Robotics, Integration, and Automation

SMARTCART VISION PART 1

Name	Class/Period	Date

1. Overview

In the previous lab activity, you integrated the Cognex IS2000 vision sensor (camera) into the SmartCart 4.0 network and created jobs for inspection of each color of block.

In this lab activity, you will integrate the previous activity's camera jobs into a Logix Designer PLC project. The PLC logic will be used to trigger the vision sensor and inspect the workpieces. Before proceeding, ensure that all of the camera jobs that you created are saved *on the vision sensor*, and each file name begins with a unique number prefix.

2. Performance Objectives

After completing this lab activity, you will be able to:

- Install an add-on profile (AOP) for Cognex vision sensors.
- Create an I/O module for the Cognex IS2000 in a Logix Designer project.
- Build a PLC programming routine that runs a sequence of machine vision-related activities.

3. Required Materials

You need the following materials to complete the lab activity:

- SmartCart 4.0
- Computer
- Ethernet cables
- Workpieces (blocks)

4. Required Software

In-Sight Explorer and **Studio 5000 Logix Designer** are required for this lab activity.

5. Inventory and Safety

Before beginning the lab activity, review this checklist and mark off each item as you complete it.

- ☐ All hardware components are available for this lab activity.
- ☐ Hands, hair, and clothing are securely away from the work area.
- ☐ The work area is clean and devoid of food or drink.
- ☐ Review the SmartCart safety guidelines.
- ☐ Read through the entirety of this lab activity to familiarize yourself with the requirements.

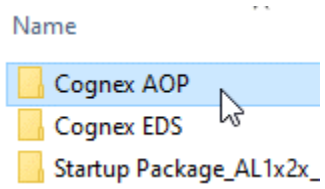
6. Lab Activity

6.1. Installing the AOP

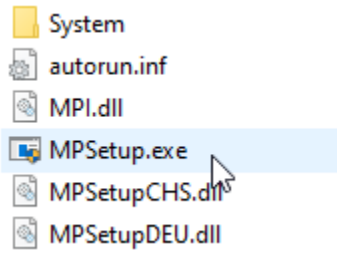
In this task, you will install Cognex add-on profiles (AOPs) for use in your Logix Designer projects.

Complete these steps:

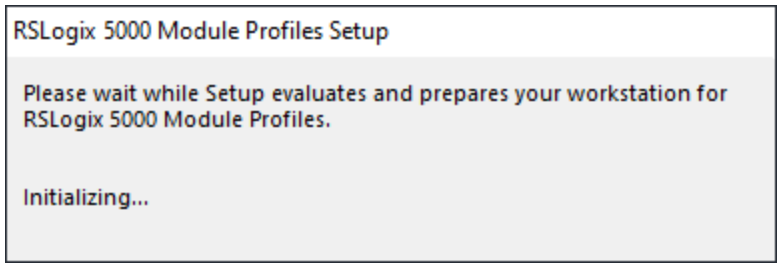
1. Ensure that all Studio 5000 applications are closed.
2. Navigate to the course page and download the Cognex AOPs resource from this section's Resources area. Once download is complete, extract the files.
3. Open the folder...



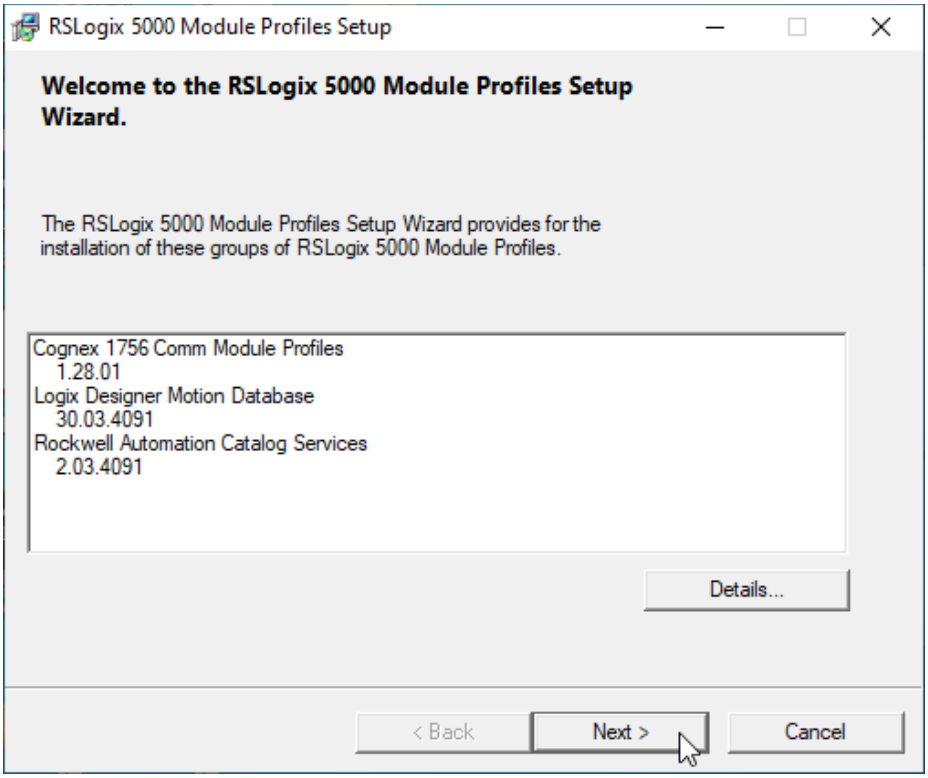
4. …and run **MPSetup.exe**.



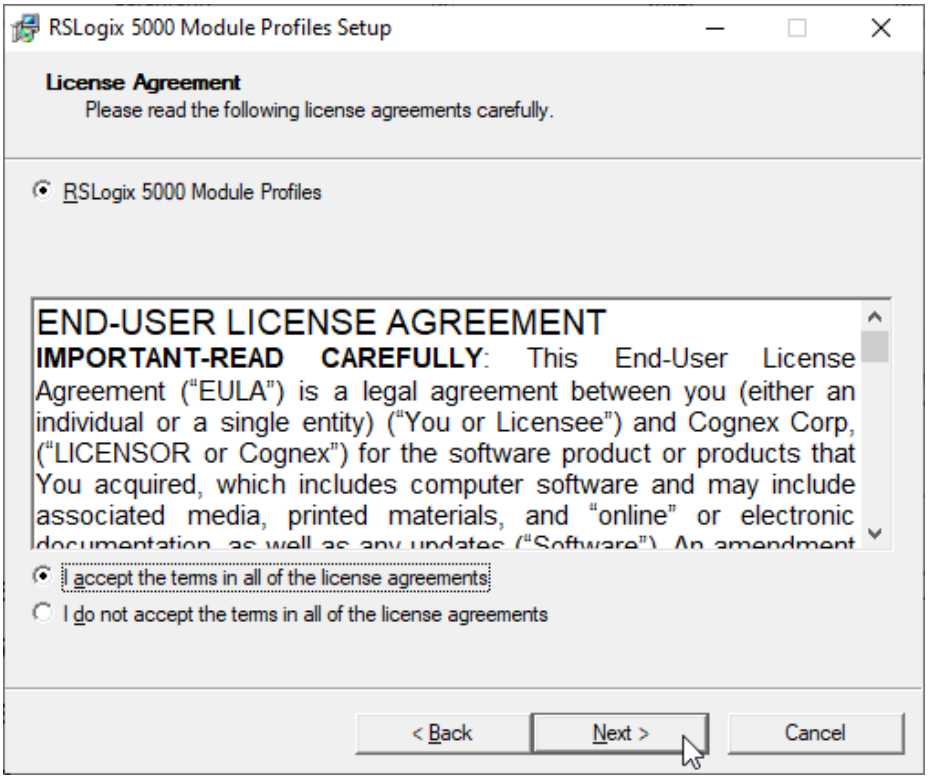
The Profile Setup application initializes.



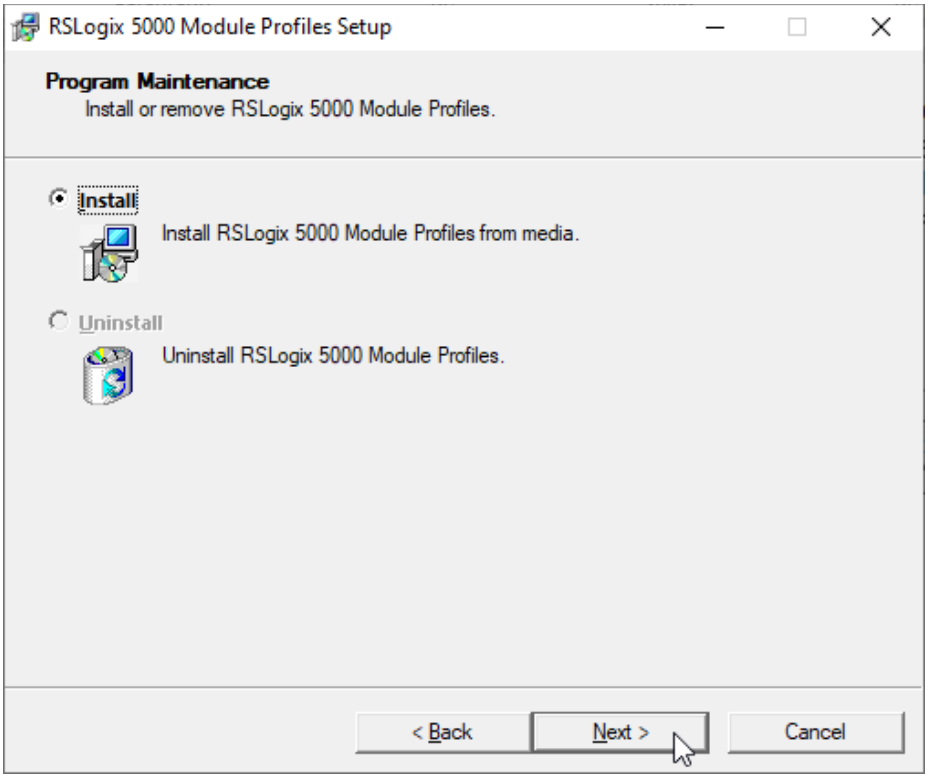
5. The setup wizard is displayed. Click **Next**.



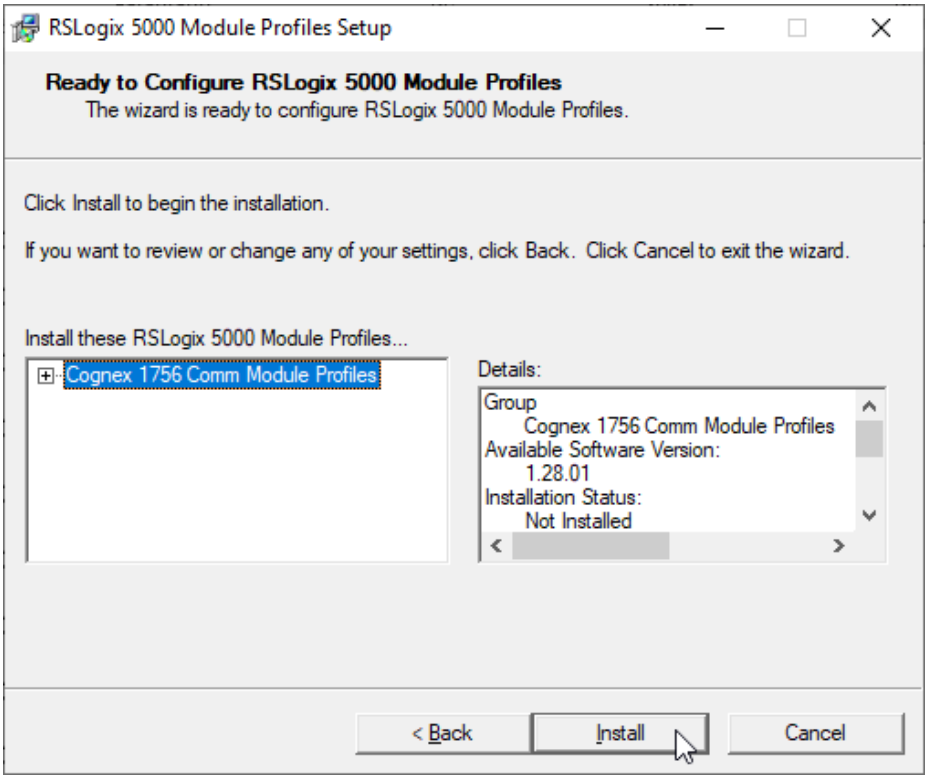
6. Accept the license agreement and then click **Next**.



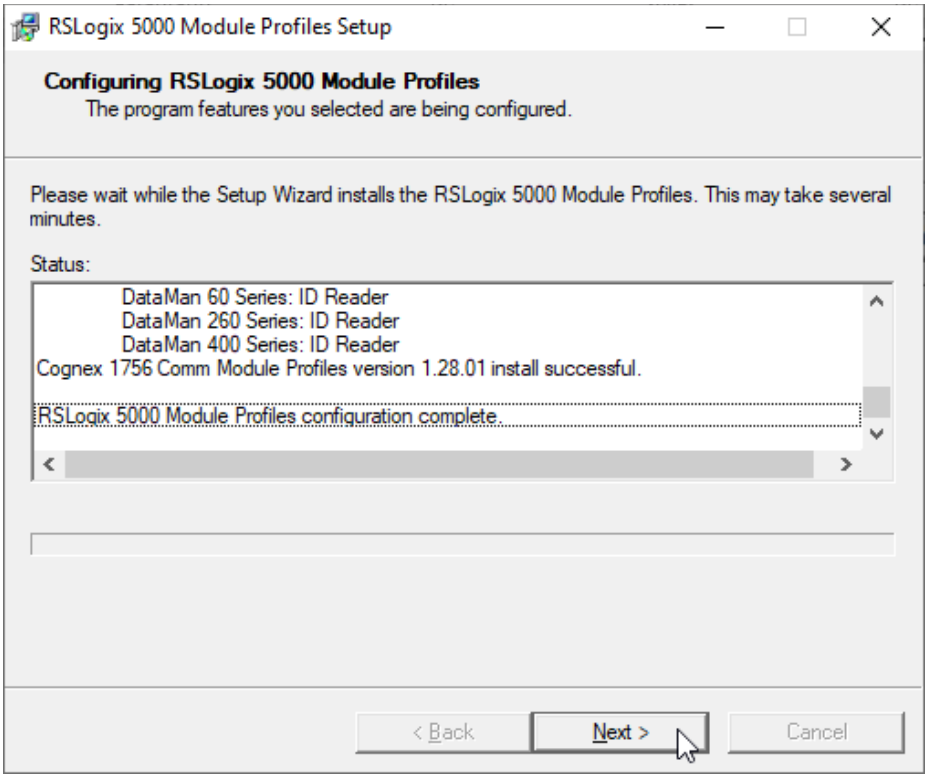
7. Select Install and then click **Next**.



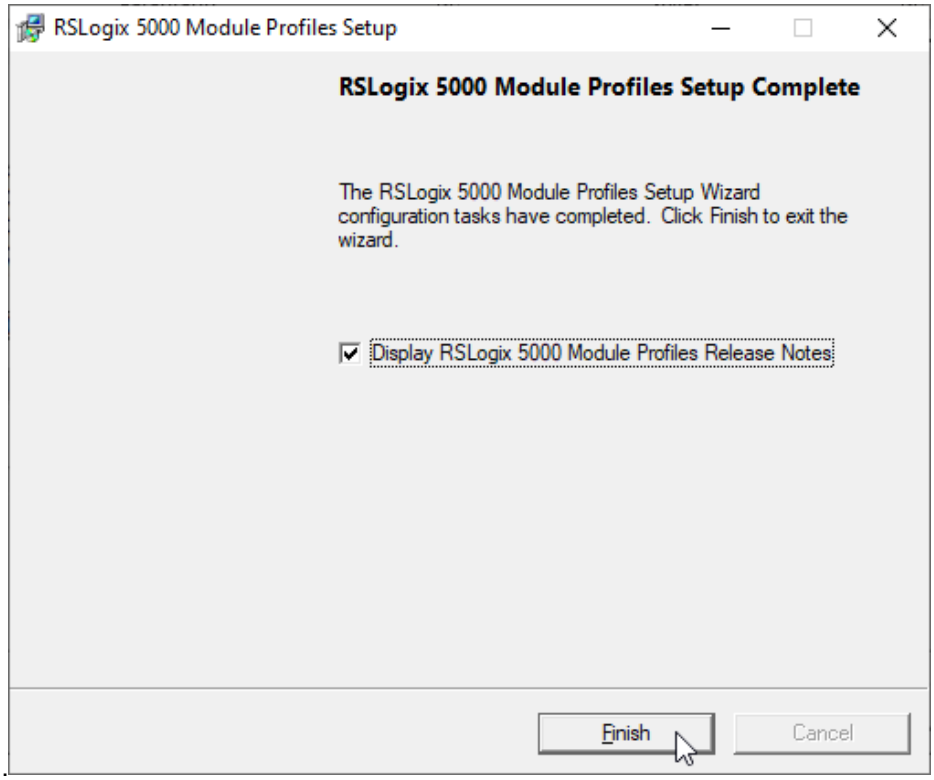
8. If desired, review the modules to be installed. Click **Install**.



9. The profiles are installed. Click **Next**.



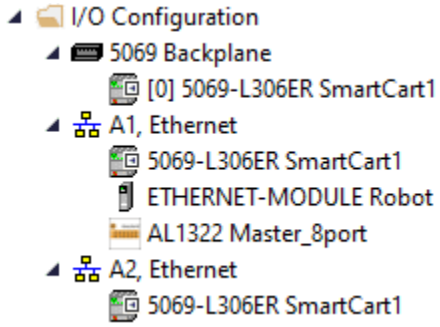
10. Click **Finish**.



6.2. Creating the Cognex Module in Logix Designer

In this task, you will add a Cognex IS2000 vision sensor module to a Logix Designer project.

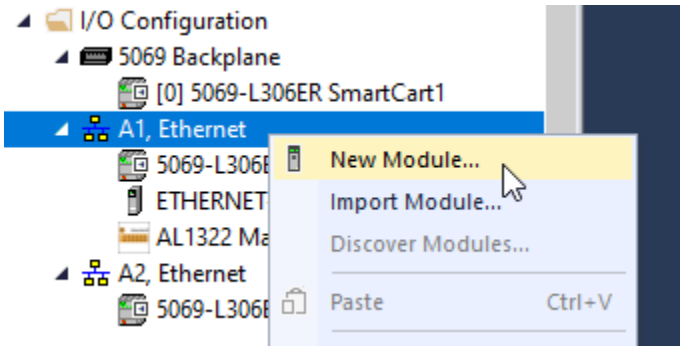
1. Power on the SmartCart's I/O box. The PLC and IS2000 vision sensor (camera) turn on.
2. Using the Windows Command Prompt, ping your IS2000 camera (IP address **192.168.0.7**) to confirm network connectivity.
3. Run **Studio 5000 Logix Designer** and open the project that you worked with in the previous section. The project should already include the PLC, the robot (ETHERNET-MODULE), and at least one IO-Link master.



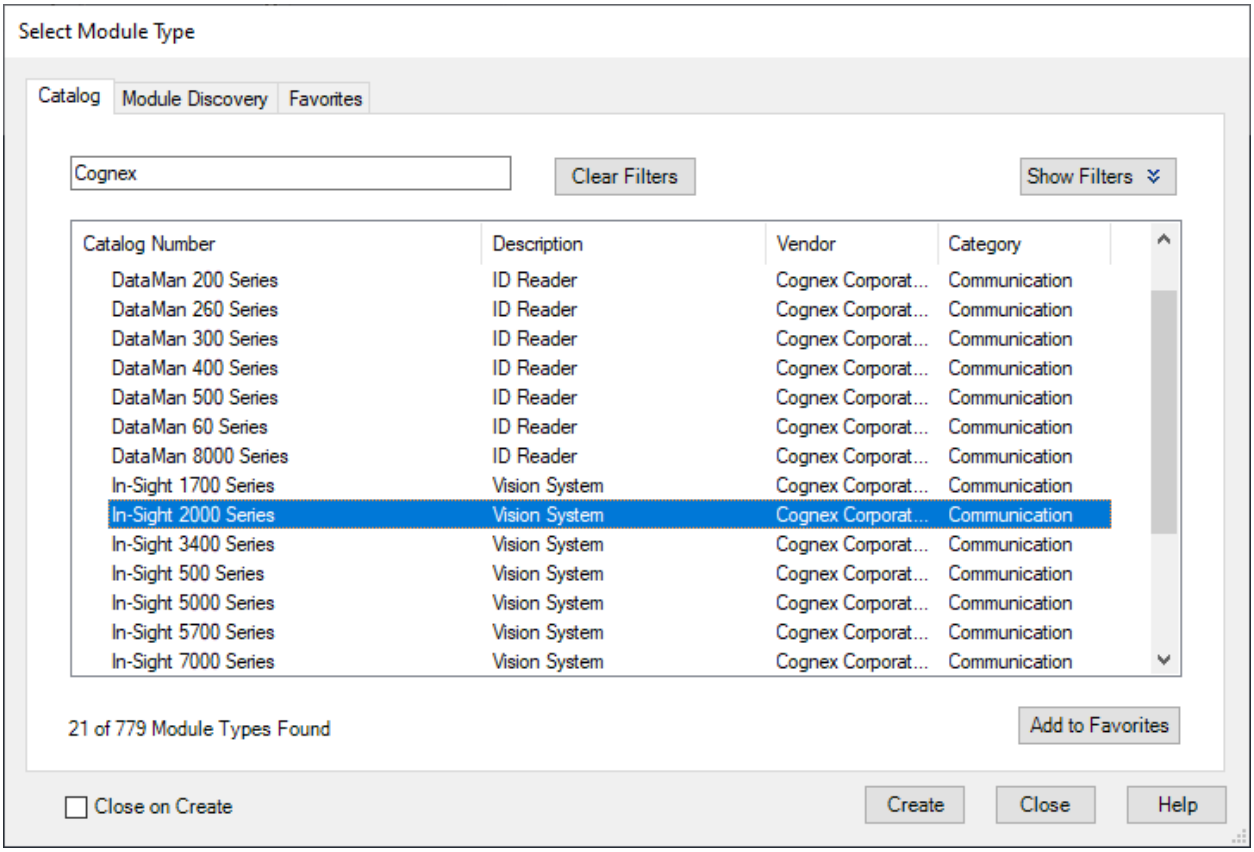
4. Save the project with a new name, such as **Machine_Vision_Test** or similar.
5. Download the project to the PLC and stay online in Rem Run mode.

   If the robot is off, there will be an I/O configuration warning. The robot is not required in this lab activity, and you can leave it off.

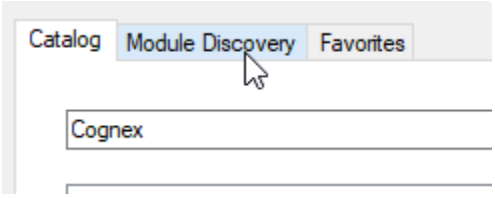
6. In the Controller Organizer, right-click the **A1, Ethernet** port and then select **New Module**.



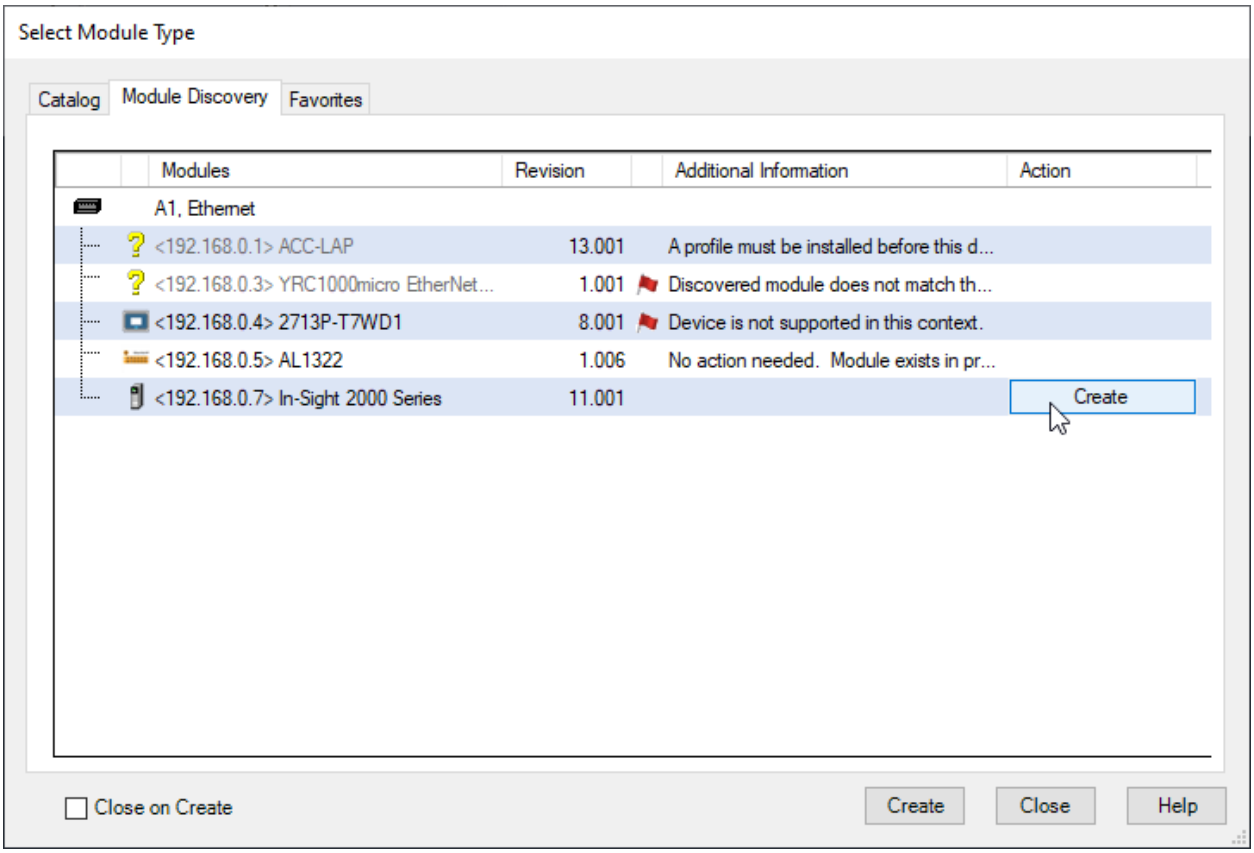
The Select Module Type window is displayed. Note that if you search *Cognex* in the Catalog tab's search field, the In-Sight devices whose AOPs you installed are displayed. Without the installation of the AOPs, these modules are neither available nor visible.



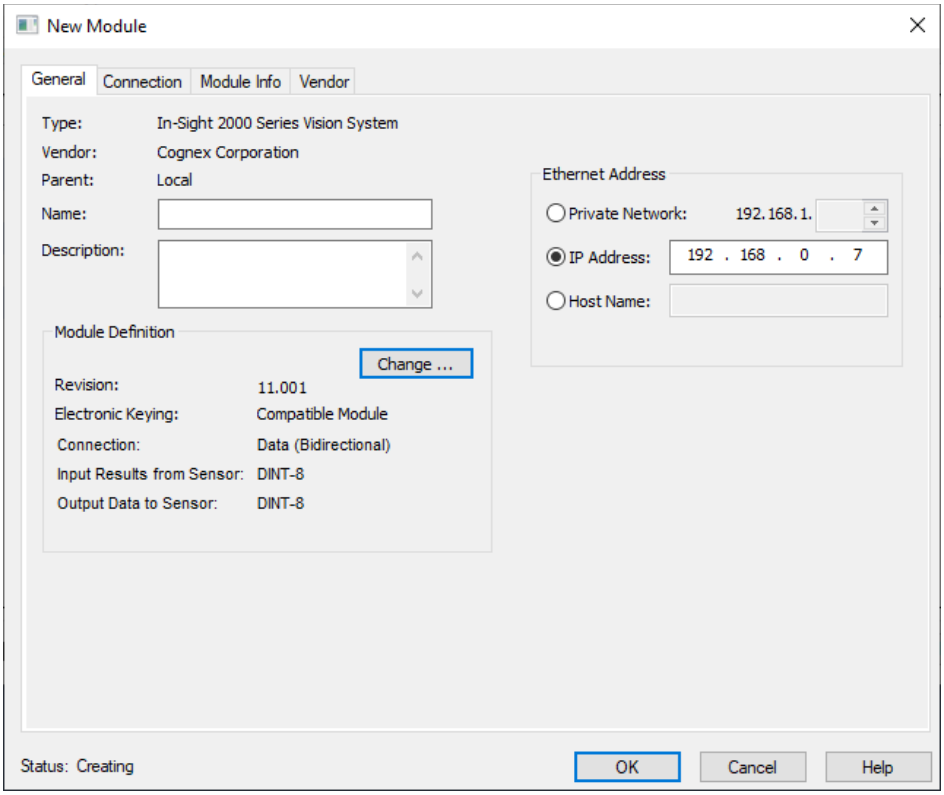
7. Select the **Module Discovery** tab.



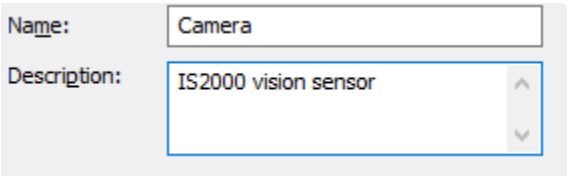
8. Wait for the modules to be displayed in the tab. If your camera is on the network, it should appear in the list. Click the **Create** button next to the In-Sight 2000 Series module.



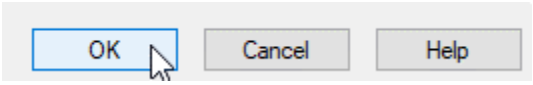
The New Module window is displayed. The IP address is entered automatically.



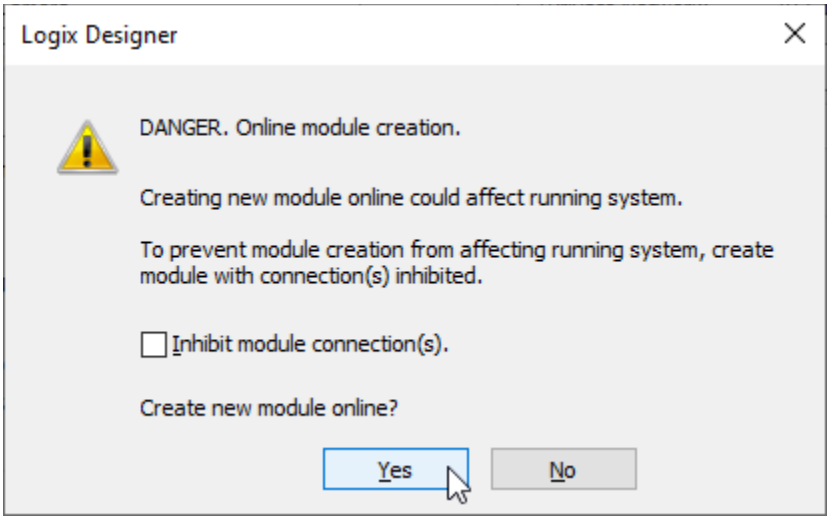
9. Add a name and description…



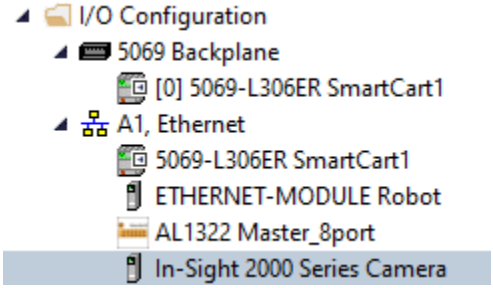
10. …and then click **OK**.



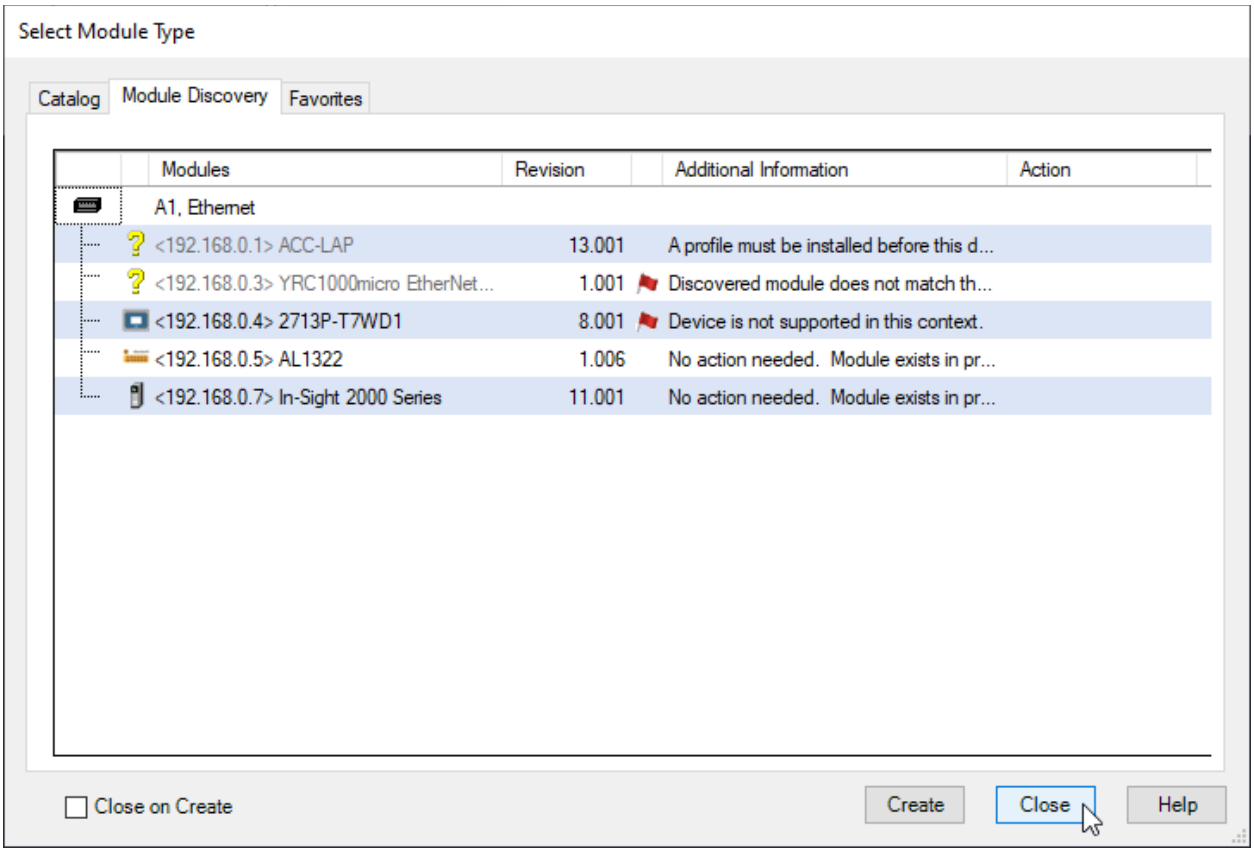
11. Accept the warning.



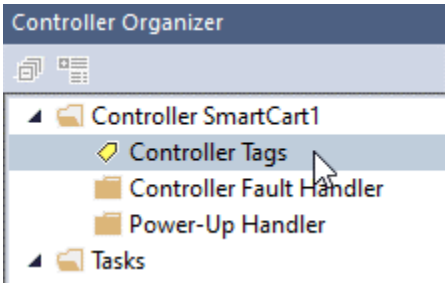
The camera is added to the Controller Organizer's I/O Configuration folder.



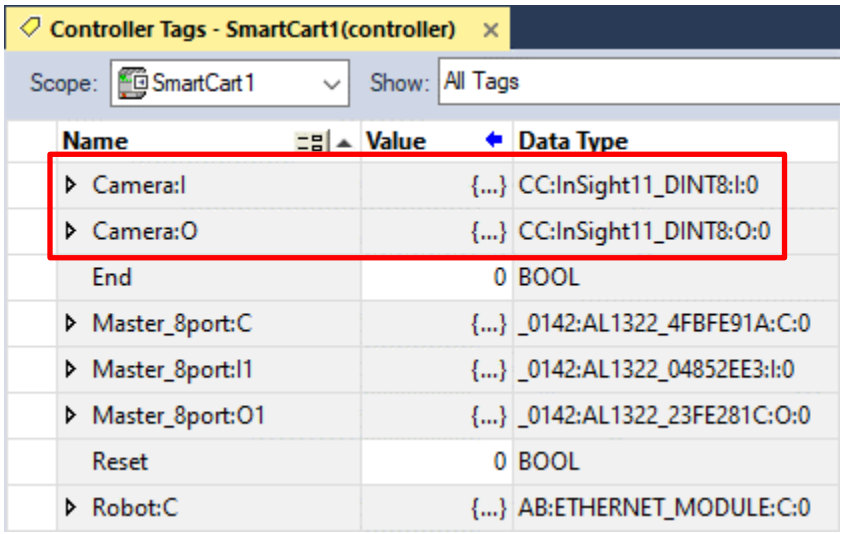
12. Close the Select Module Type window.



13. Double-click **Controller Tags**.



14. Note the addition of the new camera module tags. Take a minute to expand these tags and explore their respective tag members.



ⓘ ***Note:*** *If you named your module differently, the names of the tags will have a different prefix.*

15. Go offline with the PLC.

6.3. Building a Vision Routine

In the previous task, you added the camera module to your PLC project. The module includes a variety of input and output tags. *The tables below contain a list of the most relevant ones.* In this task, you will create a logic routine to control the actions of the camera and inspect the colored blocks.

Input Tags (Status Category)

Tag Name	Data Type	Description
Camera:I.Status.Online	BOOL	Indicates if the camera is online or offline. Being online means that the camera can send/receive data to/from the job over the network, which is something that is required in an application such as yours. The camera cannot switch jobs if it is online.
Camera:I.Status.TriggerReady	BOOL	Indicates if the camera is ready to be triggered.
Camera:I.Status.CurrentJobID	INT	Indicates the active job number.
Camera:I.Status.CommandCompleted	BOOL	Indicates that a job change has been accepted by the camera.
Camera:I.Status.ResultsValid	BOOL	Indicates that there is a job result.
Camera:I.Status.JobPass	BOOL	Indicates if the job has registered a pass or fail.

Output Tags (Control Category)

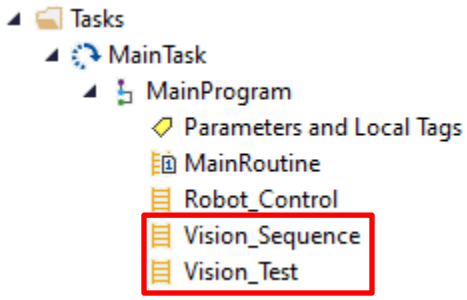
Tag Name	Data Type	Description
Camera:O.Control.SetOffline	BOOL	Switches the status of the camera to offline, which is a requirement for job switching. When this tag is turned on, Camera:I.Status.Online turns off.
Camera:O.Control.TriggerEnable	BOOL	Readies the trigger for image acquisition. When this tag is turned on, Camera:I.Status.TriggerReady turns on.
Camera:O.Control.Trigger	BOOL	Triggers an image acquisition and executes the camera job.
Camera:O.Control.Command	INT	Transfers the indicated job to the camera.
Camera:O.Control.ExecuteCommand	BOOL	Switches the active camera job to the job that was transferred by Camera:O.Control.Command.

Perform the following steps:

1. Your program will require some other tags in addition to the camera module tags. Create these tags in the Edit Tags table:

Tag Name	Data Type	Description of Use
Vision_Step	DINT	Keeps track of the vision sequence.
Vision_Command	BOOL	Starts the vision sequence.
Vision_Sequence_Complete	BOOL	Indicates if the vision sequence has been completed.
Camera_Timer	TIMER	Creates a short delay in the vision sequence.
Job_Number	DINT	For selection of the desired job.
Yellow_Block_Present	BOOL	Used as a manual trigger to test the inspection job for the yellow block.
Red_Block_Present	BOOL	Used as a manual trigger to test the inspection job for the red block.
Green_Block_Present	BOOL	Used as a manual trigger to test the inspection job for the green block.
Blue_Block_Present	BOOL	Used as a manual trigger to test the inspection job for the blue block.
ONS1	BOOL	Used for one shots.
ONS2		
ONS3		
ONS4		

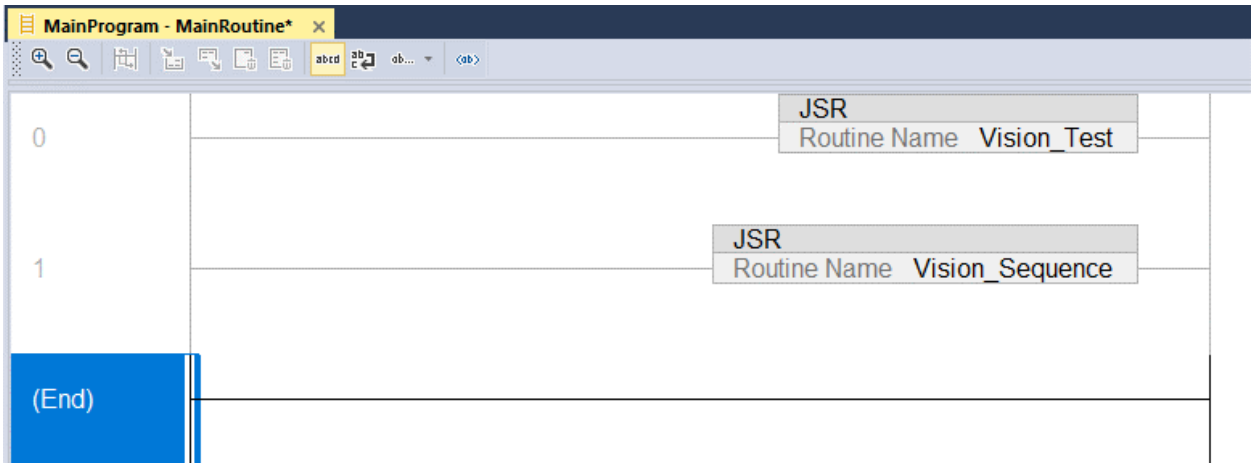
2. Add two new ladder routines to the main program. Name them **Vision_Sequence** and **Vision_Test**.



Vision_Sequence will include the series of events to enable job loading and to trigger the camera, and it is a routine that you will be able to reuse in future projects with slight modifications. Vision_Test will have instructions that will allow you to externally trigger the camera and will contain the actions that the SmartCart will take based on the inspection results of the job.

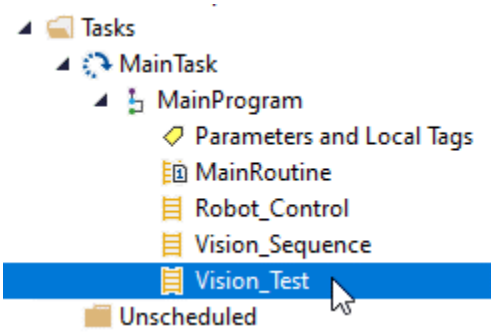
MainRoutine

3. Open MainRoutine.
4. Modify the rungs to only include JSR instructions that jump to the Vision_Sequence and Vision_Test subroutines. The main routine will ignore any other routines that you may have in your program such as the Robot_Control routine.



The Vision_Test Routine

5. Double-click the **Vision_Test** routine to open its ladder editor.

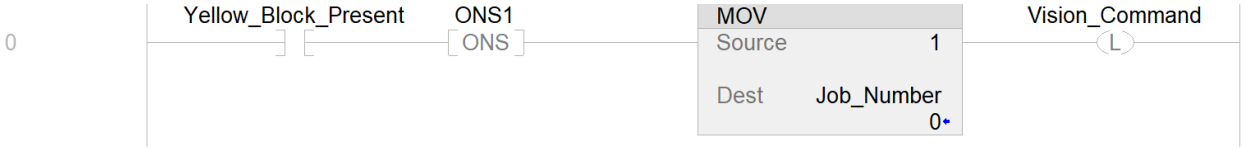


This routine will be used to trigger the camera's job switching sequence and perform actions based on the job results:

- If the result is a pass, the signal lamp should turn on with a green light.
- If the result is a fail, the lamp should illuminate a red light.

The routine will also allow for selecting the job to be loaded to the camera.

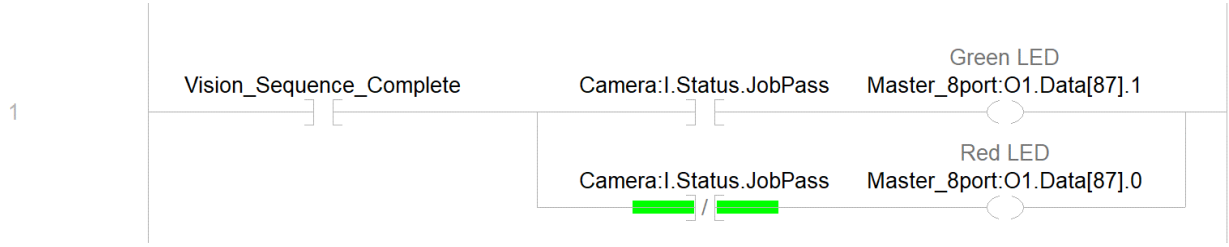
6. For rung 0, add instructions that, when the yellow block is present under the camera, will select job number 1 and initiate the vision sequence (which will be programmed in the Vision_Sequence routine).



Note the following about rung 0:

- Job 1 corresponds to the 1_Yellow_block camera job that we programmed in the previous lab activity. If your job for inspecting the yellow block has a different number, use that number as the Source in the MOV instruction.
- The XIC addressed to Yellow_Block_Present will be toggled manually. The ONS is added so that vision sequence will occur once per block.
- The Vision_Command tag is latched. It will be unlatched in the Vision_Sequence routine.

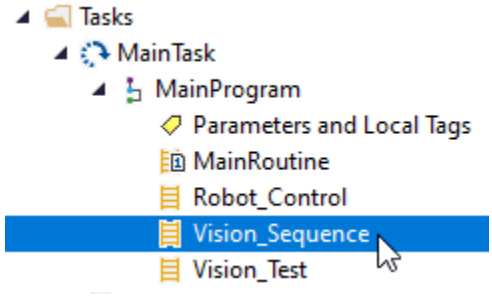
7. For rung 1, program the action of the job result after completion of the vision sequence. If the workpiece passes inspection, turn on the green LED. If the workpiece fails inspection, turn on the red LED.



8. Document the rungs of the routine.
9. Save the project.

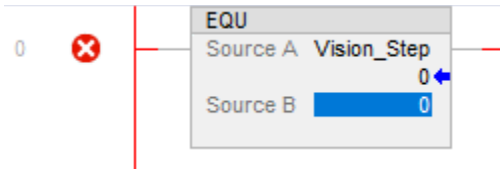
The Vision Sequence Routine

10. Double-click the **Vision_Sequence** routine to open its ladder editor.



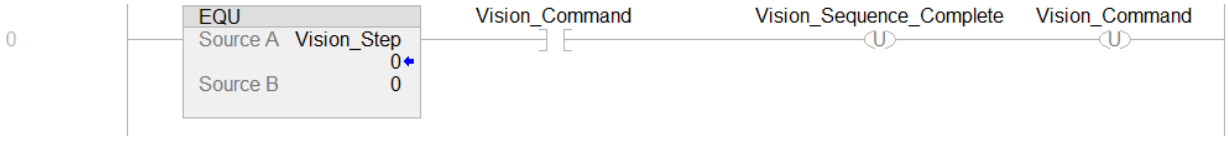
This routine will have a sequencer using EQU and MOV instructions, much like the sequencer you created for robot control. Each rung will correspond to a step in the sequence, and when the rung has completed execution, the next step will commence.

11. The first steps of the vision sequence are used to start the sequence and load the desired camera job. In rung 0, add an **EQU** instruction comparing **Vision_Step** to **0**. This is step 0 of the sequence.

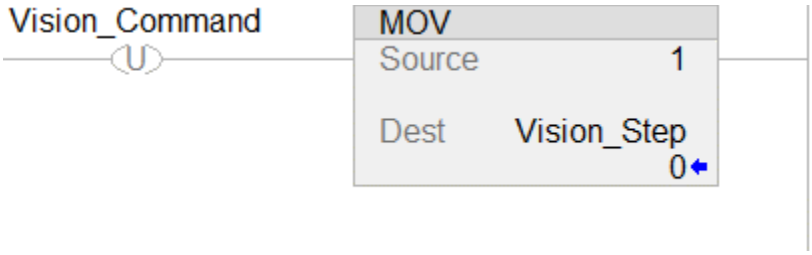


12. Insert an **XIC** instruction and address it to **Vision_Command**. This XIC will start the sequence and will be turned on in the Vision_Test routine (using an OTL instruction).

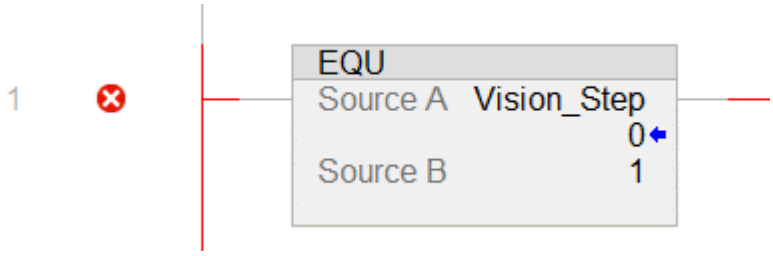
13. Add two **OTU** instructions. Address the first to the **Vision_Sequence_Complete** tag (which will be latched at the final step of the sequence) and address the second OTU to **Vision_Command**. This second OTU essentially converts the XIC into a one shot.



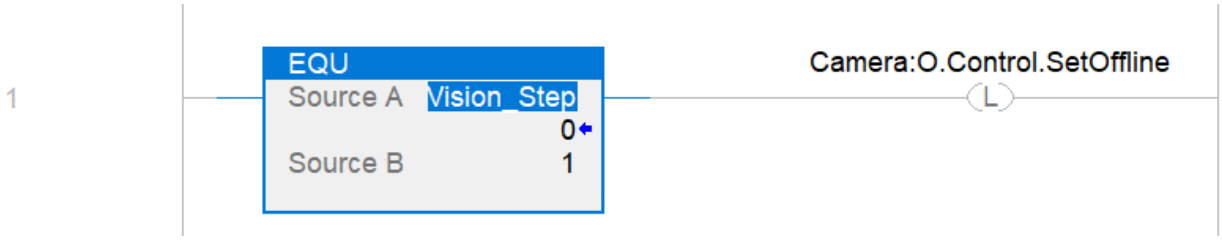
14. Complete rung 0 with a **MOV** command to advance the vision sequence to the next step, step **1**.



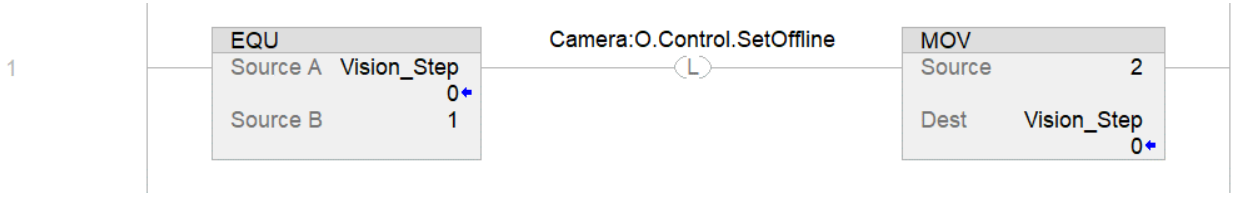
15. Start rung 1 with an **EQU** instruction to confirm the step number.



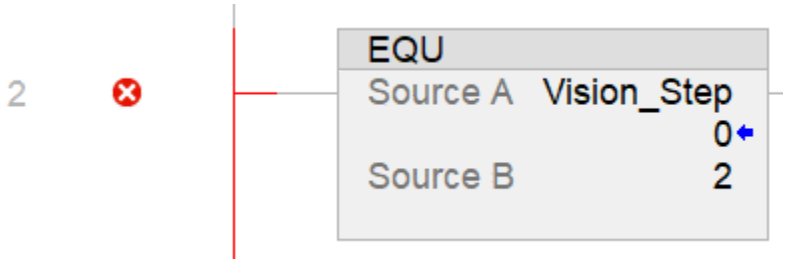
16. The camera must be switched offline in order for it to load a job. Add an OTL instruction addressed to **Camera:O.Control.SetOffline**.



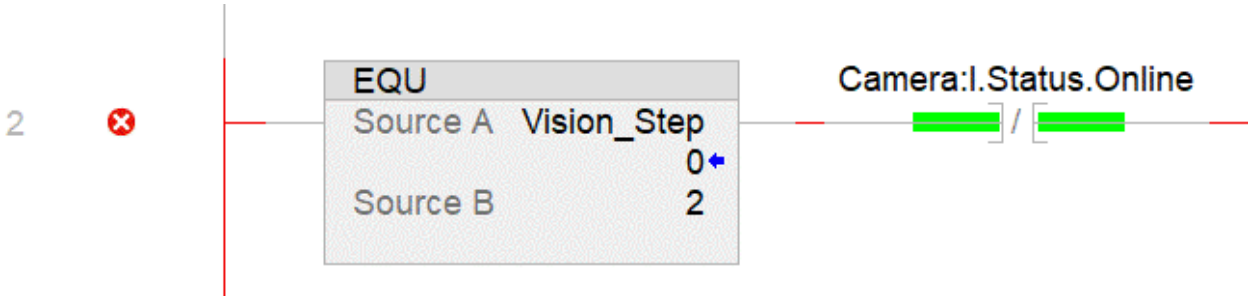
17. Complete the rung with a MOV instruction to step 2.



18. Start rung 2 (and step 2) with an **EQU** instruction comparing **Vision_Step** to **2**.

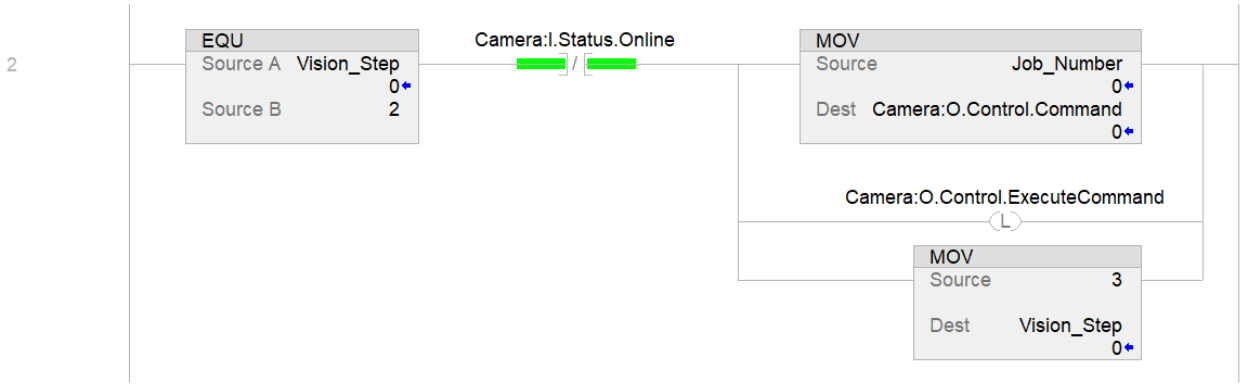


19. In step 2, once the camera's offline status is confirmed, we want to transfer the desired job number to the camera. Add an **XIO** addressed to **Camera:I.Status.Online**. This confirms that the camera is offline.

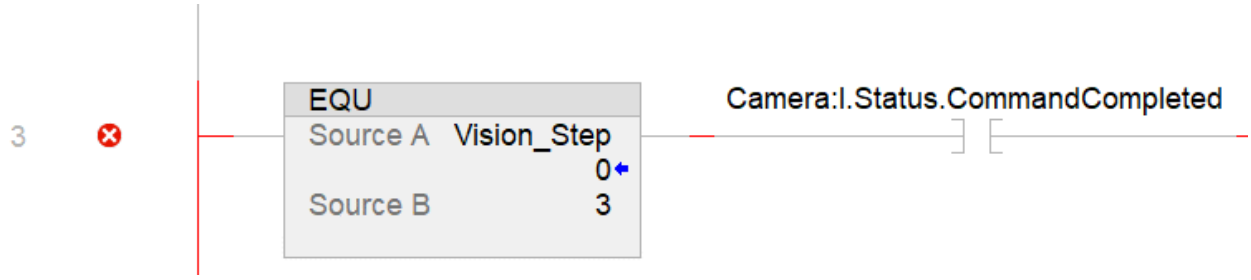


20. On the right side of the rung, add the following to a separate branch:

- A MOV instruction to transfer the **Job_Number** (which is selected in the Vision_Test routine) to **Camera:O.Control.Command**. This tag transfers the job to the camera.
- An **OTL** addressed to **Camera:O.Control.ExecuteCommand**. This changes the camera job to the one held by Camera:O.Control.Command.
- A **MOV** instruction to go to the next step (**3**).

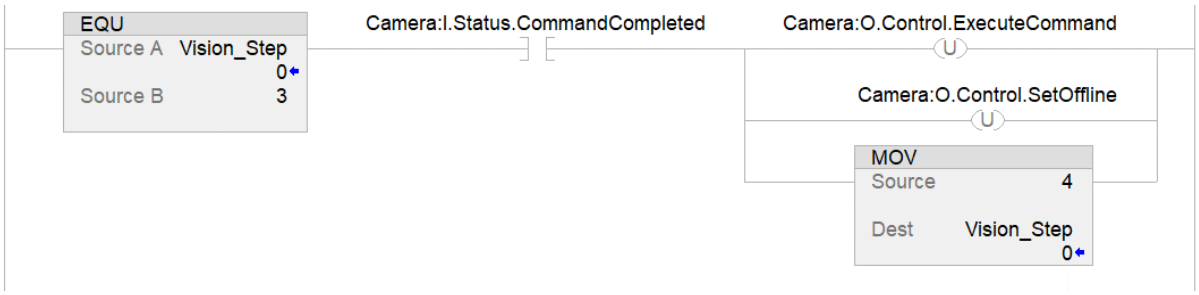


21. Start rung 3 with an **EQU** instruction confirming the start of the step and an **XIC** instruction addressed to **Camera:I.Status.CommandCompleted** to confirm the job switch/load. This tag turns on when the job switching/loading task is complete.

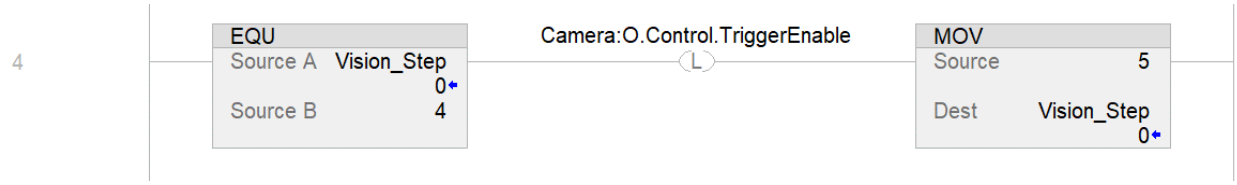


22. Again, add three instructions in parallel branches at the end of the rung:

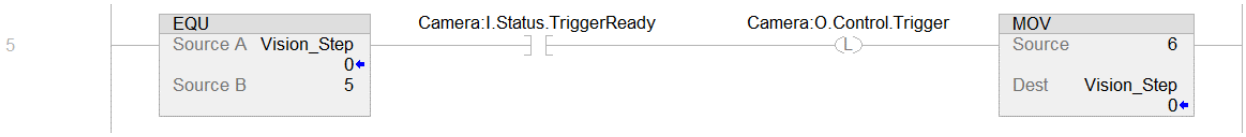
- An OTU addressed to **Camera:O.Control.ExecuteCommand** which ends the job load/switch.
- An OTU addressed to **Camera:O.Control.SetOffline** which turns the camera back online (so that it can run the job and inspect parts).
- A **MOV** instruction to go to the next step (**4**).



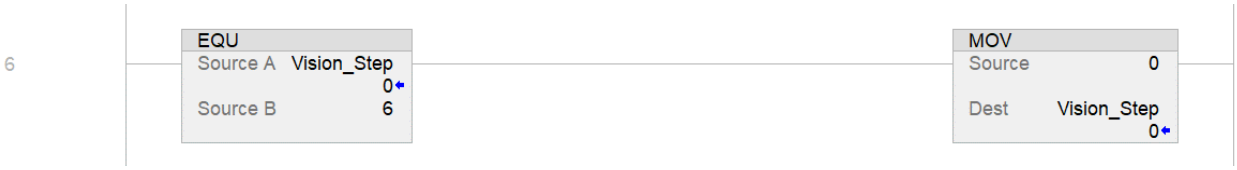
23. Step 4 is where the real camera action begins. Before the camera is triggered, though, its trigger must be enabled. Aside from the expected **EQU** and **MOV** instructions, add an **OTL** addressed to **Camera:O.Control.TriggerEnable**.



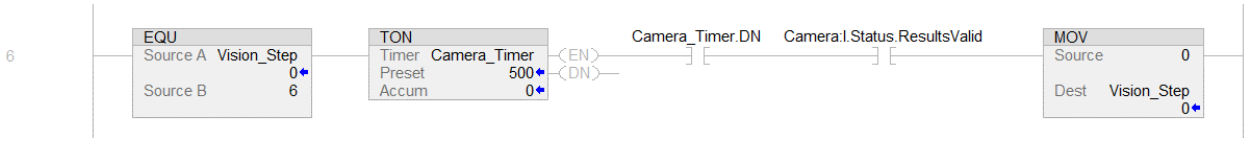
24. In step 5, besides the **EQU** and **MOV** instructions, add an **XIC** addressed to **Camera:I.Status.TriggerReady**, which confirms the trigger is ready, and an **OTL** addressed to **Camera:O.Control.Trigger**, which actually triggers the camera. When this OTL executes, you will see the camera flash and acquire an image.



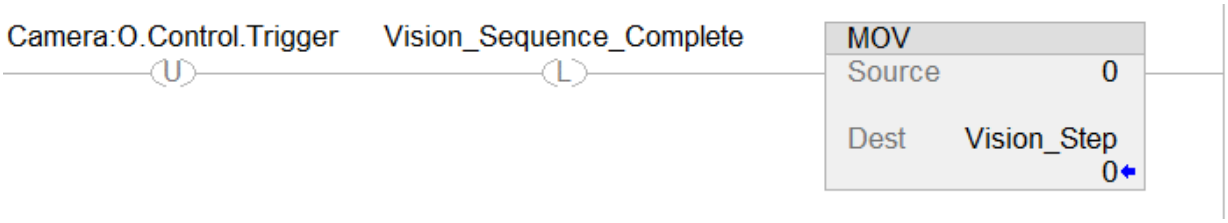
25. Step 6 will be the last step in the sequence and will be used to confirm inspection results, deactivate the trigger, and end the sequence. Add the relevant **EQU** and **MOV** instructions.



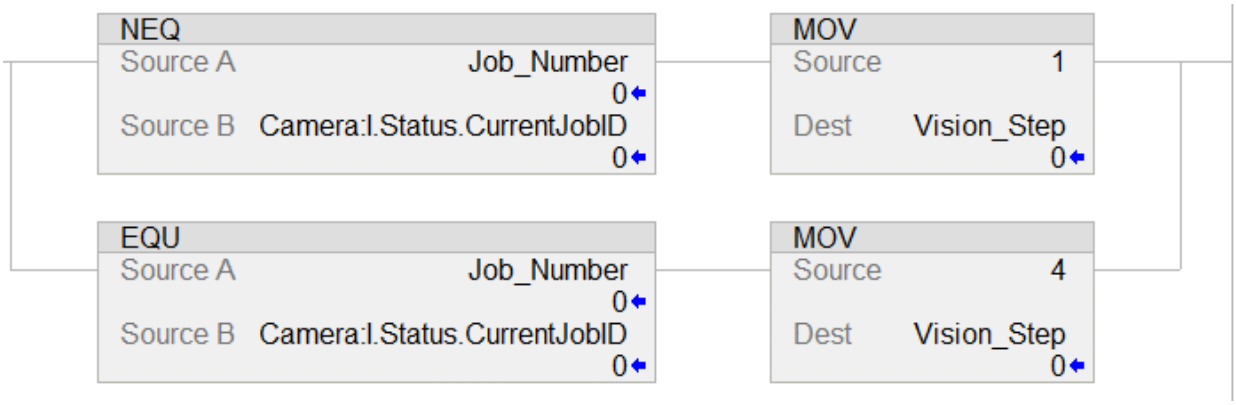
26. Between the EQU and MOV instructions, add a **half-second delay** and then an XIC addressed to **Camera:I.Status.ResultsValid**. This gives the camera a short delay in order to process the results of the inspection. This delay can be shortened if needed.



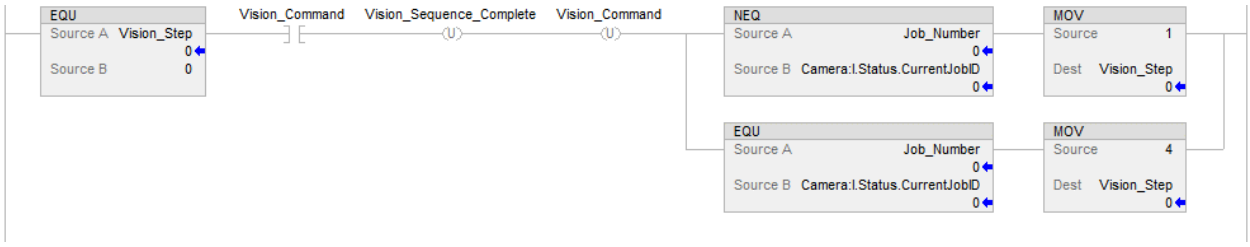
27. Complete rung 6 by unlatching **Camera:O.Control.Trigger** and latching the **Vision_Sequence_Complete** tag.



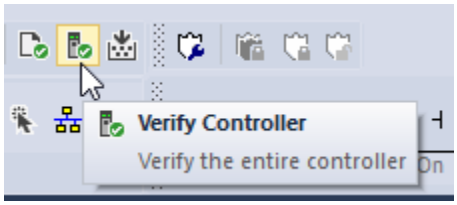
28. The sequence is now complete. However, rungs 1-3 are unnecessary if the correct job is already loaded into the camera. **Return to rung 0** and adjust it so that the sequence skips to rung 4 if the desired job is already loaded. This requires NEQ and EQU instructions that compare the desired job number to **Camera:I.Status.CurrentJobID**.



The complete rung 0 should look like this:



29. Document the rungs of the routine.
30. Verify the controller. If necessary, correct any errors. Errors related to any unused/unreferenced routines can be ignored.



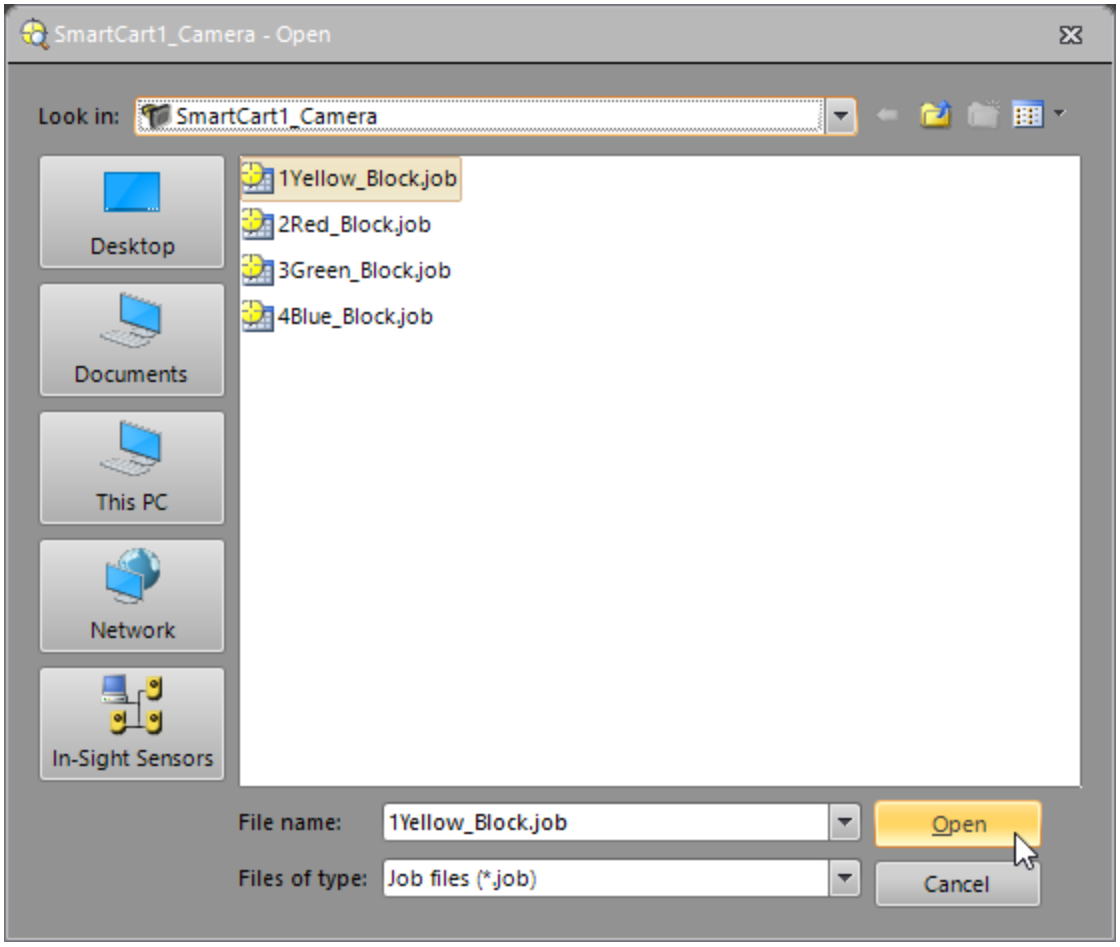
31. Save your project.
32. Download the project to the PLC. Stay online in Rem Run mode.

6.4. Readying the Camera

In this task, you will deploy the vision sensor (camera) by opening one of the jobs for block inspection, adjusting communication parameters, and switching the camera to online status.

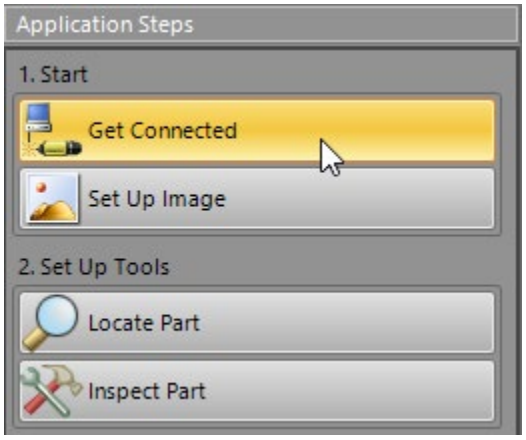
Perform these steps:

1. Remove any workpieces from underneath the camera.
2. Run In-Sight Explorer.
3. Navigate to **File > Open**. Browse to and open the job that you created in the previous activity for inspection of the yellow block.

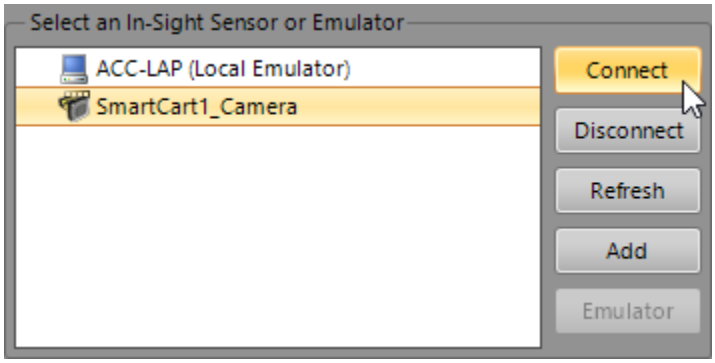


4. Ensure that you are in EasyBuilder view. If you are already connected to the vision sensor (camera), skip to step 7.

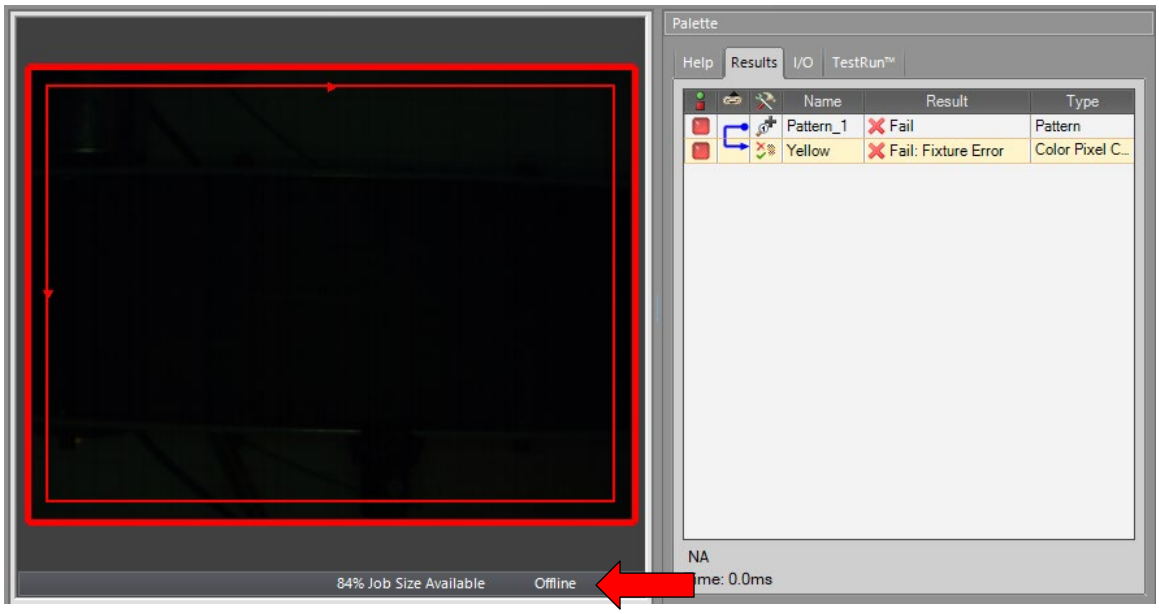
5. Click the **Get Connected** application step.



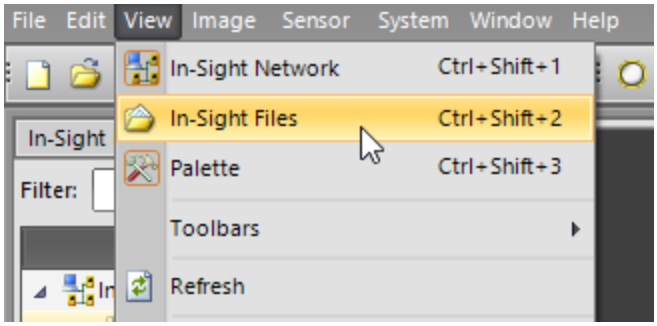
6. Select your vision sensor and then click **Connect**.



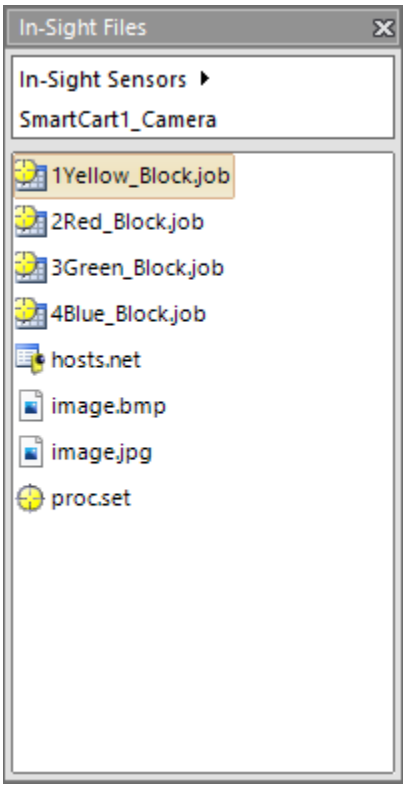
The camera is connected. Inspection results register a fail because there is no workpiece underneath the camera. Note that the camera is status is *Offline*.



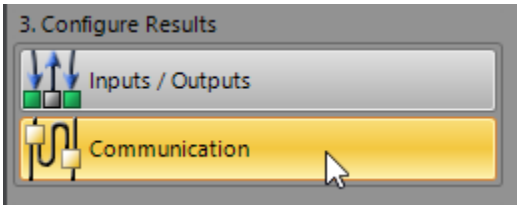
7. Select **View > In-Sight Files**.



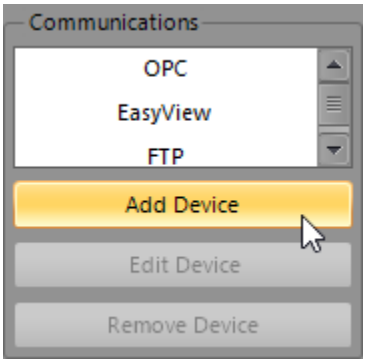
8. The In-Sight Files pane is displayed. Ensure that all of the job files that you created are shown as being saved to the camera.



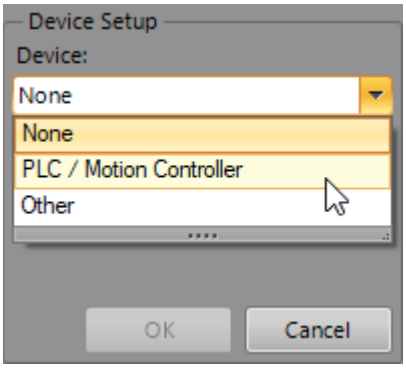
9. Select the **Communication** application step.



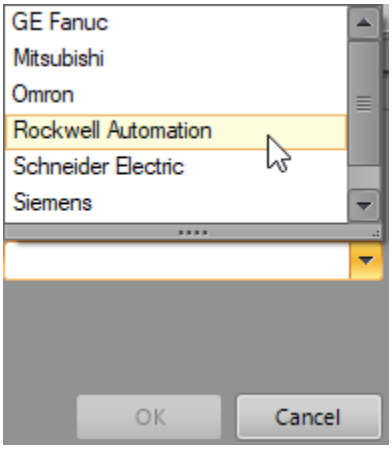
10. In the Communications area in the bottom-left corner of the settings pane, click **Add Device**.



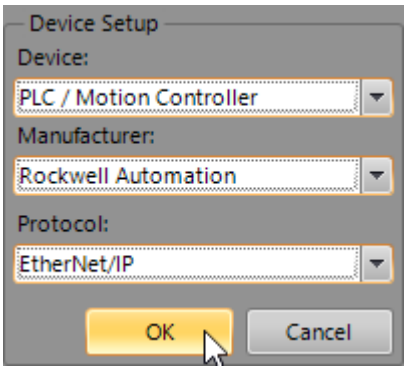
11. In the Device dropdown list, select **PLC / Motion Controller**.



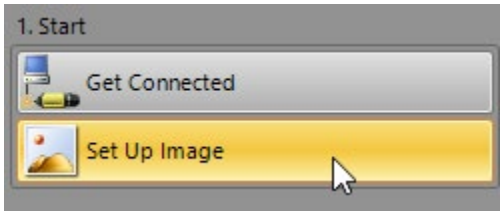
12. In the Manufacturer list, select **Rockwell Automation**.



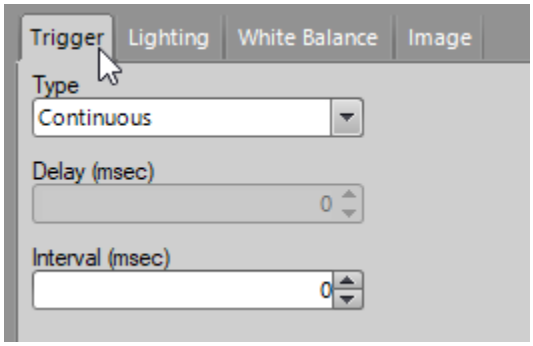
13. EtherNet/IP is automatically selected as the protocol. Click **OK**.



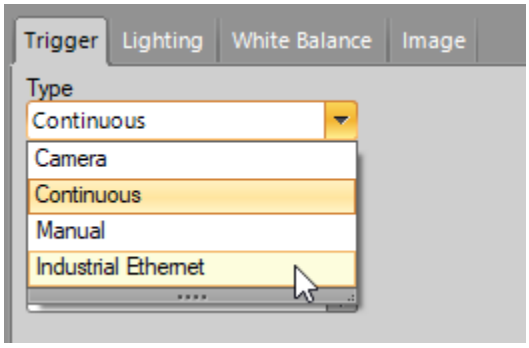
14. Select the **Set Up Image** step.



15. In the settings pane, select the **Trigger** tab.

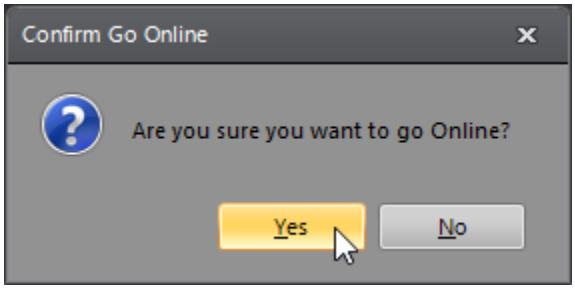


16. For Type, select **Industrial Ethernet**.



17. Click the Online button to switch the camera to Online mode.

18. In the Confirm Go Online popup dialog, click **Yes**.



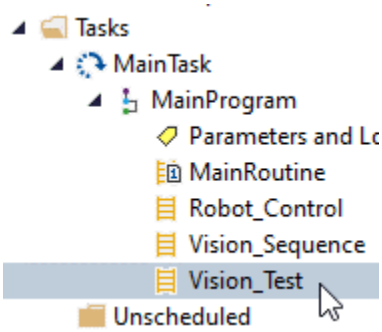
The camera is now online and can exchange data with the PLC. All trigger prompts will come from the PLC.

6.5. Testing the Vision Sequence

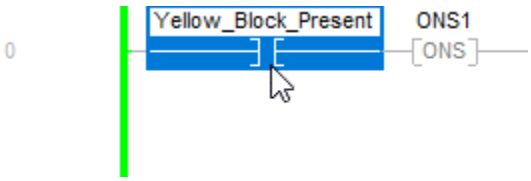
In this task, you will test the vision sequence.

Perform these steps:

1. Open the **Vision_Test** routine.

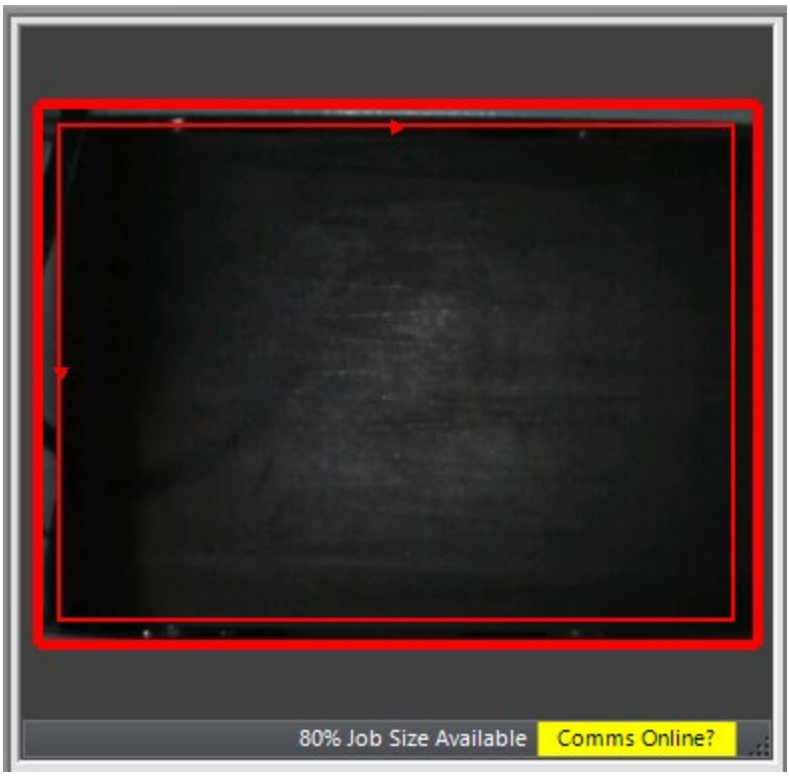


2. Select the Yellow_Block_Present XIC instruction.

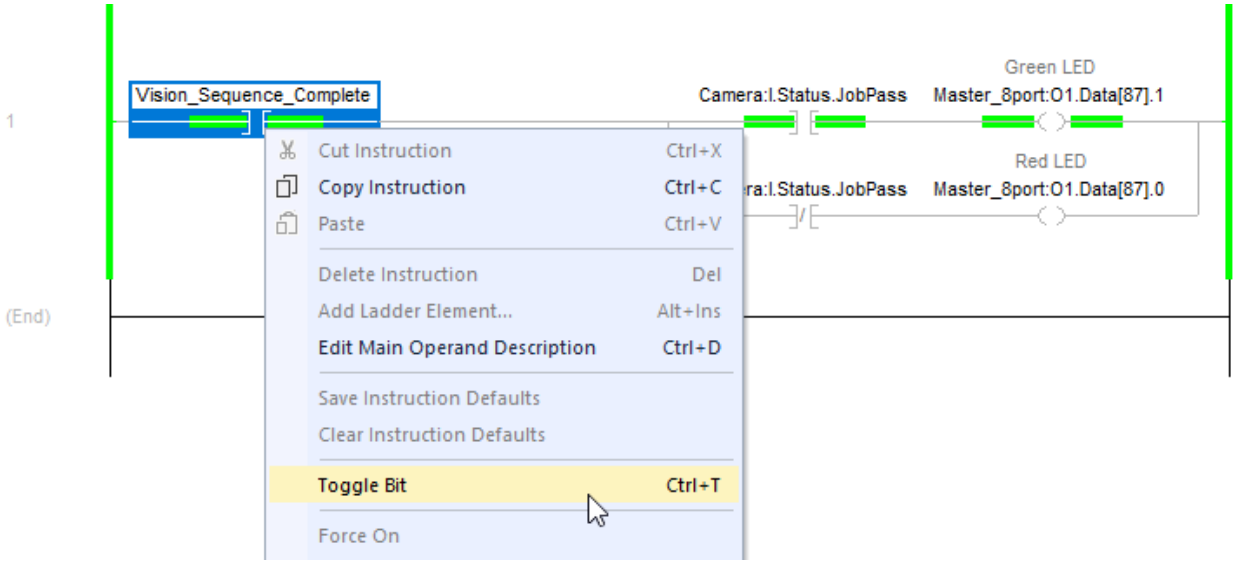


3. Press **Ctrl + T** to toggle the tag on. The camera is triggered and flashes to acquire an image. The signal lamp turns red because there is no workpiece, and the inspection has failed.

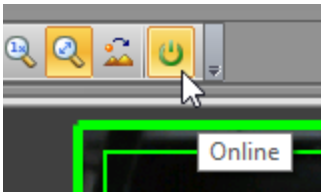
ⓘ ***Did you know:*** *When the camera went offline for the loading of the job, the status of the camera was* ***Comms Online?****, meaning that it was the Industrial Ethernet that was causing it to be offline.*



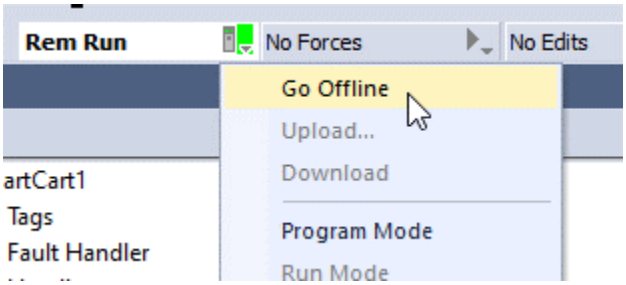
4. Press **Ctrl + T** again to toggle the tag off.
5. Place the yellow block under the camera.
6. Press **Ctrl + T** to toggle Yellow_Block_Present back on. The camera is triggered, and the signal lamp turns green because the block passes inspection.
7. Toggle the **Yellow_Block_Present** tag off.
8. Toggle the **Vision_Sequence_Complete** tag off. The signal lamp turns off.



9. Click the Online button to take the camera *offline*.



10. Take the PLC offline.



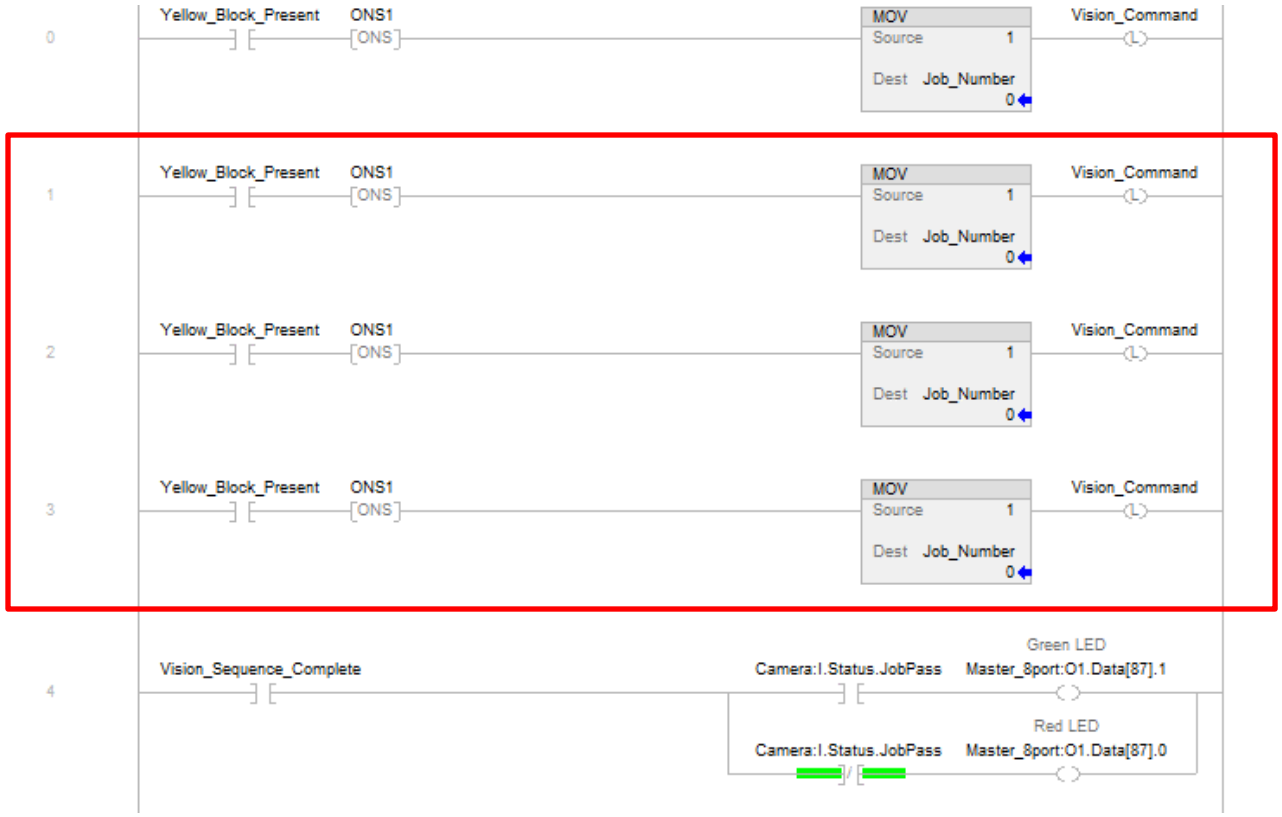
ⓘ ***Did You Know?*** *The Camera:I.Status.JobPass tag can easily be used in an HMI project.*

6.6. The Other Colors

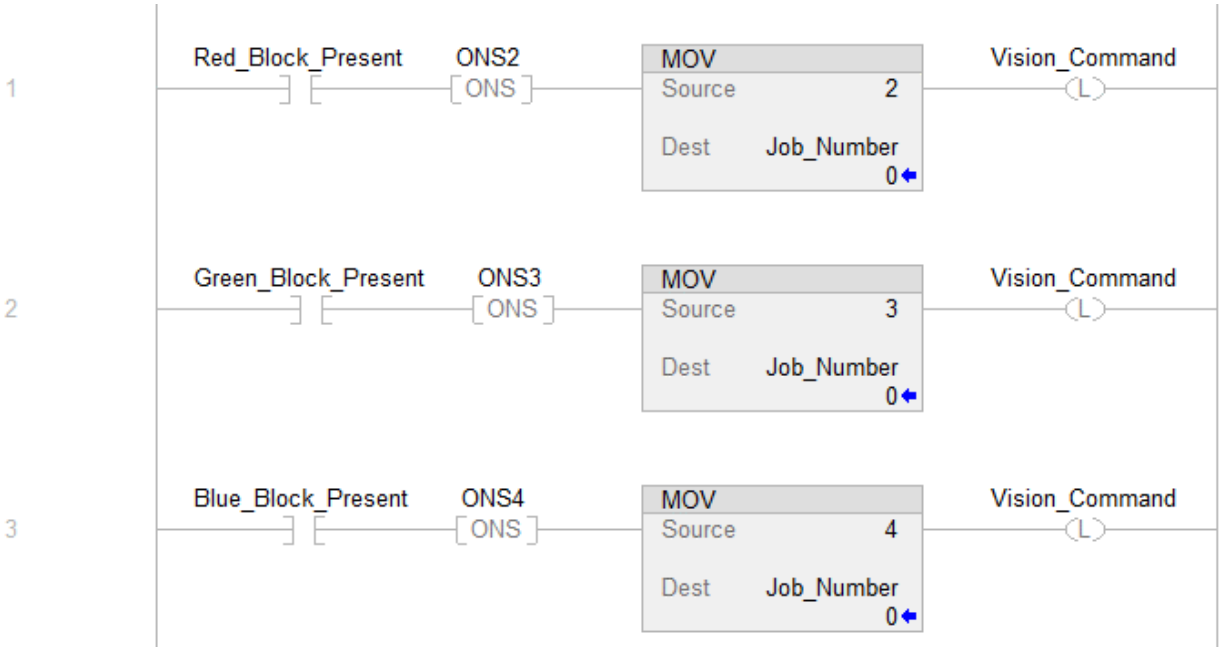
In this task, you will adjust the system in order to include the inspection jobs for the other colors of blocks.

Perform these steps:

1. In In-Sight Explorer, open the other camera jobs used for the inspection of the other colors.
2. Change the communication and trigger settings for each job as you did in Section 6.4.
3. In Logix Designer, open the Vision_Test routine and duplicate rung 0 three times.



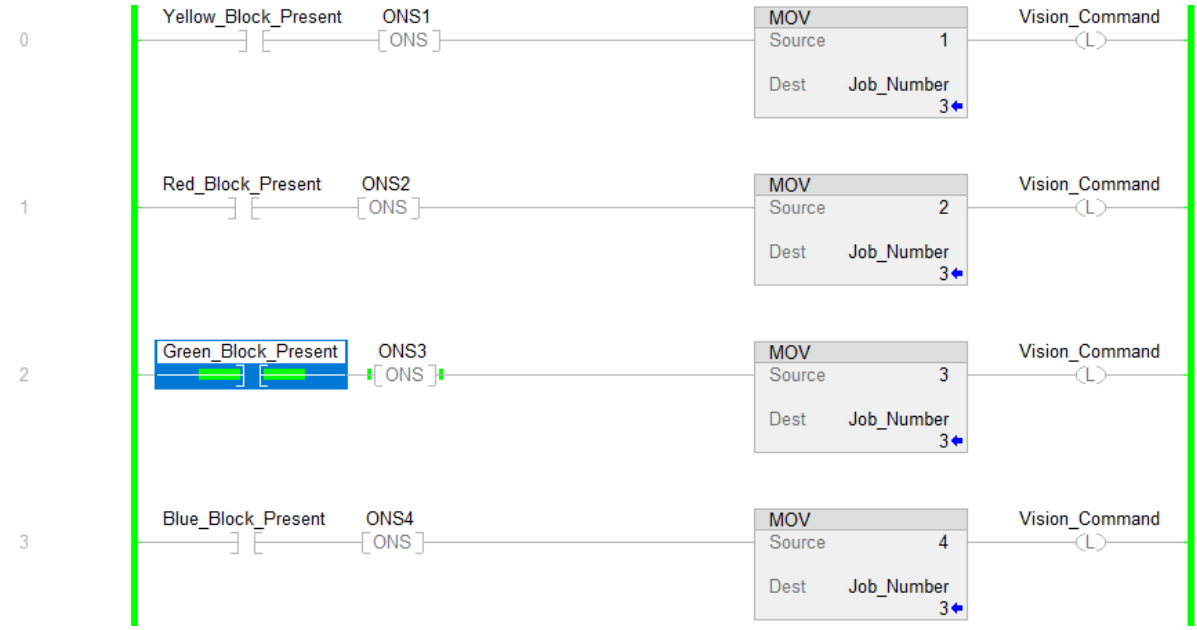
4. Modify each of these new rungs so that the tags and job numbers correspond to the various colors. Ensure also that a different tag is used for each ONS instruction.



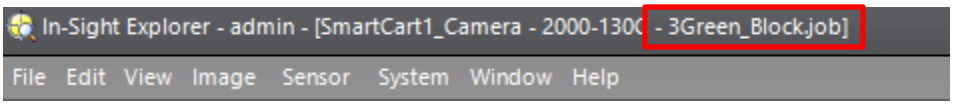
5. Verify the controller and save the project.

6. Switch both the camera and PLC online.

7. In the Vision_Test routine, toggle the different tags referenced by the XIC instructions on and off. Test with the relevant blocks present and absent under the camera. Ensure that the camera is triggered as expected and that the results of the inspection are as anticipated. Make any adjustments to the In-Sight jobs or the Logix project as required.



🛈 ***Note:*** *The loaded camera job is indicated at the top of the In-Sight Explorer window.*



8. Toggle off all XIC instructions in the Vision_Test routine, including the Vision_Sequence_Complete XIC.
9. Turn both the camera and the PLC offline.

7. Authentic Skill Assessment

Have your instructor verify that your work meets the requirements in the performance objectives and sign below. Keep this lab activity sheet for future reference.

Instructor Signature	Date

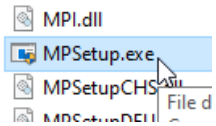
8. Reset Steps

If someone else is going to be performing this lab activity after you, perform the reset steps below. Consult with your instructor before doing so.

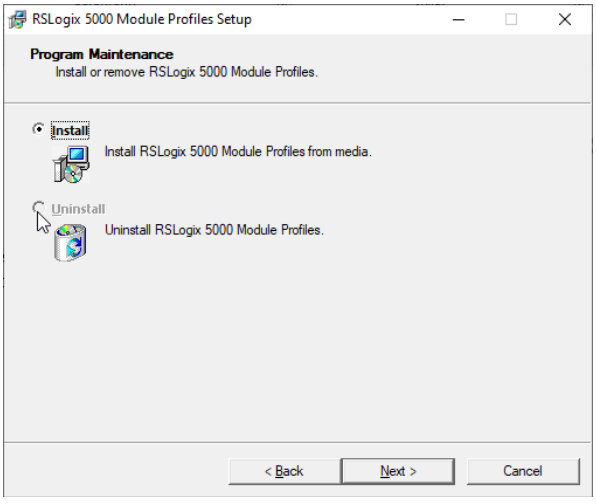
8.1. Uninstalling the AOP

Perform these steps to uninstall add-on profiles for Cognex devices:

1. Run **MPSetup.exe**.



2. Follow the steps of the install wizard. When you get to the Program Maintenance window, select **Uninstall**, and then click **Next**.



3. Select the IS2000 profile.
4. Complete the steps of the wizard.

9. Shutdown

Unless instructed otherwise by your instructor, review and complete each of the items on the checklist below.

- ☐ Return all blocks to storage.
- ☐ Close In-Sight Explorer.
- ☐ Close Logix Designer.
- ☐ Power down the I/O box.