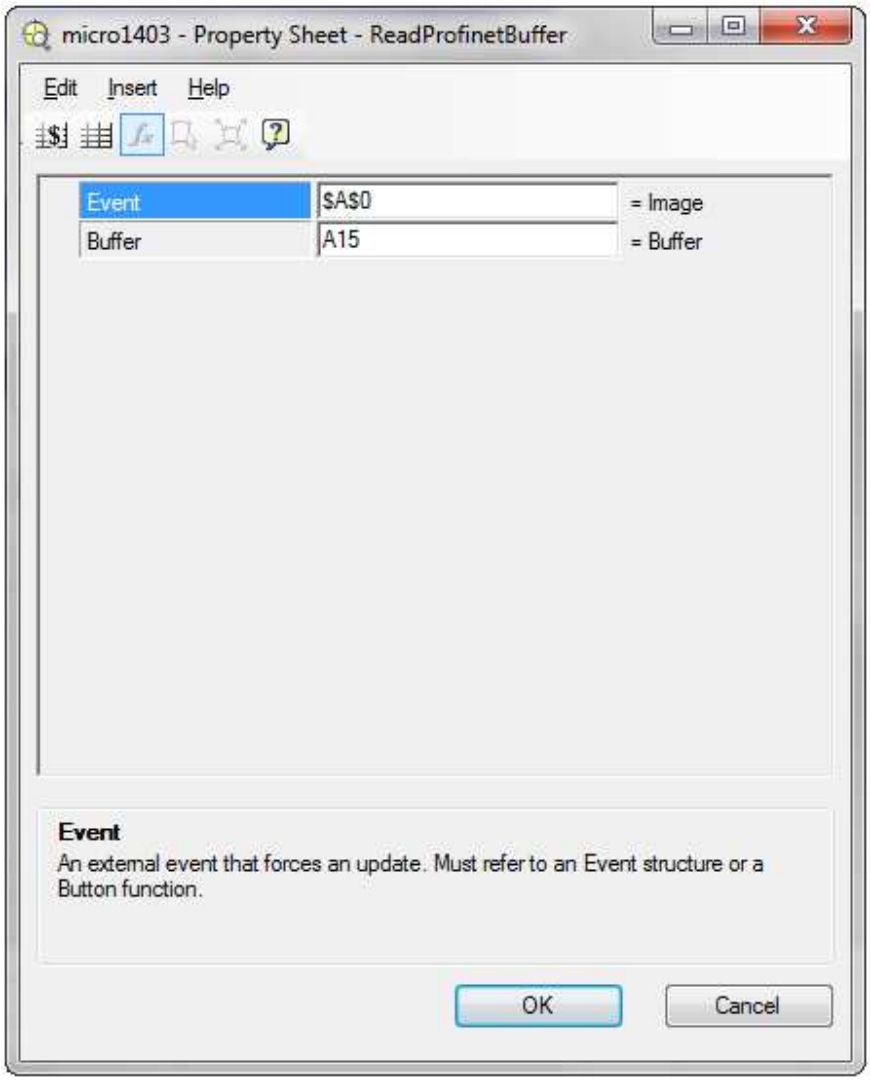
Add Camera Data

To send data to the camera, code must be added to both the PLC program and the In-Sight spreadsheet. Add a FormatInputBuffer() function to the spreadsheet.

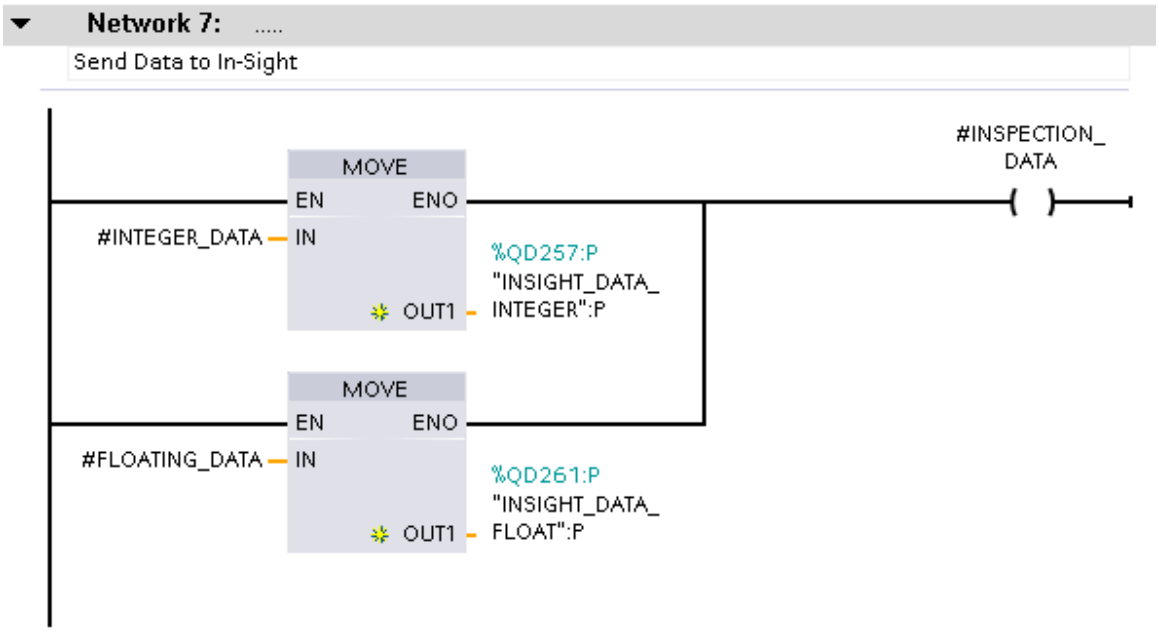


Add an integer and a floating-point register to the input buffer.

Then, add a ReadProfinetBuffer() function to the spreadsheet and reference the FormatInputBuffer() cell. The two registers are now displayed in the spreadsheet and will be updated when the camera is triggered.



Finally, add the following PLC code:

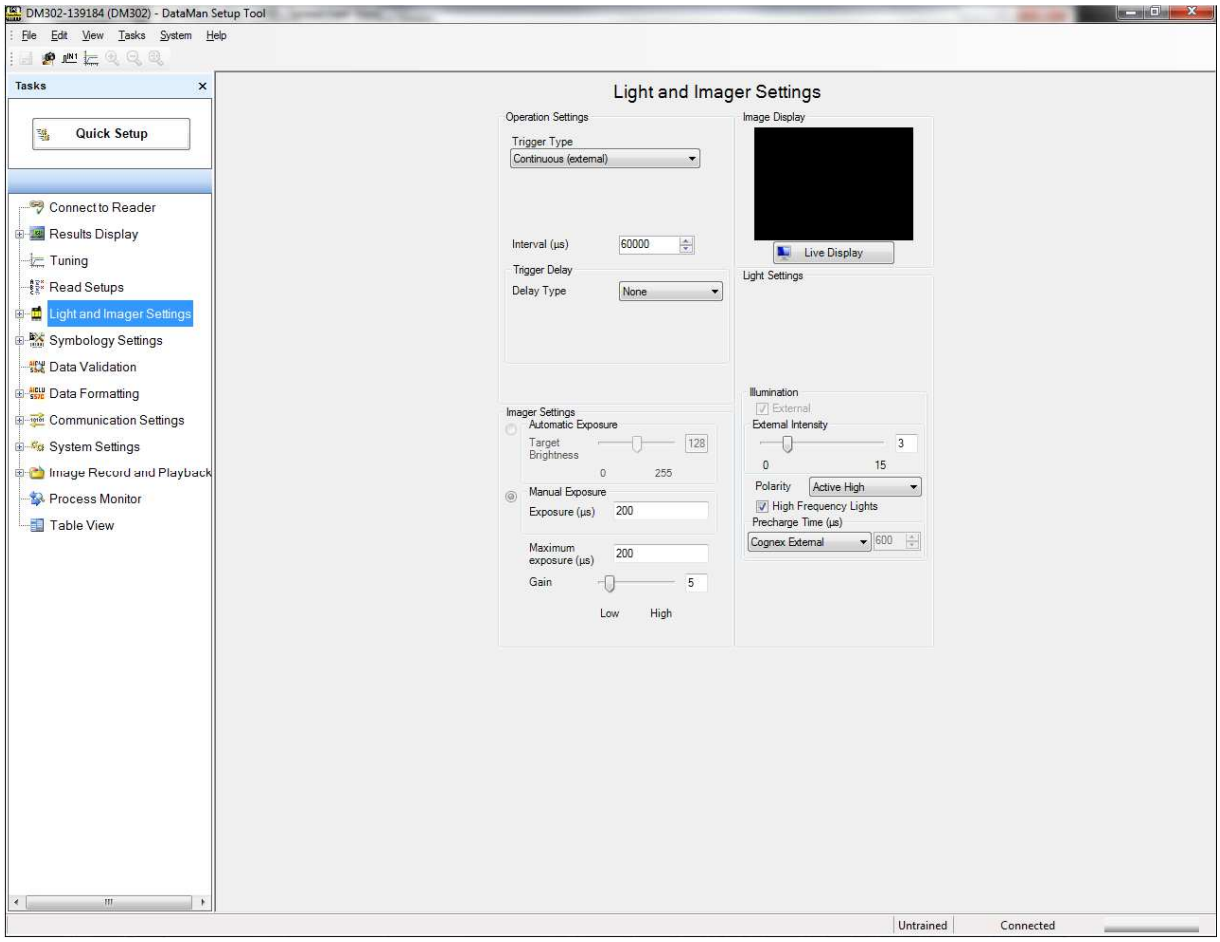


This will move data from the function block tags to the device outputs, which sends it to the camera.

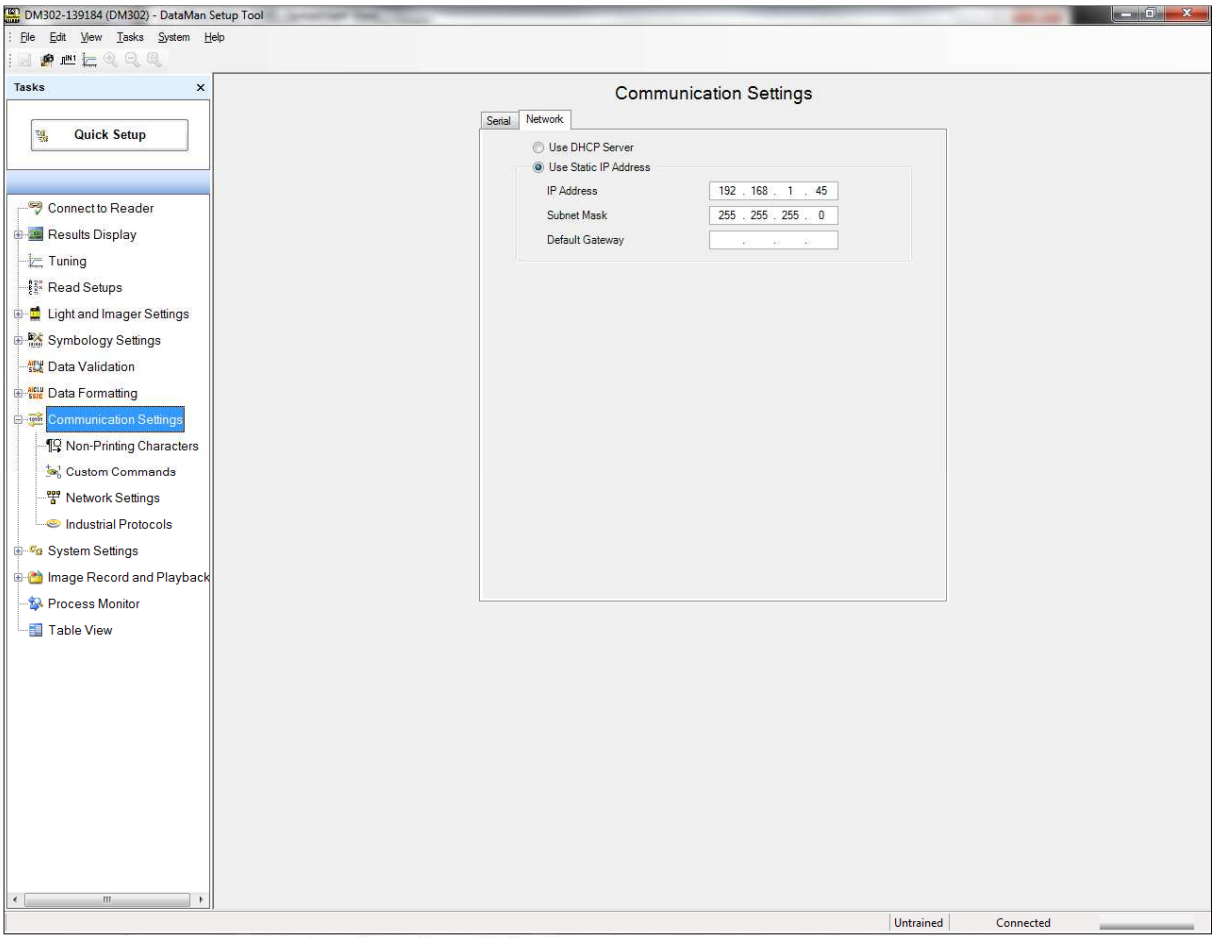
Connecting to DataMan

DataMan Configuration

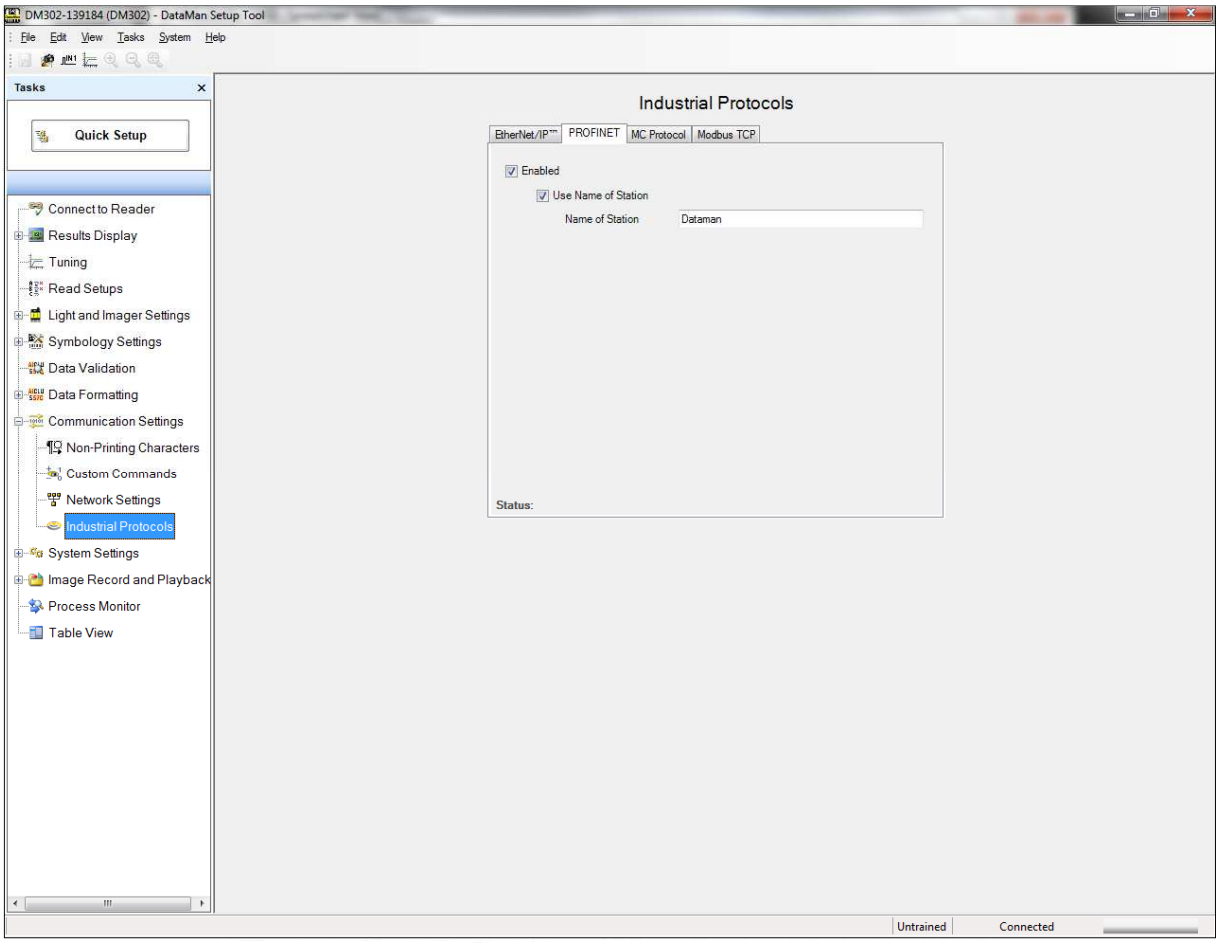
For this example, a DataMan 302X was used. There are three settings necessary to configure a DataMan for communicating to a Siemens PLC via Profinet. First, make sure the Trigger mode is set to 'Continuous (external).' This setting can be found under 'Light and Imager Settings.'



Next, set the IP Address of the DataMan. For this example, set the IP address to 192.168.1.45. This can be found under 'Communications Settings.'



Finally, select ‘PROFINET’ on the ‘Industrial Protocols’ submenu. Select the PROFINET tab.



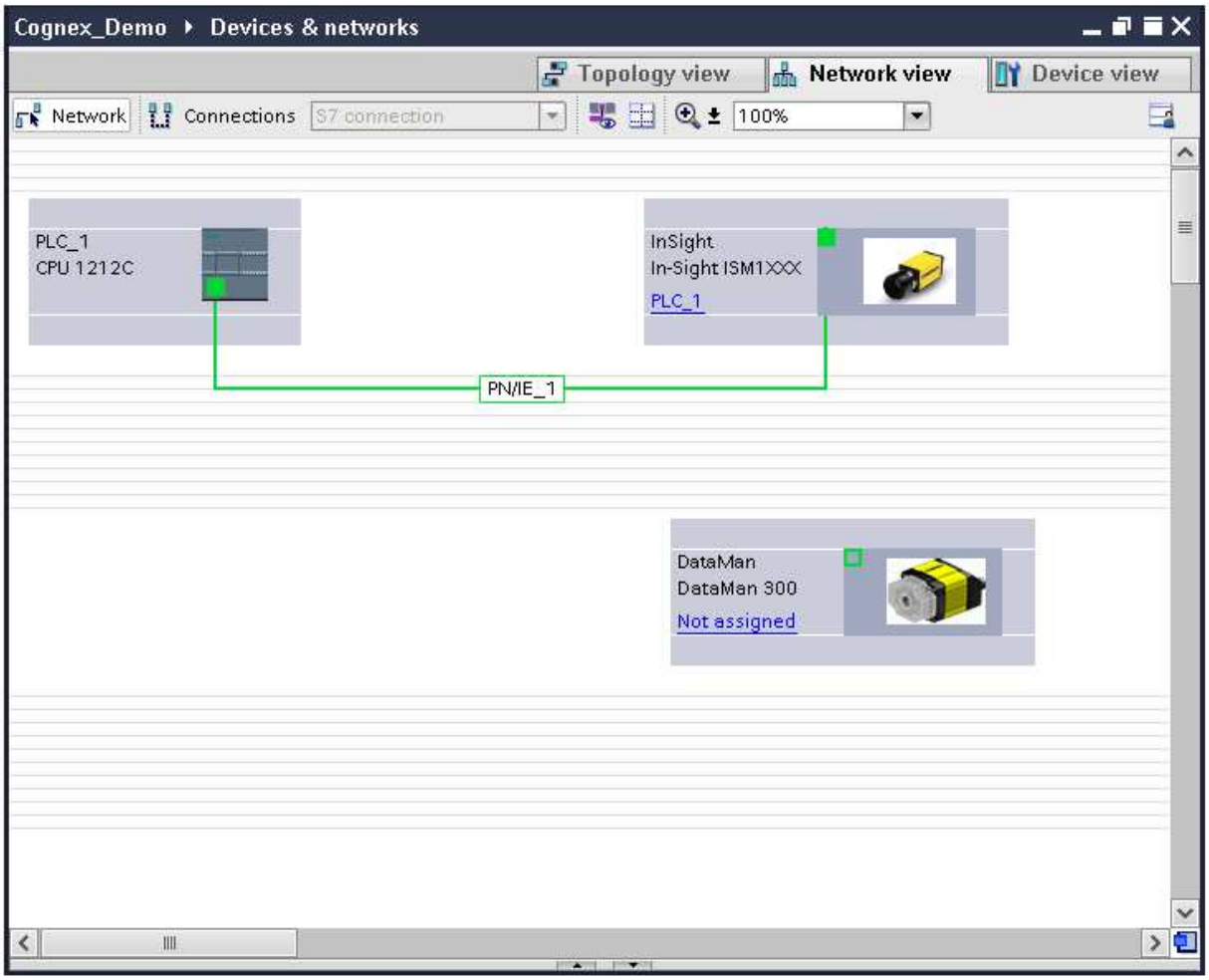
Select the ‘Enabled’ checkbox, the ‘Use of Station Name’ checkbox, and type ‘Dataman’ as the Station Name.

The system will need to be restarted to enable Profinet communications.

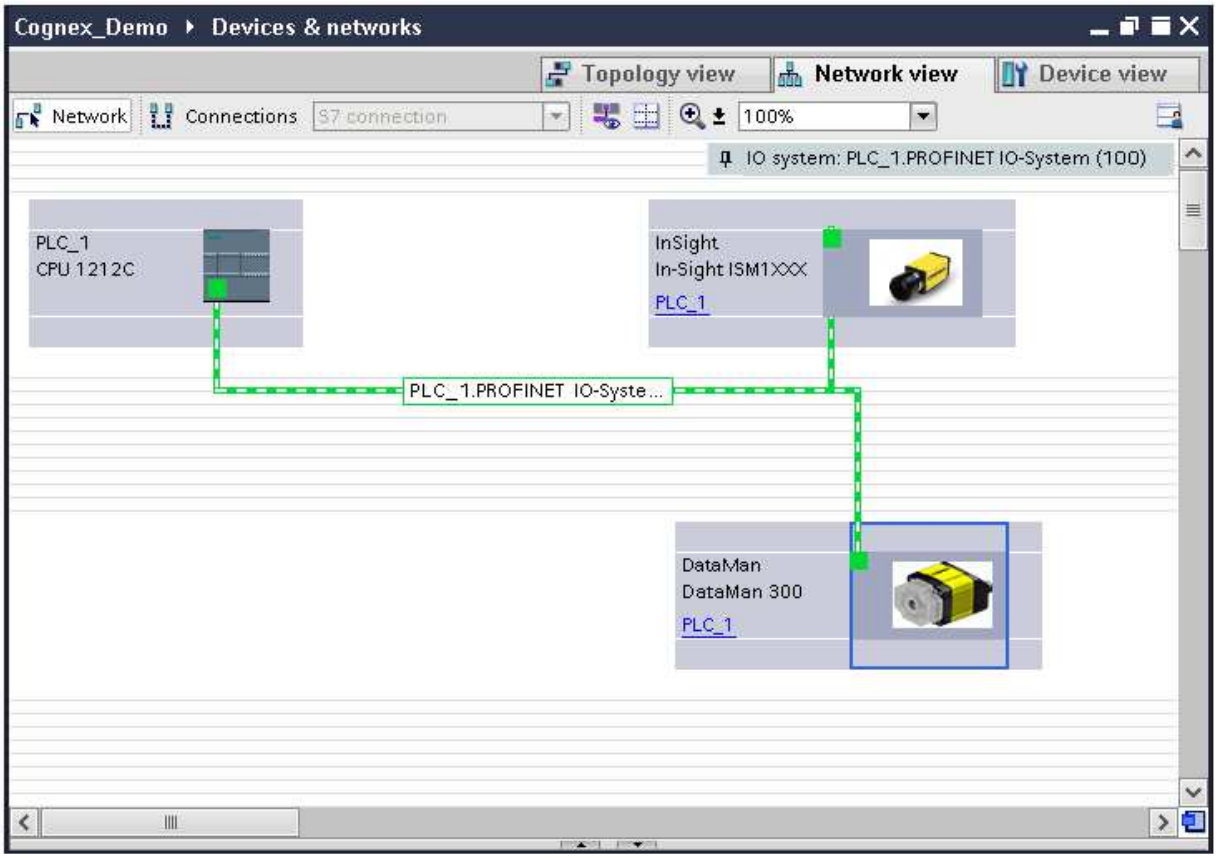
PLC Programming

Add DataMan to Network

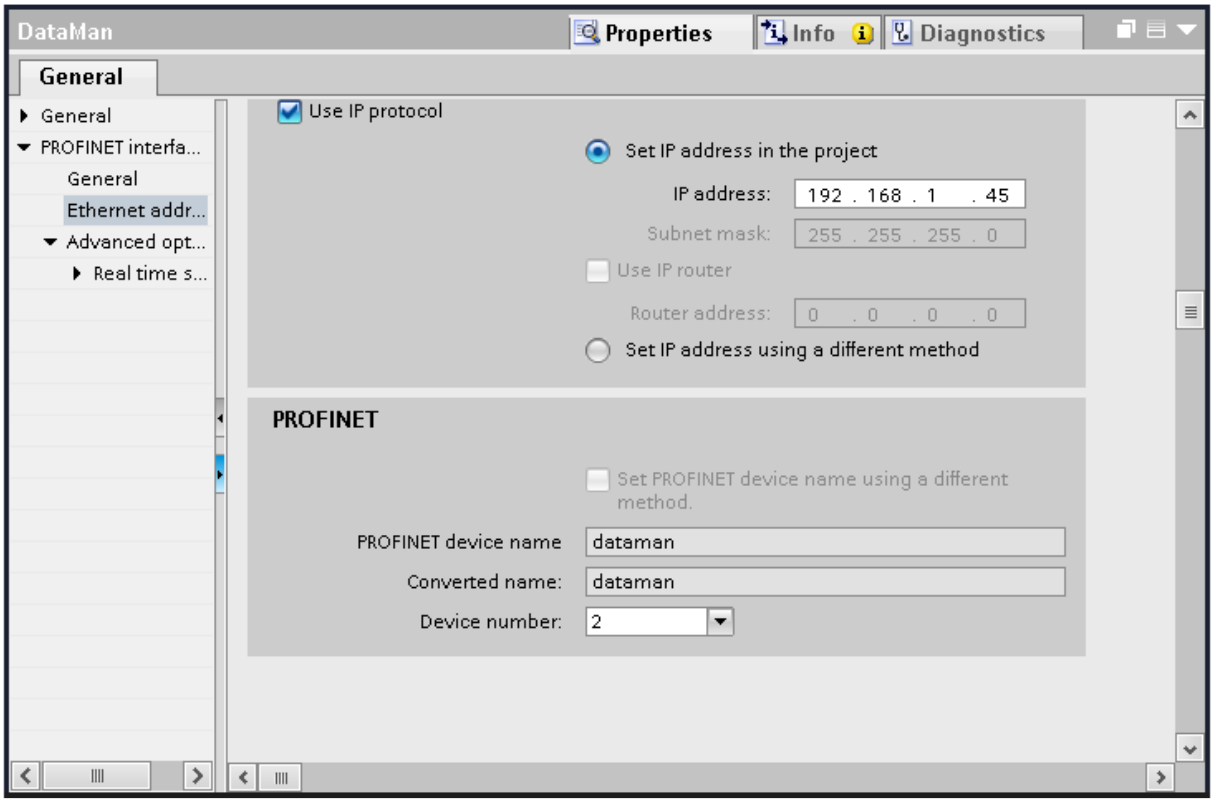
Under 'Devices & Networks,' select 'Network View.' Add DataMan 300 PDEV. This is listed under 'Other Field Devices'➔PROFINET IO➔Sensors➔Cognex Corp➔Cognex Vision Sensors. A device can be added by dragging the library object into the 'Network View.' The DataMan device will be listed as 'Not Assigned.'



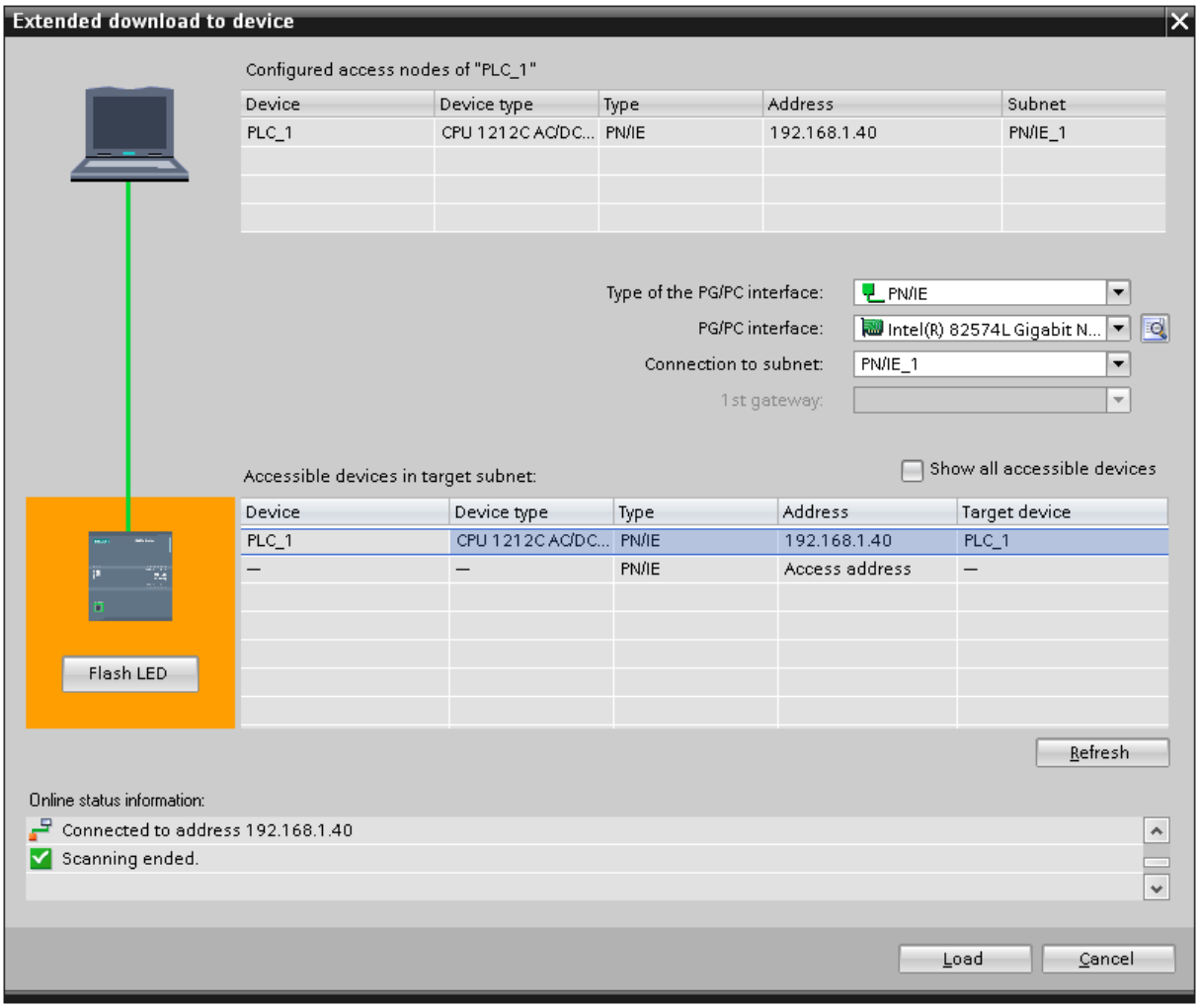
To assign the device, press and hold the left mouse button on the green box in the DataMan object, and drag your mouse to the green box in the PLC. Release the mouse button. This establishes the connection between the PLC and the DataMan.



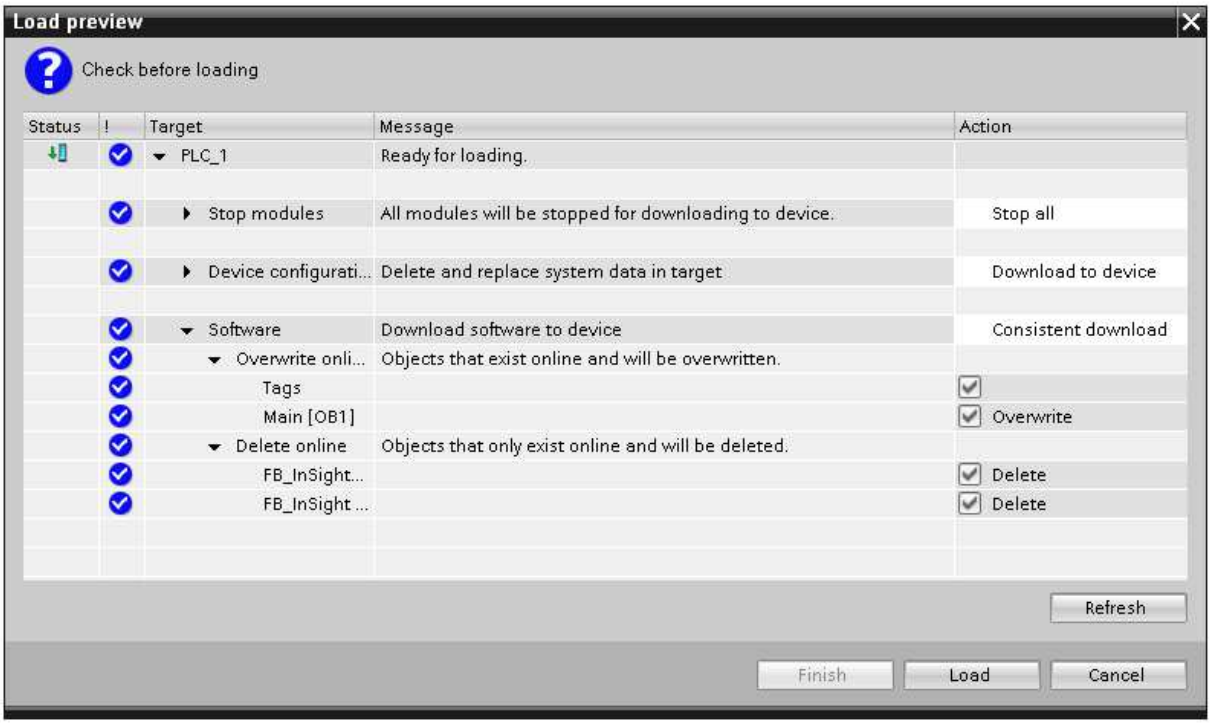
Select the camera. Under the ‘General Properties’ tab, select ‘PROFINET Interface.’ Change the name to DataMan and change the IP Address to 192.168.1.45.



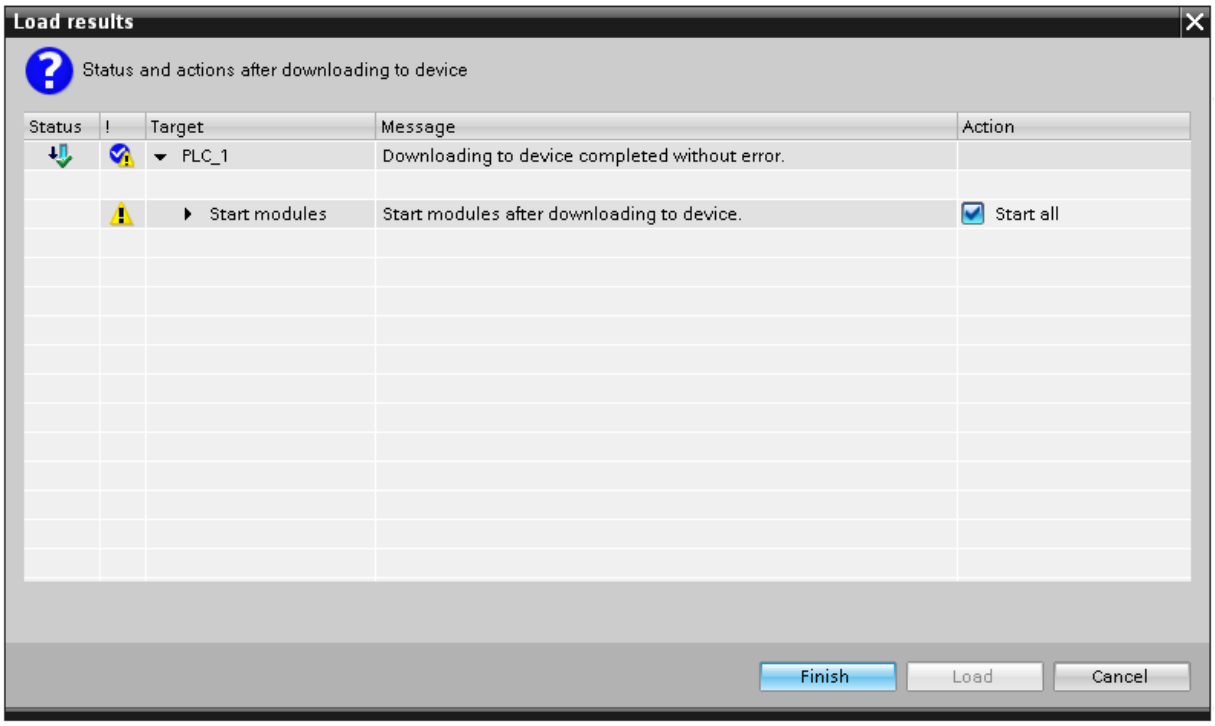
The program is ready to download to the PLC to check communication status. Select the PLC in the Project Tree. Under the 'Online' menu, select 'Download to Device.' A popup will appear.



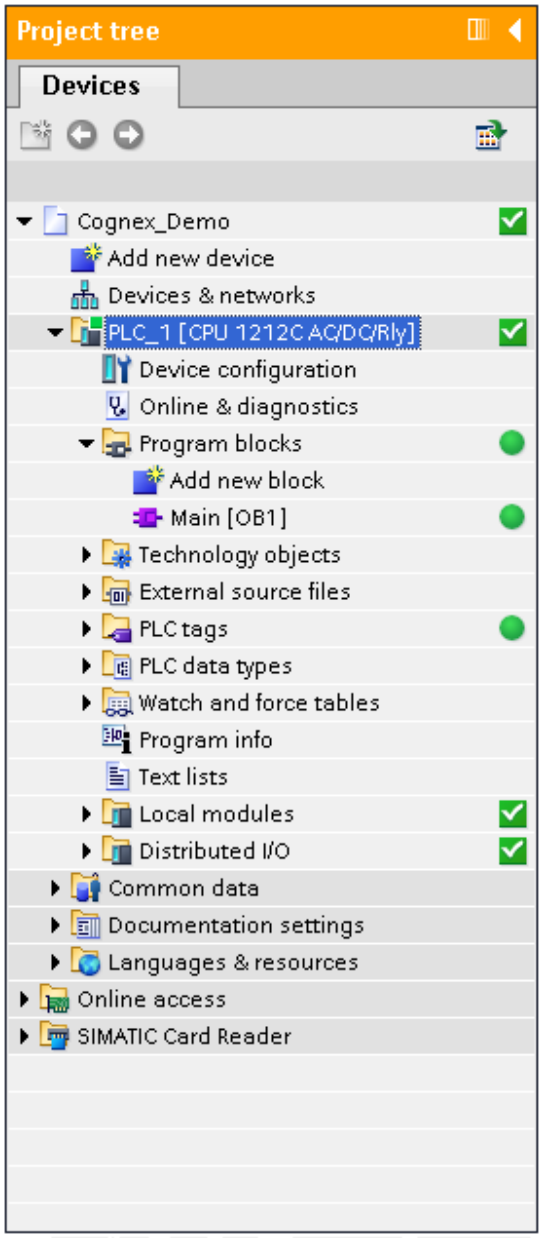
Press ‘Load.’ A new popup will appear. Do not be alarmed if your Load Preview popup or Load Results popup do not appear exactly as shown.



Press ‘Load.’ A new popup will appear.

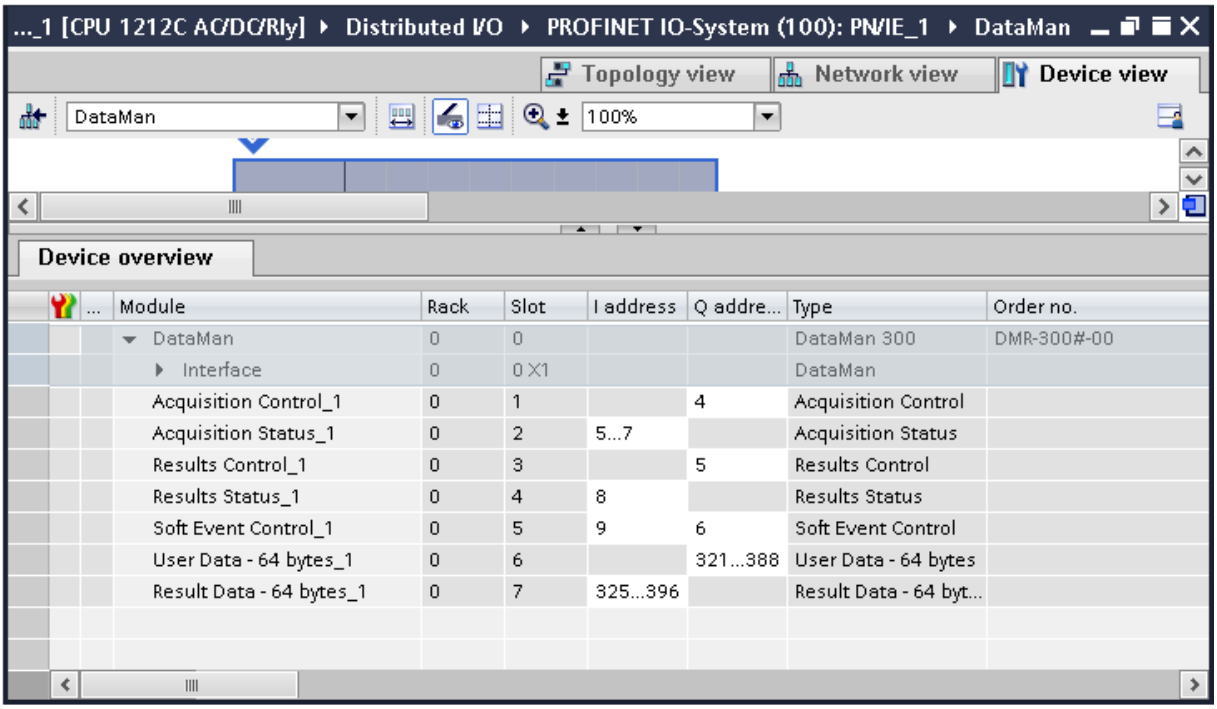


Press 'Finish' to finish the download process. Go online with the processor by selecting the PLC in the Project Tree, and select 'Go Online' under the 'Online' drop-down menu. If communications are correct, green check marks will appear next to all devices.


Project tree
Devices
Cognex_Demo
Add new device
Devices & networks
PLC_1 [CPU 1212C AC/DC/Rly]
Device configuration
Online & diagnostics
Program blocks
Add new block
Main [OB1]
Technology objects
External source files
PLC tags
PLC data types
Watch and force tables
Program info
Text lists
Local modules
Distributed I/O
Common data
Documentation settings
Languages & resources
Online access
SIMATIC Card Reader

Device I/O

When the DataMan object was added to the system, I/O was assigned for the control elements of the camera.



	Module	Rack	Slot	I address	Q addre...	Type	Order no.
	DataMan	0	0			DataMan 300	DMR-300#-00
	Interface	0	0 X1			DataMan	
	Acquisition Control_1	0	1		4	Acquisition Control	
	Acquisition Status_1	0	2	5...7		Acquisition Status	
	Results Control_1	0	3		5	Results Control	
	Results Status_1	0	4	8		Results Status	
	Soft Event Control_1	0	5	9	6	Soft Event Control	
	User Data - 64 bytes_1	0	6		321...388	User Data - 64 bytes	
	Result Data - 64 bytes_1	0	7	325...396		Result Data - 64 byt...	

The I/O assignments break down as follows:

Acquisition Control Module		
Bit	**Name**	**Description**
0	Trigger Enable	Setting this bit enables triggering via PROFINET. Clearing this bit disables triggering.
1	Trigger	Setting this bit triggers an acquisition when the following conditions are met: • The Trigger Enable bit is set. • No acquisition is currently in progress. • The device is ready to trigger.
2-7	Reserved	Unused.

Acquisition Status Module		
Bit	**Name**	**Description**
0	Trigger Ready	Indicates when the device can accept a new trigger. This field is true when the Trigger Enable bit has been set and the device is ready to accept a new trigger.
1	Trigger Acknowledge	Indicates that the DataMan has received a new trigger. This bit will remain high as long as the Trigger bit remains high.
2	Acquiring	Indicates that the DataMan is currently acquiring an image.
3	Missing Acquisition	Indicates that the DataMan was unable to successfully trigger an acquisition. The bit is cleared when the next successful acquisition occurs. It will only be set if an acquisition triggered from the Acquisition Control Module could not get executed.
4-7	Reserved	Reserved for Future Use.
8-23	Trigger ID	ID Value of the next trigger to be issued (16-bit integer). Used to match issued triggers with corresponding result data received later. This same value will be returned in Result ID of the Result data.

Results Control Module		
Bit	**Name**	**Description**
0	Results Buffer Enable	Enables queuing of Result Data. If enabled, the current result data will remain until acknowledged (even if new results arrive). New results are queued. The next set of results are pulled from the queue (made available in the Result Data module) each time the current results are acknowledged. The DataMan will respond to the acknowledge by clearing the Results Available bit. Once the Results Acknowledge bit is cleared, the next set of read results will be posted and Results Available will be set. If results buffering is not enabled, newly received read results will simply overwrite the content of the Result Data module.
1	Results Acknowledge	This bit is used to acknowledge that the PLC has successfully read the latest result data. When set to True, the Result Available bit will be cleared. If result buffering is enabled, the next set of result data will be pulled from the queue and Result Available will again be set to True.
2-7	Reserved	Reserved for Future Use.

Results Status Module		
Bit	**Name**	**Description**
0	Decoding	Indicates that the DataMan is decoding an acquired image.
1	Decode Complete	This bit is toggled on the completion of a decode operation when the new results are made available.
2	Result Buffer Overrun	This indicates that the DataMan has discarded a set of read results because the results queue is full. It is cleared when the next set of results are successfully queued.
3	Results Available	This indicates that a new set of read results are available. It is cleared when the results are acknowledged.
4-6	Reserved	Reserved for Future Use.
7	General Fault	This indicates that a fault has occurred.

Soft Event Control (Output) Module		
Bit	**Name**	**Description**
0	Train Code	Bit transition from 0→1 will cause the Train Code operation to be invoked.
1	Train Match String	Bit transition from 0→1 will cause the Train Match String operation to be invoked.
2	Train Focus	Bit transition from 0→1 will cause the Train Focus operation to be invoked.
3	Train Brightness	Bit transition from 0→1 will cause the Train Brightness operation to be invoked.
4	Untrain	Bit transition from 0→1 will cause the Untrain operation to be invoked.
5	Reserved	Reserved for Future Use.
6	Execute DMCC	Bit transition from 0→1 will cause the DMCC operation to be invoked. Note that a valid DMCC command string must first be placed in User Data before invoking this event.
7	Set Match String	Bit transition from 0→1 will cause the Set Match String operation to be invoked. Note that match string data must first be placed in User Data before invoking this event.

Soft Event Control (Input) Module		
Bit	**Name**	**Description**
0	Train Code Acknowledge	Indicates that the Train Code operation has completed.
1	Train Match String Acknowledge	Indicates that the Train Match String operation has completed.
2	Train Focus Acknowledge	Indicates that the Train Focus operation has completed.
3	Train Brightness Acknowledge	Indicates that the Train Brightness operation has completed.
4	Untrain Acknowledge	Indicates that the Untrain operation has completed.
5	Reserved	Reserved for Future Use.
6	Execute DMCC Acknowledge	Indicates that the Execute DMCC operation has completed.
7	Set Match String Acknowledge	Indicates that the Set Match String operation has completed.

User Data Module		
Byte	**Name**	**Description**
0-1	User Data Option	Currently only used by Set Match String soft event. Specifies which code target to assign the string (16-bit integer). 0: Assign string to all targets 1: Assign string to 2D codes 2: Assign string to QR codes 3: Assign string to 1D/Stacked/Postal codes
2-3	User Data Length	Number of bytes of valid data actually contained in the User Data field (16-bit integer).
4-…	User Data	Data sent from the PLC to the DataMan to support acquisition, decode and other special operations.

Result Data Module		
Byte	**Name**	**Description**
0-1	Result ID	The value of the Trigger ID when the trigger that generated these results was issued. It is used to match up triggers with corresponding result data (16-bit integer).
2-3	Result Code	Indicates the success or failure of the read that produced these results (16-bit integer). Bit 0: 1=Read, 0=No Read Bit 1: 1=Validated, 0=Not Validated Bit 2: 1=Verified, 0=Not Verified Bit 3: 1=Acquisition Trigger Overrun Bit 4: 1=Acquisition Buffer Overflow Bit 5-15: Reserved for Future Use
4-5	Result Extended	Reserved for Future Use.
6-7	Result Length	Actual number of bytes of read data contained in the Result Data field (16-bit integer).
8-…	Result Data	Decoded read result data (array of bytes).

Device Operation

Acquisition Sequence

The DataMan can be triggered to acquire images by several methods. It can be done explicitly by manipulating the Trigger bit of the Acquisition Control Module, it can be triggered by external hard wired input, and finally it can be triggered via DMCC command. Manipulating the Acquisition Control Module bits will be discussed here.

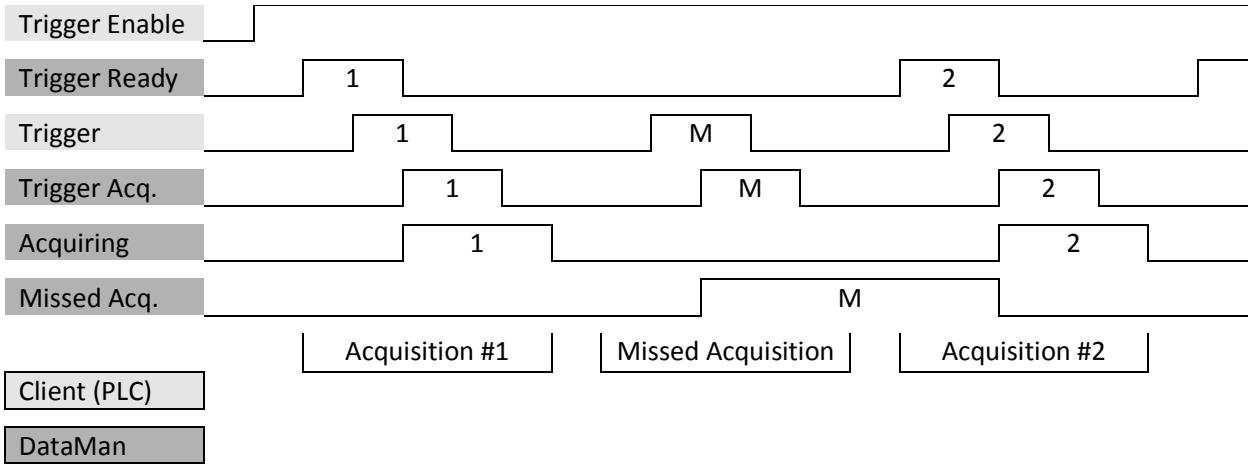
On startup, the 'Trigger Enable' bit will be false. It must be set to true to enable triggering. When the device is ready to accept triggers, the 'Trigger Ready' bit will be set to true.

While the 'Trigger Ready' bit is true, each time the reader sees the 'Trigger' bit change from 0 to 1, it will initiate an image acquisition. The client (PLC) should hold the bit in the new state until that same state value is seen back in the 'Trigger Acknowledge' bit.

During an acquisition, the 'Trigger Ready' bit will be cleared and the 'Acquiring' bit will be set to true. When the acquisition is completed, the 'Acquiring' bit will be cleared. The 'Trigger Ready' bit will again be set true once the device is ready to begin a new image acquisition.

If results buffering is enabled, the device will allow overlapped acquisition and decoding operations. 'Trigger Ready' will be set high after acquisition is complete, but while decoding is still in process. This can be used to achieve faster overall trigger rates. If result buffering is not enabled, the 'Trigger Ready' bit will remain low until both the acquisition and decode operations have completed.

To force a reset of the trigger mechanism, set the 'Trigger Enable' bit to false until the 'Trigger Ready' bit is 0. Then, 'Trigger Enable' can be set to true to re-enable acquisition.



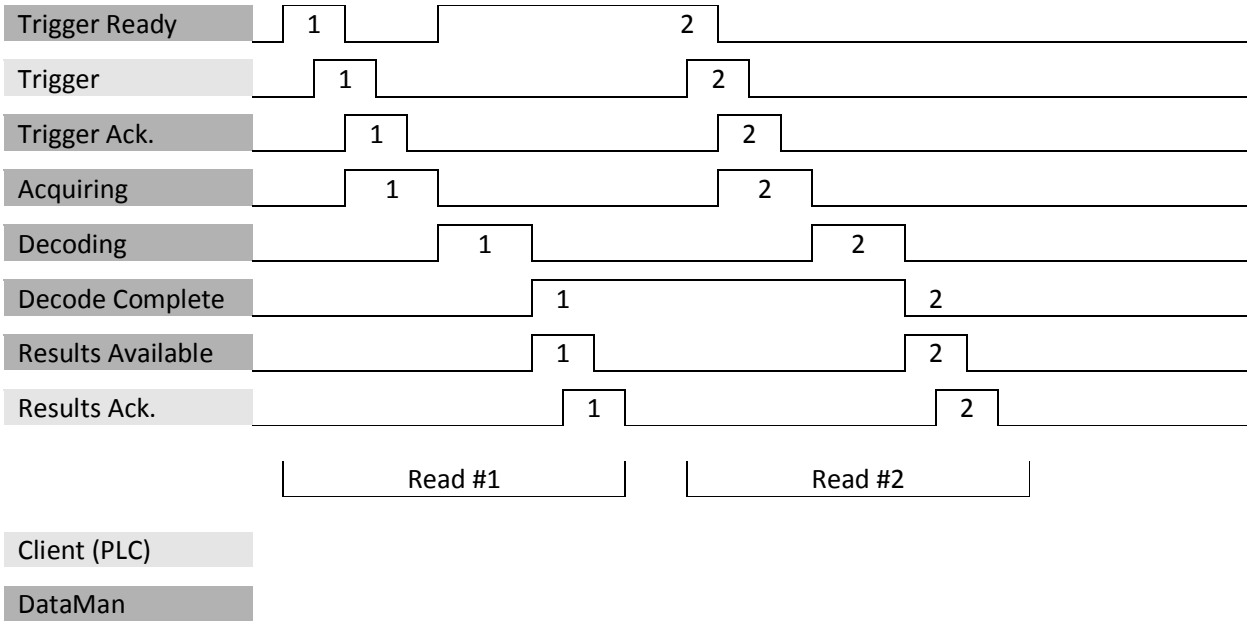
As a special case, an acquisition can be cancelled by clearing the 'Trigger' signal before the read operation has completed. This allows for the cancellation of reads in Presentation and Manual mode if no code is in the field of view. To ensure that a read is not unintentionally cancelled, it is advised that the PLC hold the 'Trigger' signal true until both 'Trigger Acknowledge' and 'Results Available' are true (or 'Decode Complete' toggles state).

Decode/Result Sequence

After an image is acquired, it is decoded. While being decoded, the 'Decoding' bit of the Result Status Module is set. When decode is complete, the 'Decoding' bit is cleared and the 'Decode Complete' bit is toggled.

The 'Results Buffer Enable' bit determines how decode results are handled by the reader. If the 'Results Buffer Enable' bit is set to false, then the decode results are immediately placed into the Results Module and 'Results Available' is set to true.

If the 'Results Buffer Enable' bit is set to true, the new results are queued. The earlier decode results remain in the Results Module until they are acknowledged by the client by setting the 'Results Acknowledge' bit to true. After the 'Results Available' bit is cleared, the client should set the 'Results Acknowledge' bit back to false to allow the next queued results to be placed in to the Results Module. This is a necessary handshake to ensure the results are received by the DataMan client (PLC).



Results Buffering

There is an option to enable a queue for decode results. If enabled, this allows a finite number of decode result data to queue up until the client (PLC) has time to read them. This is useful to smooth out data flow if the client (PLC) slows down for short periods of time or if there are surges of read activity.

Also, if result buffering is enabled, the device will allow overlapped acquisition and decode operations. Depending on the application, this can be used to achieve faster overall trigger rates. See Acquisition Sequence description above for further detail.

In general, if reads are occurring faster than results can be sent out, the primary difference between buffering or not buffering is determining which results get discarded. If buffering is not enabled, the most recent results are kept and the earlier result (which was not read by the PLC fast enough) is lost. Essentially, the more recent result will simply over write the earlier result. If buffering is enabled (and the queue becomes full), the most recent results are discarded until room becomes available in the results queue.

If the queue has overflowed and then buffering is disabled, there will be a greater than 1 difference between the 'Trigger ID' and 'Result ID' values. This difference represents the number of reads that had occurred, but could not be queued because the queue was full (number of lost reads equals 'Trigger ID' – 'Result ID' – 1). After the next read, the 'Result ID' value will return to the typical operating value of 'Trigger ID' – 1.

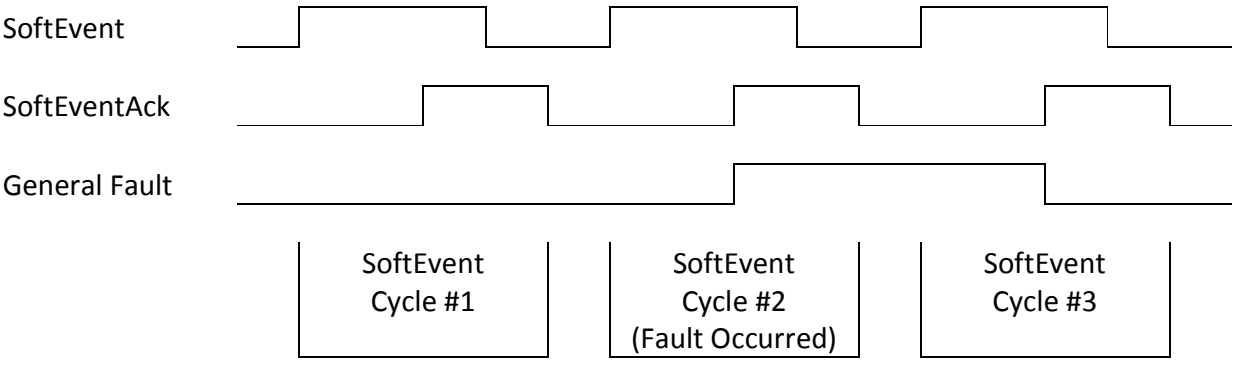
General Fault Indicator

When a communication-related fault occurs, the 'General Fault' bit will change from 0➔1. Currently, the only fault conditions supported are soft event operations. If a soft event operation fails, the fault bit will be set. The fault bit will remain set until the next soft event operation, or until triggering is disabled and again re-enabled.

SoftEvents

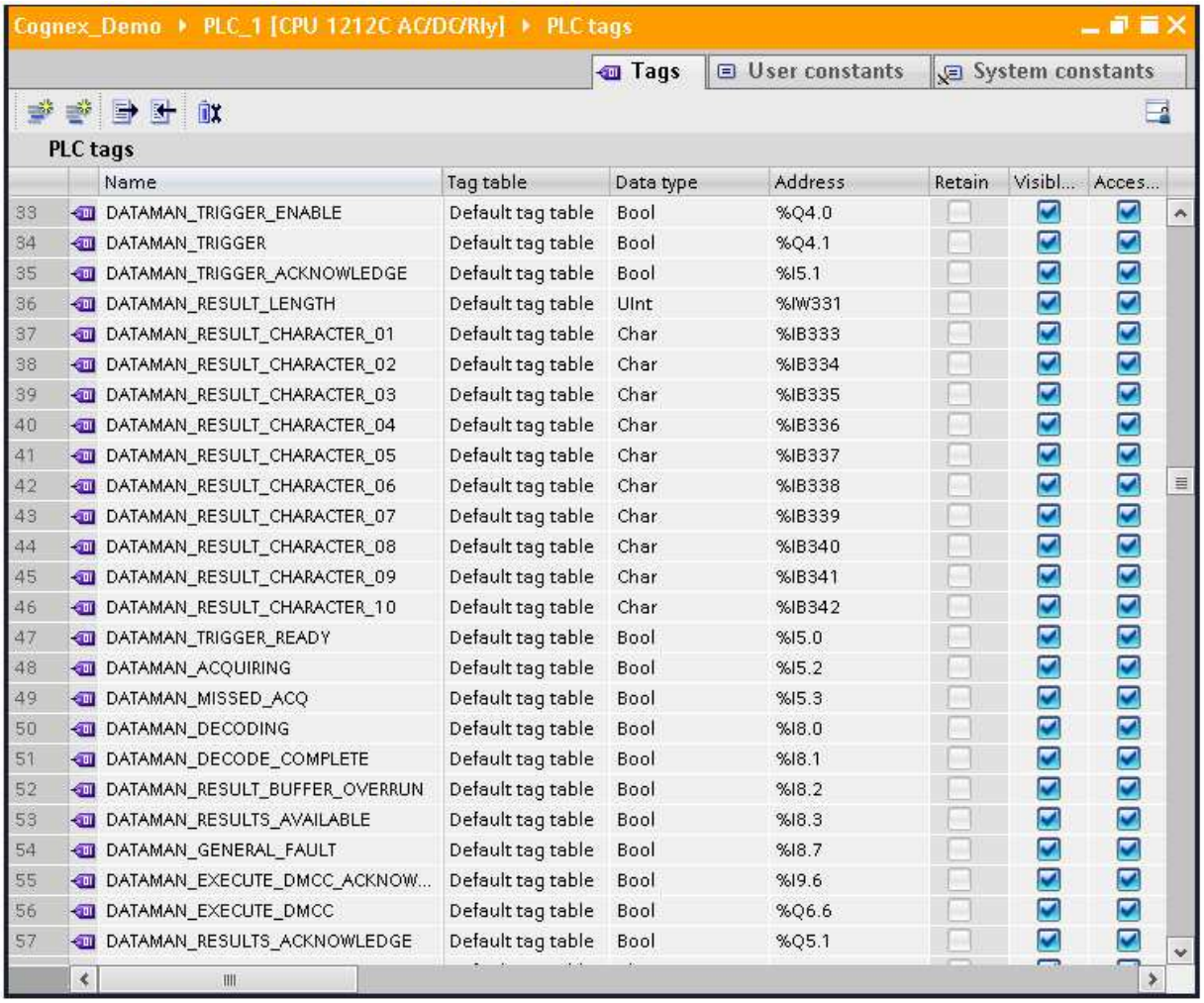
SoftEvents act as "virtual' inputs. When the value of a SoftEvent changes from 0➔1, the action associated with the event will be executed. When the action completes, the corresponding 'SoftEventAck' bit will change from 0➔1 to signal completion. The acknowledge bit will change back to 0 when the corresponding SoftEvent bit is set back to 0.

The 'Execute DMCC' and 'Set Match String' soft event actions require user supplied data. This data must be written to the 'User Data' and 'User Data Length' areas of the User Data Module prior to invoking the soft event. Since both of these soft events depend on the 'User Data,' only one may be invoked at a time.



Add PLC Tags

Under the Project Tree, a folder called 'PLC Tags' is present. Expand the folder and double-click 'Show All Tags.' Add the following tags:

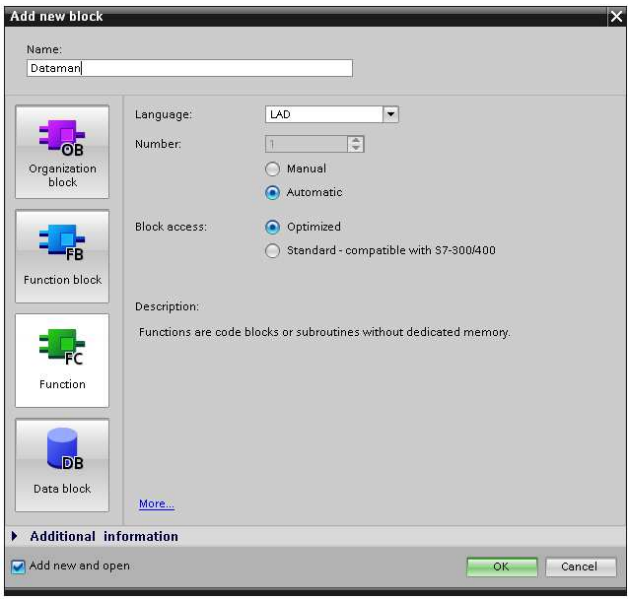


	Name	Tag table	Data type	Address	Retain	Visibl...	Acces...
33	DATAMAN_TRIGGER_ENABLE	Default tag table	Bool	%Q4.0		☑	☑
34	DATAMAN_TRIGGER	Default tag table	Bool	%Q4.1		☑	☑
35	DATAMAN_TRIGGER_ACKNOWLEDGE	Default tag table	Bool	%I5.1		☑	☑
36	DATAMAN_RESULT_LENGTH	Default tag table	UInt	%IW331		☑	☑
37	DATAMAN_RESULT_CHARACTER_01	Default tag table	Char	%IB333		☑	☑
38	DATAMAN_RESULT_CHARACTER_02	Default tag table	Char	%IB334		☑	☑
39	DATAMAN_RESULT_CHARACTER_03	Default tag table	Char	%IB335		☑	☑
40	DATAMAN_RESULT_CHARACTER_04	Default tag table	Char	%IB336		☑	☑
41	DATAMAN_RESULT_CHARACTER_05	Default tag table	Char	%IB337		☑	☑
42	DATAMAN_RESULT_CHARACTER_06	Default tag table	Char	%IB338		☑	☑
43	DATAMAN_RESULT_CHARACTER_07	Default tag table	Char	%IB339		☑	☑
44	DATAMAN_RESULT_CHARACTER_08	Default tag table	Char	%IB340		☑	☑
45	DATAMAN_RESULT_CHARACTER_09	Default tag table	Char	%IB341		☑	☑
46	DATAMAN_RESULT_CHARACTER_10	Default tag table	Char	%IB342		☑	☑
47	DATAMAN_TRIGGER_READY	Default tag table	Bool	%I5.0		☑	☑
48	DATAMAN_ACQUIRING	Default tag table	Bool	%I5.2		☑	☑
49	DATAMAN_MISSED_ACQ	Default tag table	Bool	%I5.3		☑	☑
50	DATAMAN_DECODING	Default tag table	Bool	%I8.0		☑	☑
51	DATAMAN_DECODE_COMPLETE	Default tag table	Bool	%I8.1		☑	☑
52	DATAMAN_RESULT_BUFFER_OVERRUN	Default tag table	Bool	%I8.2		☑	☑
53	DATAMAN_RESULTS_AVAILABLE	Default tag table	Bool	%I8.3		☑	☑
54	DATAMAN_GENERAL_FAULT	Default tag table	Bool	%I8.7		☑	☑
55	DATAMAN_EXECUTE_DMCC_ACKNOW...	Default tag table	Bool	%I9.6		☑	☑
56	DATAMAN_EXECUTE_DMCC	Default tag table	Bool	%Q6.6		☑	☑
57	DATAMAN_RESULTS_ACKNOWLEDGE	Default tag table	Bool	%Q5.1		☑	☑

These tags will be used for control of the DataMan.

Add DataMan Function Block

Under ‘Program Blocks,’ double-click ‘Add New Block.’ This will display a popup. Select ‘Function Block’ in the popup.



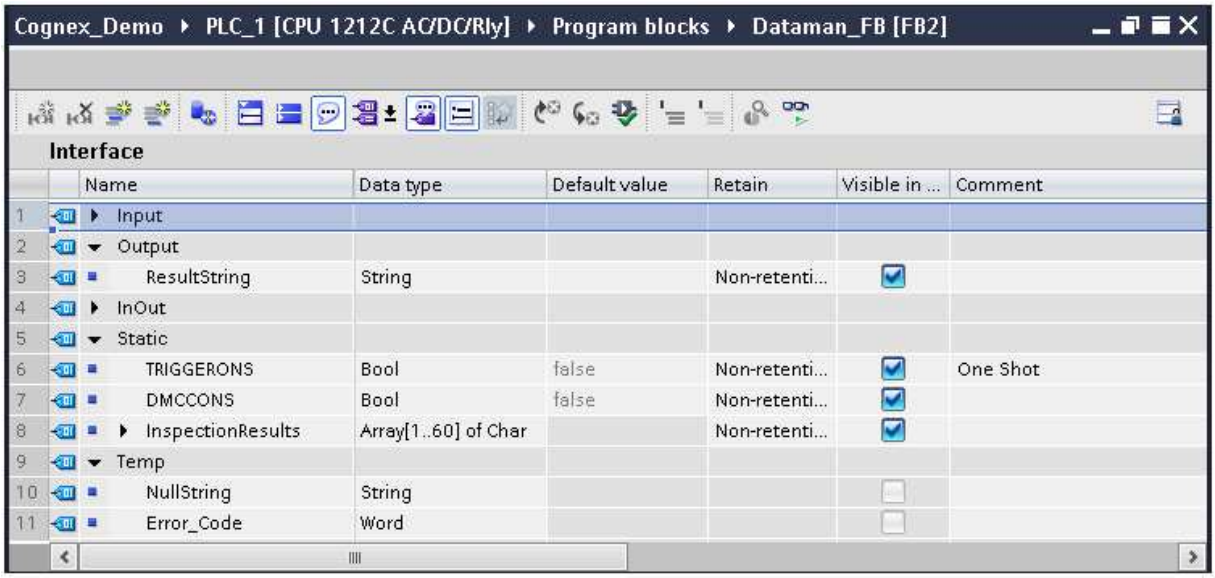
Call the Function Block Dataman_FB and press ‘OK.’ This block will be FB2.

Now, the function block needs to be called. Double-Click on ‘Main [OB1]’ in the Project Tree. Add an empty box and type FB2 and press the ‘Enter’ key. This will display a popup to create a Data Block for the Function Block.

Press ‘OK’ and the function block call is complete.

Add Function Block Tags

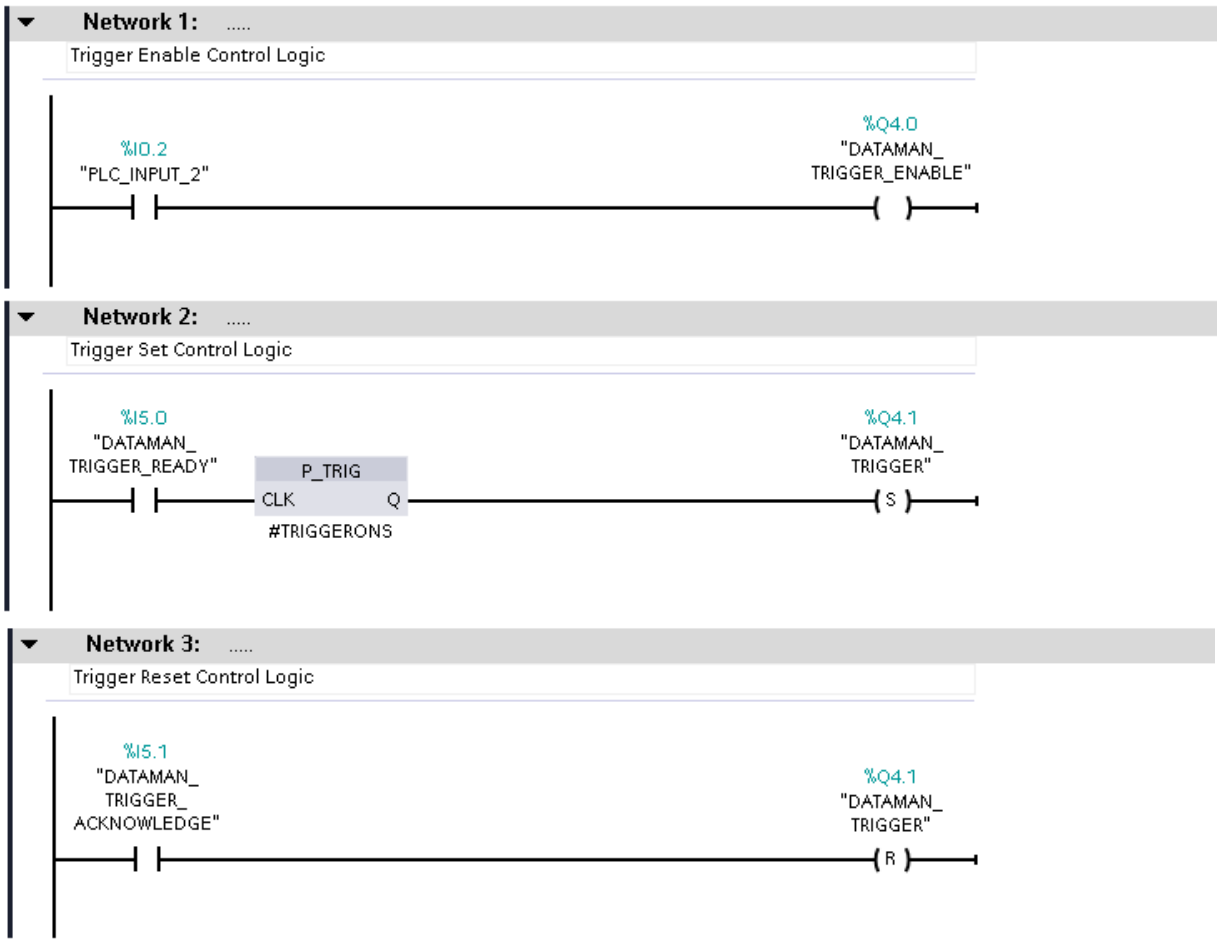
Double-click the 'DataMan_FB[FB2]' function block. Add the following tags to the Function Block:



These tags will be used in the function block to send and receive data to/from the DataMan.

Add Trigger

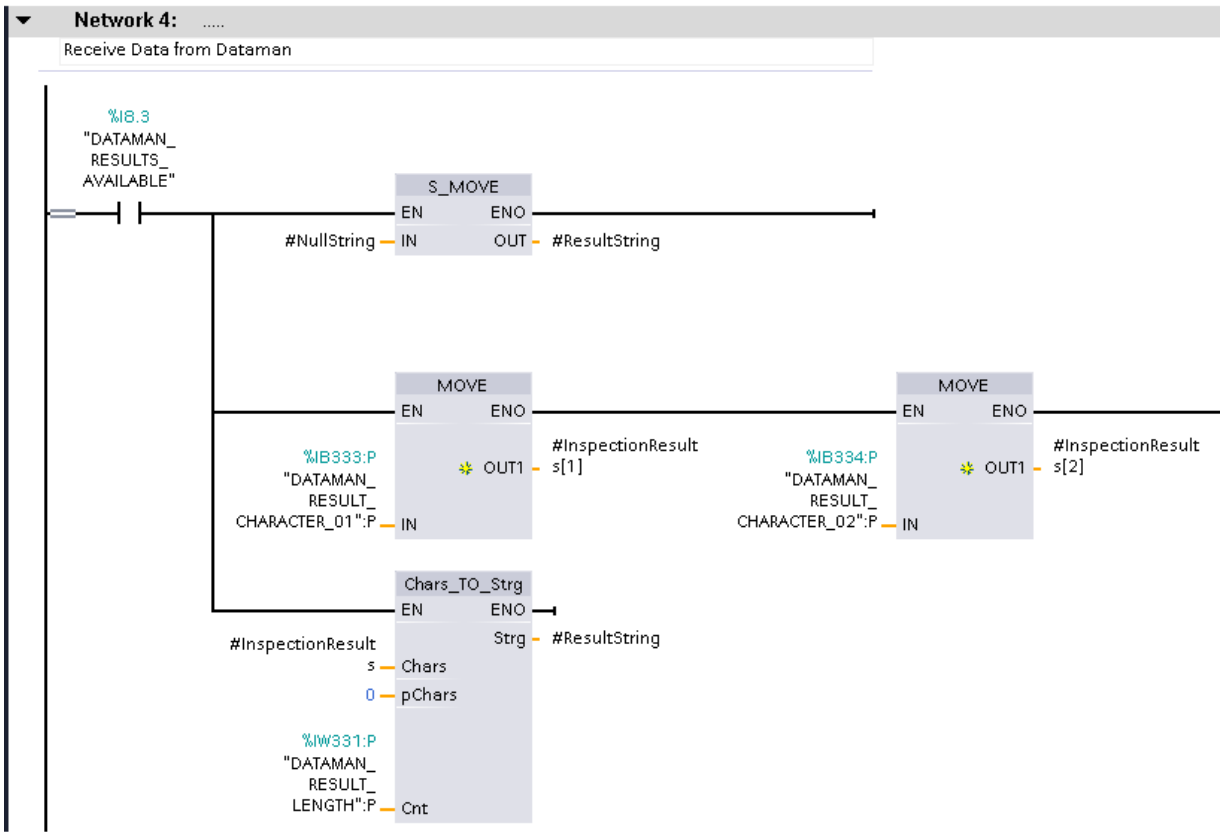
The following code shows how to add a network trigger to the PLC program:



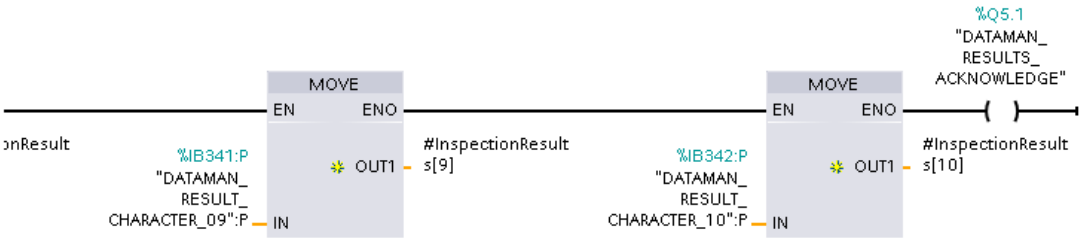
Four tags, defined by the DataMan device library, are used to control the trigger. In the first network (Network 1), the 'Trigger Enable' bit is set. When the PLC input I0.2 is high, the 'Trigger Enable' bit goes high. This allows a network trigger to be sent to the DataMan. The rising edge of the 'Trigger Ready' bit is used to set the 'Trigger' output (Network 2). The 'Trigger Ready' signal is sent by the DataMan when it receives the 'Trigger Enable' signal from the PLC. The 'Trigger' output is reset when the DataMan sends a 'Trigger Acknowledge' signal back to the PLC (Network 3). To test the trigger to the DataMan, toggle PLC input I0.2.

Add Results

When the ‘Results Available’ bit is set by the DataMan, the PLC can collect the result data. Each byte of data is combined into a string tag for further processing or display.



Continue to add MOVE instructions to get each byte of data back from the DataMan. At the end of the Network, add the ‘Results Acknowledge’ bit. This will reset the ‘Results Available’ bit sent from the DataMan.



Add DMCC Command

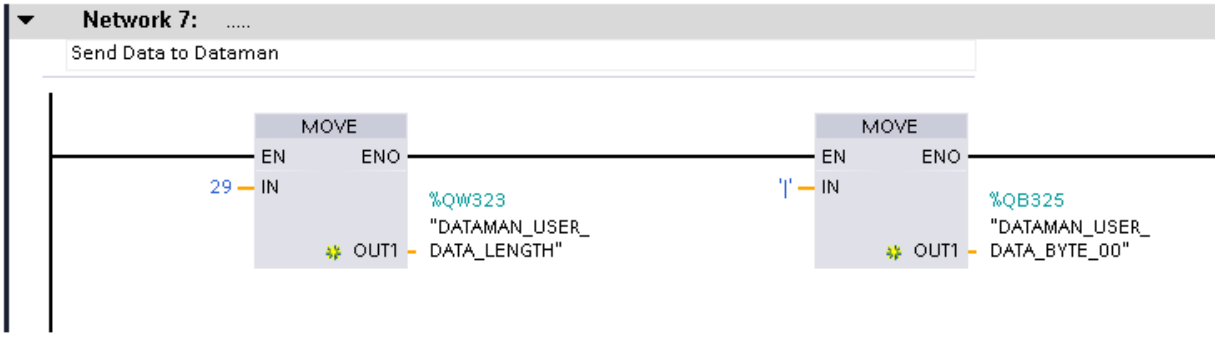
DataMan Control Commands (DMCC) can be sent to the DataMan for control of a multitude of functions. See the Command Reference help file that is installed with the DataMan Setup Tool. It will give a list of available DMCC commands.

To send a DMCC command from the PLC to the DataMan, the User Data array must be populated with the command. The first two bytes of the User Data array are used for the User Data Option. This is not required for this example. The next two bytes are used for the User Data Length. This is the total length of the data to be sent to the DataMan. The rest of the bytes in the array are the actual data. In this case, it is the DMCC Command.

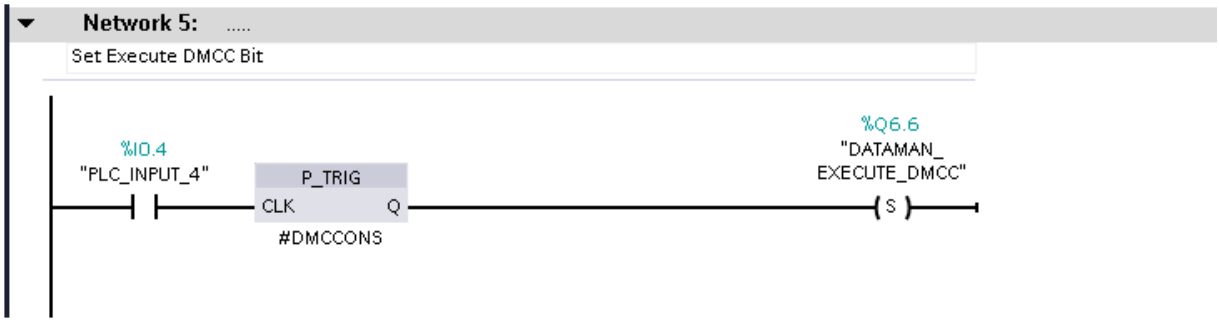
The following command will enable the 2D Data Matrix symbology:

||>SET SYMBOL.DATAMATRIX ON

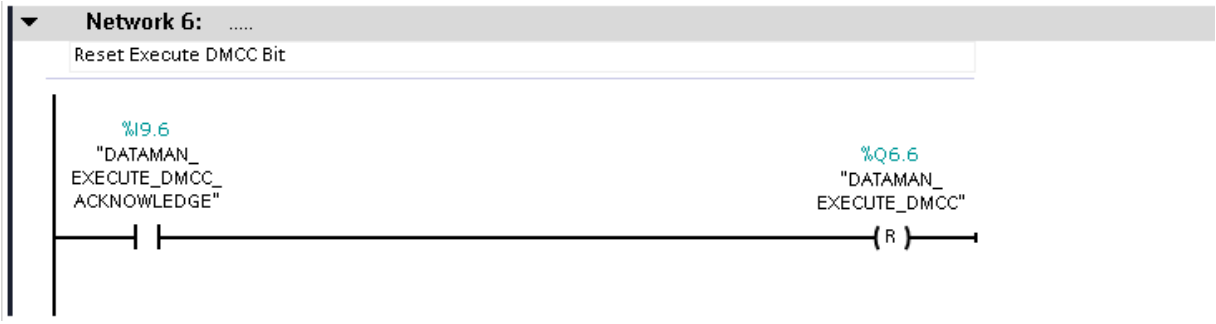
Carriage Return and Line Feed characters are added to the end of the command string. These bytes are added to the User Data Array, as well as the length of the data string (in this case, 29).



After the data is loaded, the PLC can enable the 'Execute DMCC' bit (Network 5).



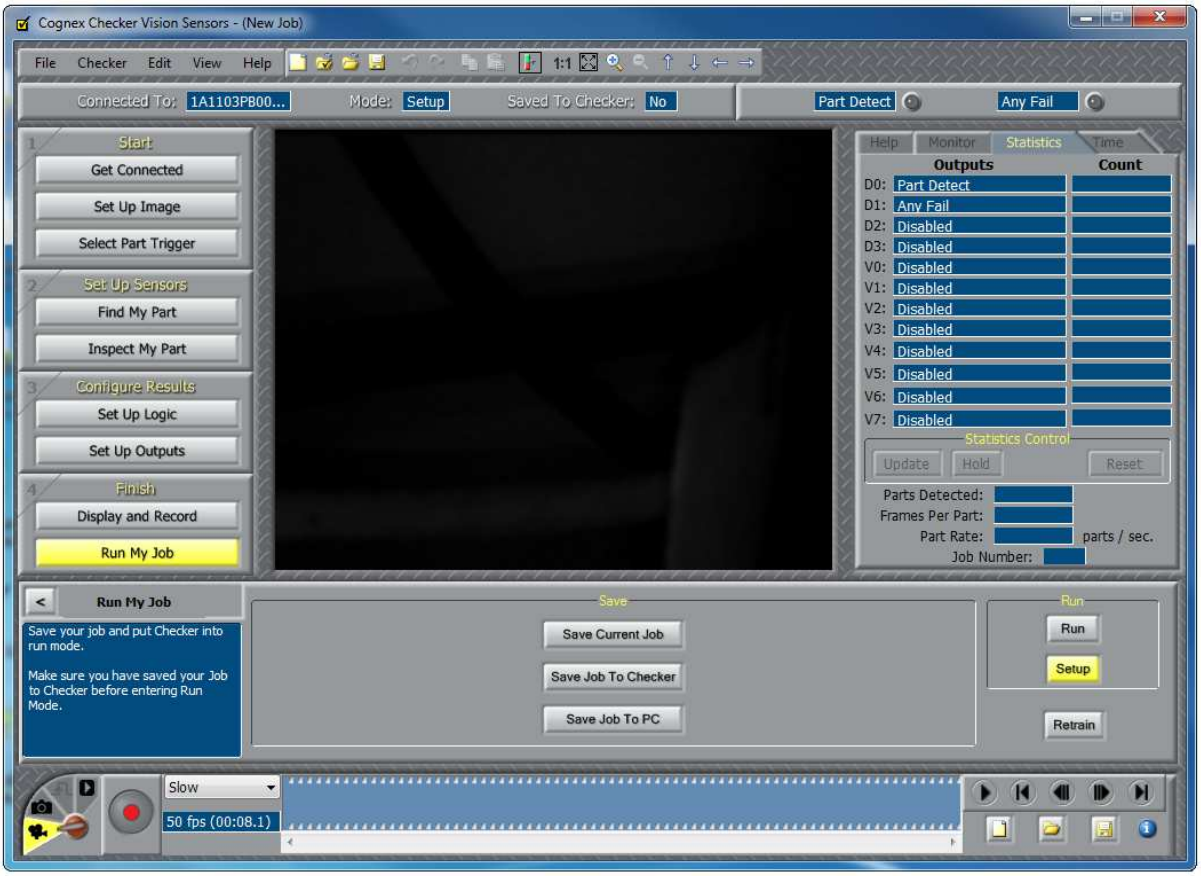
The DataMan will execute the command that is in the User Data array. After execution of the command, the DataMan will send the Execute DMCC Acknowledge signal to the PLC. This signal is used to reset the Execute DMCC bit (Network 6).



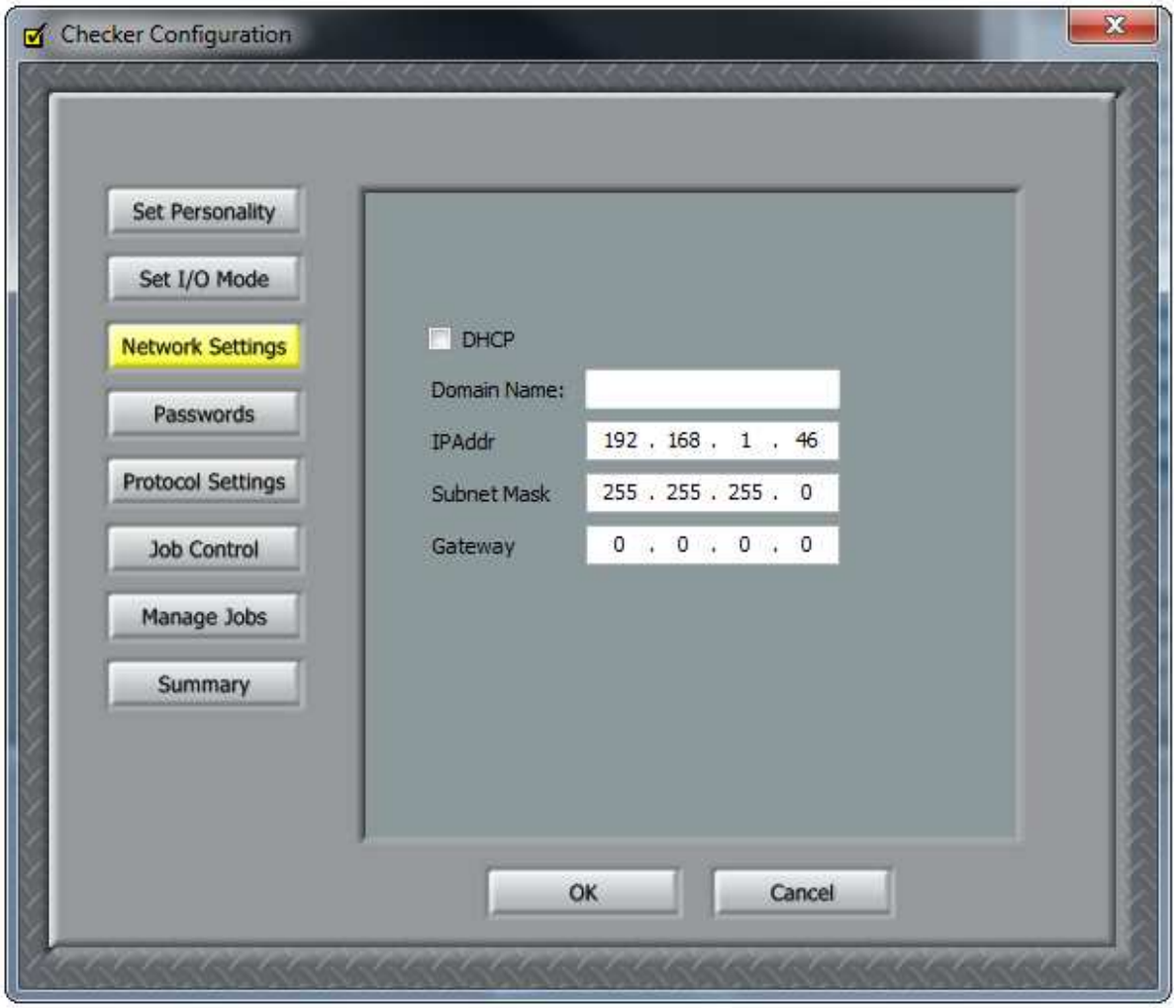
Connecting to Checker

Checker Configuration

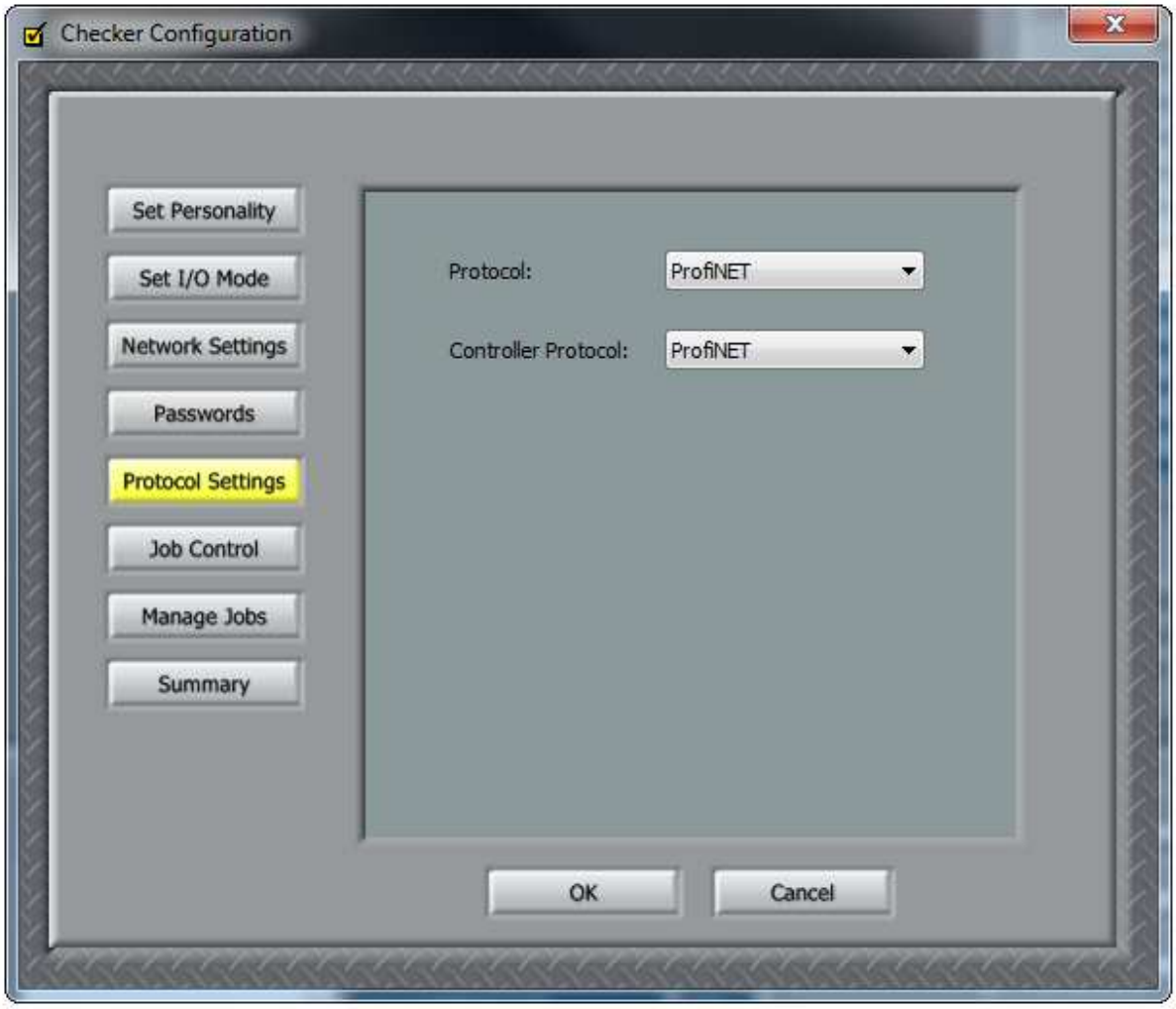
For this example, a Checker 4G1 was used. First, make sure the Checker is in Setup Mode. This can be found in the lower right corner of the Checker software, under the 'Run' section.



Next, set the IP Address of the Checker. When the Checker➔Configure Checker option is selected, a popup will appear. Select 'Network Settings' and set the IP Address. In this case, set the IP Address to 192.168.1.46.



Finally, select ‘Protocol Settings’ on the ‘Checker Configuration’ popup. Make sure both the ‘Protocol’ and ‘Controller Protocol’ settings are configured for ‘ProfiNET.’

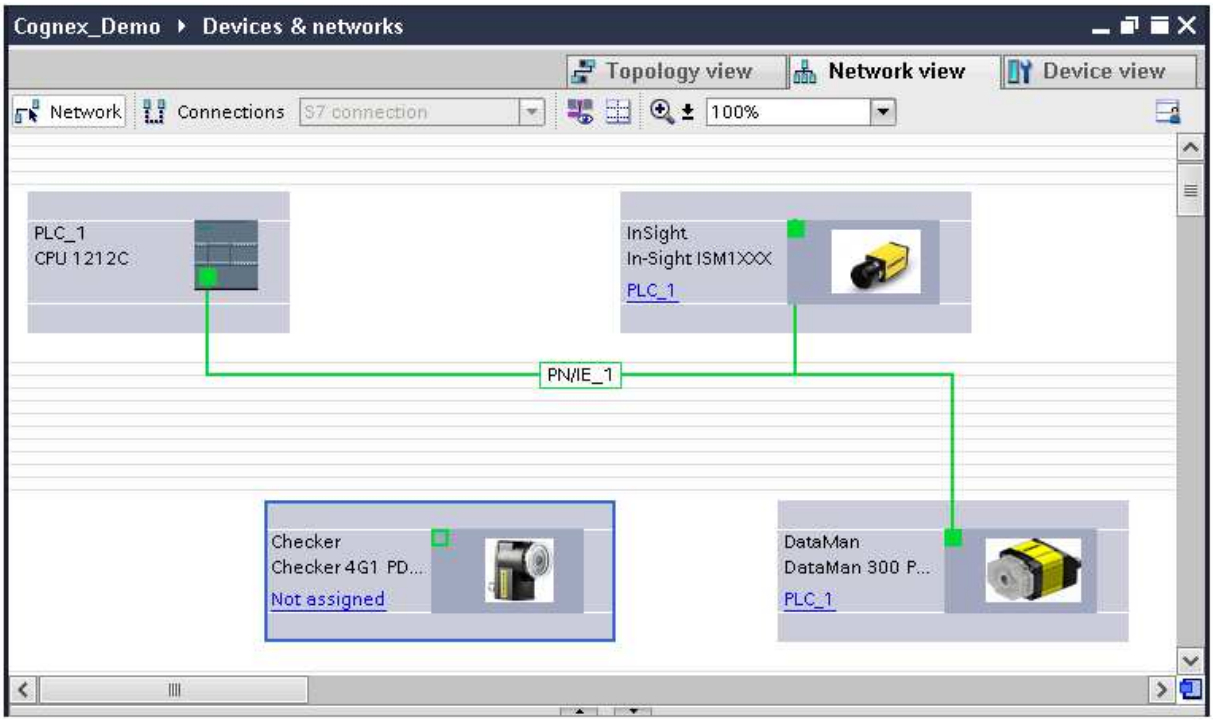


The system will need to be restarted to enable Profinet communications. This is done automatically when the ‘OK’ button is pressed.

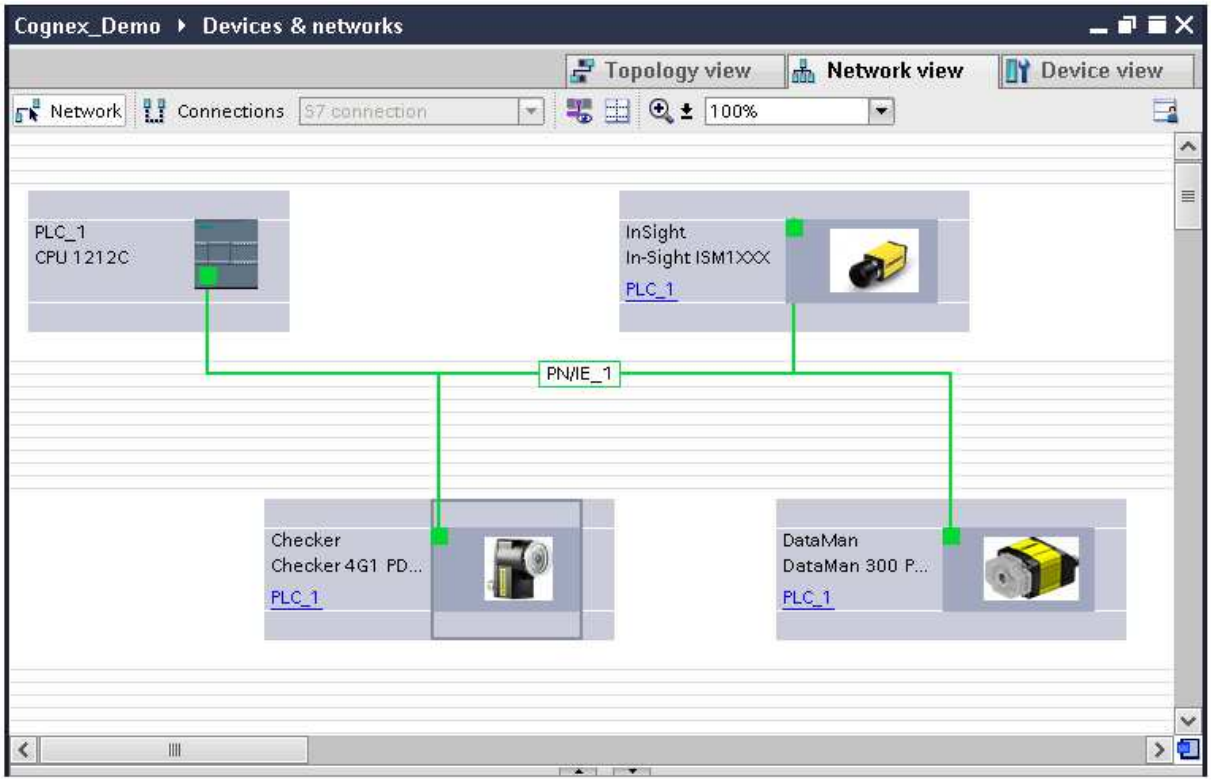
PLC Programming

Add Checker to Network

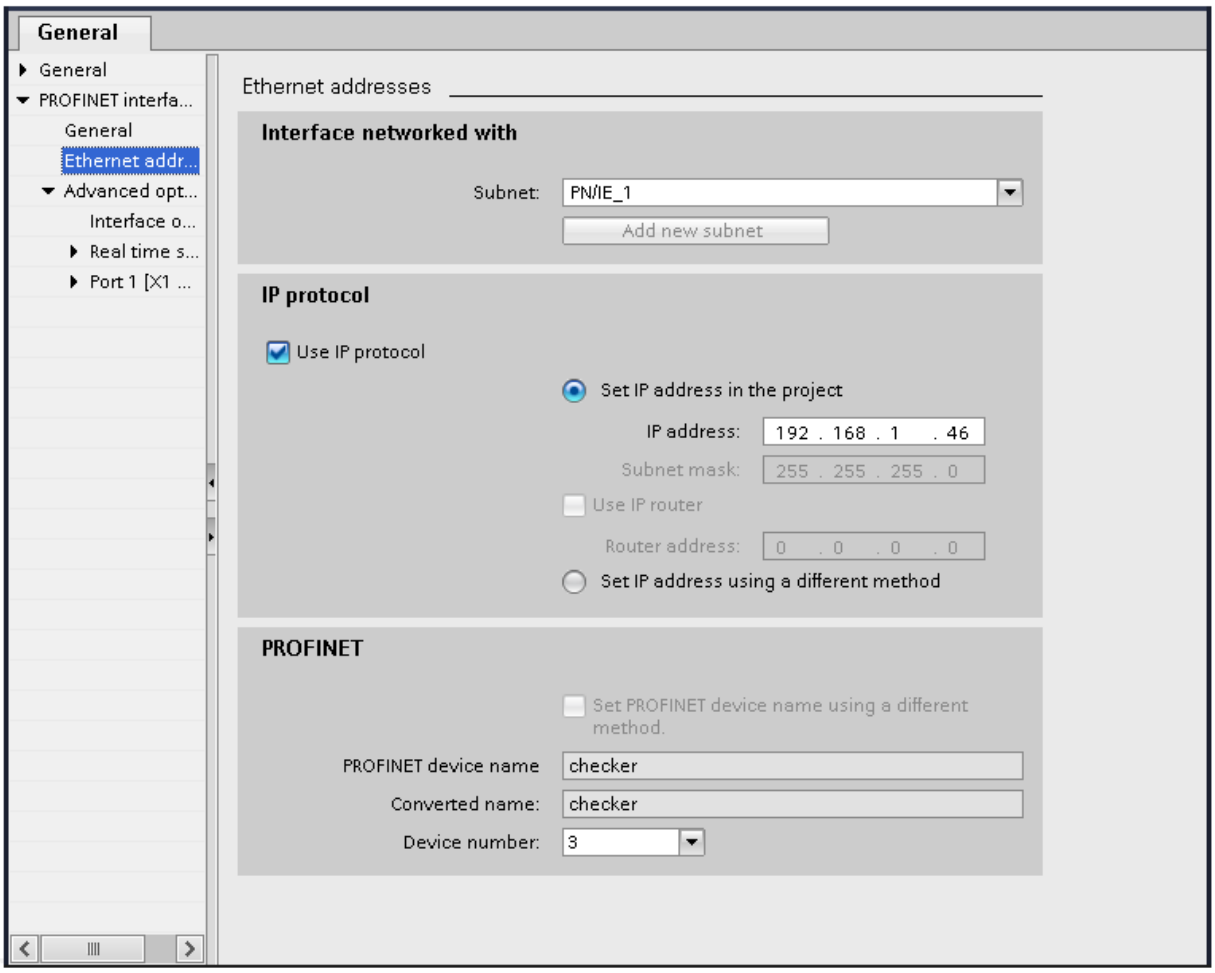
Under 'Devices & Networks,' select 'Network View.' Add Checker 4G1 PDEV. This is listed under 'Other Field Devices'➔PROFINET IO➔Sensors➔Cognex Corp➔Cognex Vision Sensors. A device can be added by dragging the library object into the 'Network View.' The Checker device will be listed as 'Not Assigned.'



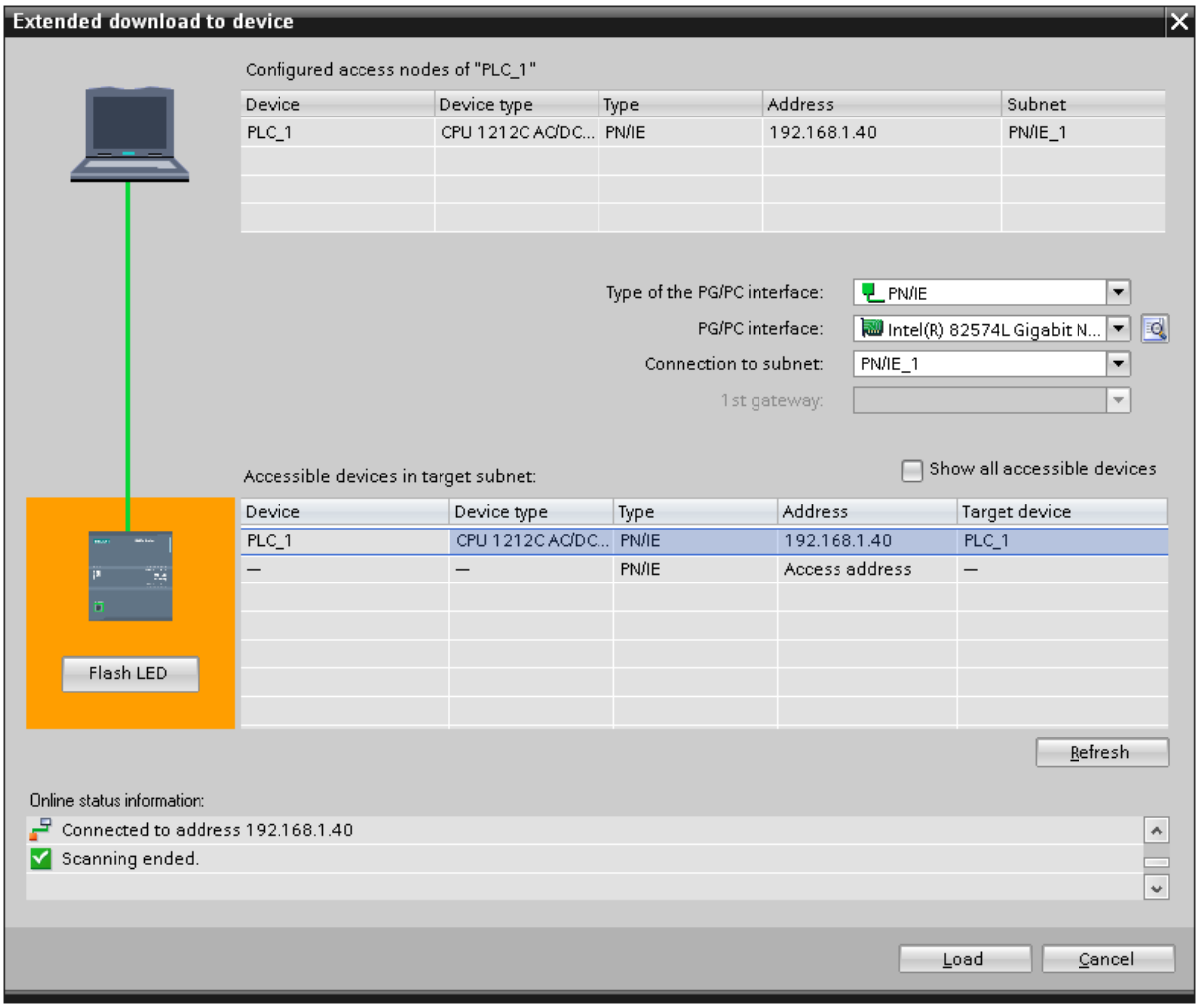
To assign the device, press and hold the left mouse button on the green box in the Checker object, and drag your mouse to the green box in the PLC. Release the mouse button. This establishes the connection between the PLC and the Checker.



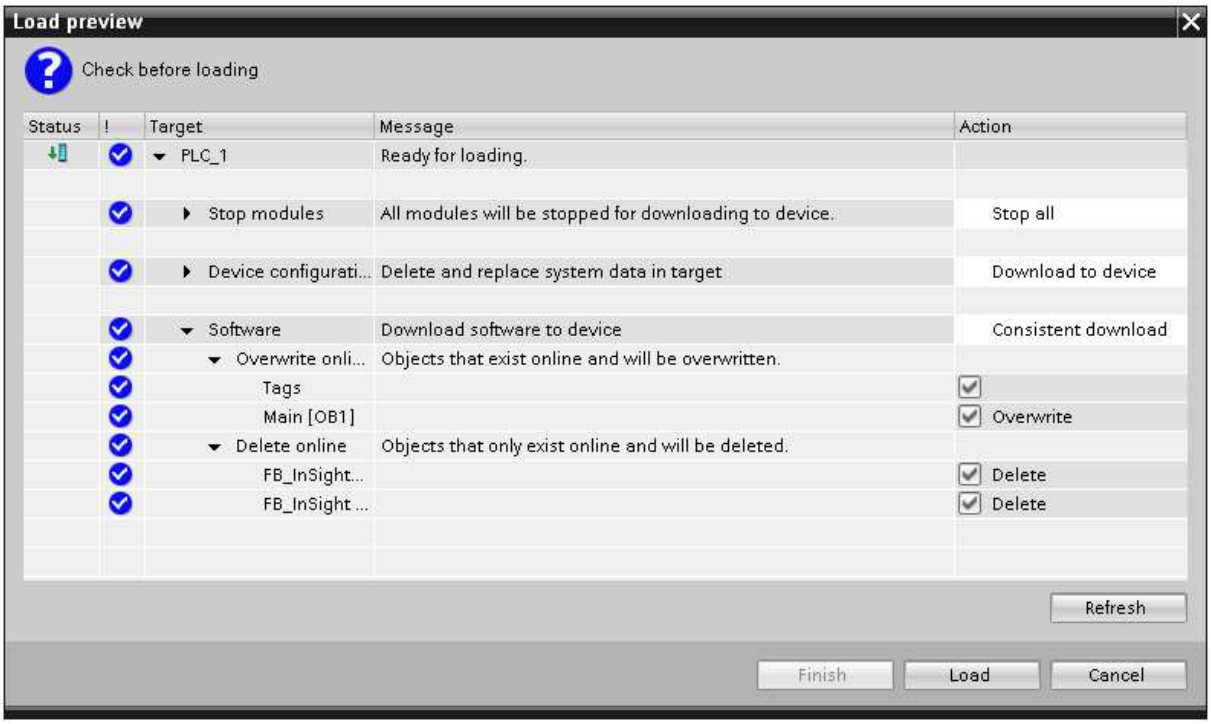
Select the camera. Under the ‘General Properties’ tab, select ‘PROFINET Interface.’ Change the name to DataMan and change the IP Address to 192.168.1.46.



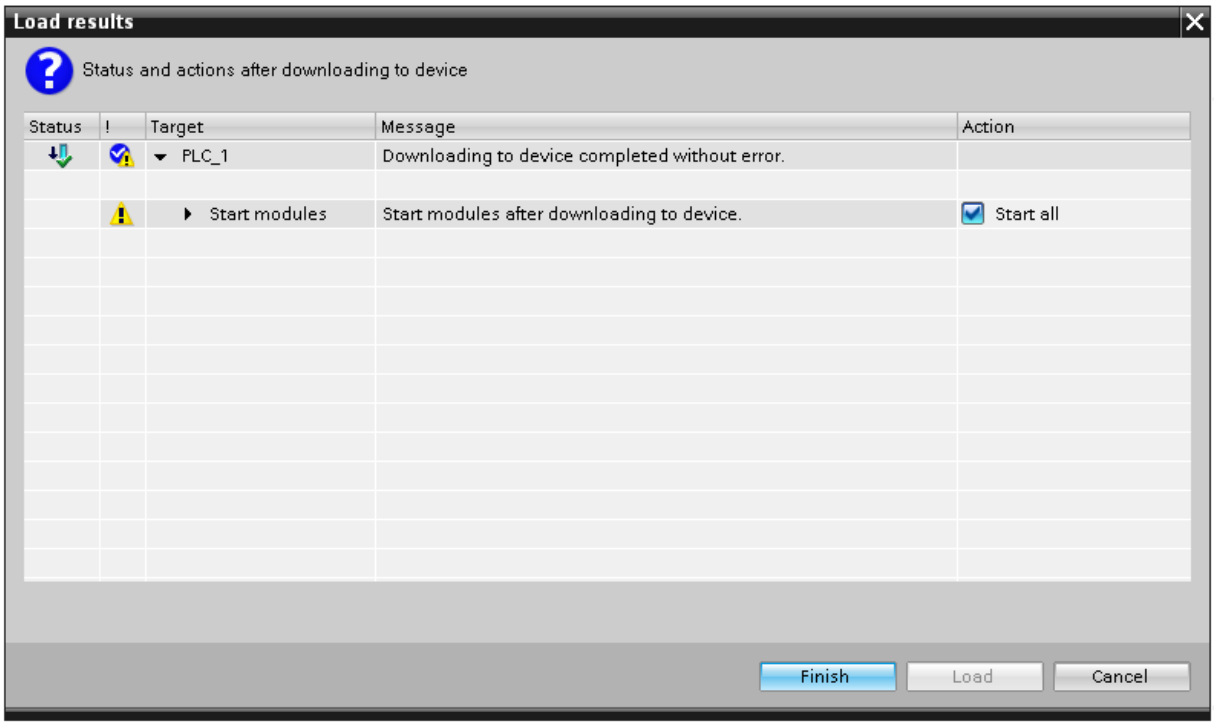
The program is ready to download to the PLC to check communication status. Select the PLC in the Project Tree. Under the ‘Online’ menu, select ‘Download to Device.’ A popup will appear.



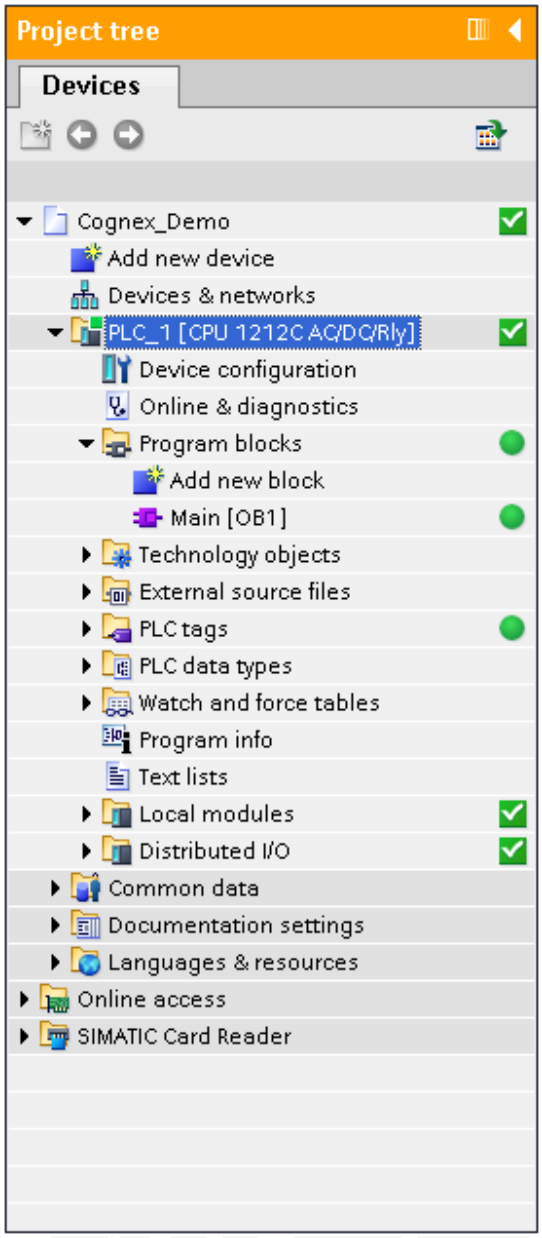
Press ‘Load.’ A new popup will appear. Do not be alarmed if your Load Preview popup or Load Results popup do not appear exactly as shown.



Press ‘Load.’ A new popup will appear.

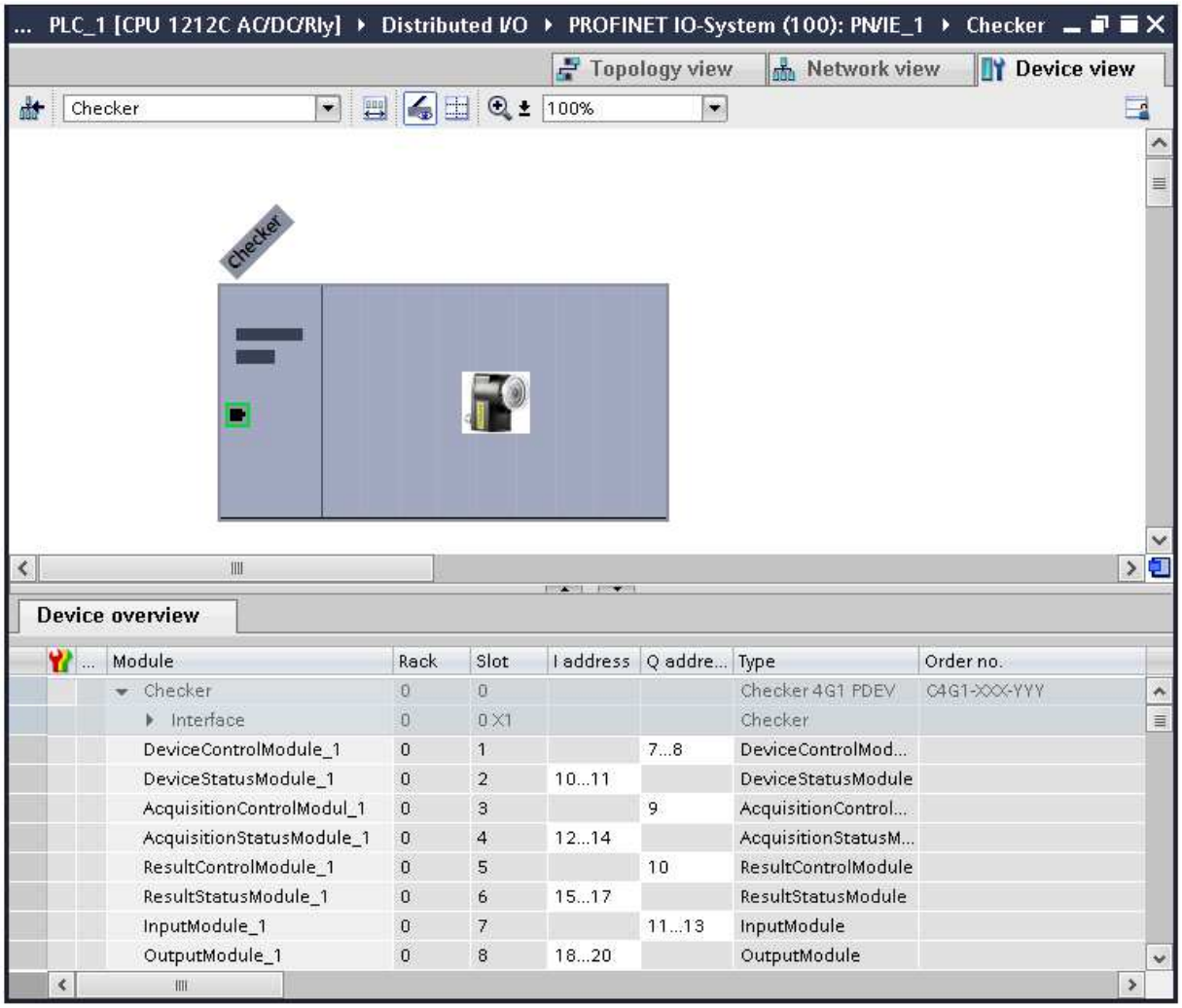


Press 'Finish' to finish the download process. Go online with the processor by selecting the PLC in the Project Tree, and select 'Go Online' under the 'Online' drop-down menu. If communications are correct, green check marks will appear next to all devices.


Project tree
Devices
Cognex_Demo
Add new device
Devices & networks
PLC_1 [CPU 1212C AC/DC/Rly]
Device configuration
Online & diagnostics
Program blocks
Add new block
Main [OB1]
Technology objects
External source files
PLC tags
PLC data types
Watch and force tables
Program info
Text lists
Local modules
Distributed I/O
Common data
Documentation settings
Languages & resources
Online access
SIMATIC Card Reader

Device I/O

When the Checker object was added to the system, I/O was assigned for the control elements of the camera.



Module	Rack	Slot	I address	Q addre...	Type	Order no.
Checker	0	0			Checker 4G1 PDEV	C4G1-XXX-YYY
Interface	0	0 X1			Checker	
DeviceControlModule_1	0	1		7...8	DeviceControlMod...	
DeviceStatusModule_1	0	2	10...11		DeviceStatusModule	
AcquisitionControlModul_1	0	3		9	AcquisitionControl...	
AcquisitionStatusModule_1	0	4	12...14		AcquisitionStatusM...	
ResultControlModule_1	0	5		10	ResultControlModule	
ResultStatusModule_1	0	6	15...17		ResultStatusModule	
InputModule_1	0	7		11...13	InputModule	
OutputModule_1	0	8	18...20		OutputModule	

The I/O assignments break down as follows:

Device Control Module		
Bit	**Name**	**Description**
0	Set Offline	Setting this bit puts the Checker into Offline mode. This means that the Checker suspends all activities and does not react to any control commands.
1	Job Change	This bit is used for changing jobs. Before (or at the same time) this bit is changed to 1 by the PLC, make sure that the Job Number is filled in.
2	Retrain	This bit is used to retrain the sensor.
3	Lights Off	Setting the Lights Off bit to 1 turns off the external illumination, regardless of job setting.
4	Volatile	Setting the Volatile bit to 0 selects the static results data that does not change for a job. Setting it to 1 selects the volatile data that varies from part to part.
8-15	Job Number	This byte represents the job number to be loaded.

Device Status Module		
Bit	**Name**	**Description**
0	Online	If the Checker is Online, its value is 1 and the Checker reacts to commands. Otherwise, its value is 0.
1	Offline Reason 1	If this is 1, the sensor is Offline. If this is 0, the sensor is Online. Setup Mode.
2	Offline Reason 2	Not Used.
3	Offline Reason 3	Not Used.
4	General Fault	It is always 0.
5	Job Load Complete	If the Job Change is successful, this changes to 1.
6	Job Load Failed	If the Job Change is unsuccessful, this changes to 1.
7	Retrain Complete	If this is 1, the Retrain operation for the sensor completed successfully.
8	Retrain Failed	If this is 1, the Retrain operation for the sensor failed.
9	Observer	If this is 1, the PROFINET controller cannot change the status of the Checker and can only observe its state.

Acquisition Control Module		
Bit	**Name**	**Description**
0	Trigger	Setting this bit triggers an acquisition when the following conditions are met: • Trigger Enable is set. • No acquisition is currently in progress. • The device is ready to trigger.
1	Trigger Enable	Setting this bit enables triggering via PROFINET. Clearing this bit disables triggering.
2-7	Reserved	Reserved for Future Use.

Acquisition Status Module		
Bit	**Name**	**Description**
0	Trigger Ready	Indicates when the device is ready to accept a new trigger. This bit is true when the Trigger Enable has been set and the device is ready to accept a new trigger.
1	Trigger Acknowledge	This indicates that the Checker has received a new Trigger and the trigger process has started. This bit will remain true as long as the Trigger bit remains true (that is, it is interlocked with the Trigger bit).
2	Acquiring	Indicates that the Checker is currently acquiring an image.
3	Missed Acquisition	Indicates that the Checker was unable to successfully trigger an acquisition. The bit is cleared when the next successful acquisition occurs.
4-7	Reserved	Reserved for Future Use.
8-23	Acquisition ID	This is the ID value of the next trigger to be issued (16-bit integer). It is used to match issued triggers with corresponding result data received later. This same value will be returned in Result ID of the Results Status Module.

Results Control Module		
Bit	**Name**	**Description**
0	Results Buffer Enable	This enables queuing of Result Data. If enabled, the current result data will remain until acknowledged (even is new results arrive). New results are queued. The next set of results are pulled from the queue (made available in the Result Data module) each time the current results are acknowledged. The Checker will respond to the acknowledgement by clearing the Results Available bit. Once the Results Acknowledge bit is cleared, the next set of read results will be posted and Results Available will be set to true. If results buffering is not enabled, newly received read results will simply overwrite the content of the Result Data module.
1	Results Acknowledge	This bit is used to acknowledge that the PLC has successfully read the latest result data. When set to true, the Result Available bit will be cleared. It is only used if Results Buffering is enabled.
2-7	Reserved	Reserved for Future Use.

Results Status Module		
Bit	**Name**	**Description**
0	Part Detect	Its value is 1 if the Part Finding Sensor was successful. If unsuccessful, its value is 0. It is valid if Results Available is 1.
1	Inspecting	Its value is 1 when the Checker is inspecting. Otherwise, its value is 0.
2	Inspection Complete Toggle	It changes value when the inspection is complete.
3	Result Buffer Overrun	This bit is only valid in case of Result Buffering (Buffer Result Enable = 1). Its value changes to 1 if there is no place in the buffer for the last inspection result.
4	Result Available	Its value changes to 1 when the inspection is finished. See also Inspection Complete Toggle.
5	Any Fail	Its value is 1 if the inspection of at least one of the sensors defined in the current job is unsuccessful. If the inspections of all sensors are successful, its value is 0. This bit is valid if Result Available is 1.
6	All Pass	Its value is 1 if all the inspections of all the sensors defined in the current job are successful. If any of them is not successful, its value is 0. Valid when Result Available is 1.
7-22	Result ID Register	This bit is the pair of Acquisition ID.

Input Module		
Bit	**Name**	**Description**
0-23	Inputs 0 through 23	These bits are used as Inputs in the Ladder Logic of the Checker GUI.

Output Module		
Bit	**Name**	**Description**
0-23	Outputs 0 through 23	The function of these bits can be changed in the Checker GUI. They can be any of the following virtual outputs: • Part Detect • All Pass • All Fail • Coil • External Trigger • External Retrain Pass • External Retrain Fail • Job Change Pass • Job Change Fail

Sensor Meter Module		
Byte	**Name**	**Description**
0-255	Meter Information	The order of the sensor meter values available through PROFINET is determined by the order of the sensors you set in the Sensor Meter table.

Device Operation

Acquisition Sequence

Checker can be triggered to acquire images implicitly via the Assembly object.

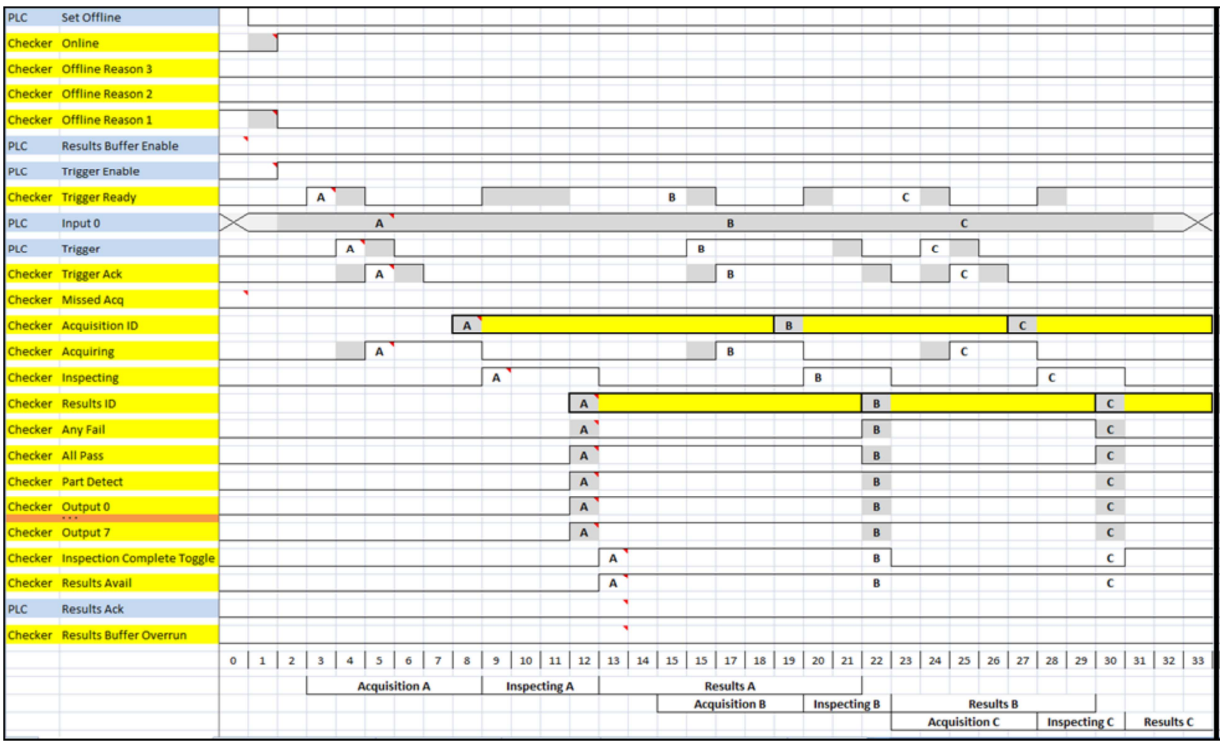
On startup the 'Trigger Enable' attribute will be false. It must be set to true to enable triggering. When the device is ready to accept triggers, the 'Trigger Ready' bit in the Acquisition Status Module will be set to true.

While the 'Trigger Enable' attribute is true and the 'Trigger Ready' bit is true, each time the Checker object sees the 'Trigger' attribute change from 0 to 1, it will initiate an image acquisition. When setting this via the assembly objects, the attribute should be held in the new state until that same state value is seen in the 'Trigger Acknowledge' bit of the Acquisition Status Module.

During an acquisition, the 'Trigger Ready' bit in the Acquisition Status Module will be cleared and the 'Acquiring' bit will be set to true. When the acquisition is completed, the 'Acquiring' bit will be cleared and the 'Trigger Ready' bit will again be set true once the acquisition system is ready to begin a new image acquisition.

To force a reset of the trigger mechanism, set the 'Trigger Enable' attribute to false until the Acquisition Status Module is 0. Then, 'Trigger Enable' can be set to true to re-enable acquisition.

Triggering is only available when Checker is in External Trigger Mode. See the following figure for a typical acquisition sequence where results buffering is disabled.



NOTE: The state of the 'Any Fail,' 'All Pass,' 'Part Detect,' and 'Output 0-7' lines depends on the results of the inspection. The states shown are just examples.

'Results Buffer Enable' is false will disable 'Results Available Acknowledge' and 'Results Buffer Overrun.' 'Results Available' will stay true after the first set of results. Use 'Inspection Complete Toggle' to gate the results from the I/O buffers into PLC memory.

'Missed Acknowledge' will go true if 'Trigger Enable' is true, 'Trigger Ready' is false and a leading edge of 'Trigger' occurs. 'Missed Acknowledge' will stay true until a successful acquisition occurs.

If 'Set Offline' is true, or the Checker application goes into Setup mode, then 'Online' will go false and 'Offline Reason 1' will go true.

'Trigger Enable' should only go true if 'Online' is true first. 'Trigger Enable' is true if one of the signals required for 'Trigger Ready' goes true.

'Trigger Ready' will be true if 'Online' is true, 'Trigger Enable' is true, and 'Acquiring' is false.

'Trigger' can go true any time 'Trigger Ready' is true. 'Trigger' should go false once 'Trigger Acknowledge' goes true. The leading edge of 'Trigger' going true will cause 'Trigger Ready' to go false and will cause 'Acquiring' to go true.

The 'Input 0' line can be in any state during the sequence. The value at the leading edge of the 'Acquiring' signal is the value that will be used.

'Trigger Acknowledge' goes true when 'Trigger' goes true. 'Trigger Acknowledge' goes false when 'Trigger' goes false.

'Acquiring' will go true if 'Trigger Ready' is true and after the leading edge of the 'Trigger' signal. 'Acquiring' means that the Checker is taking a picture and moving the image to an acquire buffer. The trailing edge of 'Acquiring' is the gated signal for the 'Acquisition ID' and will cause 'Inspecting' to go true.

The 'Acquisition ID' must be valid and stable when the 'Acquiring' signal goes false. This number will be used to match up the acquired image with the inspection output by using the same number for the 'Results ID.'

'Inspecting' will go true immediately after 'Acquiring' goes false, as it is the next process in the pipeline. The trailing edge of 'Inspecting' is the gating signal for 'Results ID' and all of the results data. 'Inspection Complete Toggle' and 'Results Available' are also activated by the falling edge of 'Inspecting.'

'Results ID' must be valid and stable prior to the trailing edge of 'Results Available.' The 'Inspection Complete Toggle' will transition and 'Results Available' will go true on the trailing edge of 'Inspecting.' The 'Result ID' value must be the same as the 'Acquisition ID' value for the matching camera image.

'Any Fail' must be valid and stable prior to the trailing edge of 'Results Available.' The 'Inspection Complete Toggle' will transition and 'Results Available' will go true on the trailing edge of 'Inspecting.'

'All Pass' must be valid and stable prior to the trailing edge of 'Results Available.' The 'Inspection Complete Toggle' will transition and 'Results Available' will go true on the trailing edge of 'Inspecting.'

'Part Detect' must be valid and stable prior to the trailing edge of 'Results Available.' The 'Inspection Complete Toggle' will transition and 'Results Available' will go true on the trailing edge of 'Inspecting.'

'Output 0-7' must be valid and stable prior to the trailing edge of 'Results Available.' The 'Inspection Complete Toggle' will transition and 'Results Available' will go true on the trailing edge of 'Inspecting.'

'Inspection Complete Toggle' changes at the same time as the trailing edge of 'Inspecting.' The 'Results ID' is incremented every time 'Results Available' becomes true.

'Results Available' immediately follows the trailing edge of 'Inspecting' and must only go true once the results data is solidly stable and valid.

NOTE: For 'Results Buffer Enable' = false, after the first result, the 'Results Available' signal will always be true. Use 'Inspection Complete Toggle' as the enable signal to copy new result data from the I/O result buffers to PLC memory.

'Results Acknowledge' is not used if 'Results Buffer Enable' is false.

'Results Buffer Overrun' is not used if 'Results Buffer Enable' is false.

Inspection/Result Sequence

After an image is acquired, it is inspected. While being inspected, the 'Inspection' bit of the 'Result Status Module' is set. When the inspection is complete, the 'Inspection' bit is cleared and the 'Inspection Complete' bit is toggled.

The 'Results Buffer Enable' bit determines how inspection results are handled by the sensor. If the 'Results Buffer Enable' bit is set to false, then the inspection results are immediately placed into the 'Results Module' and 'Results Available' is set to true.

If the 'Results Buffer Enable' bit is set to true, the new results are queued. The earlier inspection results remain in the 'Results Module' until they are acknowledged by the client by setting the 'Results Acknowledge' bit to true. After the 'Results Available' bit is cleared, the client should set the 'Results Acknowledge' bit back to false to allow the next queued results to be placed into the 'Results Module.' This is a necessary handshake to ensure the results are received by the Checker client (PLC).

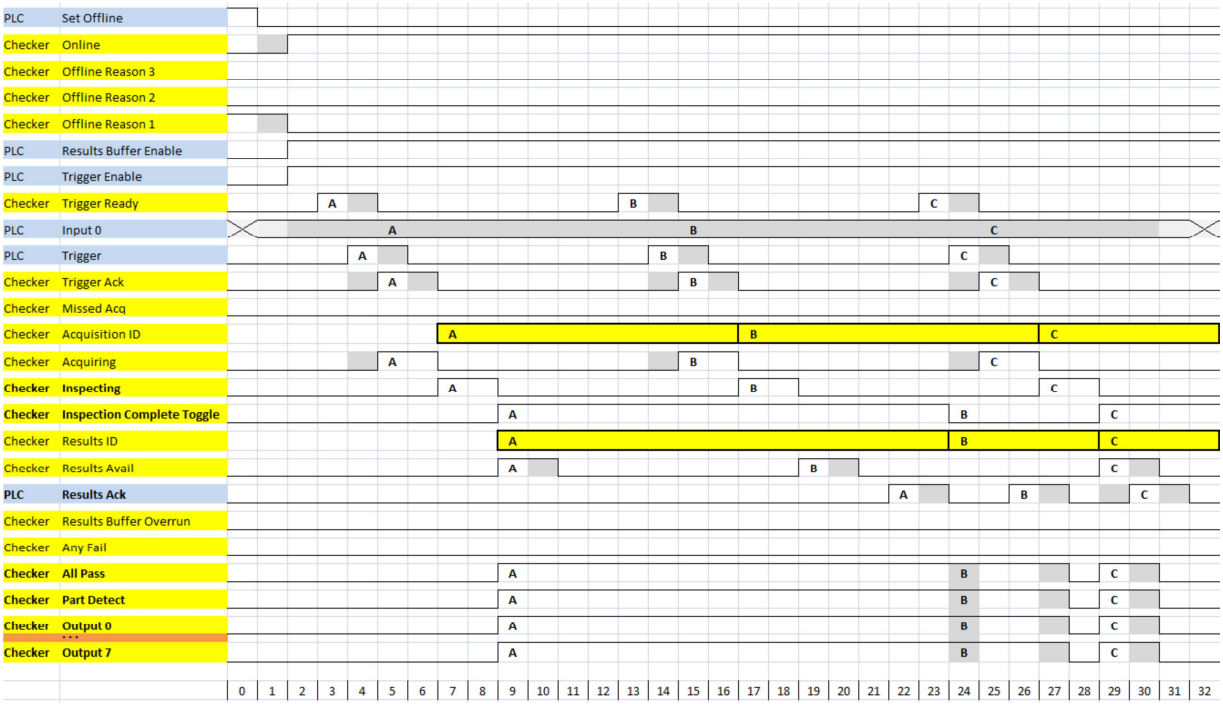
Results Buffering

There is an option to enable a queue for inspection results. If enabled, this allows a finite number of inspection result data to queue up until the client (PLC) has time to read them. This is useful to smooth out data flow if the client (PLC) slows down for short periods of time, or if there are surges of read activity.

Also, if result buffering is enabled, the device will allow overlapped acquisition and inspection operations. Depending on the application, this can be used to achieve faster overall trigger rates. See Acquisition Sequence description above for further detail.

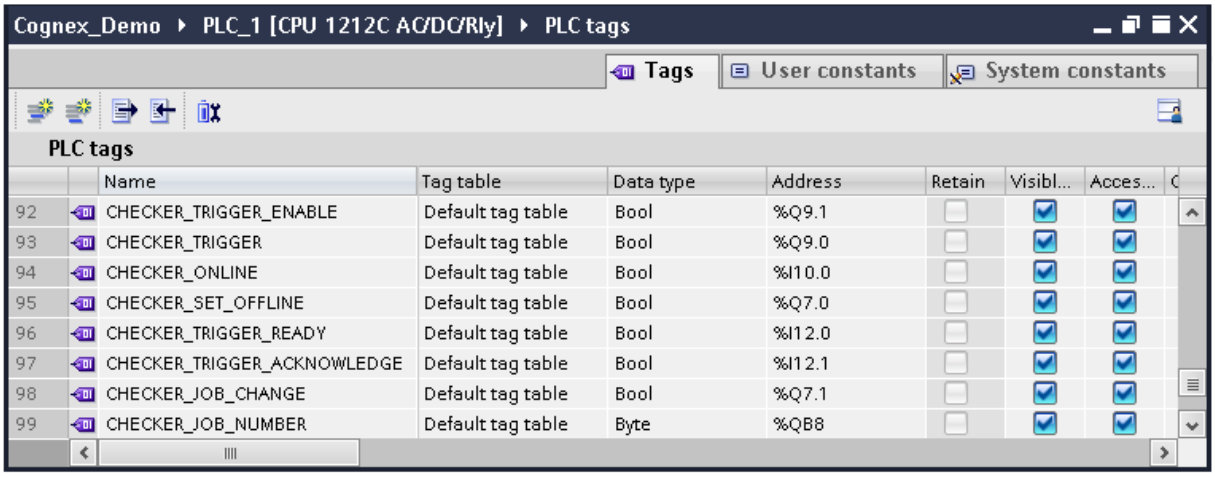
In general, if reads are occurring faster than results can be sent out, the primary difference between buffering or not buffering determines which results get discarded. If buffering is not enabled, the most recent results are kept and the earlier result (which was not read by the PLC fast enough) is lost. Essentially, the more recent result will simply over write the earlier result. If buffering is enabled (and the queue becomes full), the most recent results are discarded until room becomes available in the results queue.

As shown in the following figure, first the results corresponding to the 'A' acquisition go to the results buffer and to the output lines. This is followed by the results of 'B' going to the result buffer, but still the results of 'A' remain in the output lines. After the PLC read the results of 'A,' it disappears from both the buffer and the output lines.



Add PLC Tags

Under the Project Tree, a folder called 'PLC Tags' is present. Expand the folder and double-click 'Show All Tags.' Add the following tags:

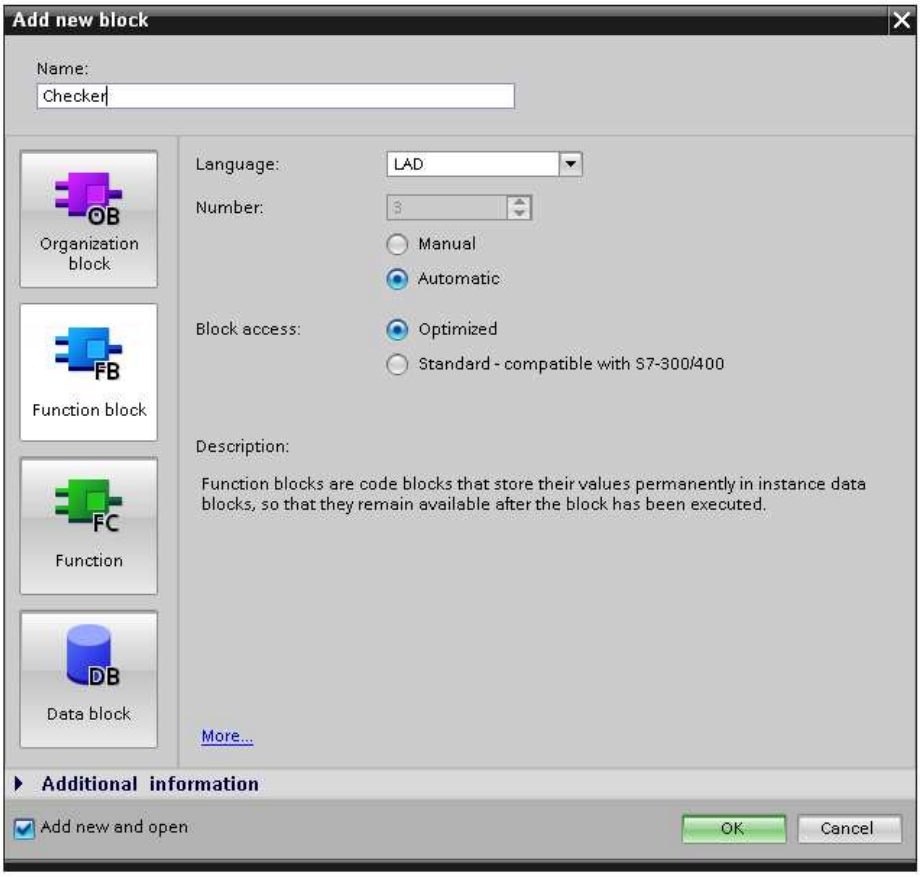


	Name	Tag table	Data type	Address	Retain	Visibl...	Acces...
92	CHECKER_TRIGGER_ENABLE	Default tag table	Bool	%Q9.1	☐	☑	☑
93	CHECKER_TRIGGER	Default tag table	Bool	%Q9.0	☐	☑	☑
94	CHECKER_ONLINE	Default tag table	Bool	%I10.0	☐	☑	☑
95	CHECKER_SET_OFFLINE	Default tag table	Bool	%Q7.0	☐	☑	☑
96	CHECKER_TRIGGER_READY	Default tag table	Bool	%I12.0	☐	☑	☑
97	CHECKER_TRIGGER_ACKNOWLEDGE	Default tag table	Bool	%I12.1	☐	☑	☑
98	CHECKER_JOB_CHANGE	Default tag table	Bool	%Q7.1	☐	☑	☑
99	CHECKER_JOB_NUMBER	Default tag table	Byte	%QB8	☐	☑	☑

These tags will be used for control of the Checker.

Add Checker Function Block

Under 'Program Blocks,' double-click 'Add New Block.' This will display a popup. Select 'Function Block' in the popup.



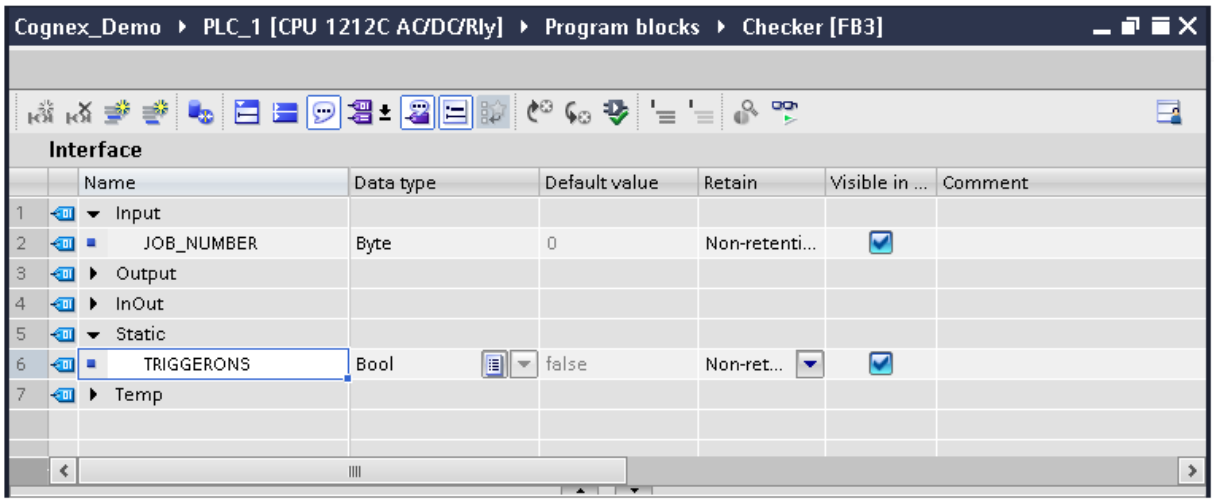
Call the Function Block Checker_FB and press 'OK.' This block will be FB3.

Now, the function block needs to be called. Double-Click on 'Main [OB1]' in the Project Tree. Add an empty box and type FB3 and press the 'Enter' key. This will display a popup to create a Data Block for the Function Block.

Press 'OK' and the function block call is complete.

Add Function Block Tags

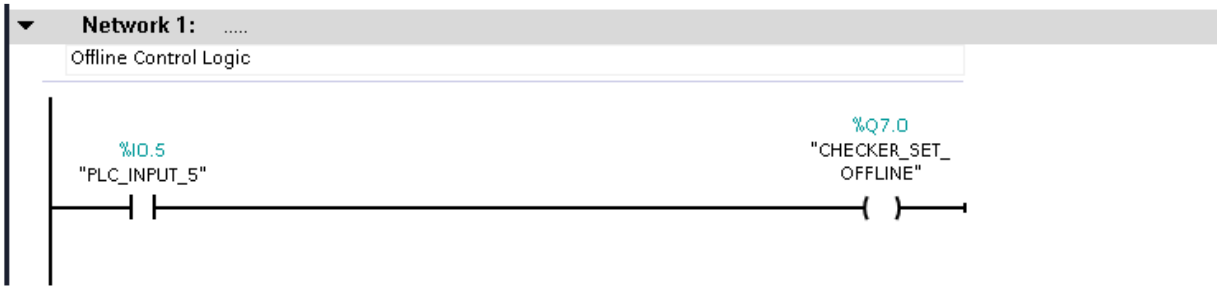
Double-click the 'Checker_FB[FB3]' function block. Add the following tags to the Function Block:



These tags will be used in the function block to send and receive data to/from the Checker.

Add Online/Offline Control

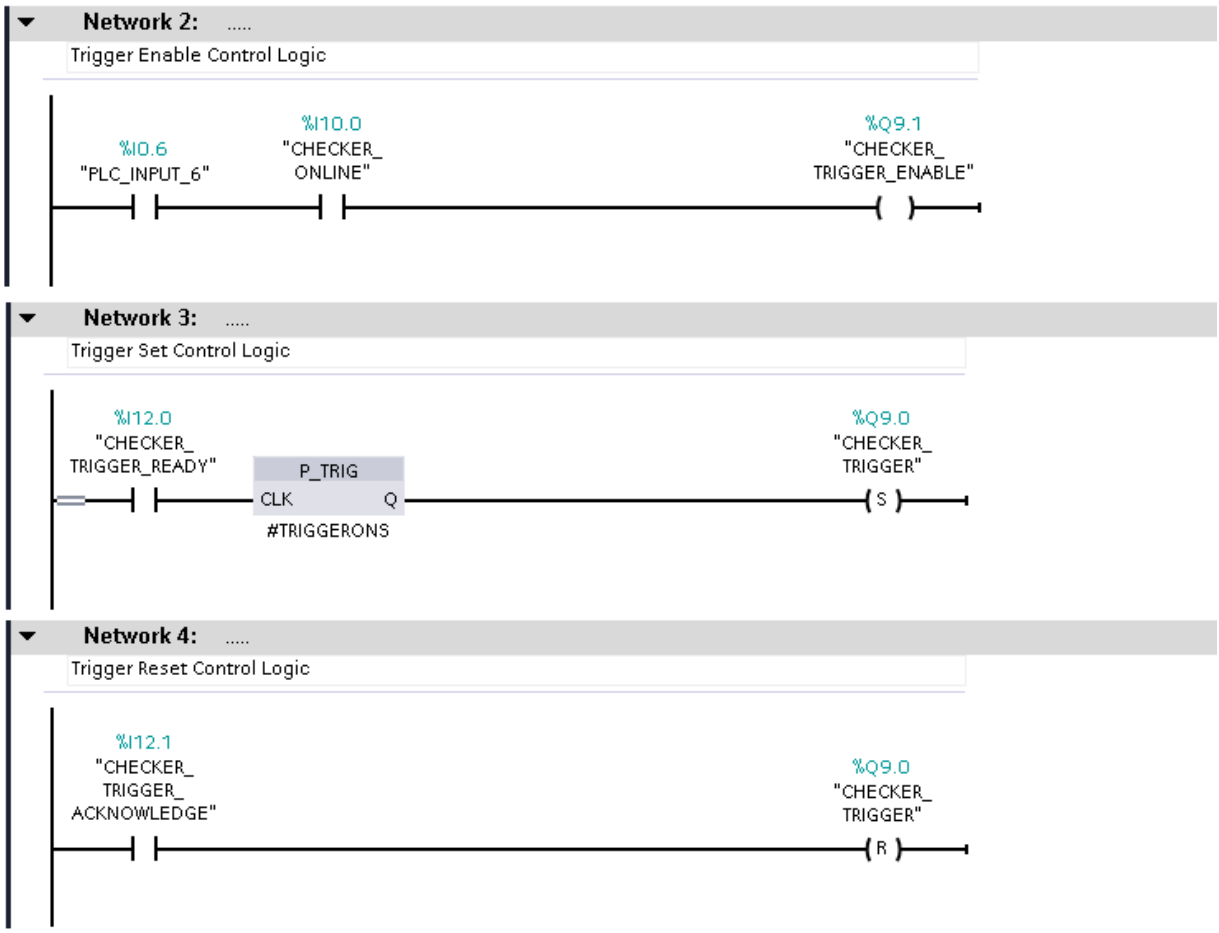
The following code shows how to add Online/Offline control to the PLC program:



The Online bit can be monitored, as well as the Offline Reason bits, to verify that the sensor is in fact, Offline. That logic is not shown.

Add Trigger

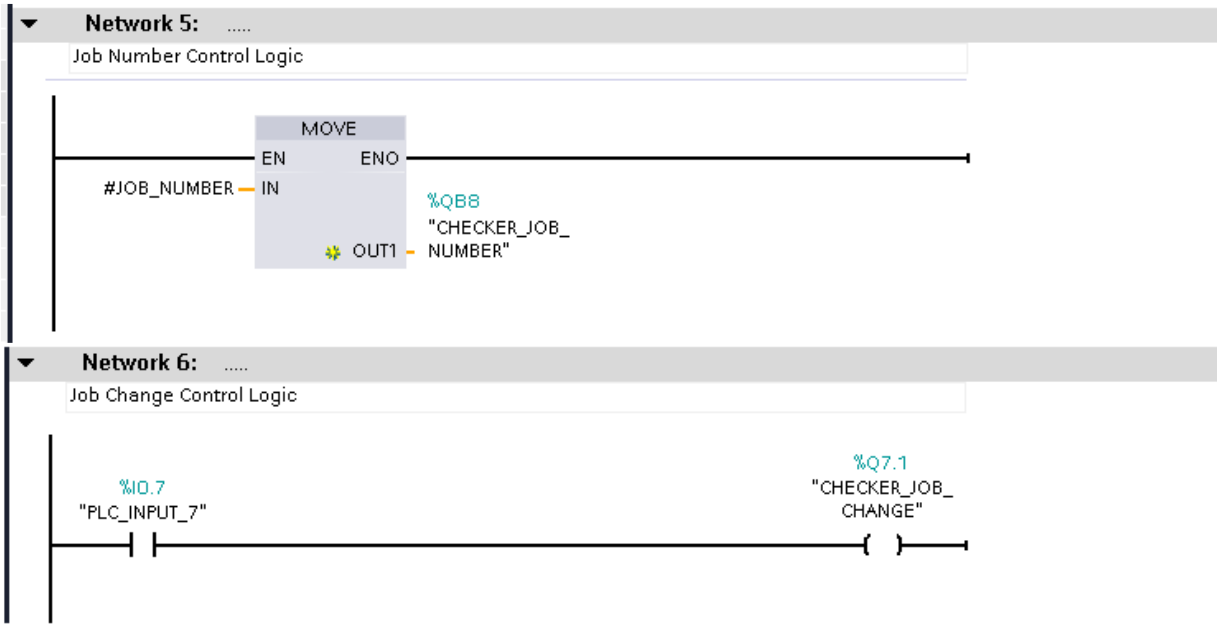
The following code shows how to add a network trigger to the PLC program:



Four tags, defined by the Checker device library, are used to control the trigger. In the first network (Network 2), the 'Trigger Enable' bit is set. When the PLC input I0.6 is high, the 'Trigger Enable' bit goes high. This allows a network trigger to be sent to the Checker. The rising edge of the 'Trigger Ready' bit is used to set the 'Trigger' output (Network 3). The 'Trigger Ready' signal is sent by the Checker when it receives the 'Trigger Enable' signal from the PLC. The 'Trigger' output is reset when the Checker sends a 'Trigger Acknowledge' signal back to the PLC (Network 4). To test the trigger to the Checker, toggle PLC input I0.6.

Add Job Change

There are two steps to changing a job in Checker: Loading the Job Number and the Job Change trigger.



The 'Checker Job Number' byte must contain the Job Number before the 'Job Change' bit is triggered. To change a job, toggle PLC input I0.7.