Contents

Lab Exercise 1.1a – Getting Connected

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

- Identify the camera system components
- Launch the In-Sight Explorer software
- Change the In-Sight interface from the EasyBuilder to the Spreadsheet view

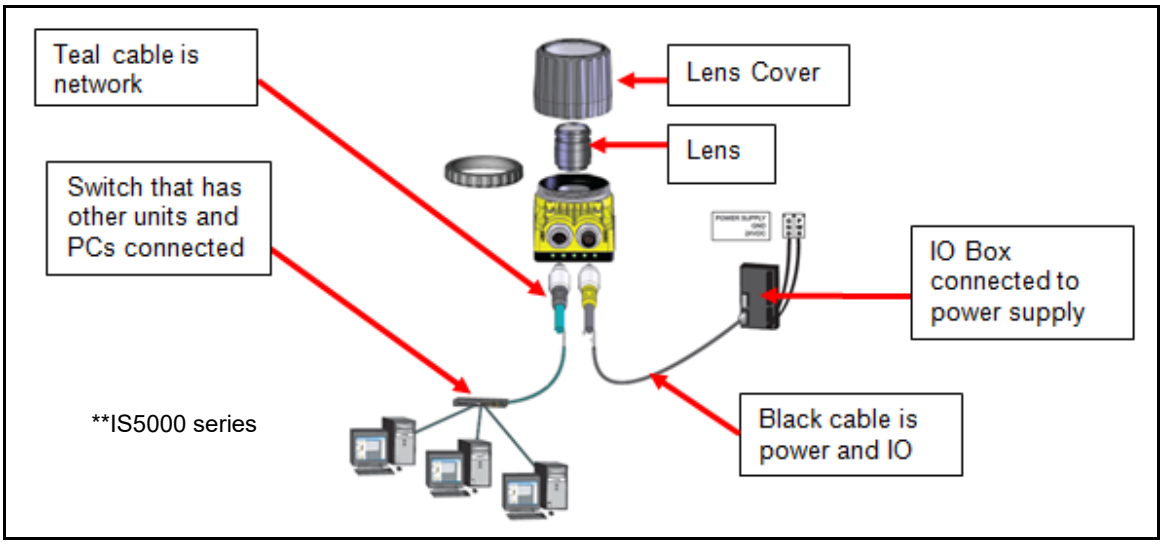
The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- Software Menus
- Spreadsheet
  - ImageAcquire cell

NOTE: *If you make a mistake or want to stop editing a cell, you can click the <Esc> key on your keyboard to back out.*

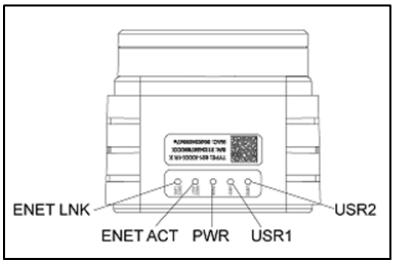
Follow the steps below to complete the lab exercise (if using an In-Sight Camera):

1. Assemble the hardware components.
   NOTE: *The tripod should have the top four portions of its legs pushed back in to get the proper height. The unit should be directly above the part below it with the lens pointing down.*

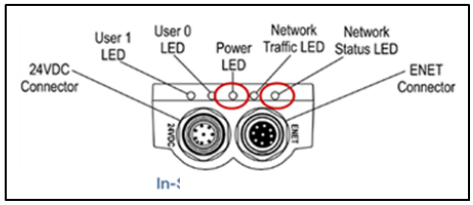


2. Confirm there is power and network connectivity to the unit.

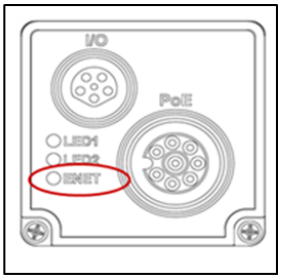
Normal Led Pattern:
7000 Series – power LED and ENET connector should be lit in green
2000 Series & 7000 Gen II Series – power LED green, network LED yellow
5000 Series – power LED and ENET connector should be lit in green
Micro Series – ENET LED should be green



In-Sight 7000 Series



In-Sight 5000 Series



In-Sight Micro


In-Sight 2000 Series and 7000 Gen II Series

3. Look at the set-up at your work station and make note of which item is the In-Sight sensor (camera) and which is the I/O Expansion Module.

   Look at the type of hardware that you are using and make note of it below:

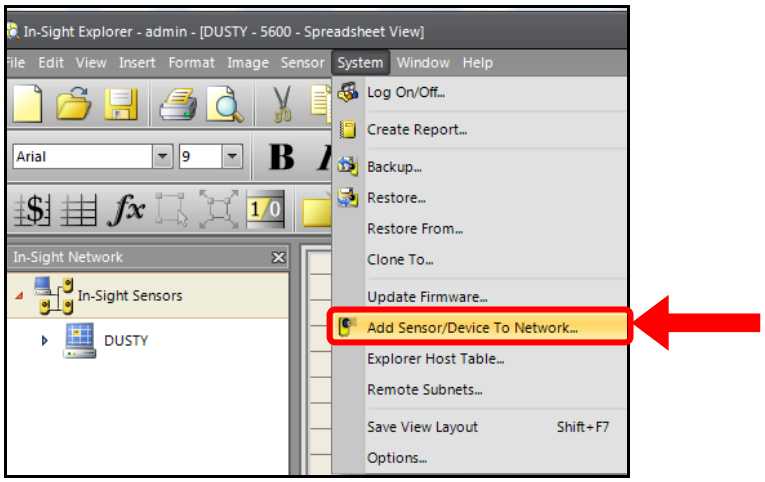
   In-Sight Sensor Type:_____

   I/O Module Type:_____

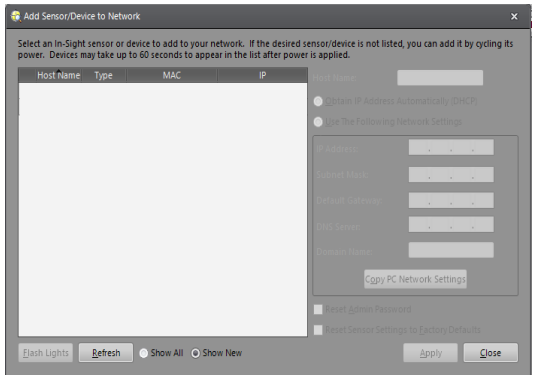
4. Click the **In-Sight Explorer icon** on your desktop to launch In-Sight Explorer. Or, Start Menu → Programs → Cognex → In-Sight Explorer (ISE) on your PC.

5. Next, you are going to set your camera to factory defaults, which will remove any changes in settings made to your camera by a previous class. (If you already did this earlier in the class, skip to step 7.)

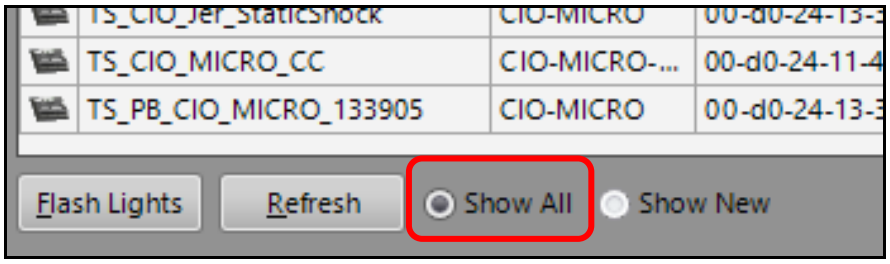
Go the **System→Add Sensor** menu



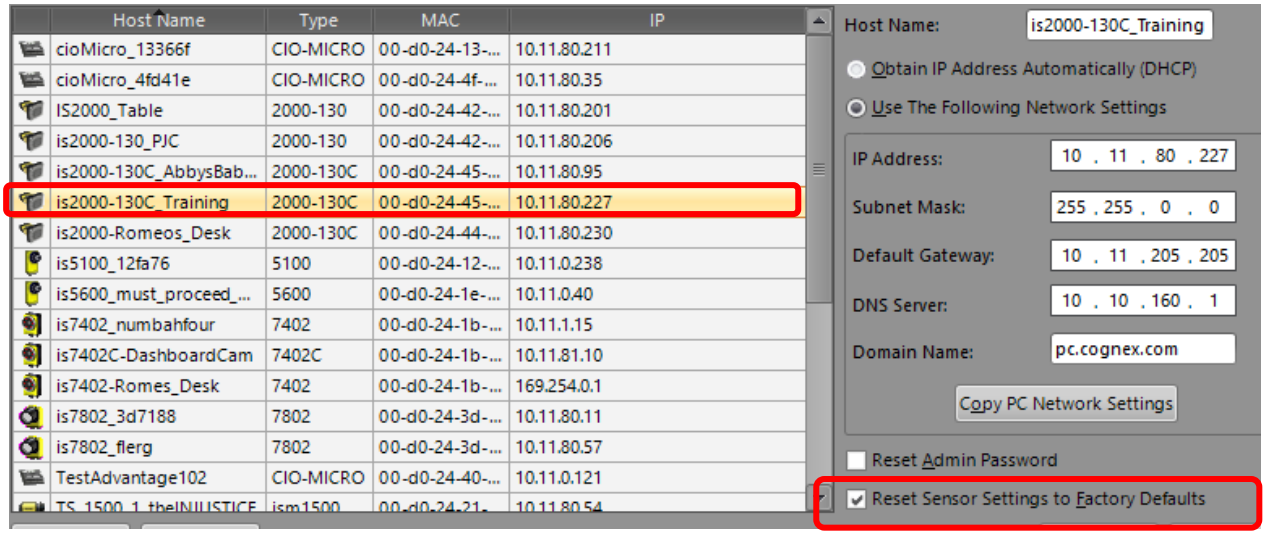
6. This will bring you to the **Add Sensor/Device to Network** window. Your camera's name should *not* appear, because it is already properly configured.



To show all cameras that are properly networked, click on **Show All.**

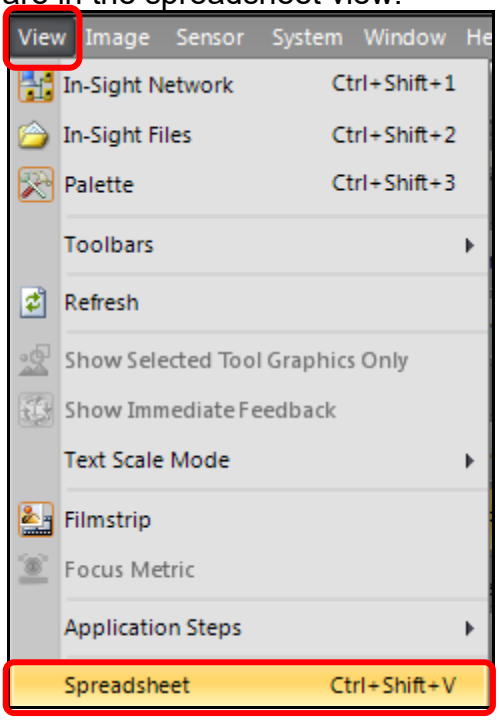


7. After a few seconds, a list of all cameras on the network, including yours, should appear. Click on yours, then click on the checkbox labeled **Reset Sensor Settings to Factory Defaults**.

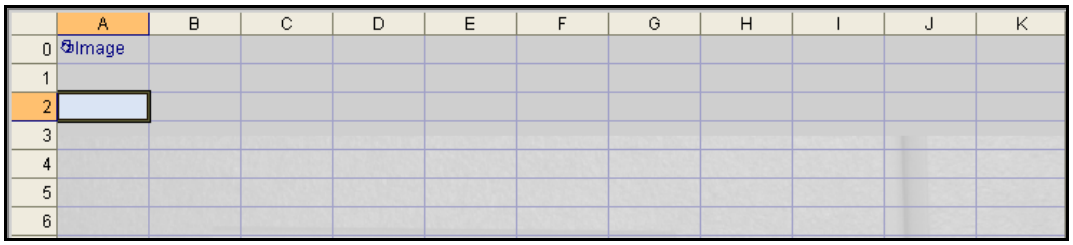


Click **Apply** and follow the resulting instructions to cycle power on your camera. This will take about 2 minutes, at which point a message will indicate that the reset was successful. **Close** the **Add Sensor** window.

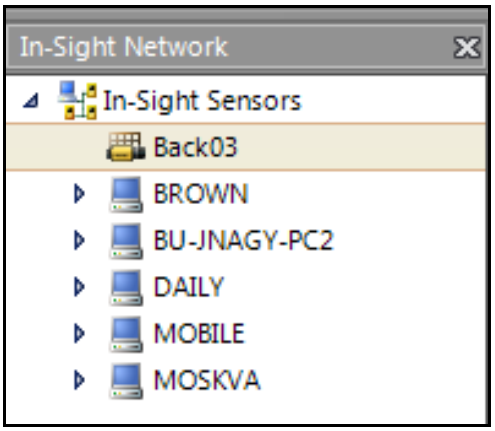
8. Log on to your In-Sight camera.
9. Confirm that you are in the **Spreadsheet View**.
10. In the Application Menu, click **View → Spreadsheet**. If you do not see this option in the View menu then you are in the spreadsheet view.



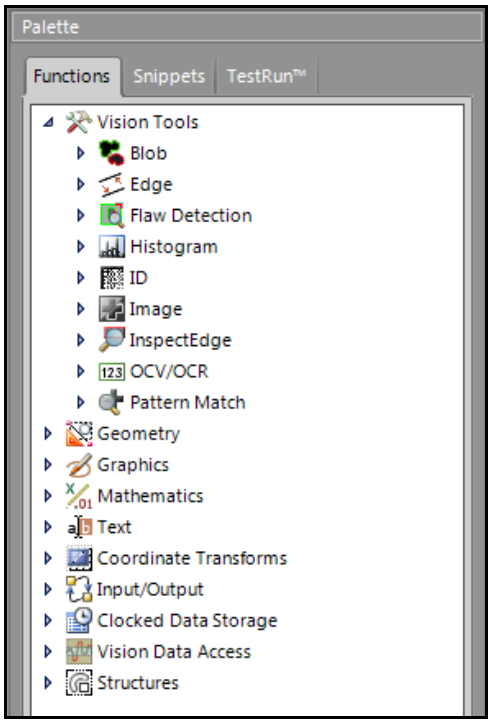
11. Click the **New Job** button.
A blank spreadsheet displays.



12. Click **View → In-Sight Network** to see all of the cameras and emulators that are on the network.



13. Click **View → Palette** to view all of the tools available.



Lab Exercise 1.1b – Getting Connected

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

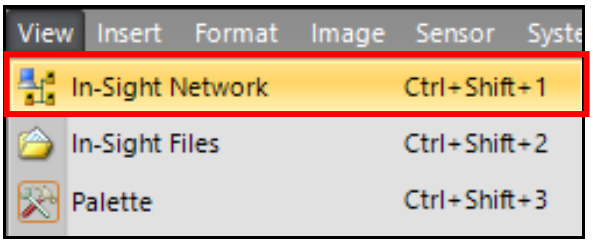
- Emulate the In-Sight 7802

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

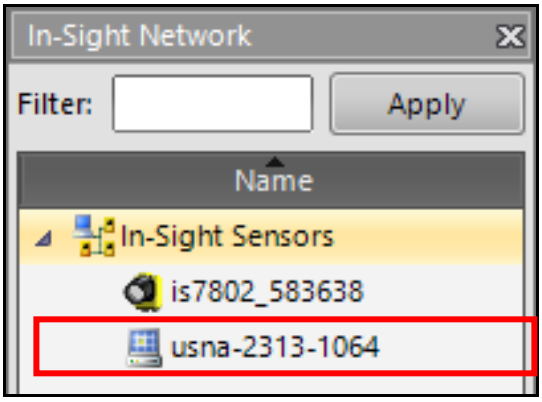
- Options / Emulation

Follow the steps below to complete the lab exercise (using an Emulator):

1. Launch In-Sight Explorer (ISE) and enter the Spreadsheet view.
   The Spreadsheet view displays. (If not, specify View→Spreadsheet.)
2. Click the **View** menu and select **In-Sight Network** from the list.

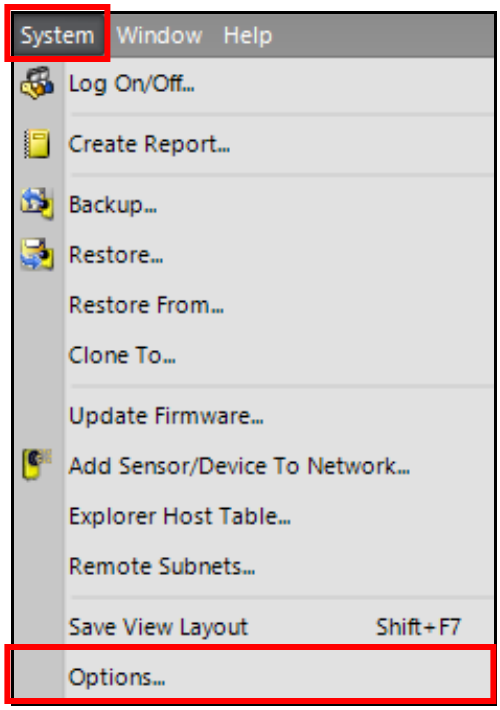


   The **In-Sight Network** Pane displays.



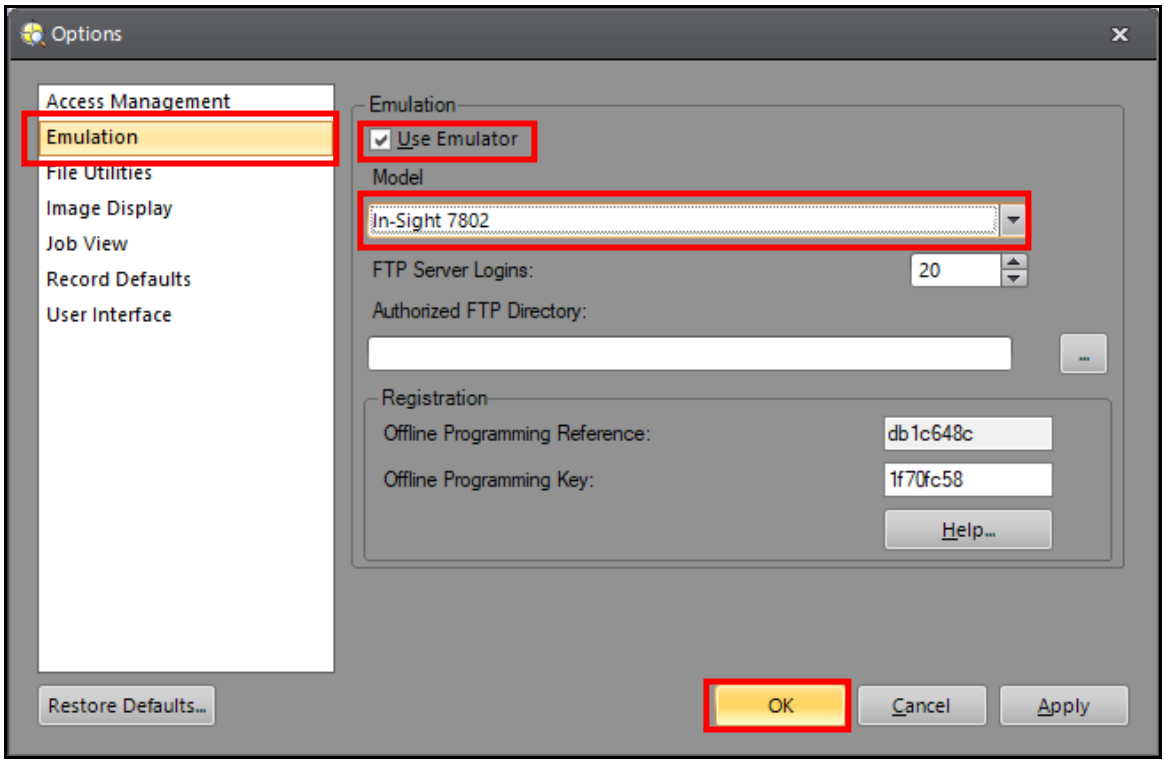
3. Select your **Emulator** from the list of available sensors.
   NOTE: *The emulator has a computer icon, while the sensor has a camera icon.*

4. Select **Options** from the **System** menu.

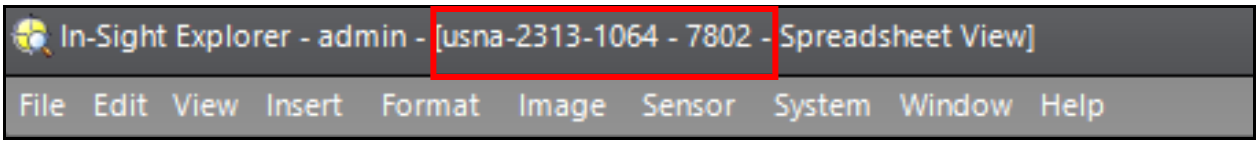


5. The **Options** dialog displays:
   - Select **Emulation** from the menu.
   - Check the **Use Emulator** checkbox
   - Select **In-Sight 7802** for the Model

   Click the **OK** button.



6. If the **Save Current Emulator Job?** dialog displays, click the **No** button to continue.
7. The **Emulator** model will display at the top of the In-Sight Explorer window.



Lab Exercise 2.1a – Software and Image Acquisition

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

- Log into the camera and put it into Live Mode
- Acquire a good image on the camera

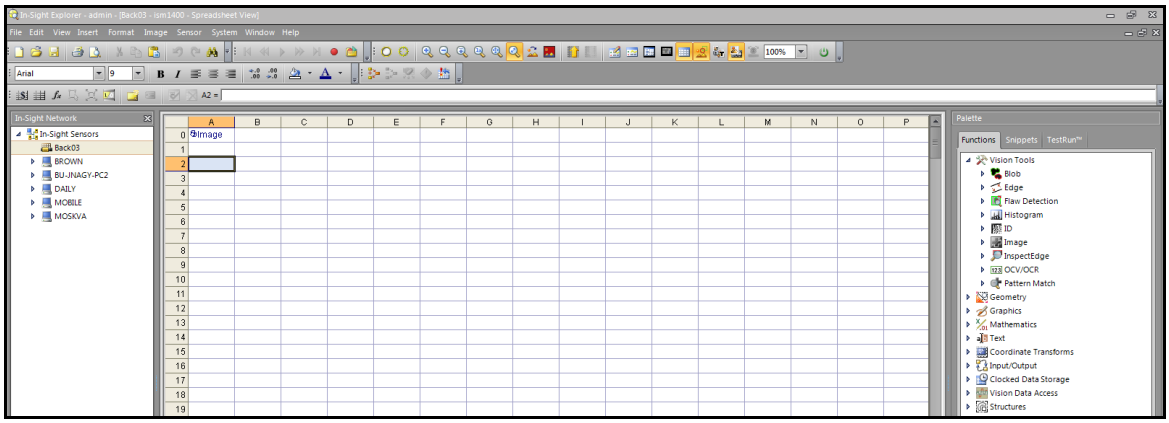
The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- Logging on
- Live Mode

Follow the steps below to complete the lab exercise (*using an In-Sight Camera*):

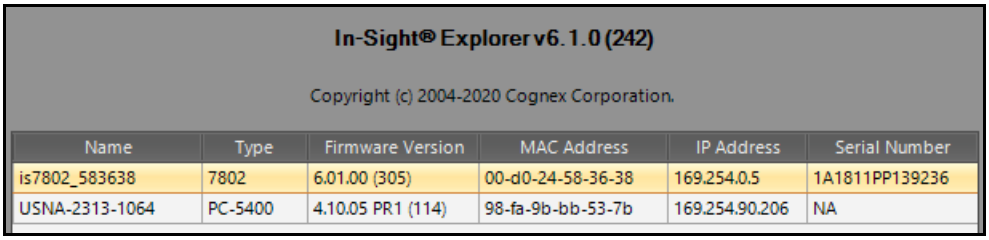
1. Launch In-Sight Explorer (ISE) and enter the Spreadsheet view.
2. Create a folder on your desktop named **InSightClass** with the date appended e.g. **InSightClass012720**. This is where you will save the jobs that you will develop in the lab exercises.

   The Spreadsheet view displays. (If not, specify View→Spreadsheet)



3. Double-click on your camera to log into it and control it.

   HINT: *If you don't know which camera is yours, click Help → About In-Sight Explorer and match your MAC address (printed on the camera) to your camera's name.*



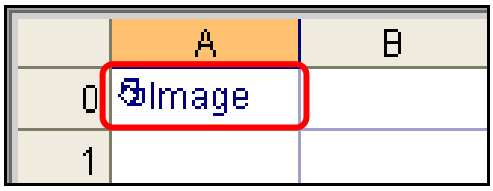
Name	Type	Firmware Version	MAC Address	IP Address	Serial Number
is7802_583638	7802	6.01.00 (305)	00-d0-24-58-36-38	169.254.0.5	1A1811PP139236
USNA-2313-1064	PC-5400	4.10.05 PR1 (114)	98-fa-9b-bb-53-7b	169.254.90.206	NA

3. Start a new job.

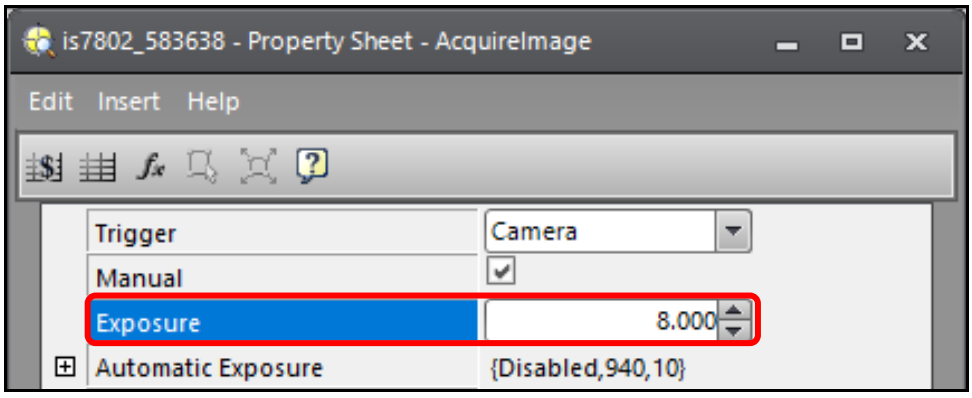
   Click the **Live Video** button to start a live image.
4. Move the part under the camera to confirm the image is updating.

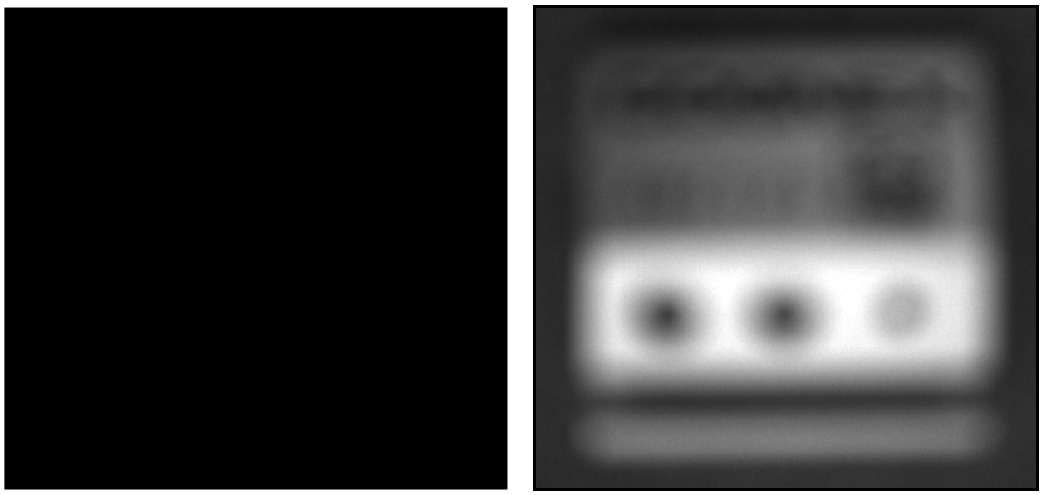
   NOTE: *If there is too much glare on the part, try moving the tripod so it is not directly under a ceiling light. As a last resort, try tilting the camera or part.*

5. If you are using an **Autofocus** lens, continue with **step 6**.
   If you are using a **C-Mount** lens, skip to **step 15**.
   NOTE: *Click anywhere in the Image view window to stop the live acquisition.*
6. Double-click on cell **A0**, the **Image** cell.



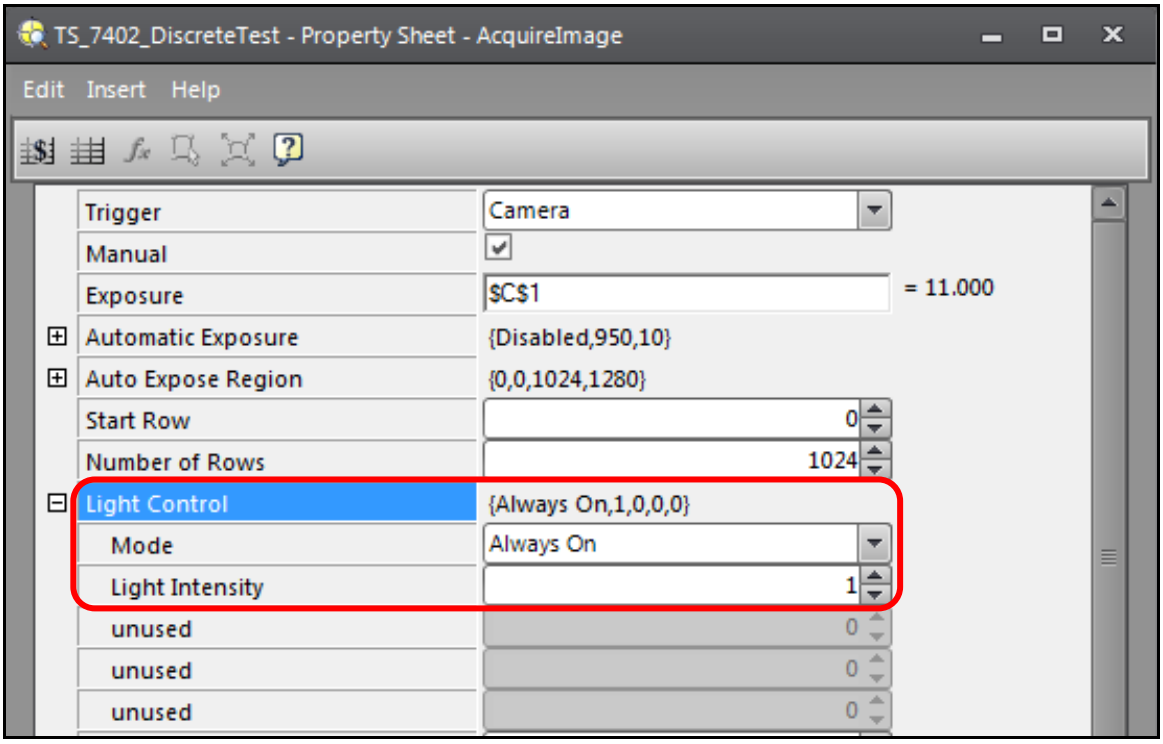
7. Adjust the **Exposure** setting to establish light and dark pixels.
   NOTE: *Increase exposure for a lighter image.*



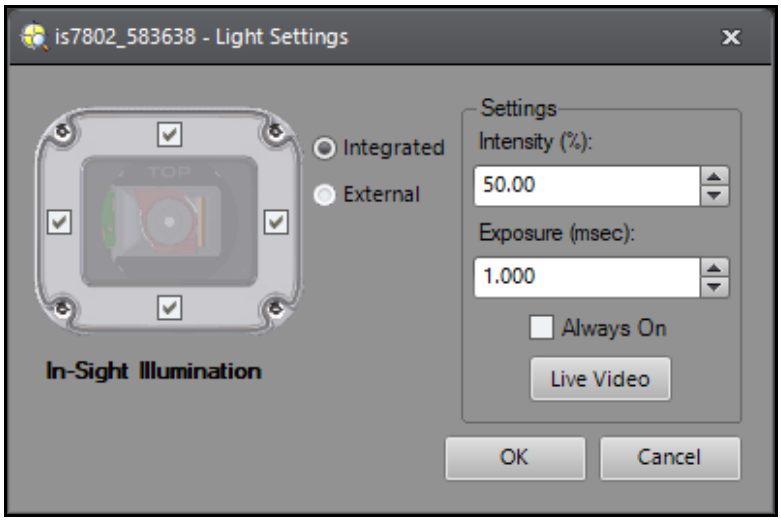


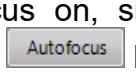
7a. If you are using a camera with a ring light accessory, adjust the **Light Control** settings to establish light and dark pixels.

NOTE: *Ensure either Always On or Exposure Control is selected, and then adjust the Light Intensity to establish light and dark pixels.*

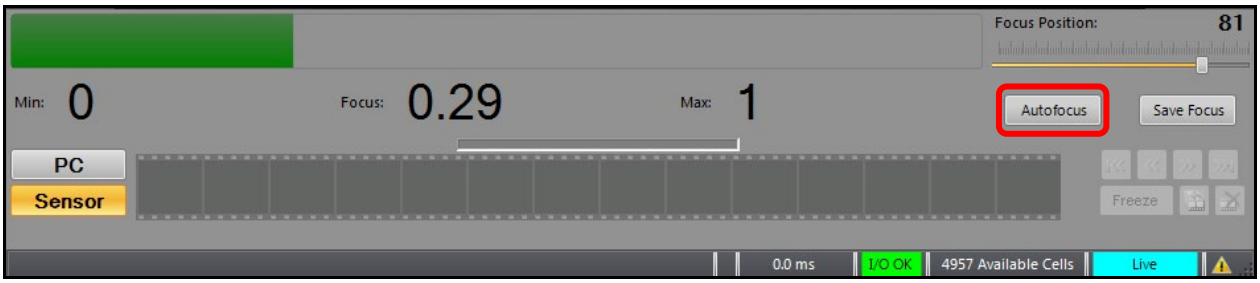


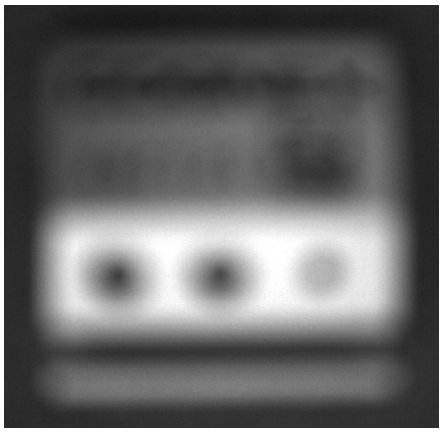
7b. If you are using a camera with the four-bank integrated lighting accessory, check the settings under **Sensor→Light Settings.**



8. Set the Focus Region in **AcquireImage**. You will need to decide which area of the part to focus on, since it is a 3D part. Then adjust the focus by using the **Autofocus**  button.

NOTE: *The button is in the lower right-hand corner of the live video view.*

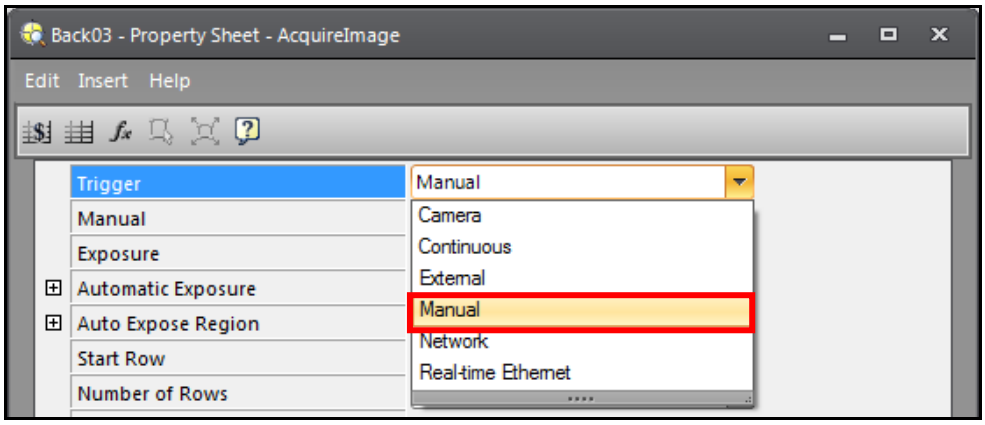




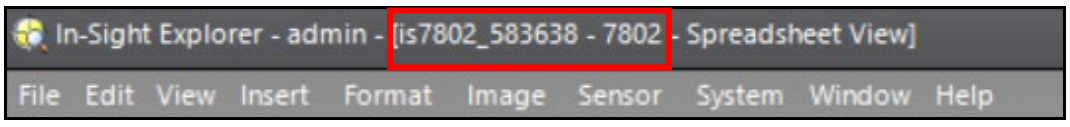
9. Click the **Live Video** button to close the window.
10. Double-click on cell **A0**, the Image cell.

    The **Property Sheet – AcquireImage** dialog box displays.
11. Set the Trigger to **Manual** and click the **OK** OK button.

    NOTE: *Use the **Trigger** button or use the **<F5>** key to trigger.*



NOTE: *The top border of ISE will indicate what camera you are logged onto. Confirm that you are logged onto your camera.*



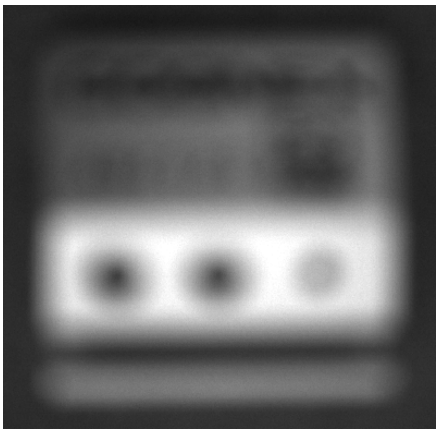
NOTE: *If you are not logged onto your camera, select your camera from the In-Sight Network list and double-click on it. (Lab 1 – step #8)*

12. Trigger the camera; you should see the last image that your camera acquired. Move your hand under the camera – since you are not in Live Mode you should not see any movement.
13. Click on the **Live Video** button. Wave your hand under the camera, you should now see movement.
14. Place your good block with the Cognex side up so that the whole part appears in your view.

15. If you are using a C-mount lens, adjust the two ring controls on the lens to adjust the aperture and focus.

5000, 7000 and Micro cameras with C-mount lens:

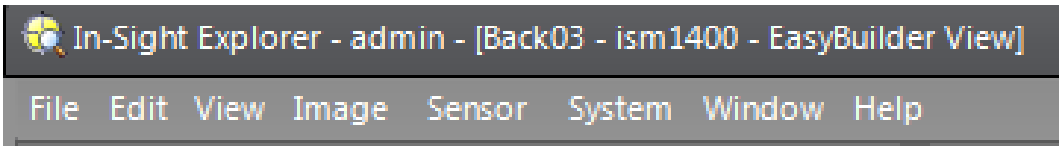
Aperture – adjusts the amount of light allowed to pass through the lens.
Focus – adjusts the sharpness of the image.



To verify the writing on the block is dark and the metallic background light on your image, click the **Show Image Saturation** button (top icon bar) to assist with this.

NOTE: *Too much blue means that the image is too dark and too much red means that the image is too light. To remedy this, adjust aperture setting, exposure or light control (LEDs).*

16. If the cameras in the training room are on a network, find another In-Sight system in the room and ask its user if it is Okay for you to try and log into it.
17. With the Set up Image button selected, activate **Live Mode** on their system to verify that you have logged into the correct system. Offer them the same courtesy.
18. Browse through the various drop-down menus in In-Sight Explorer to compare what is available on both the Icon bar and within the Menu bar.



19. Save the job as **MyFocus** in the folder that you created in Step 1.

Lab Exercise 2.1b – Software and Image Acquisition

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

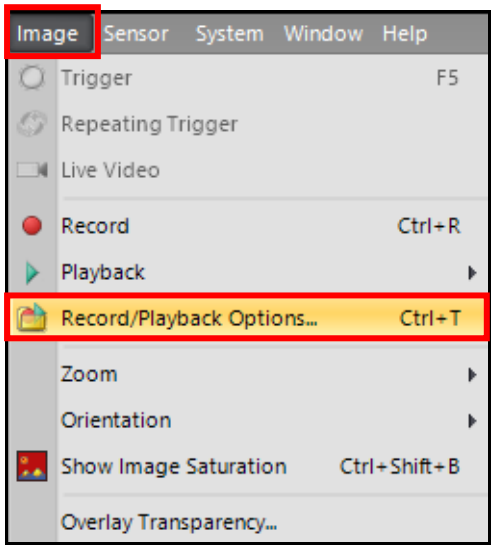
- Direct your emulator to the image database used in the lab exercises

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- Image

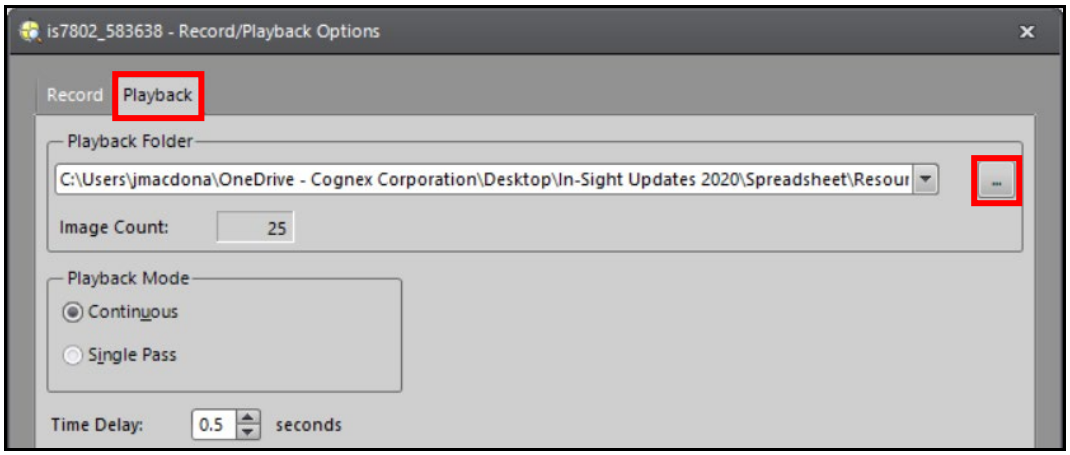
Follow the steps below to complete the lab exercise (*using an Emulator*):

1. Create a folder on your desktop named **InSightClass** with the date appended e.g. **InSightClass012720**. This is where you will save the jobs that you will develop in the lab exercises.
2. Select **Record/Playback Options** from the **Image** Menu.



The **Record/Playback Options** dialog displays.

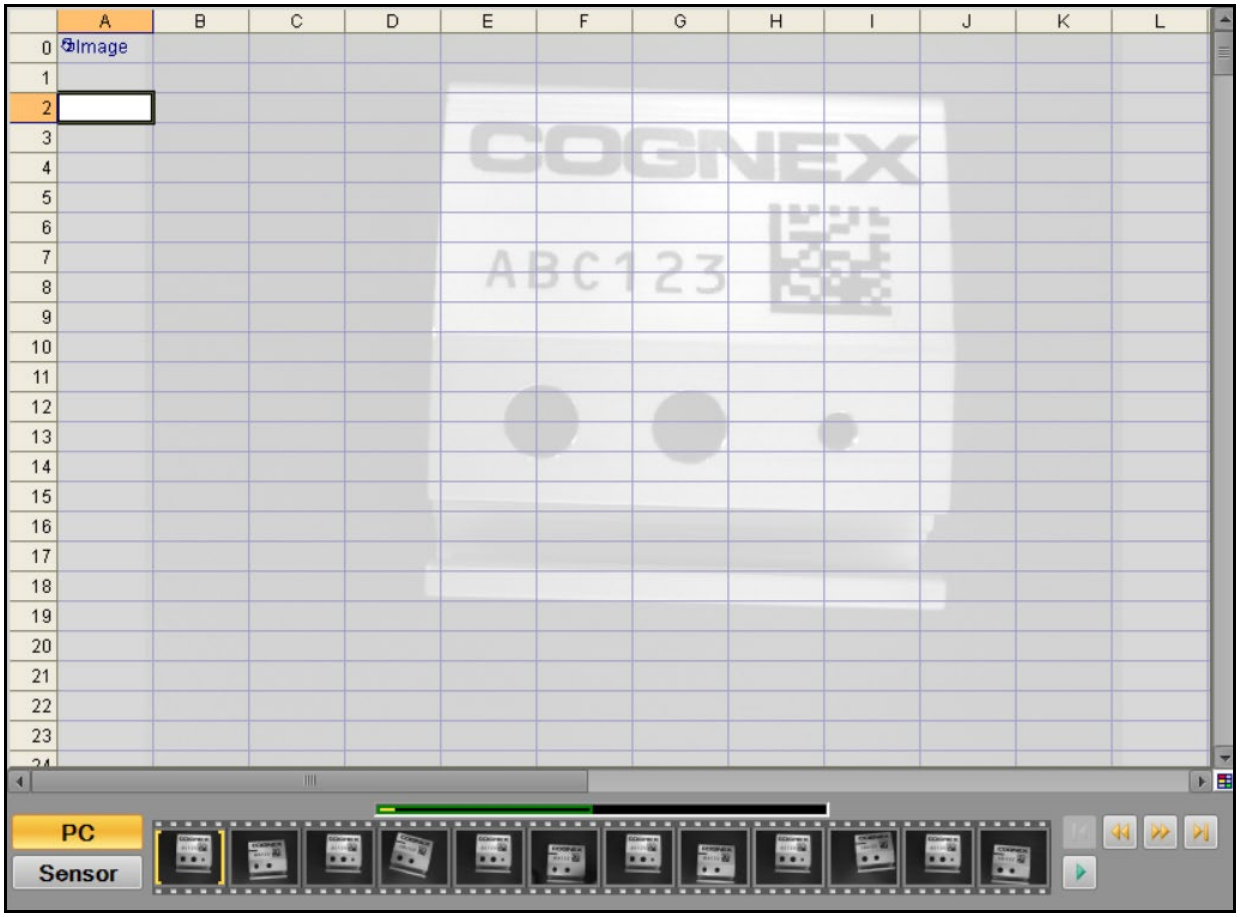
3. On the Playback tab click the ellipsis to direct the playback folder to the image database to be used for the lab exercises.



NOTE: *The Instructor will tell you where the images are located.*

4. Click the **OK** button at the bottom of the dialog.
   The images display in the **PC Filmstrip** below the spreadsheet.

5. Click the first image in the filmstrip.
   The image displays behind the spreadsheet.



6. Save the job as **MyFocus** in the folder that you created in Step 1.

Lab Exercise 2.2 – References

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

- Insert Absolute and Relative references into a spreadsheet and observe the differences

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

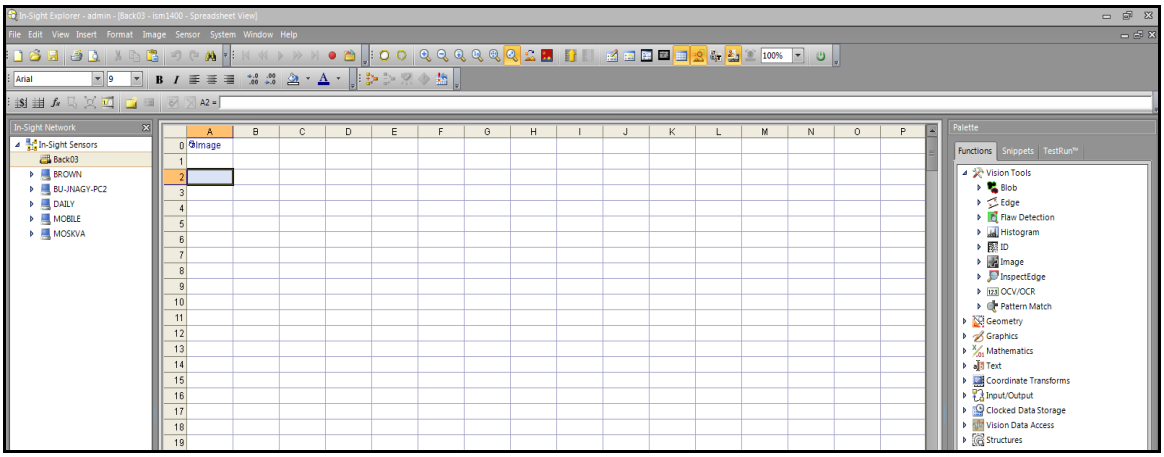
- Absolute Reference
- Relative Reference

Absolute References

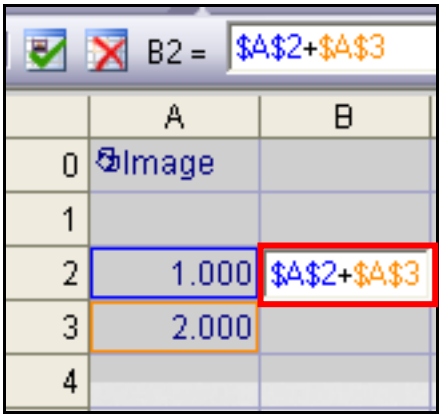
Follow the steps below to complete the lab exercise:

1. Click the **New Job** 

button to begin a new job.
A blank Spreadsheet displays.
NOTE: *We will not be using an image for this lab.*



2. Enter a value of 1.0 in cell **A2**.
3. Enter a value of 2.0 in **A3**.
4. Enter a formula into cell **B2** that adds cells **A2** and **A3** using **Absolute References**.



Relative References

Follow the steps below to complete the lab exercise:

1. Enter a value of -1.0 in cell **A5**.
2. Enter a value of 4.0 in **A6**.
3. Enter a formula into cell **B5** that adds cells **A5** and **A6** using **Relative References**.

Copying and Pasting Relative References

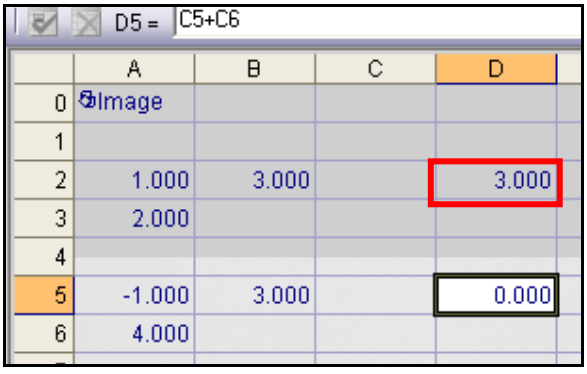
Follow the steps below to complete the lab exercise:

1. Highlight cell **B2**.
2. Copy and paste it to cell **D2**.
3. Highlight cell **B5**.
4. Copy and paste it to cell **D5**.
5. Examine the formula in cell **D5** and compare it to the original formula in **B5**.

   How do they differ? _____

   Why do they differ? _____

   Why is cell **D2** showing a number? _____



6. Save the job as **MyCells** in the folder on the desktop created in Lab #1.

Lab Exercise 3.1 – PatMax

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

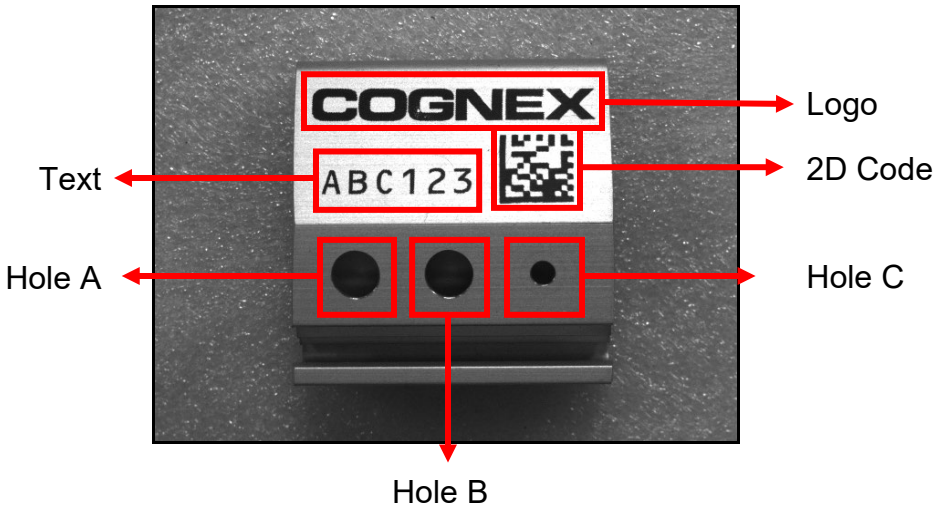
- Utilize the TrainPatMaxPattern and FindPatMaxPatterns to locate the Cognex block in the Field of View
- Report the location based on row, column, and angle
- Apply the location information for fixturing in other vision functions

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- Live Video
- FindPatterns
- Profiler

We will use the following terminology to identify the parts of the Cognex block.

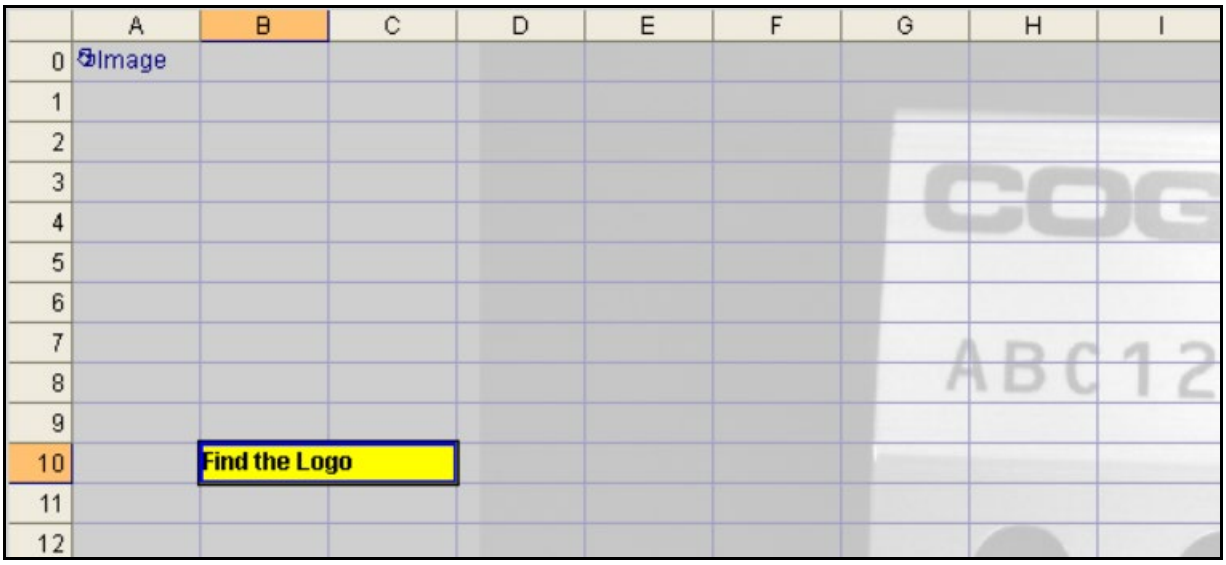
- Logo – Cognex logo on the front of the part
- Text – Human readable code
- 2D Code – Data matrix code of human readable text
- Holes A, B, and C – The 3 holes in the flat portion of the block



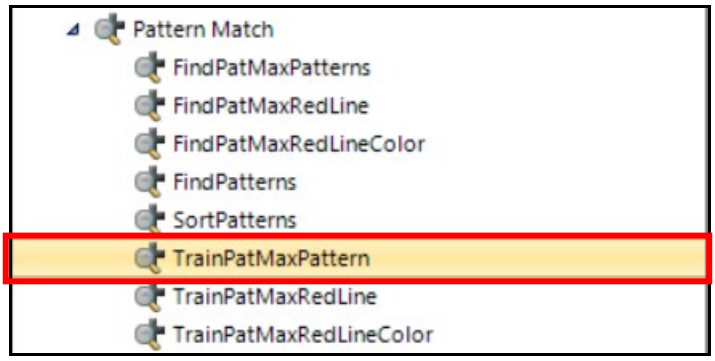
Follow the steps below to complete the lab exercise:

1. Open the **MyFocus** job from Lab Exercise 2.
2. Save the job as **MyPatMaxPatterns** in the folder that you created in Lab #1.
3. To verify the block is in the Field of View, click the **Live Video** button and position the block under the camera so that it is centered in the field of view, as shown above.
   NOTE: *Make it as large as possible in the FOV for good resolution but leave some room for part movement.*
4. Exit Live Video mode.

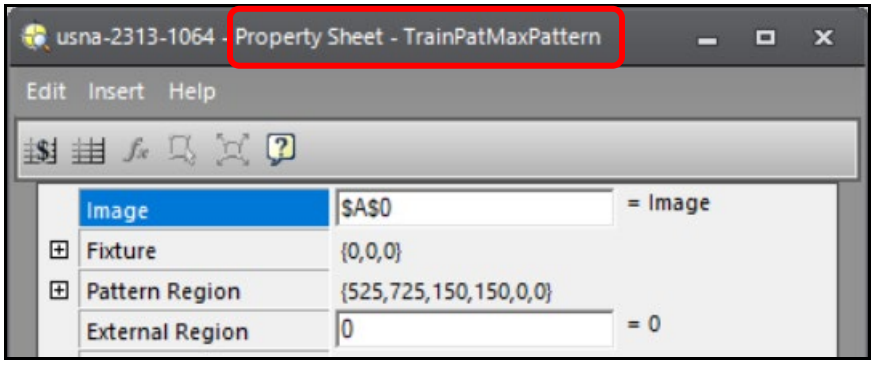
5. Leave the first 10 spreadsheets rows (numbered 0 – 9) blank (except for AcquireImage).
   NOTE: *We will use these rows in a later lab to create an operator interface.*
6. Enter the comment **Find the Logo** in cell **B10**. Be sure to start with an apostrophe (‘).
   NOTE: *You can format the cells with the comments so that they are more noticeable in your spreadsheet.*



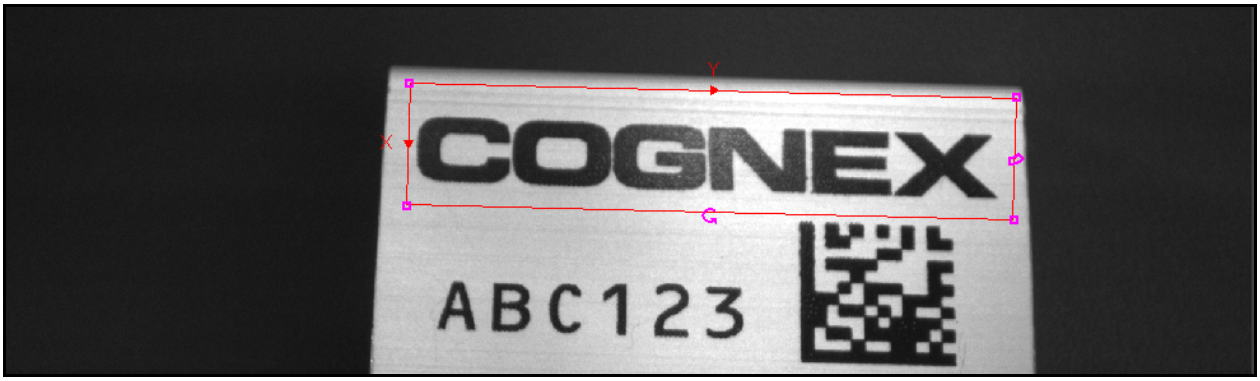
7. Insert a **TrainPatMaxPattern** function into cell **C11** of the spreadsheet.



The **TrainPatMaxPattern Property Sheet** displays.



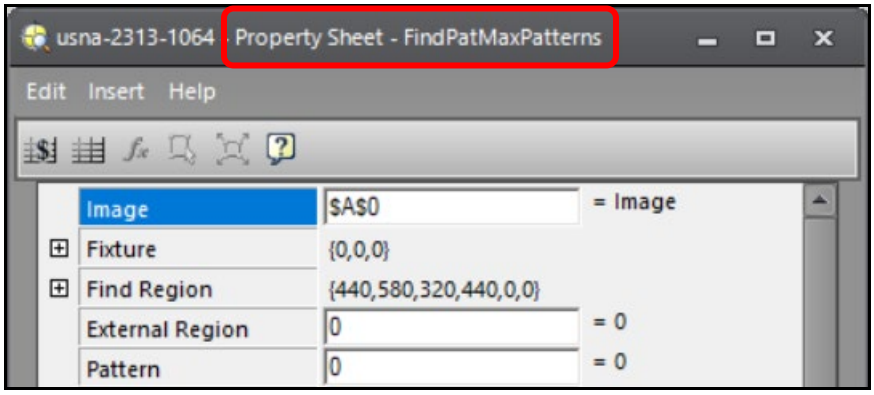
8. Double-click the **Pattern Region** parameter and position the region around the Cognex logo, as shown below.



9. Allow the defaults to remain for the other parameters and click the **OK** 

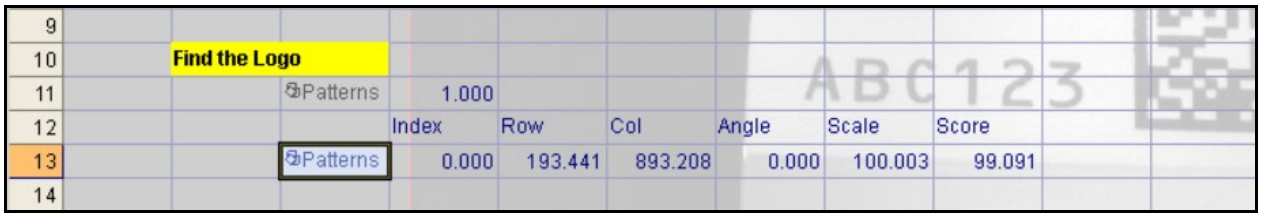
button.
The **TrainPatMaxPattern** result displays in the spreadsheet.

10. Add a **FindPatMaxPatterns** function into cell **C13**.
The **FindPatMaxPatterns Property Sheet** displays.

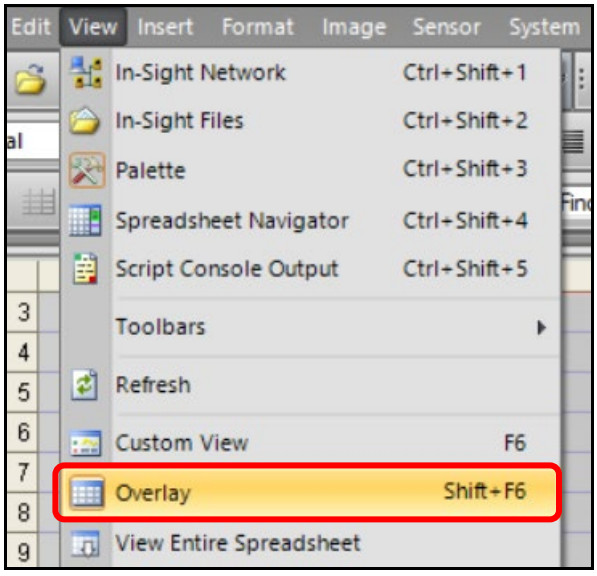


11. Double-click the **Find Region** parameter.
12. Click the **Maximize Cell Region**  button in the toolbar and the **OK**  button to set the region.
13. Configure the Parameters of the FindPatMaxPatterns Property Sheet as follows:
Pattern = reference to cell C11
Find Tolerances = Angle Start -45, Angle End 45, Scale Start 98, Scale End 100
Show = input and result graphics
Allow the other parameters to remain as their default values.

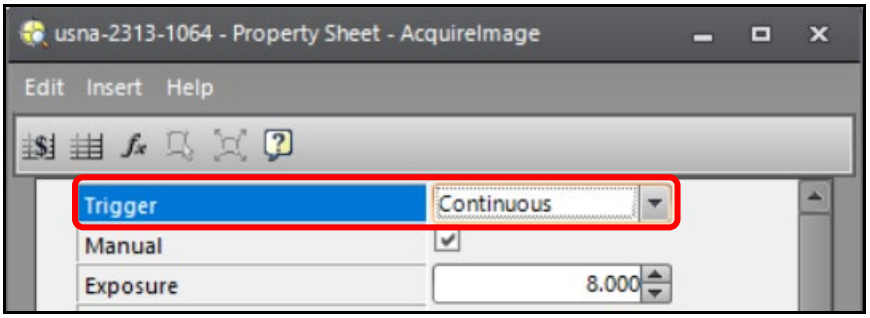
The FindPatMaxPatterns result displays in the spreadsheet.



14. Click **Overlay** in the View menu or click the **Overlay** icon in the toolbar to turn off the overlay to see the image without the spreadsheet blocking it.



15. When done, turn the Overlay back on.
16. Double-click on cell **A0** to change the AcquireImage to **Manual** Trigger.



17. Click the **Repeating Trigger** button to go online.

18. Move the part around, rotate it and observe the FindPatMaxPatterns' results in the spreadsheet when the model is within the Field of View (FOV) and outside of the FOV.

9								
10	Find the Logo							
11		Patterns	1.000					
12			Index	Row	Col	Angle	Scale	Score
13		Patterns	0.000	193.481	893.220	0.000	100.006	99.113
14								

19. Observe the **Angle** value as you rotate the block. Repeat with the bad block.

9								
10	Find the Logo							
11		Patterns	1.000					
12			Index	Row	Col	Angle	Scale	Score
13		Patterns	0.000	318.318	846.775	4.444	100.012	96.476
14								

9								
10	Find the Logo							
11		Patterns	1.000					
12			Index	Row	Col	Angle	Scale	Score
13		Patterns	0.000	190.379	876.693	3.115	100.020	99.174
14								

9								
10	Find the Logo							
11		Patterns	1.000					
12			Index	Row	Col	Angle	Scale	Score
13		Patterns	0.000	193.481	893.220	0.000	100.006	99.113
14								

20. Turn the **Repeating Trigger** off by clicking the button  once.
21. Create an **If** logic statement that uses 1 to Pass and 0 to Fail for the FindPatMaxPatterns result in cell **K13**.

10	Find the Logo								
11		Patterns	1.000						
12			Index	Row	Col	Angle	Scale	Score	Logic
13		Patterns	0.000	193.495	893.226	0.001	100.003	99.170	If(I13>97,1,0)

22. Save the job.

Lab Exercise 3.2 – PatMax RedLine Tools (if time allows)

Use the PatMax tools **TrainPatMaxRedLine** and **FindPatMaxRedLine** to locate the block.

Lab Exercise 4.1 – ExtractHistogram

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

- Utilize the FindSegment tool to determine the distance (in pixels) across the block
- Fixture both vision functions to the Row, Column, and Angle returned by FindPatterns
- Use If functions to specify pass (1) or fail (0) for both tests

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- ExtractHistogram
- AcquireImage
- If
- Fixturing

Follow the steps below to complete the lab exercise:

1. Load **MyPatMaxPatterns** from the previous lab.
2. Save the job as **MyHistogram** in the folder that you created in Lab #2.

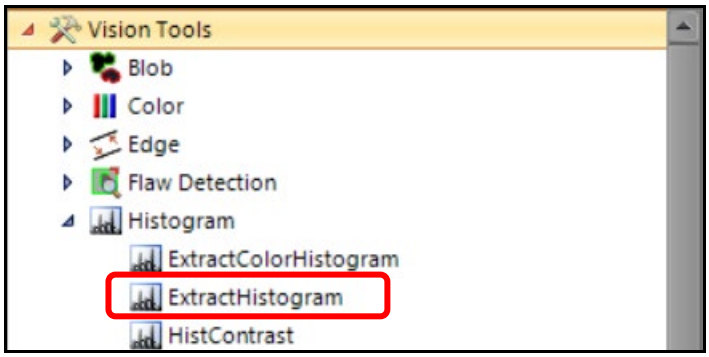
   NOTE: *You will analyze the area indicated in the image below.*

3. Enter the Comment **Check for Gouge** in cell **B14**.

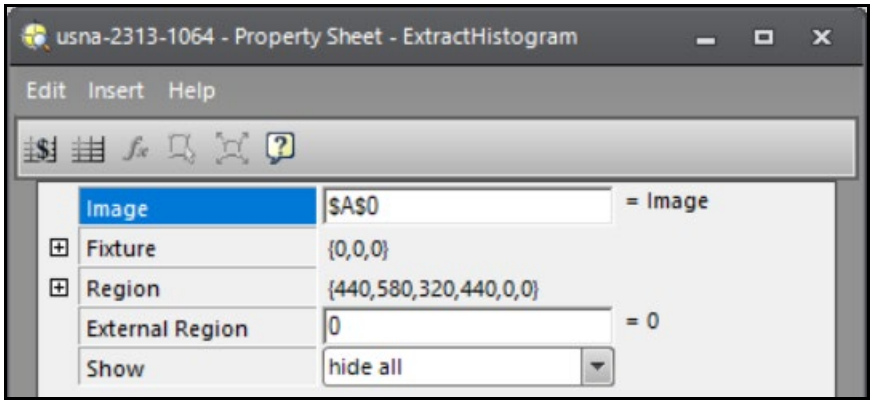
   NOTE: B*est practice is to enter the comments in as you go along.*

9								
10	**Find the Logo**							
11		Patterns	1.000					
12			Index	Row	Col	Angle	Scale	Score
13		Patterns	0.000	190.379	876.693	3.115	100.020	99.174
14	**Check for Gouge**							
15								

4. Insert an **ExtractHistogram** function into cell **C16** of the spreadsheet.

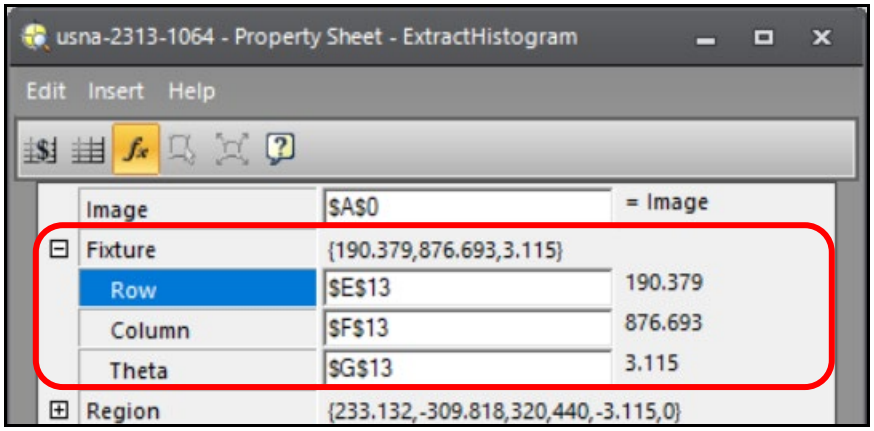


The **ExtractHistogram Property Sheet** displays.

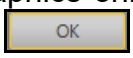


5. Fixture it to the Row, Column, and Angle reported by the FindPatMaxPatterns function by double-clicking on Row under fixture.
NOTE: *This is done by clicking the left mouse button and pulling across the results from your FindPatMaxPatterns tool that was created in the last lab (it will highlight with a red box) and clicking the* ***<Enter>*** *key.*

10		**Find the Logo**							
11			Patterns	1.000					
12				Index	Row	Col	Angle	Scale	Score
13			Patterns	0.000	190.379	876.693	3.115	100.020	99.174
14		**Check for Gouge**							



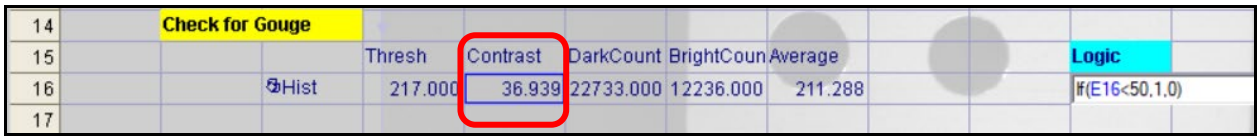
6. Double-click on the word **Region** in the Property Sheet and position the Region as shown below.

7. Set the **Show** parameter to input graphics only. This will allow you to always be able to see the Region. Click the **OK**  button.

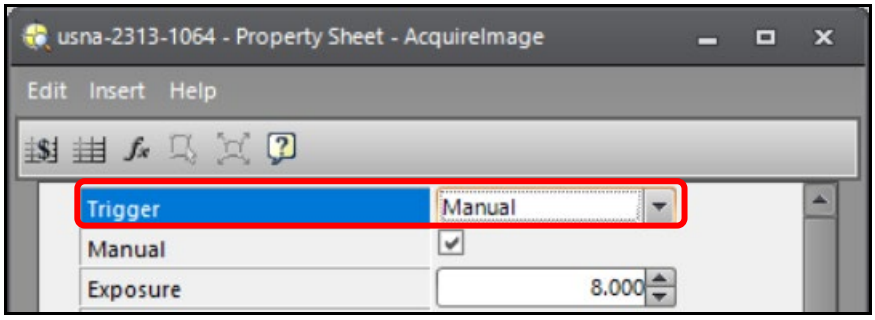
NOTE: *We will look at Contrast and Average as possible parameters to use for determining Pass/Fail.*

8. Write down the Contrast and Average values returned when there are no gouges (Good Block) and when there are gouges (Bad Block).
   No Gouges:
   Contrast = _____pixels Average = _____pixels
   Gouges:
   Contrast = _____pixels Average = _____pixels
9. Pick an appropriate threshold (limit) for it to distinguish between these two cases.
10. Enter a comment in cell **K15** that indicates that you are creating a logic statement.
11. Use the threshold limit that you determined above in an If function (under Mathematics → Logic) in cell **K16** that gives you a value of 1 for no gouges and 0 for gouges.

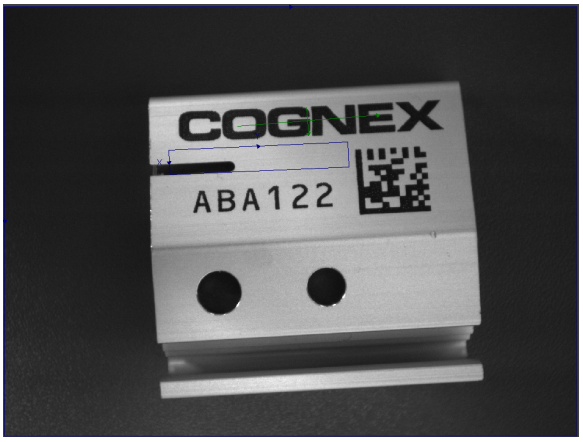
NOTE: *We will use this value later to generate color indicators, green for pass and red for fail.*



12. Confirm **AcquireImage** is still set in Manual Mode.



13. Move the block a little in the Field of View and trigger (F5). Repeat several times.
14. Verify that the Region for ExtractHistogram region follows the movement of the block. Try this with both the good and the bad block.



15. Check the value of the If function for a Good Block (1 = no gouges) and a Bad Block (0 = gouges present).

16. Save the job.

Lab Exercise 4.2 – FindSegment

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

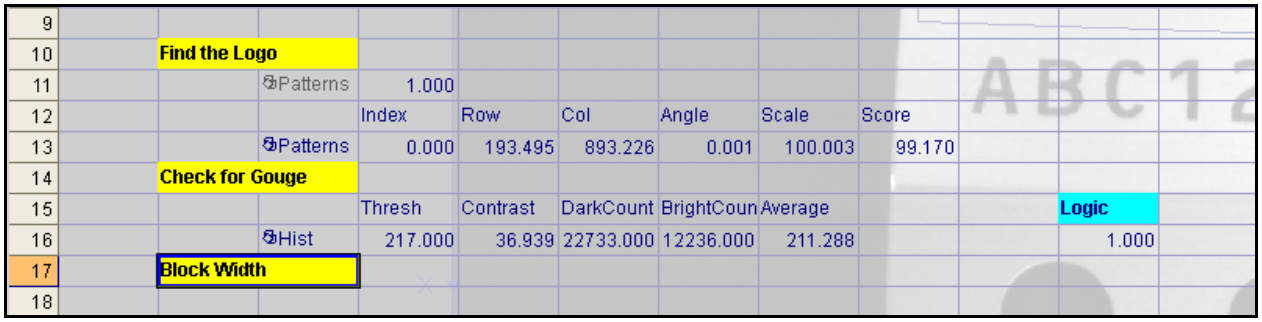
- Utilize the FindSegment tool to determine the distance (in pixels) across the block
- Fixture both vision functions to the Row, Column, and Angle returned by the FindPatMaxPatterns function
- Use InRange functions to specify pass (1) or fail (0) for both tests

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

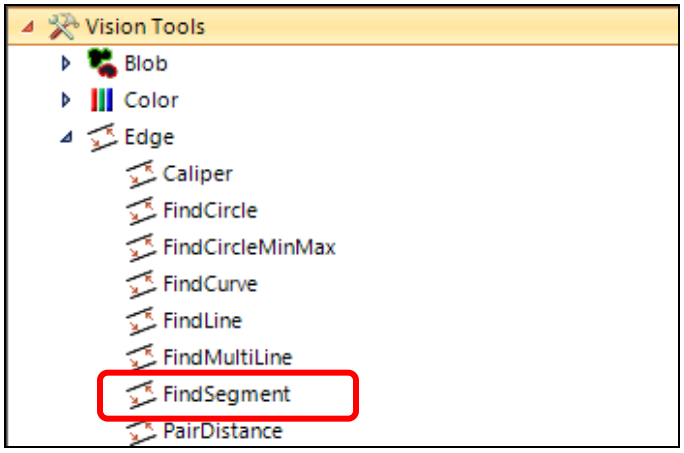
- FindSegment
- AcquireImage
- InRange

Follow the steps below to complete the lab exercise:

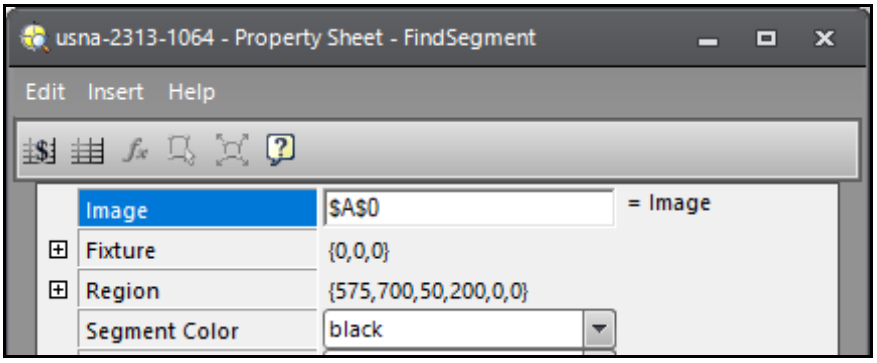
1. Continue with **MyHistogram** from the previous lab.
2. Save the job as **MyEdges** on the In-Sight camera and your own folder on the PC.
3. Enter the Comment **Block Width** in cell **B17**.



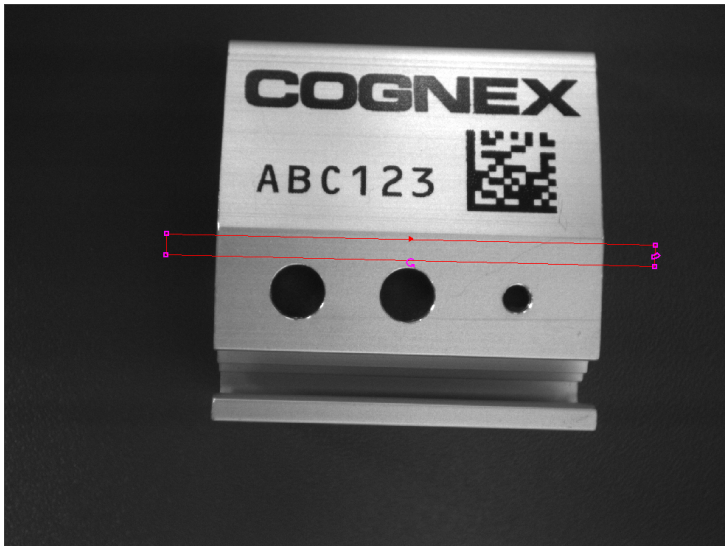
4. Insert a **FindSegment** function into cell **C19**.



The **FindSegment Property Sheet** displays.

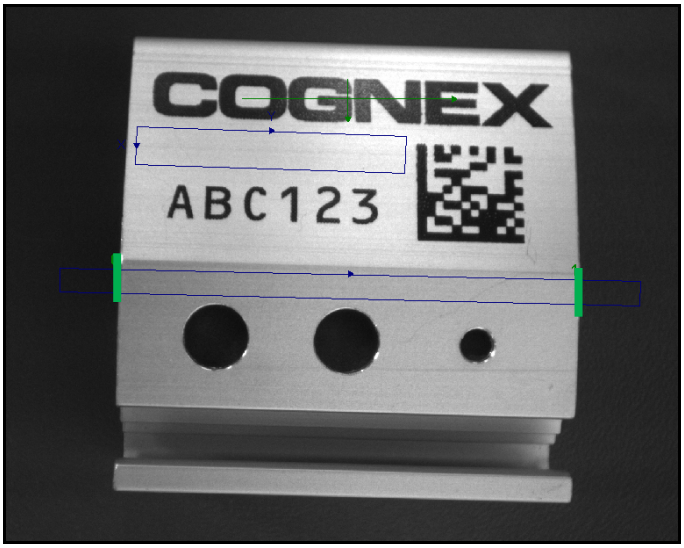


5. Configure the parameters of the FindSegment Property Sheet as follows:
   Fixture – Reference the Row, Column, and Angle returned by FindPatMaxPatterns (follow the same steps as in the Histogram lab)
   Region – Set its size to span the length of the block, and be perpendicular to the edges of the cutout, as shown below.
   Segment Color – The segment between these two edges is white, compared to the darker background of the block, so specify **white**.
   Find By – widest segment
   Angle Range – 5
   Edge Width - 6
   Show – input and result graphics
   Allow the remainder of the defaults to remain.
6. Click the **OK**  button.

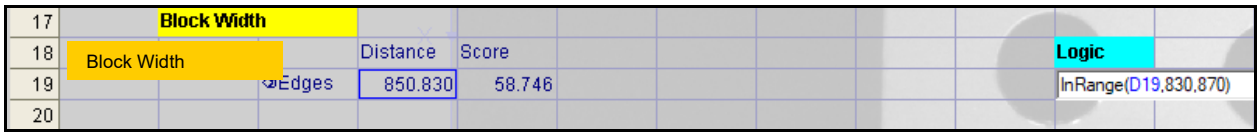


NOTE: *The direction of the red arrow needs to be perpendicular to the edge.*

7. Record the distance returned for a good block and a bad block:
   Correct Gap Width: _____



8. Pick an appropriate minimum and maximum tolerance for the gap width.
9. Enter a comment in cell **K18** that indicates that you are creating a logic statement.



10. Use the tolerance that you determined above in an **InRange** function (under Mathematics → Logic) in **K19** that gives a value of 1 for Pass, 0 for Fail.
11. Confirm **AcquireImage** is still set in Manual Mode.
12. Move the block around in the Field of View, triggering with <F5> each time you do.
13. Verify that the Region for FindSegment follows the movement of the block.
14. Check the value of the InRange function for a good block and a bad block.
15. Save the job.

Lab Exercise 4.3 – (if time allows)

What if you placed the Find Segment's Region across the holes on the block and now potential unwanted edges are being selected.

Which of the four choices for Find By in FindSegment's Property Sheet would be best to avoid misinterpreting the holes as an edge?

Try implementing two FindLine functions, one for each edge, to handle this situation.
HINT: *Direction of search is important here.*

Lab Exercise 5.1 – DetectBlobs

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

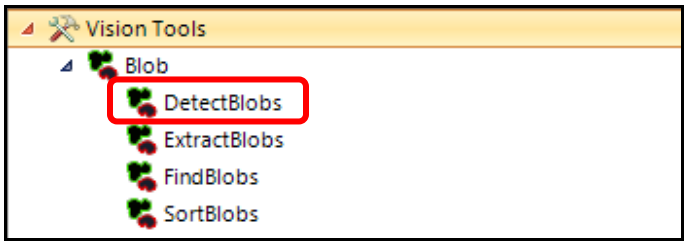
- Utilize DetectBlobs to check for size of holes

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

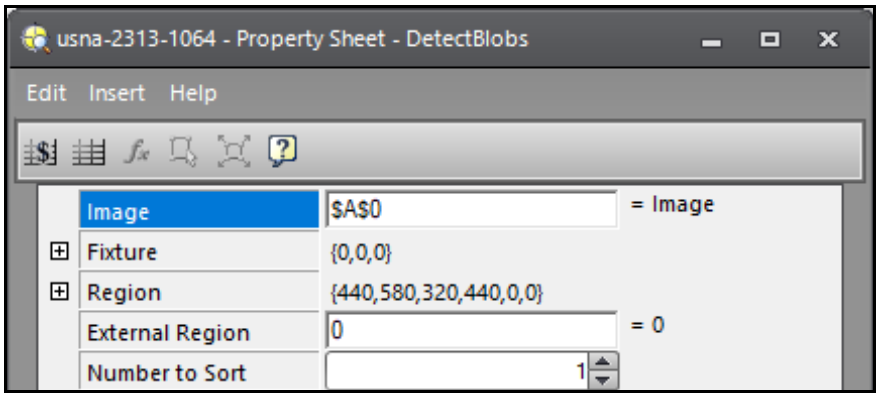
- DetectBlobs

Follow the steps below to complete the lab exercise:

1. Load **MyEdges** from a previous lab.
2. Save the job as **MyBlobs** in the folder that you created in Lab #1..
3. Enter the comment **Check Holes** in cell **B20**.
4. Insert a **DetectBlobs** function in cell **C22** in the spreadsheet.



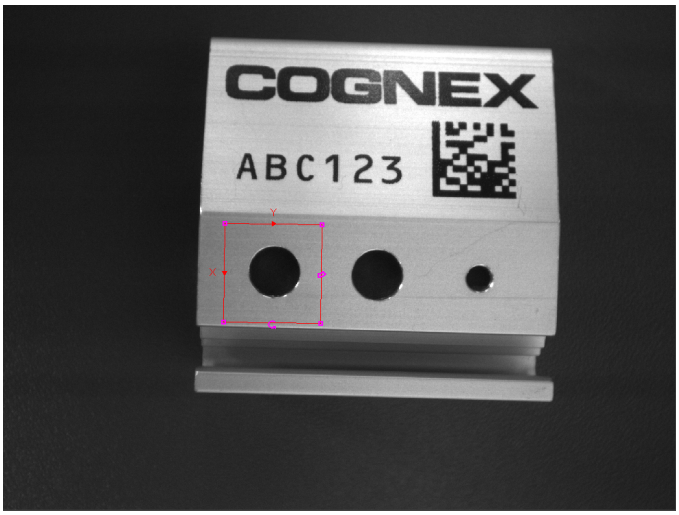
The **DetectBlobs Property Sheet** displays.



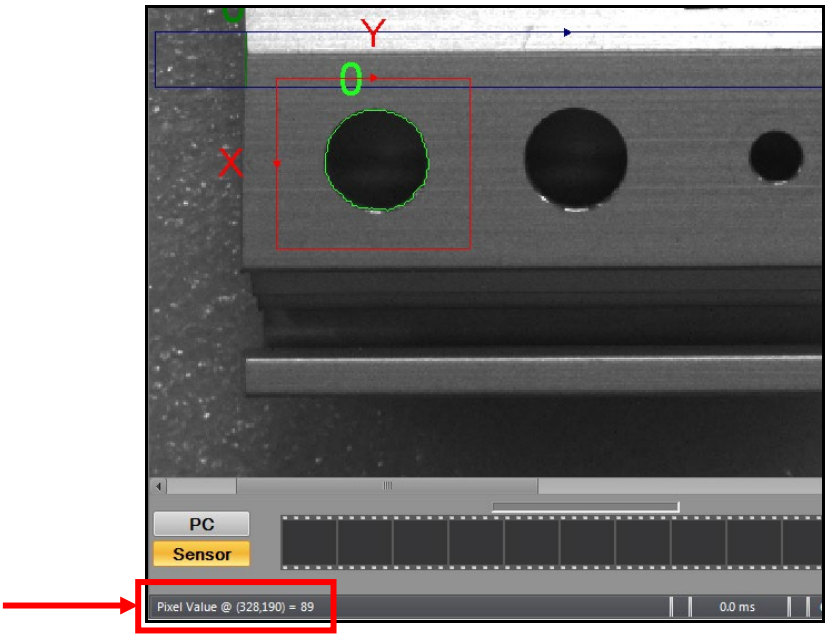
5. Fixture the tool to the same result from the FindPatMaxPatterns tool.

   NOTE: *Please refer to previous labs if you need assistance with fixturing.*

6. Set the Region to be a square around Hole A (see below).



7. Leave the **Number to Sort** = 1.
8. Determine the grayscale values of you blob (the hole) and your background (the block).
   NOTE: *This can be done by removing the Overlay. As your mouse moves across the image, the Row and Column results along with the grayscale value of the current pixel will be shown in the bottom left corner of the image.*

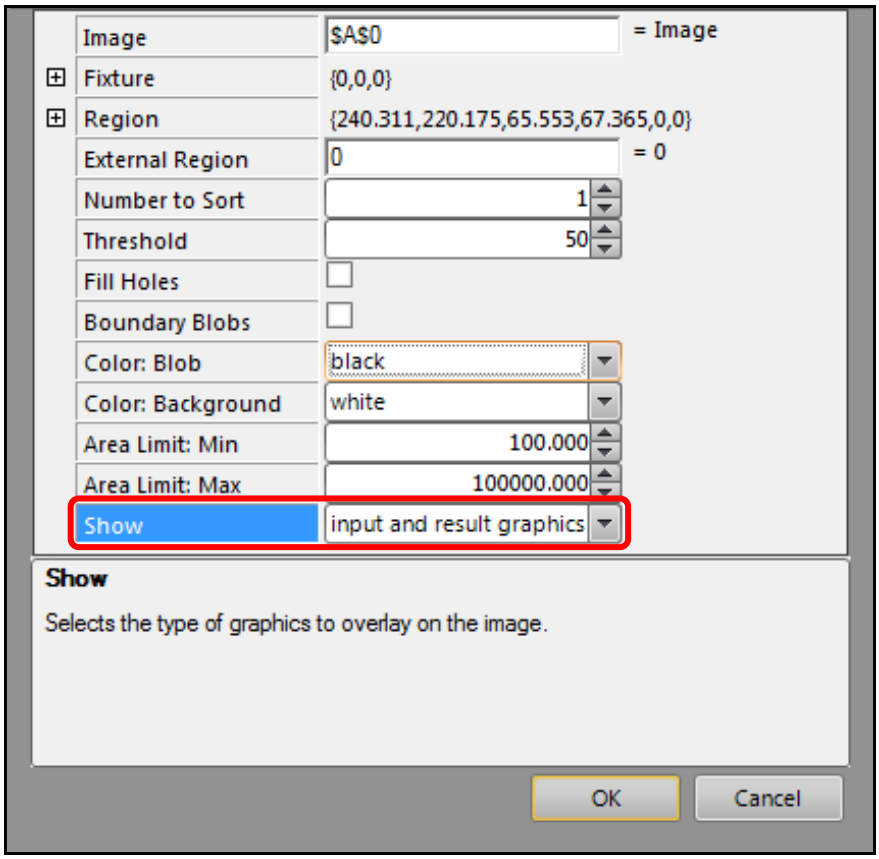


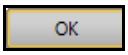
9. Write the approximate grayscale value of the following:

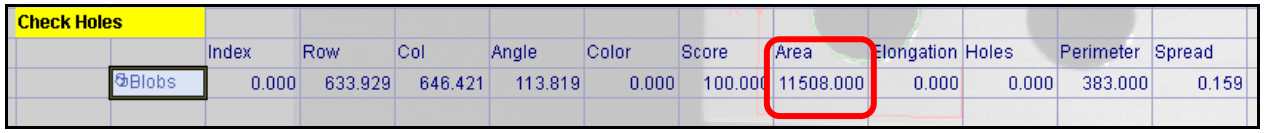
   Blob Grayscale:_____

   Background Grayscale: _____

10. Determine a good threshold value using the data from step 7.
    HINT: *Pick a value in between the blob grayscale and the background grayscale*.
11. Deselect **Boundary Blobs**.
12. Set the proper **Blob Color**.
13. Set the proper **Blob Background**.
14. Set the **Show** Parameter to **input and result graphics**.



15. Click the **OK**  button to finalize the DetectBlobs settings.
16. Notice the **Area** reported for a good hole.



17. Calculate what ± 10% of that area value should be:

    -10%: _____

    +10%: _____

18. Try this on a good block and a bad block. (In the next section, you will use the data calculated in the step above to set the proper tolerance for the hole.)

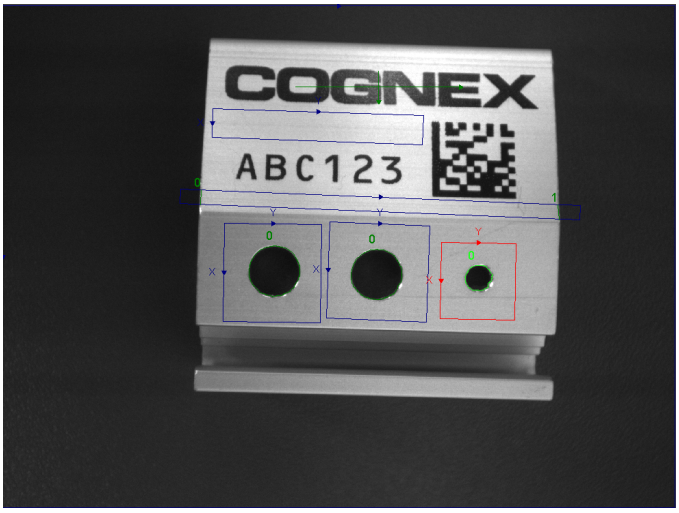
19. Repeat steps 3 – 17 for the other two holes (different spreadsheet cells) and write the 10% limits here:

	Middle hole:	Right hole:
-10%:	_____	_____
+10%:	_____	_____

Your spreadsheet should look similar to this:

Check Holes												
		Index	Row	Col	Angle	Color	Score	Area	Elongation	Holes	Perimeter	Spread
	Blobs	0.000	633.929	646.421	113.819	0.000	100.000	11508.000	0.000	0.000	383.000	0.159
	Blobs	0.000	642.093	889.365	114.720	0.000	100.000	11532.000	0.000	0.000	388.000	0.159
	Blobs	0.000	650.230	1131.448	126.745	0.000	100.000	3111.000	0.000	0.000	206.000	0.160

And the image should look like this:

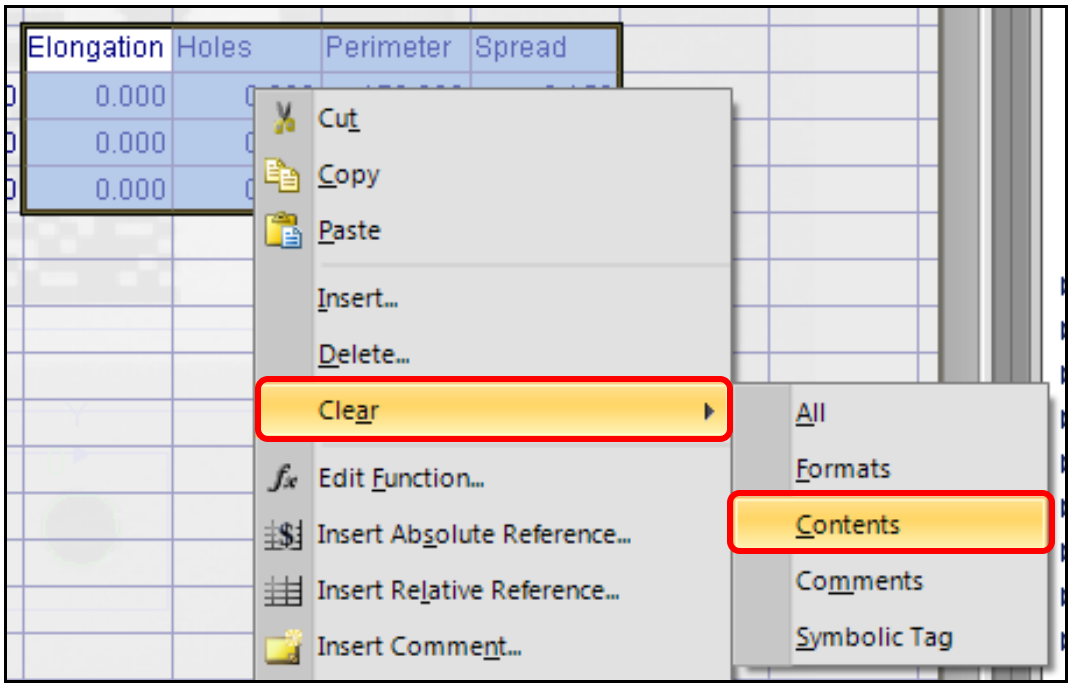


20. In order to reduce unused results from the spreadsheet and make it more readable, you can remove the results for Elongation, Holes, Perimeter, and Spread by selecting cell **K21**, keeping the left mouse button depressed, and moving down to cell **N24**.

Check Holes												
		Index	Row	Col	Angle	Color	Score	Area	Elongation	Holes	Perimeter	Spread
	Blobs	0.000	633.929	646.421	113.819	0.000	100.000	11508.000	0.000	0.000	383.000	0.159
	Blobs	0.000	642.093	889.365	114.720	0.000	100.000	11532.000	0.000	0.000	388.000	0.159
	Blobs	0.000	650.230	1131.448	126.745	0.000	100.000	3111.000	0.000	0.000	206.000	0.160

21. Right click and select **Clear → Contents** from the menu.

    NOTE: *This is the same as <Delete> on your keyboard.*



22. Save the job.

Lab Exercise 5.2 – Snippets

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

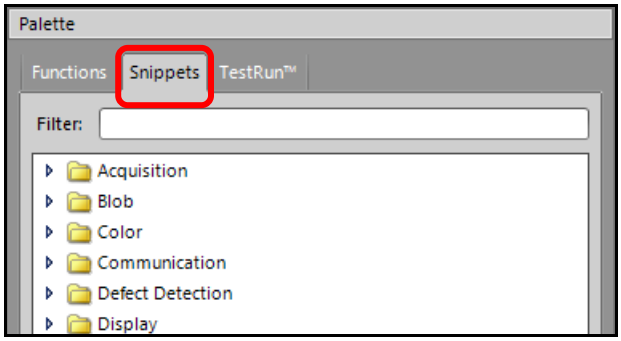
- Utilize Snippets to quickly create tolerances and graphics

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

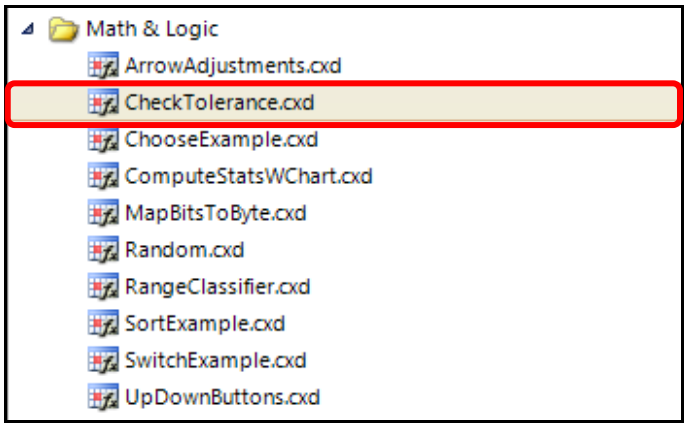
- CheckTolerance snippet to check for pass/fail
- Use a snippet to check for pass/fail and display color indicators

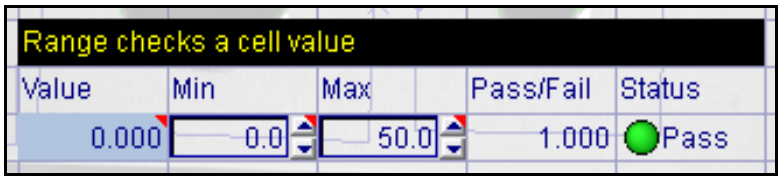
Follow the steps below to complete the lab exercise:

1. Continue with **MyBlobs**.
2. Save the job as **MySnippet** in the folder that you created in Lab #1.
3. Click on the **Snippets** tab in the Palette on the right side of the In-Sight Explorer interface.



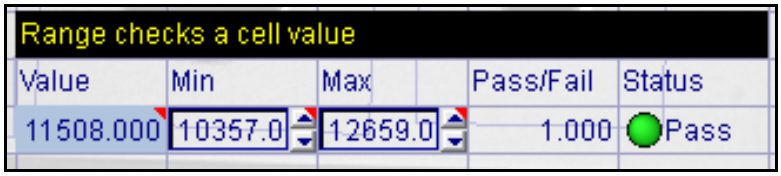
4. Insert a **CheckTolerance.cxd** Snippet (under Math & Logic) into cell **L20**.
   NOTE: *This is one row higher than you may think, but it is to accommodate the two rows of headers in this snippet.*



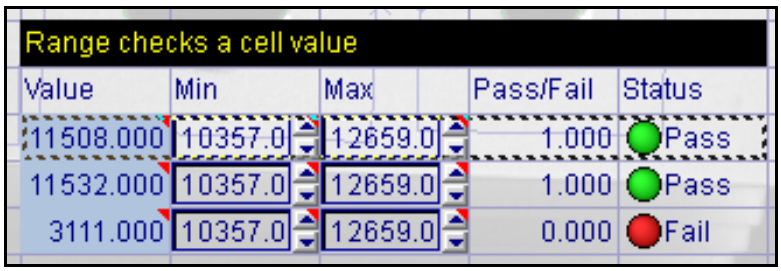


5. Double-click in cell **L22** and have it relative reference the result of the first blob area in cell **J22**.

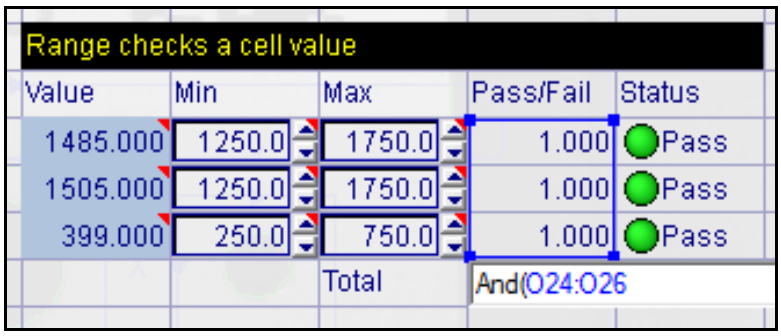
6. Set the Min and Max values to the -10% and +10% values you calculated earlier in this lab.



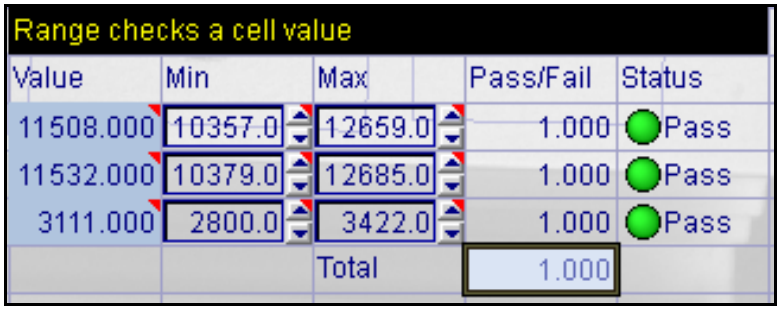
7. Copy and paste the single row of cells **L22 – P22** into cells **L24** and **L25**.



8. Tweak the Min and Max values to 10% for the remaining blobs.



9. In cell **O25**, insert an And statement to determine if all blobs had passed (don't forget to add a comment in cell **N25**).



10. Save the job.

Lab Exercise 5.3 – Image Tools

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

- Use an Image Tool to improve an image for inspection by a vision tool

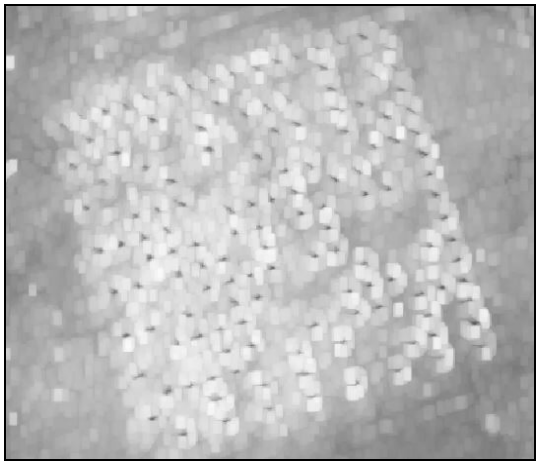
The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- Use the **Erode** filter operation to improve a degraded image of a Data Matrix code, and then read the filtered image using ReadIDMax

Follow the steps below to complete the lab exercise:

1. Connect to your Emulator and make sure you are emulating a standard resolution model – one whose model number ends in 00, for example, 5400. Be sure you have saved the job from the previous lab, and then start a new job.
2. Find the Student Folder on your desktop (It will be named **IS_Student** or **Student** or something similar).
   Navigate through subfolders named **Classes→ In-Sight Spreadsheet Standard→Resources→Images** (or something similar). Then drag an image file named **DegradedDataMatrix** image into the spreadsheet pane.

   This is a very degraded code:

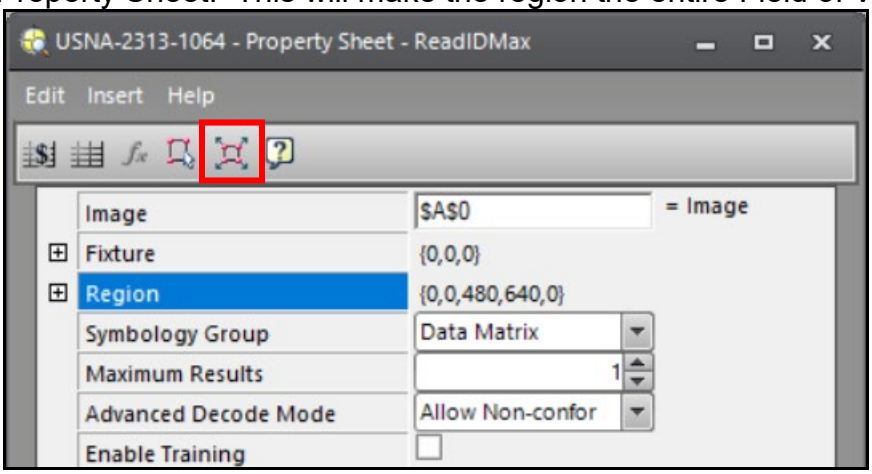


3. Enter the comment **Read Data Matrix code** in cell **B2.**
4. Skip a row and enter a **ReadIDMax** tool into cell **C4**.

   NOTE: *The tool is found in the Palette under **Vision Tools → ID → ReadIDMax**.*

   The ReadIDMax Property Sheet displays.

5. Click the **Region** parameter and then click the **Maximize Region** button at the top of the Property Sheet. This will make the region the entire Field of View:

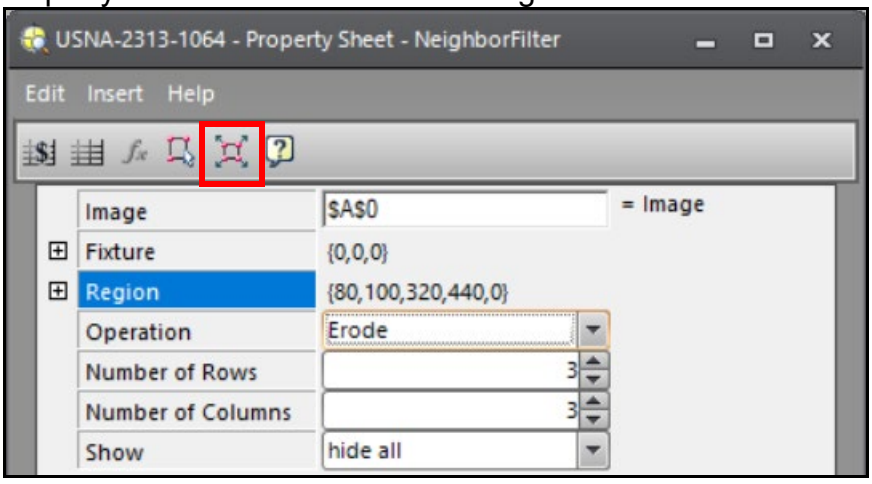


6. In the Property Sheet, set parameters as follows:
   - **Symbology Group** = *Data Matrix*
   - **Advanced Decode Mode** = *Allow Non-conformant Modules*
   - Leave other parameters at their defaults.
7. Click the **OK** button to close the Property Sheet.
   The tool should **not** be able to decode the degraded image:

Next, we are going to create a better image using the filter type called **Erode**. Then we will change **ReadIDMax** so that it references the filtered image.

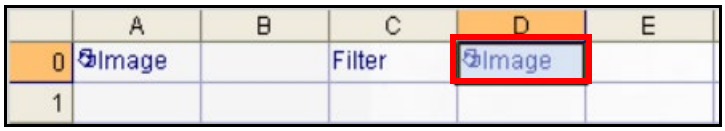
9. Enter the comment **Filter** in cell **C0**.
10. Enter a **NeighborFilter** tool in cell **D0**.

    NOTE: *The NeighborFilter tool is found in the Palette under **Vision Tools → Image → NeighborFilter.***
11. Click the **Region** parameter and then click the **Maximize Region** button at the top of the Property Sheet. This will make the region the whole Field of View.

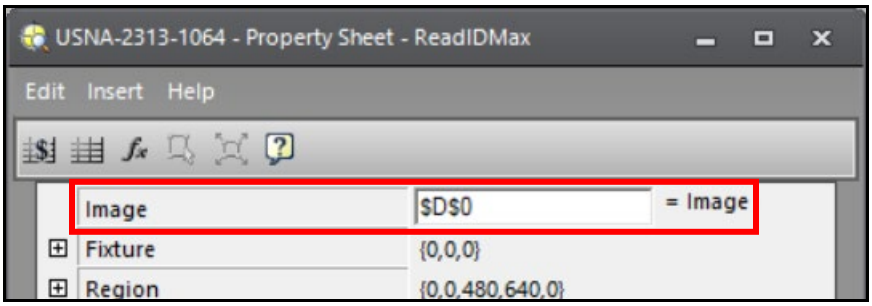


12. In the Property Sheet, set the parameters as follows:
    - **Operation** = *Erode*
    - **Kernel Rows** = *3*
    - **Kernel Columns** = *3*
    - Leave other parameters as their defaults.
13. Click the **OK** button to close the Property Sheet.

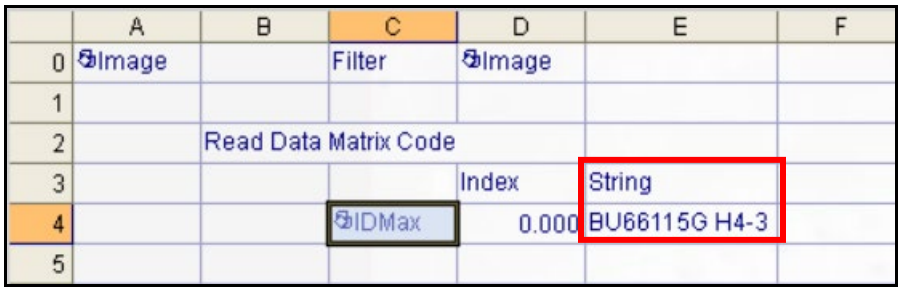
    The Filtered Image is entered in the spreadsheet.



14. Open the **ReadIDMax** tool and direct the **Image** parameter to the filtered image in cell **D0**. Click the **OK** button to close the Property Sheet.



    The Data Matrix string displays.



15. The choice of kernel size (Kernel Rows and Kernel Columns) can affect whether a tool is successful. In the above example, we left the kernel size at the default value (3x3) and **ReadIDMax** was successful.

    Try a kernel size of 5x5. Is **ReadIDMax** successful?

    Try a kernel size of 15x15. Is **ReadIDMax** successful?

16. We do not use this job in subsequent labs, so there is no need to save it.

Exercise 5.4 – (if time allows)

Try other Filter Types such as Binarize, Clip, Stretch, Erode, and Dilate.

Lab Exercise 6.1 – Error Handling

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

- Utilize the ErrFree tool to ensure that the final tool result is free of all errors

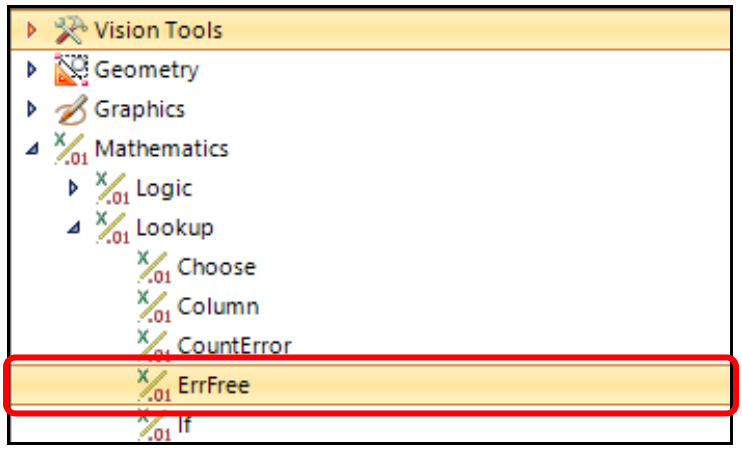
The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- ErrFree

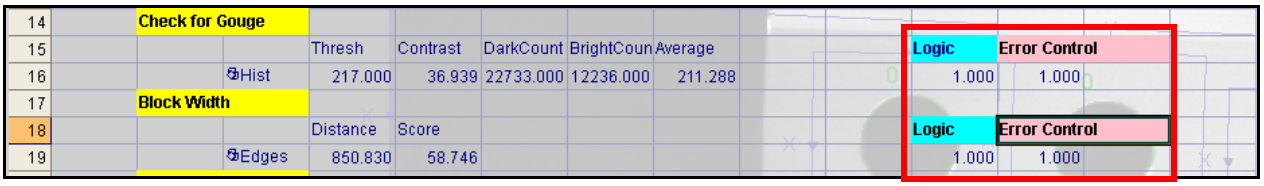
Follow the steps below to complete the lab exercise:

1. Continue with the **MySnippet** job.
   NOTE: *For the Logic statements that we created throughout the spreadsheet job, we need to ErrFree each one so that the result never goes to #ERR as Output functions will not know what to do with that result.*
2. Save the job as **MyErrorHandling** in the folder that you created in Lab #1.
3. Enter the comment **Error Control** in cell **L15**.

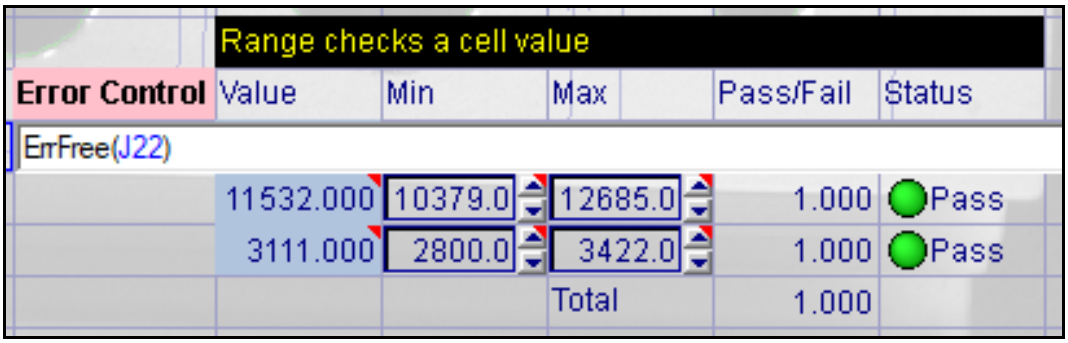
4. Insert an **ErrFree** function into cell **L16** (Histogram result) that references the Logic result in cell **K16**.



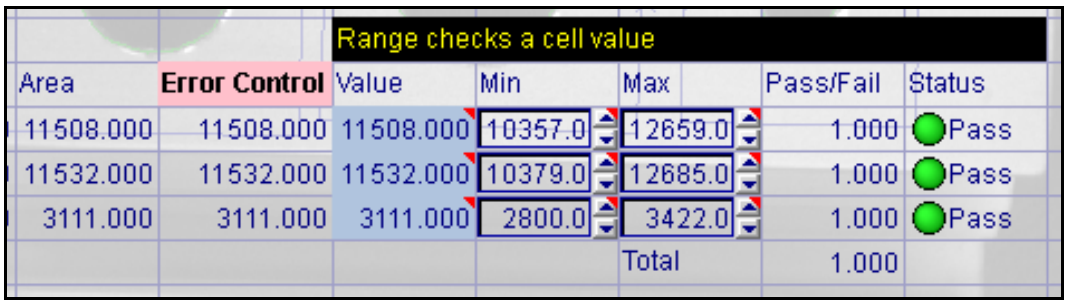
5. Repeat step 3 for the Edge tool result.



6. To remove the #ERRs in the Blob tool, you will ErrFree the Area results.
7. Enter the comment **Error Control** in cell **K21**. You may need to make the column a little wider.
8. Insert an **ErrFree** function into cell **K22** that references cell **J22 (**the result of the Area from the first blob).



9. Insert an **ErrFree** function into cell **K23** that references cell **J23 (**the new ErrFree value).
10. Enter an ErrFree function for the final blob area result.



11. Test with the good and bad blocks to ensure that no #ERRs are propagating through the final tool result.
12. Save the job.

Lab Exercise 6.2 – Cell State

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

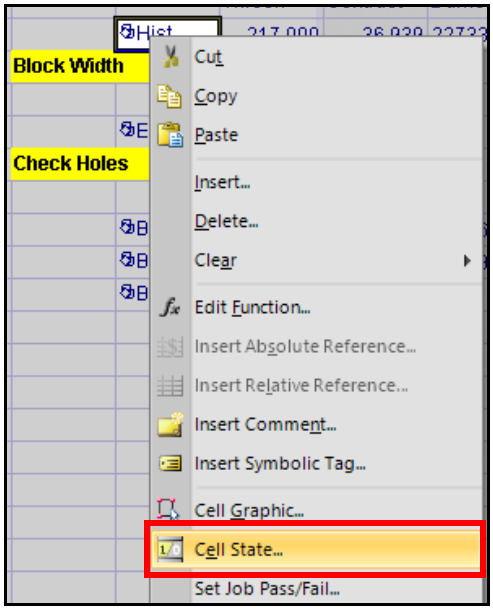
- Integrate error handling and proper use of cell state

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

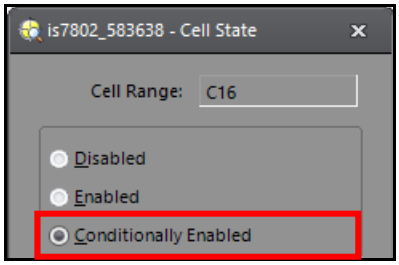
- Cell State

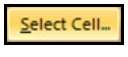
Follow the steps below to complete the lab exercise:

1. Continue with the **MyErrorHandling** job.
2. Save the job as **MyCellState** in the folder that you created in Lab #1. (You will not use the MyCellState.job again until section 6.4.)
3. Select the **Hist** tool and right click. Select **Cell State** from the fly out list.



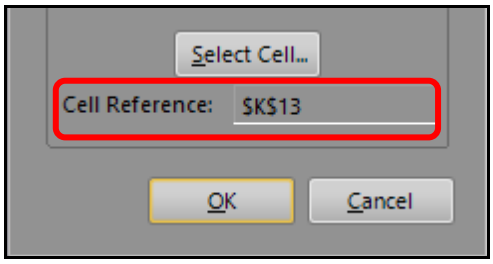
The **Cell State** dialog displays.



4. Select **Conditionally Enabled** and then click the **Select Cell**  button.

5. Select cell **K13**, the PatMax tool logic statement and click **<Enter>**.

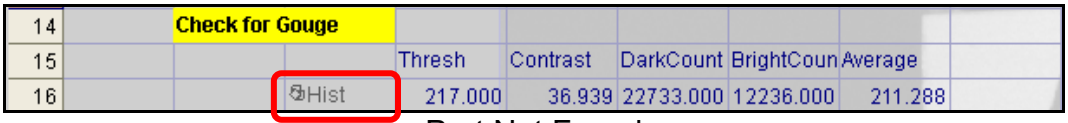
The absolute reference to cell **K13** will display in the **Cell Reference** field.



6. Click the **OK**  button.

Notice the cell when the block is found and when it is not found.

Part Found



Part Not Found

7. Repeat the same process to control the cell state for the FindSegment and DetectBlobs tools.
8. Save the job.

Lab Exercise 6.3 – Dependencies Viewer

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

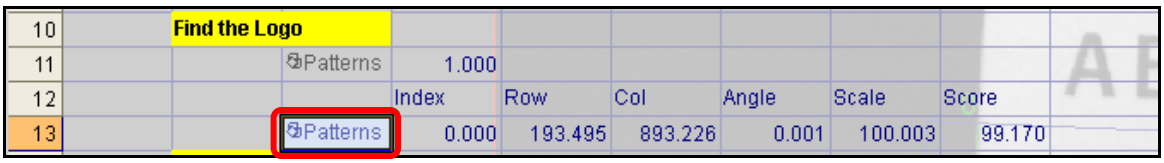
- Explain how to view multiple levels of dependencies within the spreadsheet

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

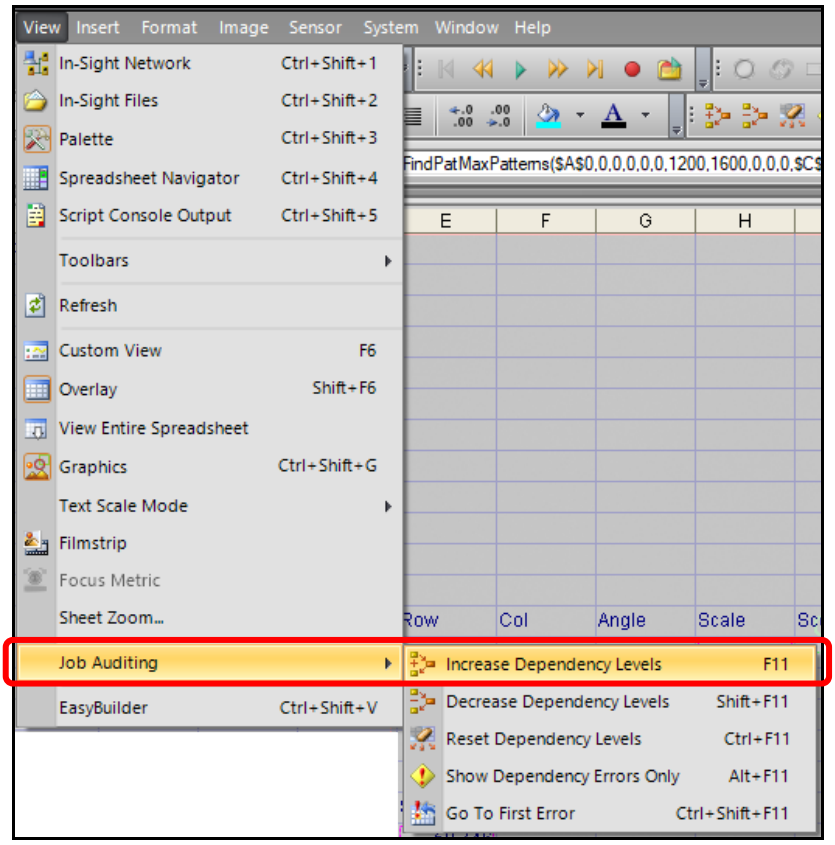
- Dependencies

Follow the steps below to complete the lab exercise:

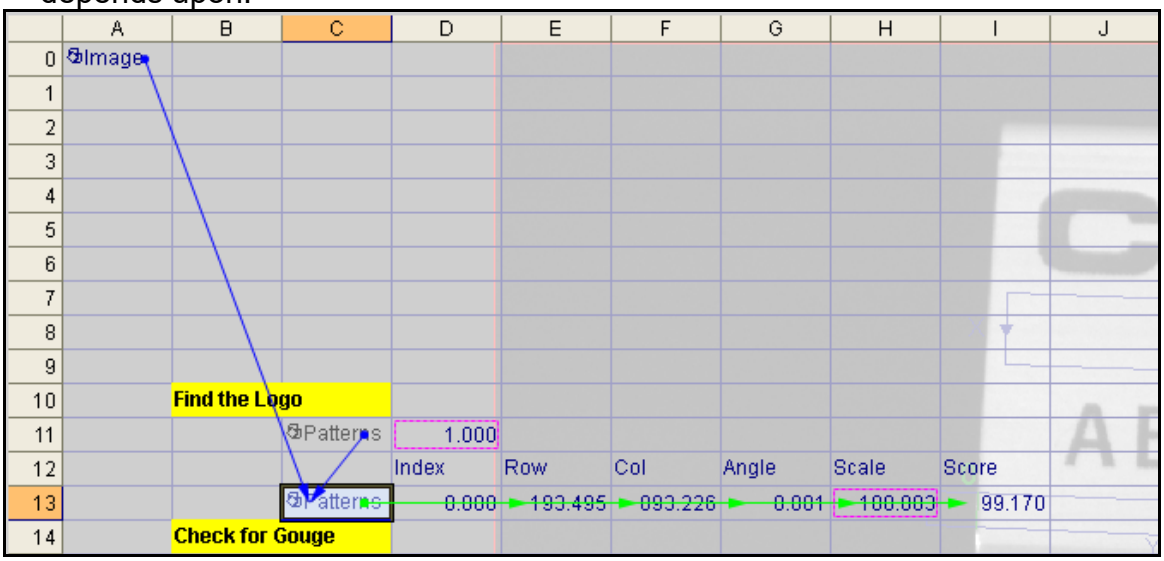
1. Continue with the **MyCellState** job.
2. Highlight the **FindPatMaxPatterns** structure.



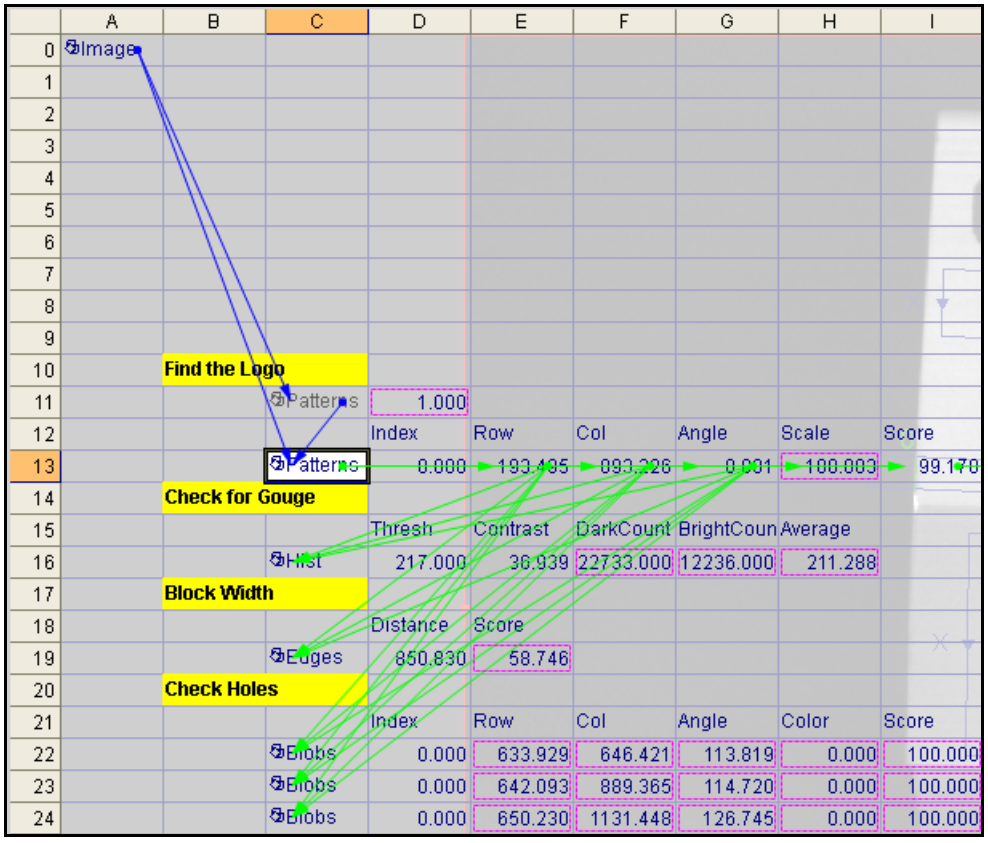
3. Click the **Show Dependency Levels Increase** button.
   Or: Click **View → Job Auditing → Increase Dependency Levels**.



4. Notice the graphics showing which cells depend on the FindPatMaxPatterns structure (those in green) and which cells the FindPatMaxPatterns (those in blue) depends upon.



5. Click the **Show Dependency Levels Increase** button again.
   A second level of dependencies displays.



6. Click the **Show Dependency Levels Reset** button to remove the dependency arrows. *Do not save this job*.

Lab Exercise 6.4 – Calibration

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

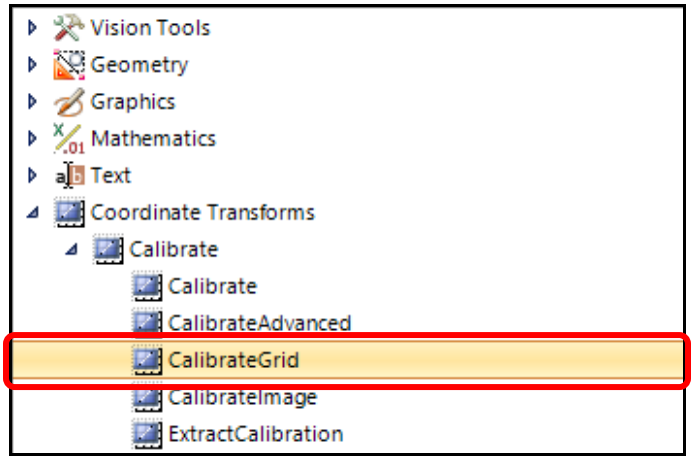
- Utilize the CalibrateImage to transform the pixel locations of the Image **A0** to calculate the real world positions
- Use the image reference in the Edges (FindSegment) structure to convert the width of the block from the previously found distance to millimeters

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

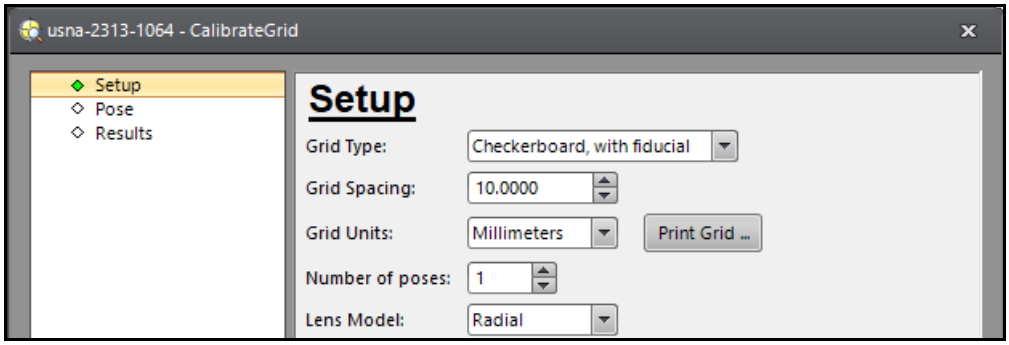
- CalibrateGrid
- CalibrateImage

Follow the steps below to complete the lab exercise:

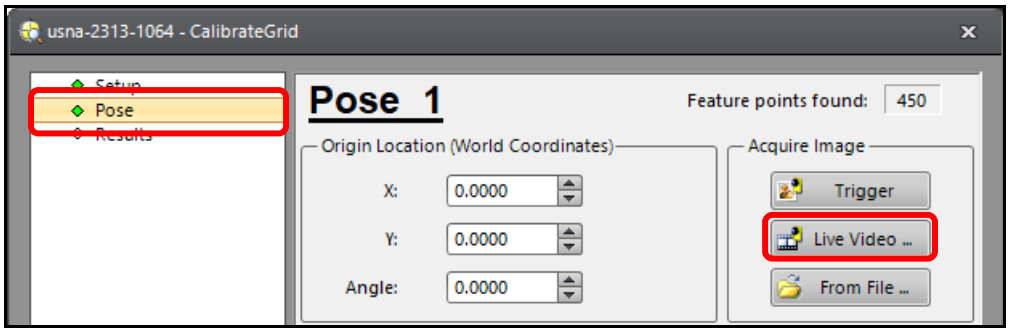
1. Open the **MyErrorHandling** job.
2. Your instructor will provide a calibration grid. Position the calibration grid under the camera at the same distance to the lens as the block.
3. Save the job as **MyCalibration** in the folder that you created in Lab #1.
4. Enter the comment **Calibration** in cell **B26** and insert a **CalibrateGrid** function into cell **C27.**

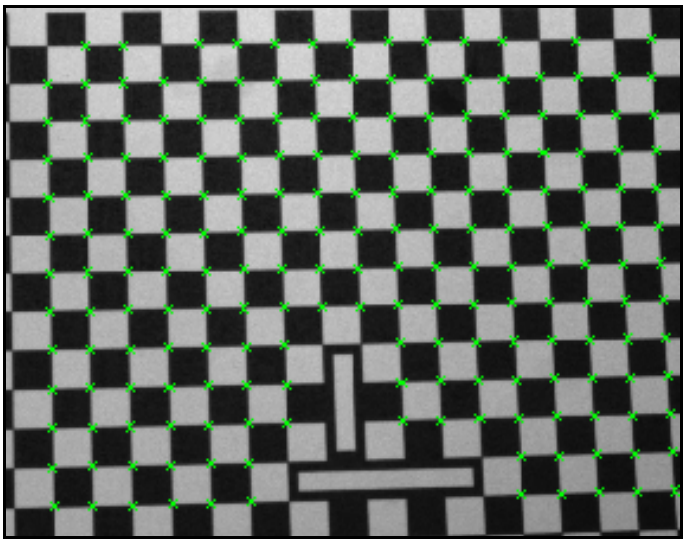


The **CalibrateGrid Wizard** displays.

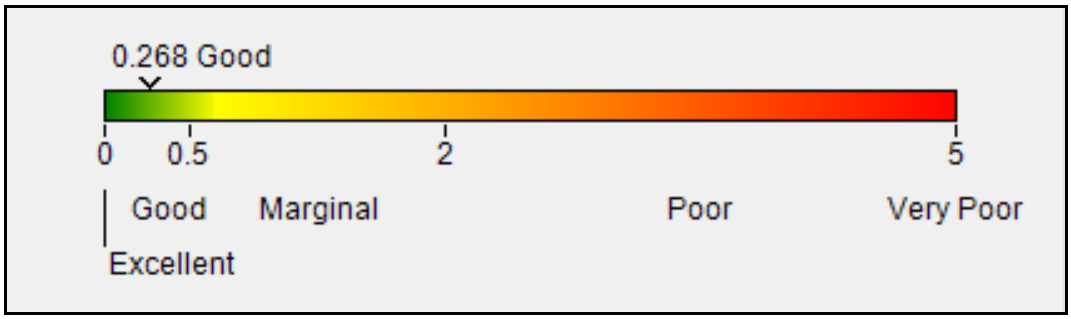


5. Configure the **Setup** step. The instructor will tell you the Grid Spacing, usually 5 mm.
6. Select **Pose** – acquire an image of the calibration plate using **Live Video**. Once you are happy with the image, click anywhere in the image to stop **Live Video**, then click the **Calibrate** Calibrate button.



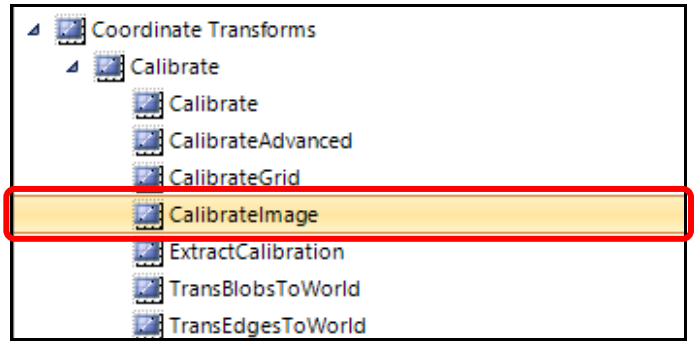


7. Select **Results** – the quality of the calibration is returned. Once complete click the **OK** OK button.

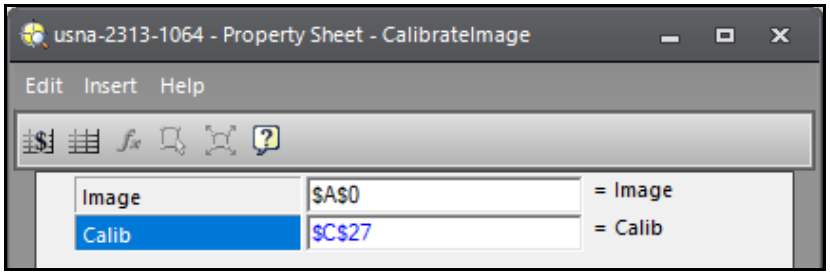


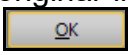
8. Remove the calibration grid and return the block under the camera.

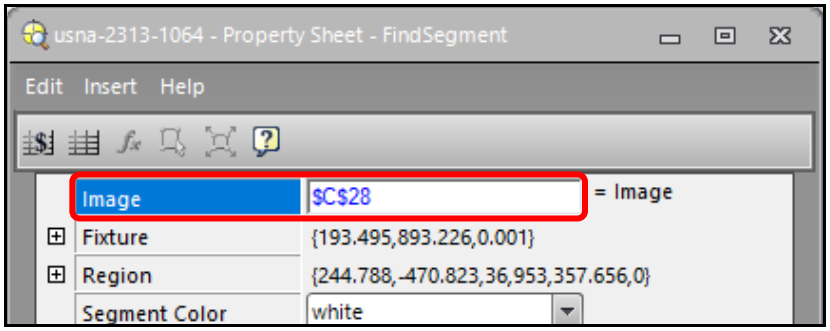
9. Insert a **CalibrateImage** function into cell **C28** to create an image based on real world units.



The **CalibrateImage Property Sheet** displays.

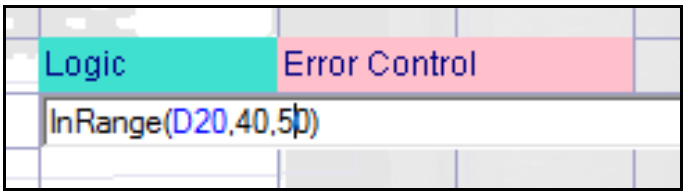


10. The CalibrateImage must reference the original image cell **A0** and the CalibrateGrid cell **C27**.
11. In the original FindSegment (cell **C19**) – change the reference to the CalibrateImage cell **C28** instead of the original image cell of A0 to determine the gap width in millimeters and click the **OK**  button.



12. The value is returned in millimeters, does this make sense in terms of your calibration?

13. Adjust your logic statement for the gap in cell **K19** to account for the new results.



14. Save the job.

Lab Exercise 7.1 – Discrete I/O – Input

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

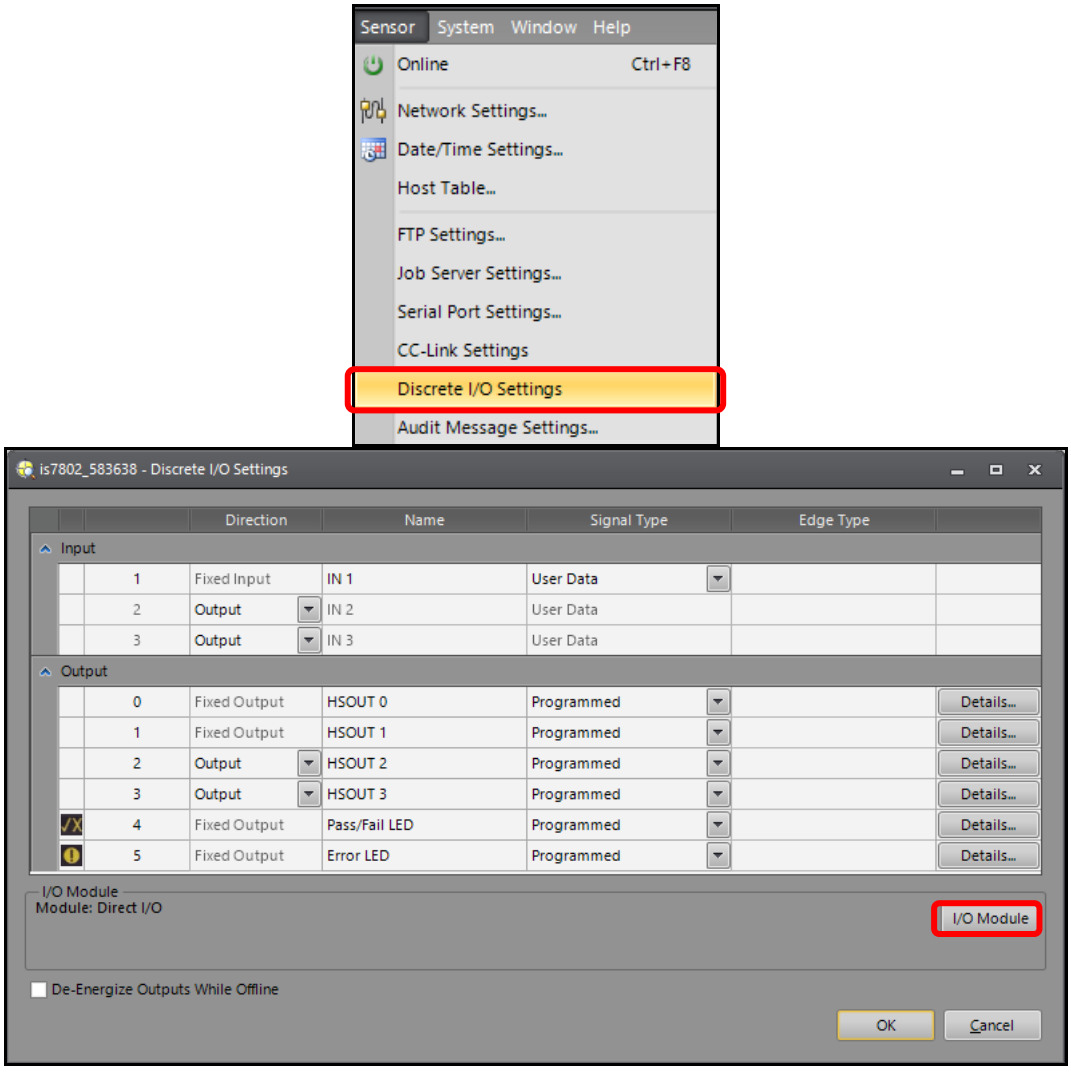
- Use Input0 (or Input1) to trigger an asynchronous event
- Create WriteDiscrete functions to signal pass or fail over a discrete output line

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- Event
- Count

Follow the steps below to complete the lab exercise:

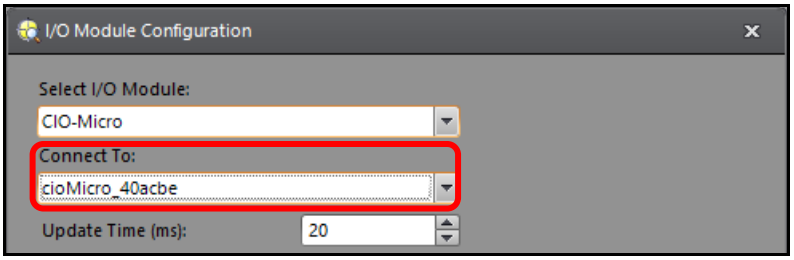
1. Open the **MyCalibration** job from the last lab exercise.
2. Save the job as **MyInput** in the folder that you created in Lab #1..
3. Go to **Sensor → Discrete I/O Settings** to confirm the appropriate I/O expansion module is selected.



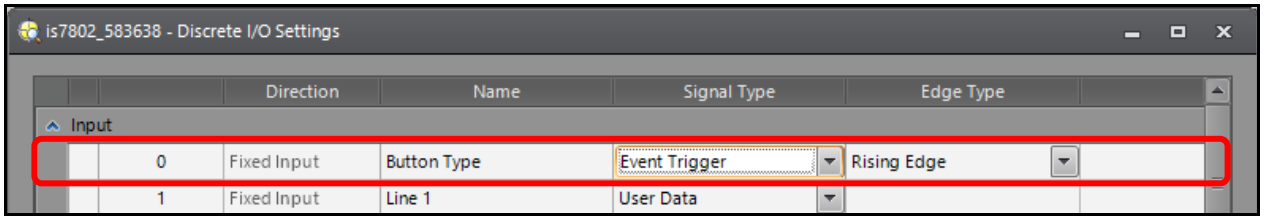
The **I/O Module Configuration** dialog displays.

NOTE: *Select the **CIO-Micro** for all cameras except the IS5000*

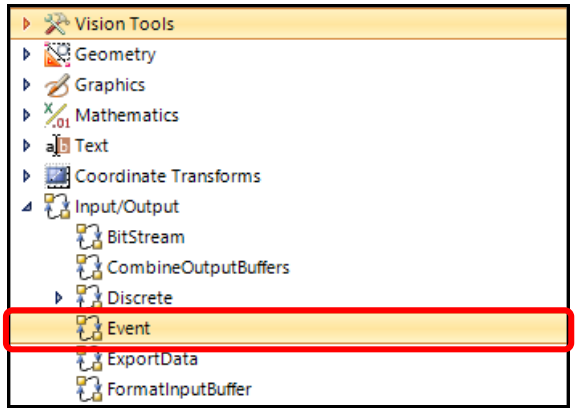
4. Select the CIO-Micro model in the **Connect To** field and click the **OK** button.



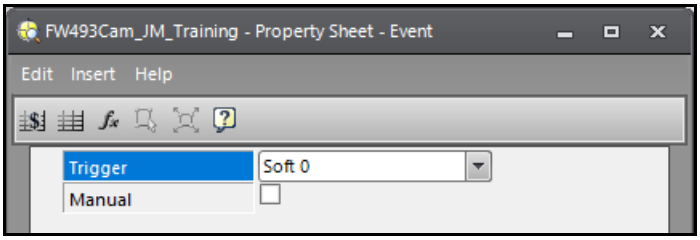
5. Go to your Sensor's Input Settings and change the name for Input Line 0 to **Button Push**, the Signal Type to **Event Trigger** and the Edge Type to **Rising Edge** and click the **OK** button.



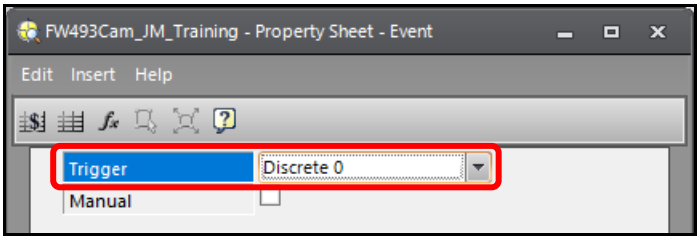
6. Enter the Comment **Count Button Pushes** in cell **B29**.
7. Insert an **Event** function into cell **C30** of the spreadsheet.



The **Event Property Sheet** displays.

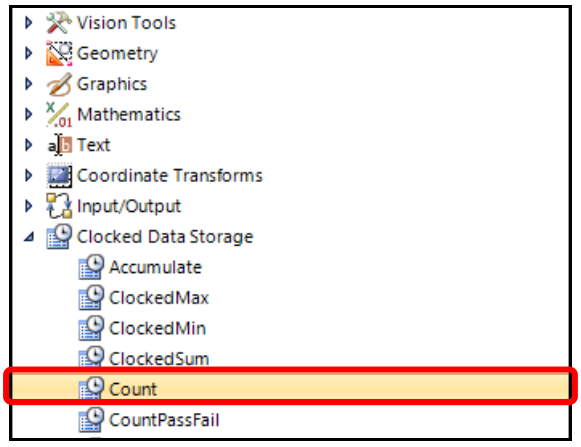


8. Select **Discrete 0** as the Trigger and click the **OK** button.

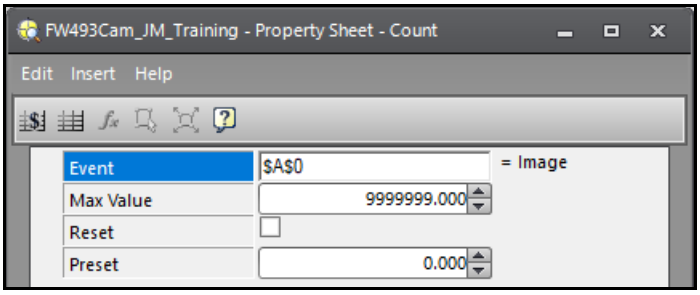


The Event will now trigger every time a signal is detected on Discrete Input Line 0.

9. Insert a **Count** tool into cell **D30** of the spreadsheet.

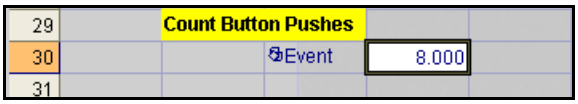


The **Count Property Sheet** displays.



10. Set the **Event** to reference the Event in cell **C30** and click the **OK** button.

11. Go Online and notice the Count tool changes as you press the button connected to the I/O module.



NOTE: *The 24 VDC enters input line 0, input line 0 triggers the Event, and the Event then activates the Count.*

12. Go Offline and save the job.

Lab Exercise 7.2 – WriteDiscrete

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

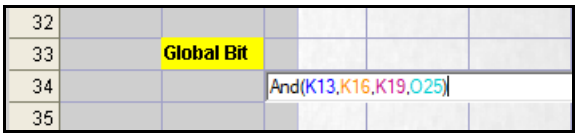
- Create WriteDiscrete functions to signal pass or fail over a discrete output line

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

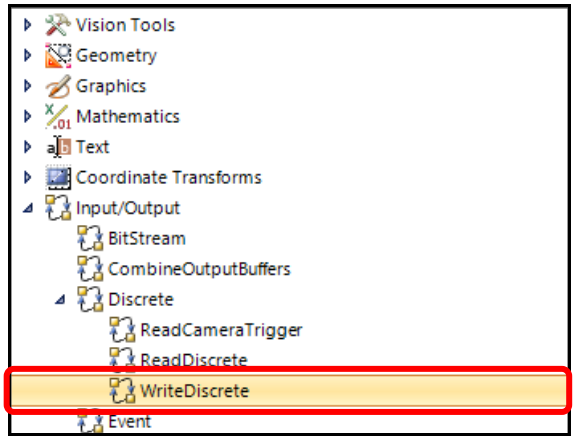
- Global Bit
- WriteDiscrete

Follow the steps below to complete the lab exercise:

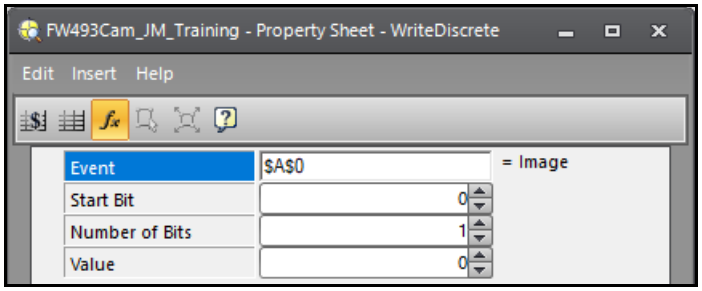
1. Continue with the **MyInput** job.
2. Save the job as **MyOutput** in the folder you created in Lab #1.
3. Enter the comment **Global Bit** into cell **B31.** Then insert an **AND** function in cell **C32** that references all of the tool results.



4. Enter the comment **Write Discrete** into cell **B33**. Insert a **WriteDiscrete** function into cell **C34** of the spreadsheet.

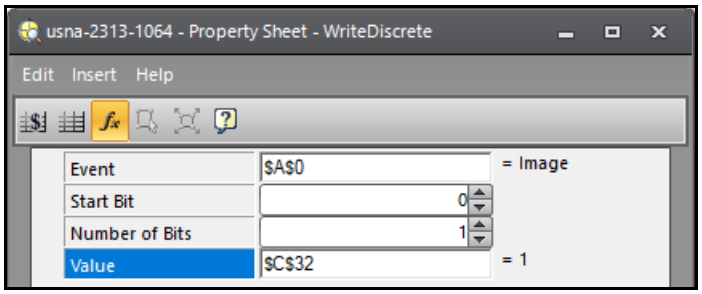


The **WriteDiscrete Property Sheet** displays.



5. Set the **Start Bit** to 0.

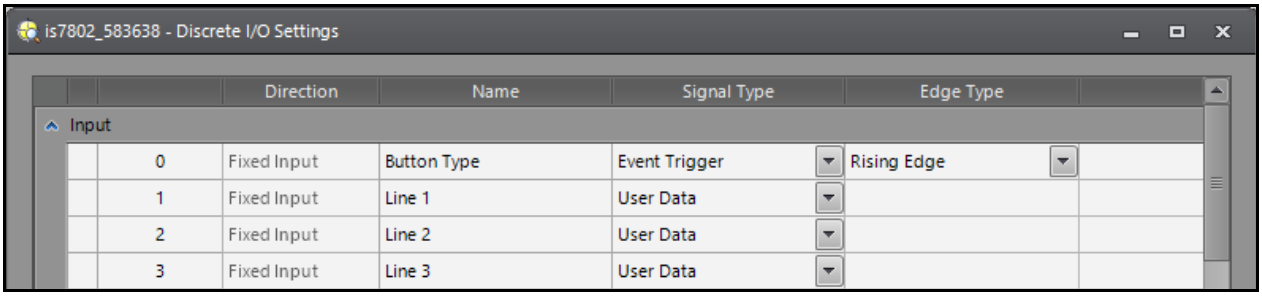
6. Set the **Number of Bits** to 1.
7. Reference the **Value** parameter to the logic that determines a Pass or Fail for the part (toggles between 0 and 1). This is cell **C32**.



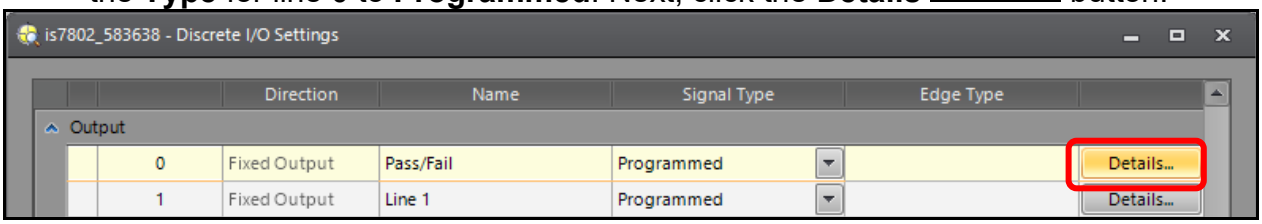
NOTE: *WriteDiscrete will not show the current value of the logic until the system is online and triggered. So its value might not match the Global Bit until then.*

8. Click the **OK** OK button.
9. Go to **Sensor → Discrete I/O Settings** to set the pulse duration.

   The **Discrete I/O Settings** dialog box displays.

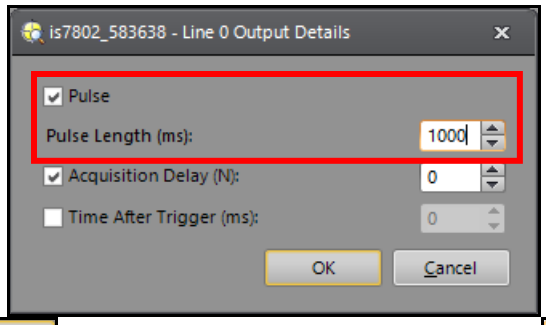


10. Scroll down to the Output portion of the dialog. Change **Name** to Pass/Fail and set the **Type** for line 0 to **Programmed**. Next, click the **Details** Details... button.



   The **Line 0 Output Details** displays.

11. Check the **Pulse** checkbox and set the **Pulse Length** to 1000 ms (1 second).



12. Click the **OK** OK button *twice* and click the **Online** button to go Online.

13. Place the *good* block under the camera and do a manual trigger. As you do, watch the LEDs on the I/O Expansion module. OUT 0 should go on for a second.
    NOTE: *The results displayed by the WriteDiscrete function should be a 1 for the good block.*
14. Place the *bad* block under the camera and do a manual trigger.
    NOTE: *The results displayed by the WriteDiscrete function should be a 0 for the bad block. Notice the LEDs on the I/O Expansion module. OUT 0 should not go on.*
15. Save the job.

Lab Exercise 8.1 – Network Communication

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

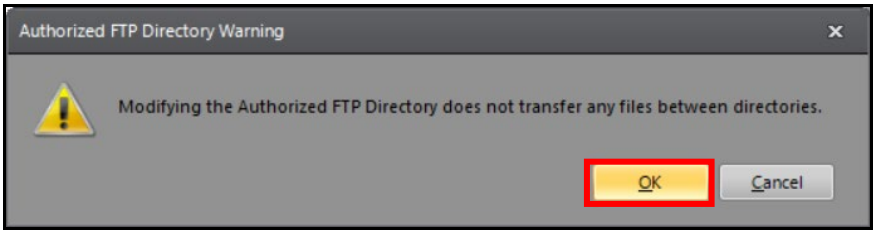
- Utilize the WriteImageFTP function to add images to an Authorized FTP Directory
- Utilize the FormatString function to write the text string
- Utilize the WriteFTP function to add text to an Authorized FTP Directory

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- WriteImageFTP
- FormatString
- WriteFTP

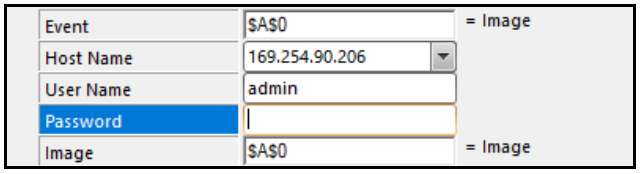
Follow the steps below to complete the lab exercise:

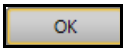
1. Open the **MyOutput** job from the last lab exercise.
2. Save the job as **MyComm** in the folder you created in Lab #1.
3. Create a new folder named **FTP** in the class folder on your desktop. Copy the path of this folder.
   NOTE: *This folder will be the Authorized FTP Directory where your images will be saved.*
4. Enter the copied path into the **Authorized FTP Directory** field and click the **Apply** button.
   NOTE: *This is found under **System → Options → Emulation**.*
5. The **Authorized FTP Directory Warning** dialog displays. Click the **OK** button to close.



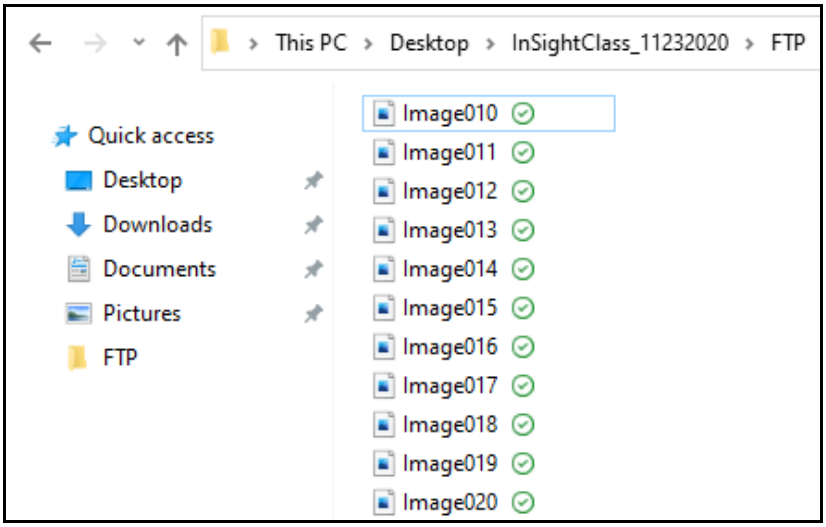
6. Enter the comment **FTP** into cell **B35** and insert a **WriteImageFTP** function into cell **C36** of the spreadsheet.
   NOTE: *This is found under **Input/Output → Network → WriteImageFTP**.*
7. Click the **OK** button to close the Options dialog.
   The **WriteImageFTP Property Sheet** displays.
8. Select the name of your **PC** for the **Host Name**.
   NOTE: *Use the IP Address if your PC name is not listed.*
9. Enter a **User Name** and **Password** to log into the FTP Server.

NOTE: *If using the In-Sight Explorer FTP server, the default username is admin, and the default password is blank*.

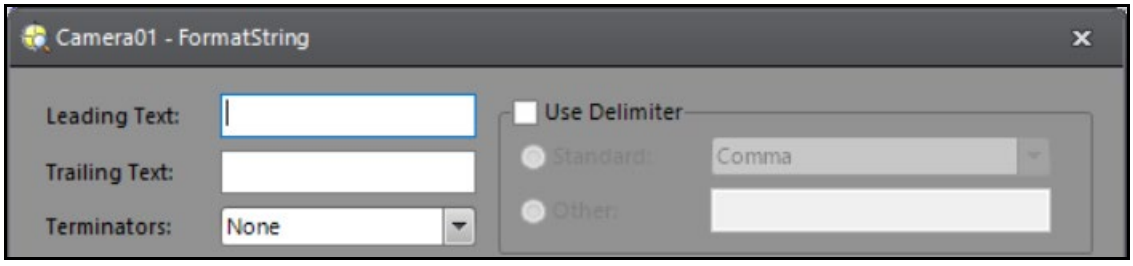


10. Enter the FTP folder path into the **File Name** field and add **\Image** to the end.
11. Click the **OK**  button.

12. Go **Online**.
13. Click the **Trigger**  button to acquire images. For each acquired image, a bitmap file is stored on the specified host.
14. Once you have acquired several images go **Offline**.
15. Open the **FTP** folder on your desktop and review the results.

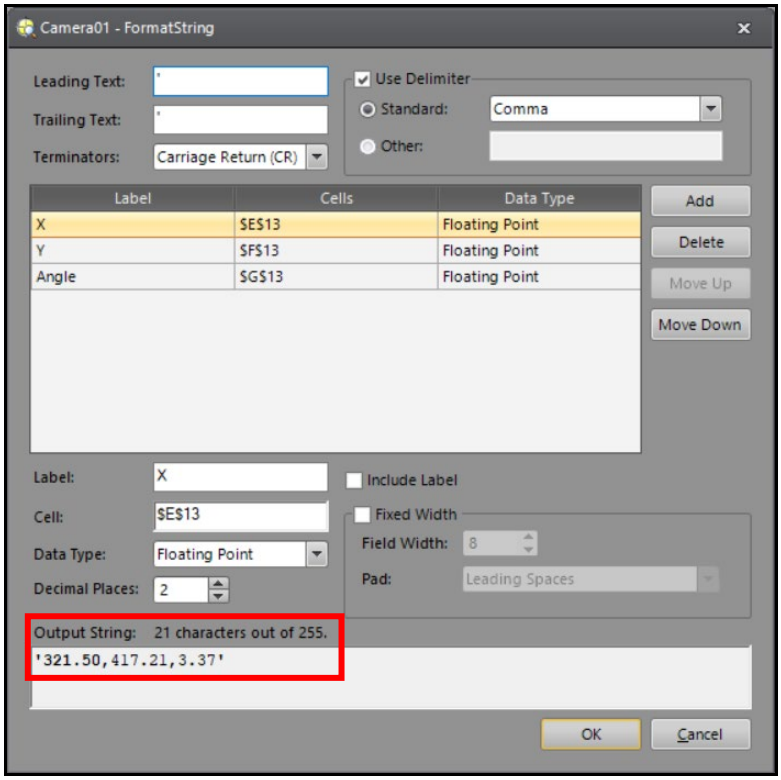


16. Click the first image, <Ctrl+A> to select all images and click the <Delete> key. Close the FTP folder.
17. Enter the comment **Format String** into cell **B37** and insert a **FormatString** function into cell **C38** of the spreadsheet.
NOTE: *This function is found under Text → String → FormatString. This will allow you to select multiple values as well as control the formatting of the string.*
The **FormatString Property Sheet** displays.

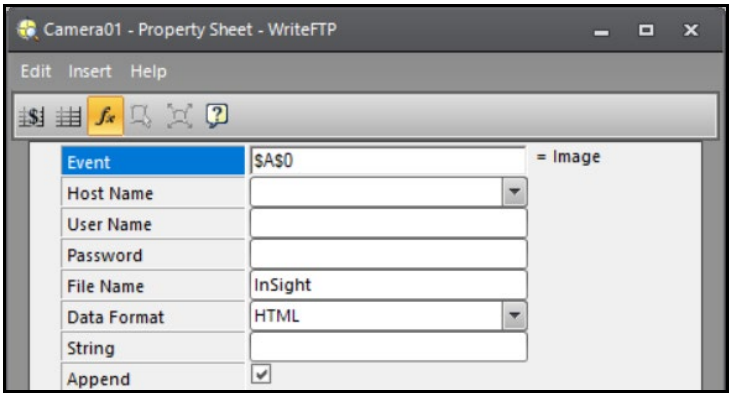


18. Click the **Add** button and format your Output String as follows:
    Leading Text = ‘ (single quote)
    Trailing Text = ‘ (single quote)
    Terminators = CR+LF
    Use Delimiter = check the checkbox
    Data = The three fixture values (Row, Column and Angle of the FindPatMaxPatterns tool)
    Decimal Points = 2

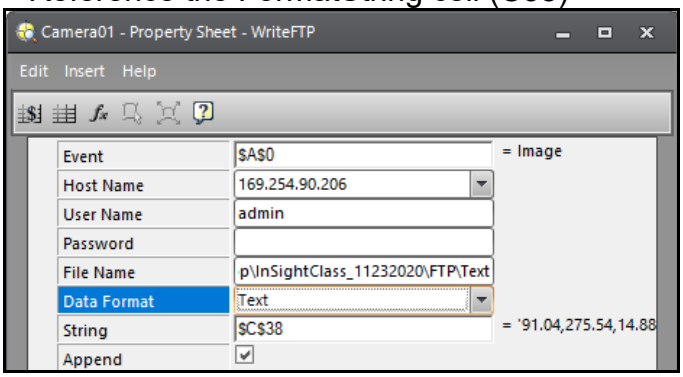
    NOTE: *Review the bottom of the dialog box.*



19. Click the **OK** button to close the FormatString dialog.
20. Trigger the camera and review the changes in cell **C38**.
21. Enter the comment **Text** into cell **B39** and insert a **WriteFTP** function into cell **C40** of the spreadsheet.
    The **WriteFTP Property Sheet** displays.

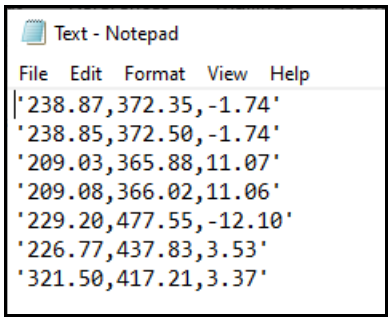


22. Set the Property Sheet parameters as follows:
    Host Name = IP Address of your computer
    User Name = admin
    Password = (leave blank)
    File Name = FTP folder path and add **\Text** to the end
    Data Format = Text
    String = Reference the FormatString cell (C38)



23. Click the **OK** button to close the property sheet.
24. Go **Online**.
25. Click the **Trigger** button to acquire images. For each acquired image, a bitmap file is stored on the specified host.
26. Once you have acquired several images go **Offline**.
27. Open the **FTP** folder on your desktop and review the results.
    Your folder will include images and a text file.

28. Open the text file and review the results.



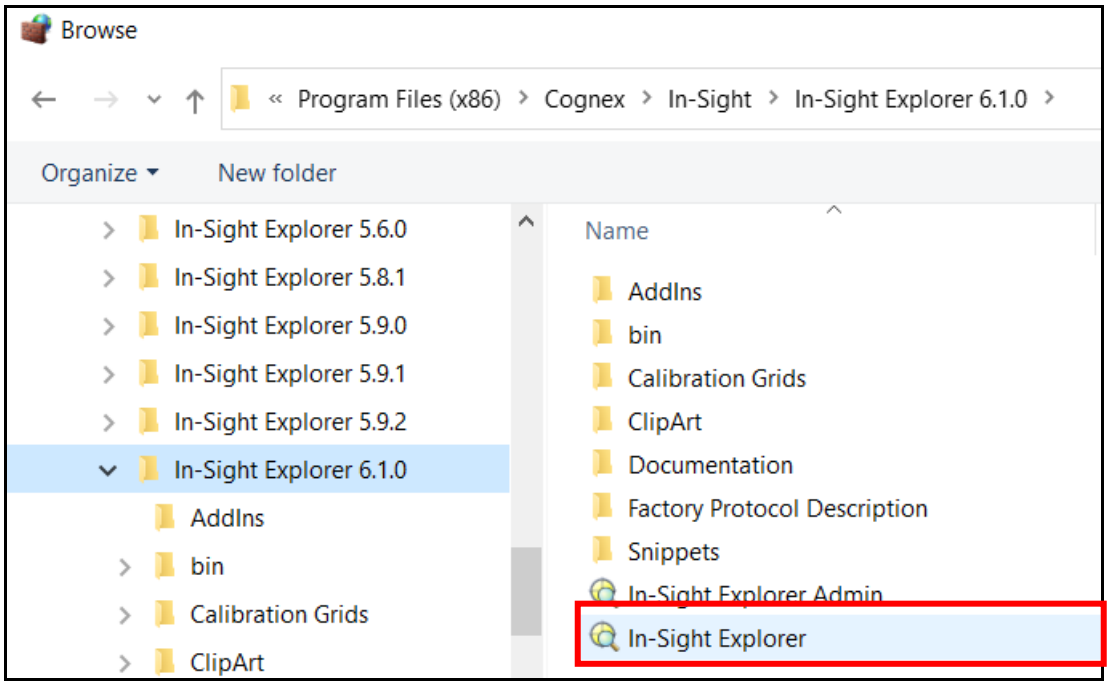
29. Save your job.

In-Sight Explorer and In-Sight OPC Classic Server are not automatically added to the Windows Firewall Programs exception list. To add the programs, either allow access through the warning dialogs that may display, or manually add the following to the Windows Firewall exception list.

For In-Sight Explorer, add the following applications:

- [Program Files (x86)]\Cognex\In-Sight Explorer 6.1.0\In-Sight Explorer.exe
- [Program Files (x86)]\Cognex\In-Sight\Emulators\6.1.0\In-Sight.exe

1. From the **Start** menu, click **Control Panel**, click **System and Security**, and then click **Windows Defender Firewall**.
2. Click **Add an app or feature through Windows Defender Firewall** from the left-hand menu.
3. Click the **Allow another app…** button.
4. Click the **Browse** button.



5. Click the **Open** button.
   The Program is added to the Apps: list.
6. Repeat step 4 and 5 for the remaining app.
   The apps will be added to the list of **Allowed apps and features.**

Lab Exercise 9.1 – Profiler

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

- Create an Operator Interface in the spreadsheet

The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

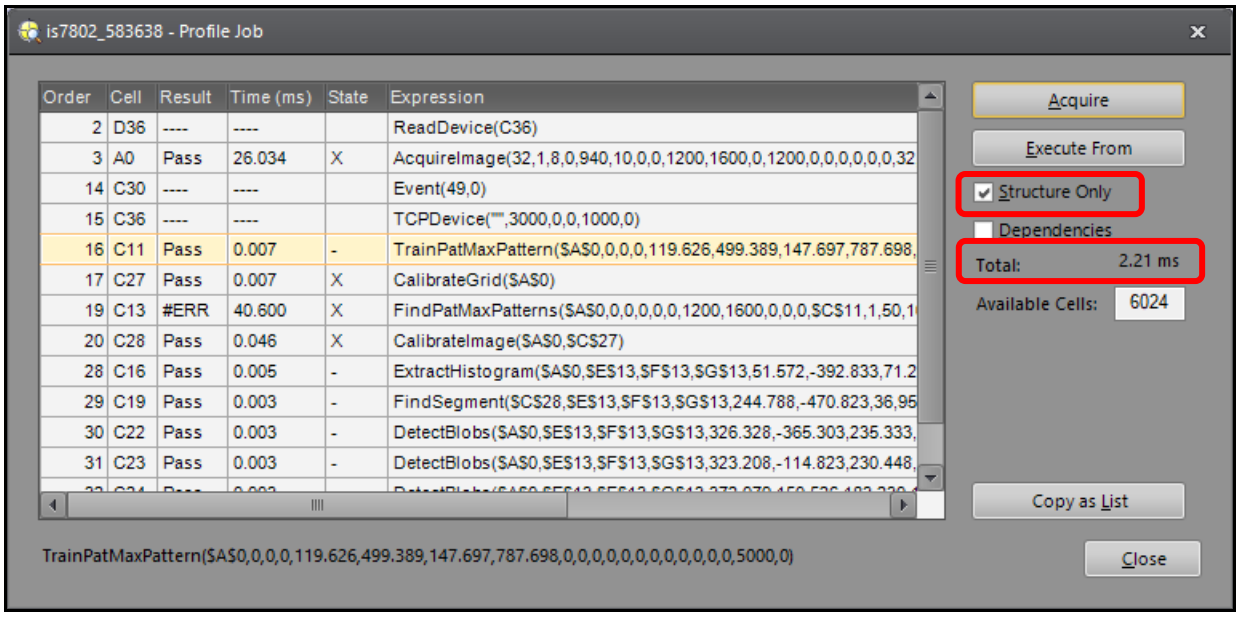
- Status
- Button
- Custom View
- References

Follow the steps below to complete the lab exercise:

1. Continue with the **MyComm** job from the from the previous lab exercise.
2. Click **Sensor → Profile Job**.

   The **Profile Job** dialog box displays.

   NOTE: *By default you see All cells – to see the time each tool takes to run, highlight the row and click the **Acquire*** [Acquire] *button.*
3. Click the **Structure Only** checkbox. This shows just those cells that have a structure in them (your tools).



4. Click the **Acquire** [Acquire] button again to run the inspection once.

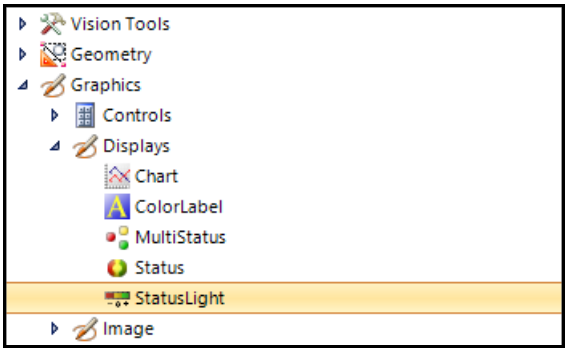
   NOTE: *You will see the updated time for each tool as well as a total cycle time in the lower right corner. The total time in the Profiler will be greater than the time in the lower right-hand corner of the spreadsheet. This discrepancy is the rendering of graphics on the display.*

Lab Exercise 9.2 – Operator Interface

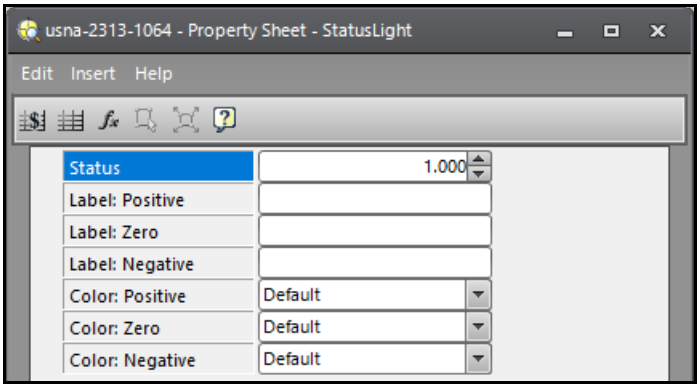
In an earlier lab, we deliberately skipped the first 10 rows of the spreadsheet. Now, we are going to use those cells to define a **Custom View**, which will reference a number of cells you already set up.

Though we will walk you through setting this up, please keep in mind that these are suggestions. If you want to try your own design, please keep the data in a logical fashion and in a way that will make it easy for an operator to interpret.

1. Continue with the **MyComm** job from the previous lab.
2. Save the job as **MyOp** on the In-Sight camera and your own folder on the PC.
3. Enter the comment **Part:** into cell **B2**.
4. Insert a **StatusLight** function into cell **C2**.



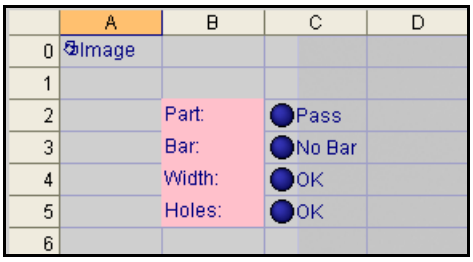
The **StatusLight Property Sheet** displays.



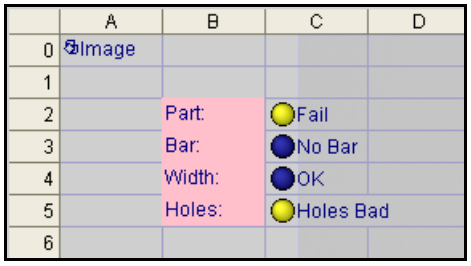
5. Configure the StatusLight Property Sheet as follows:
   Status – Reference the Global Bit (B34) result
   Label: Positive – Pass
   Label: Zero – Fail
   Label: Negative – Error
   Color: Positive – Dark Blue
   Color: Zero – Yellow
   Color: Negative – Red

6. Click the **OK**  button.
7. Enter the comment **Bar:** into cell **B3**.
8. Insert a **StatusLight** function into cell **C3**.
9. Configure the StatusLight Property Sheet as follows:
   Status – Reference the Histogram (K19) result
   Label: Positive – No Bar
   Label: Zero – Bar Found
   Label: Negative – Error
   Color: Positive – Dark Blue
   Color: Zero – Yellow
   Color: Negative – Red
10. Repeat the steps above for the Width and Holes.

    The display should look similar to below:

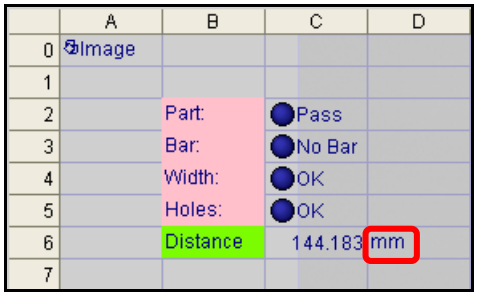


Good Block



Bad Block

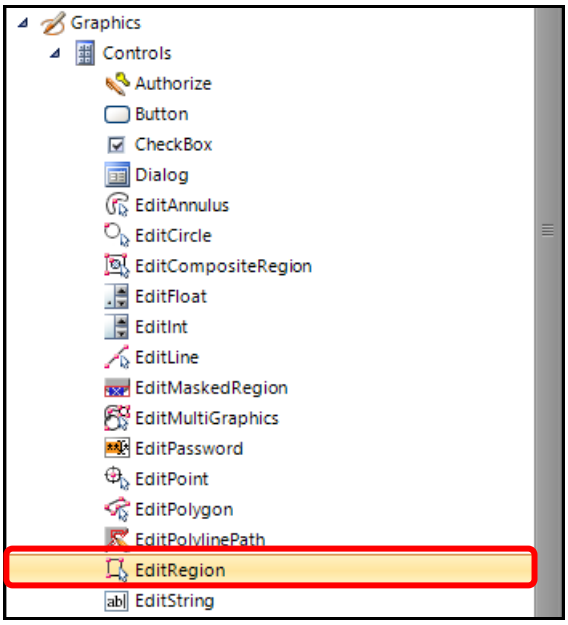
11. Enter the comment **Distance** in cell **B6**.
12. Reference the dimensional measurement of the block in cell **D19**.



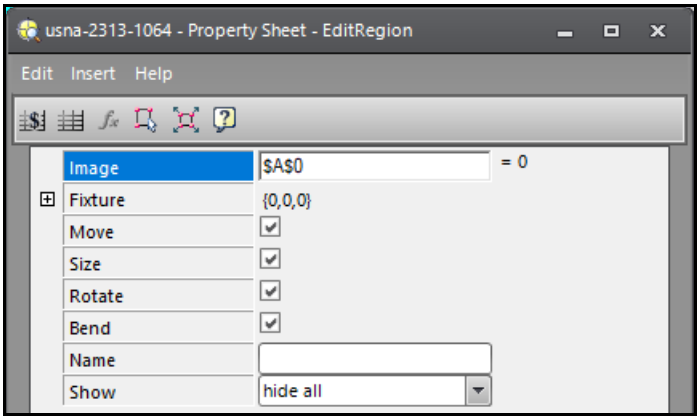
NOTE: *You may want to enter your units of measure (mm) into cell D6 so that the results make sense to your operator.*

13. Enter the comment **Adjust for Bar Region Online** in cell **B38.**

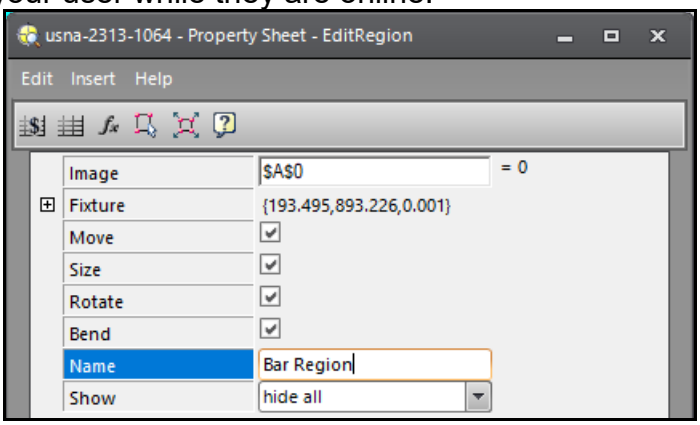
14. Insert an **EditRegion** function into cell **C40** in the spreadsheet.



The **EditRegion Property Sheet** displays.



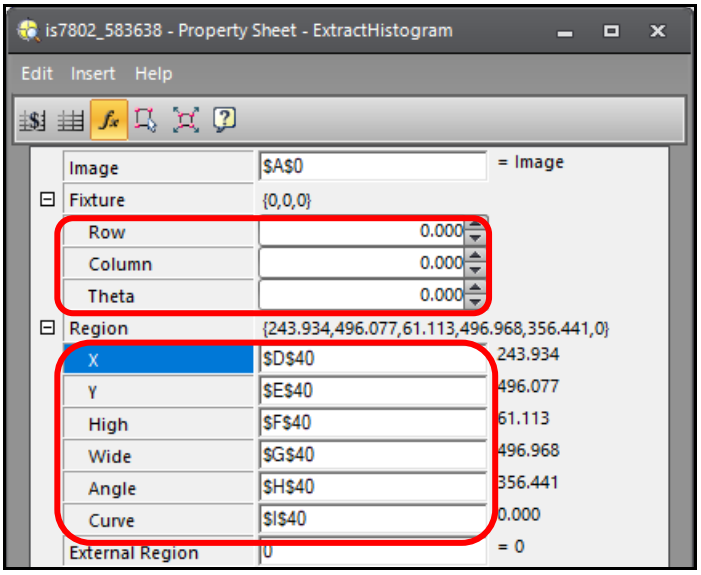
15. Fixture to the FindPatMaxPatterns tool (E13, F13, G13) and enter the Name which will appear on the button as **Bar Region**. Select or deselect options that will be available to your user while they are online.



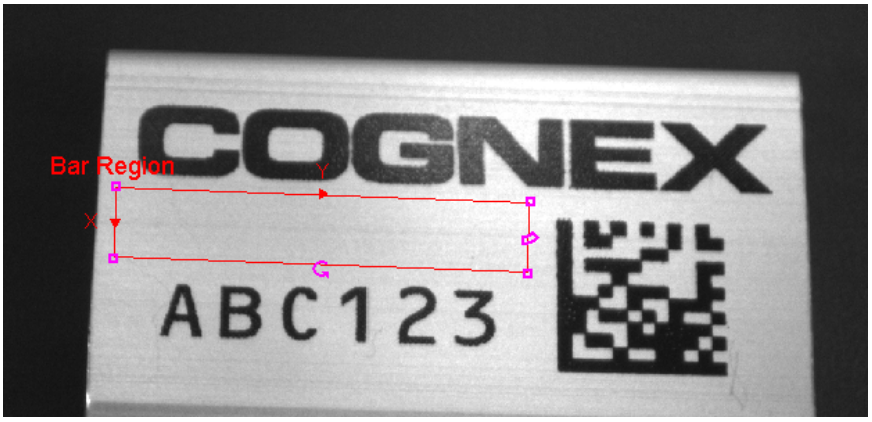
16. Click the **OK** button and note the Bar Region button and auto-inserted functions that are created.

38		Adjust for Bar Region Online						
39			Row	Col	High	Wide	Angle	Curve
40		Bar Region	440.000	580.000	320.000	440.000	0.000	0.000
41								

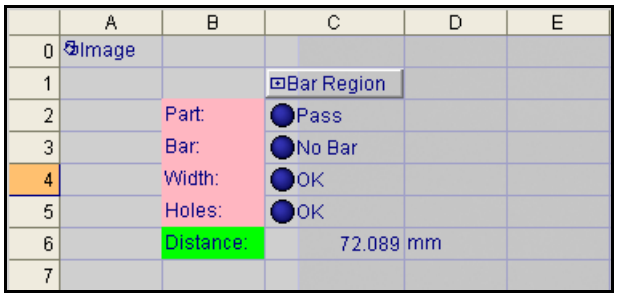
17. Attach the Region to the ExtractHistogram tool (C16) by going into the Histogram Property Sheet and referencing the Region parameters to the output of the EditRegion tool.
Because the region is now fixtured in EditRegion, we need to disable fixturing in the ExtractHistogram tool itself. To do this, zero out the fixture information in the Histogram tool.



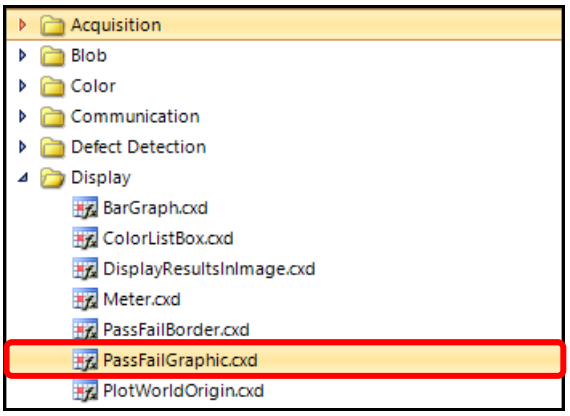
18. Click the **OK** button.
NOTE: *Now when the Bar Region button is pushed the region for the Histogram tool is able to be changed.*
19. Click the **Bar Region** button to reset the region to be similar to the area originally set for the Histogram tool, being sure not to get too close to the edge of the block.



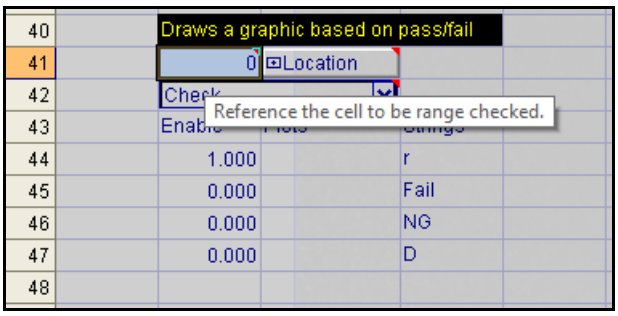
20. Cut the Bar Region button and Paste it into cell **C1**, directly on top of the Custom View Status Lights area.



21. Insert a **PassFailGraphic** snippet into cell **B41** of the spreadsheet. This will add an image display to your application to quickly show the user if the part is good or bad.

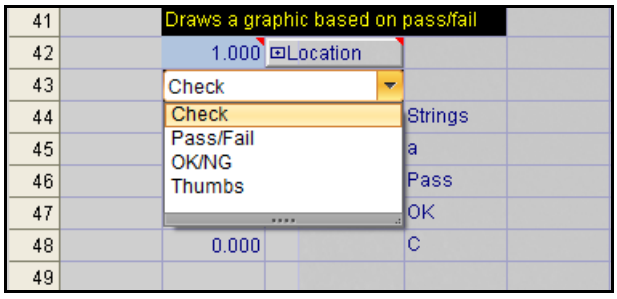


22. To attach the snippet to your program, you need to make a reference from cell **B42** to your Global Bit Result (C34).

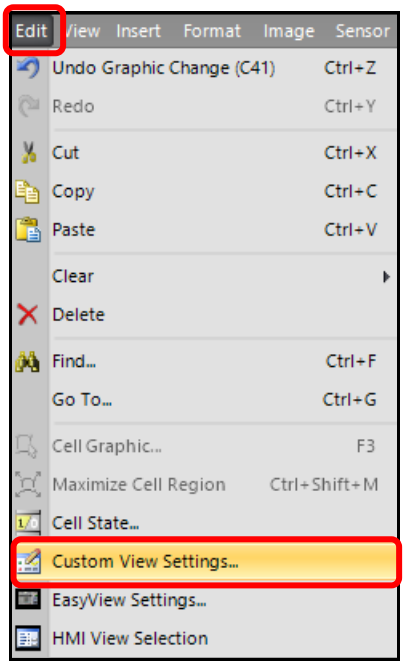


23. **Location** (C42) is an **EditPoint** function that allows you to set where the string will be displayed on the image based on the upper left-hand corner. It displays a point that can be moved. Leave it at the default for now.

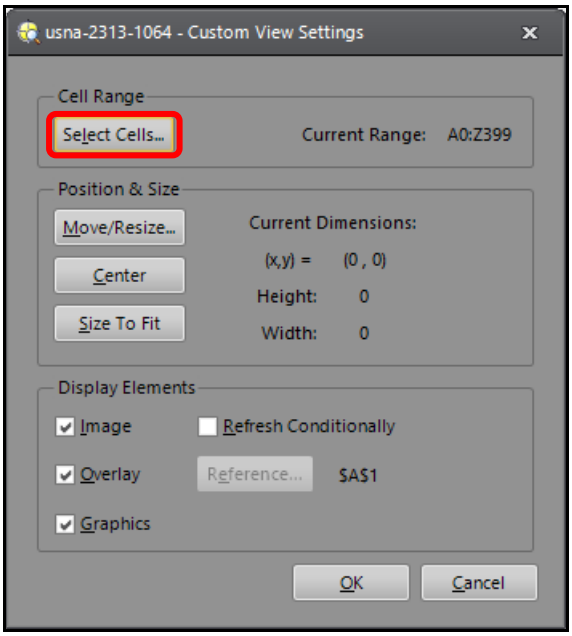
24. From the drop-down list, select the type of graphic that you would like to use to show the Pass/Fail status of the job. (In our example below, we use **Thumbs.**)



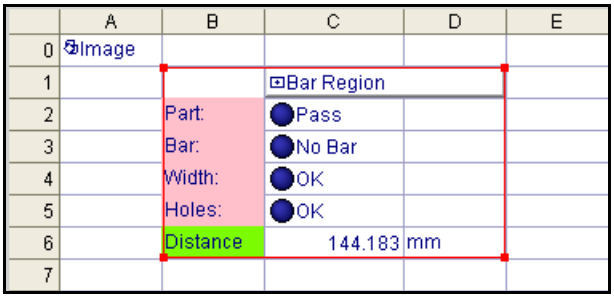
25. Select **Custom View Settings** under the Edit menu to finish creating your Custom View.



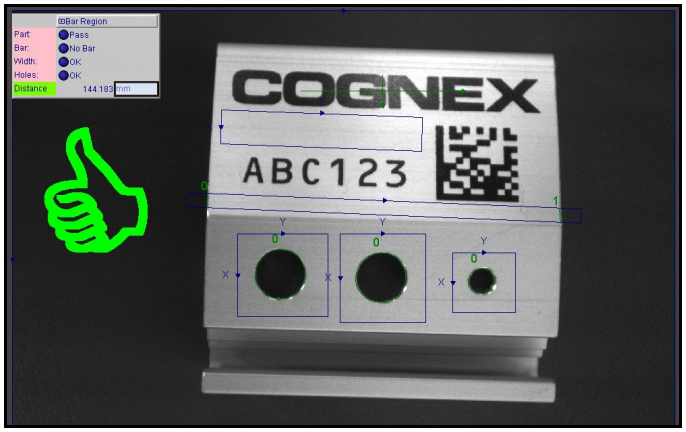
The **Custom View Settings** displays.



26. Click the **Select Cells** Select Cells... button and choose cells B1 through D6.



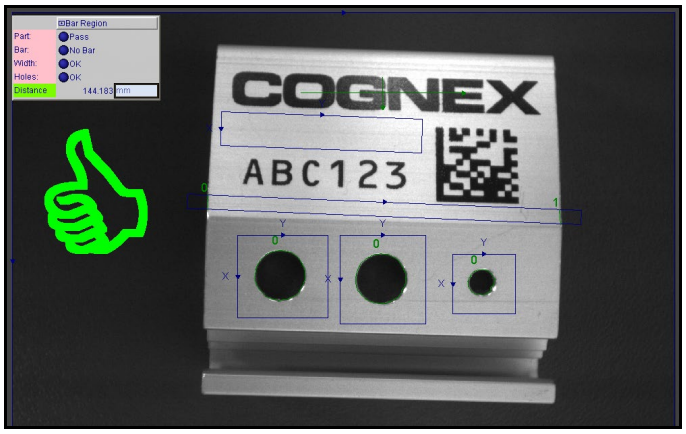
NOTE: *The selected cells will be outlined in red.* Once the cells are highlighted click the **<Enter>** key. The cells will display in the upper left. You can move/resize the window as you would like.



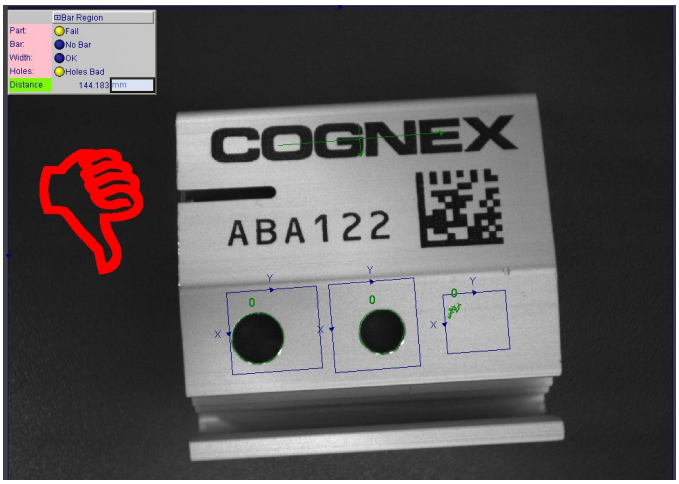
27. Once you have positioned and sized the Custom View as you would like, click the **OK** OK button.

NOTE: *To toggle between the spreadsheet and Custom View click the* ***<F6>*** *key.*

28. Test the good block and the bad block and review the results.

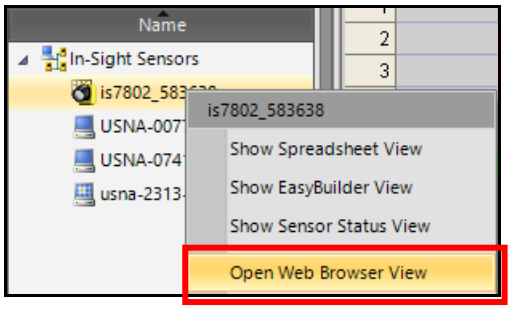


Good Part



Bad Part

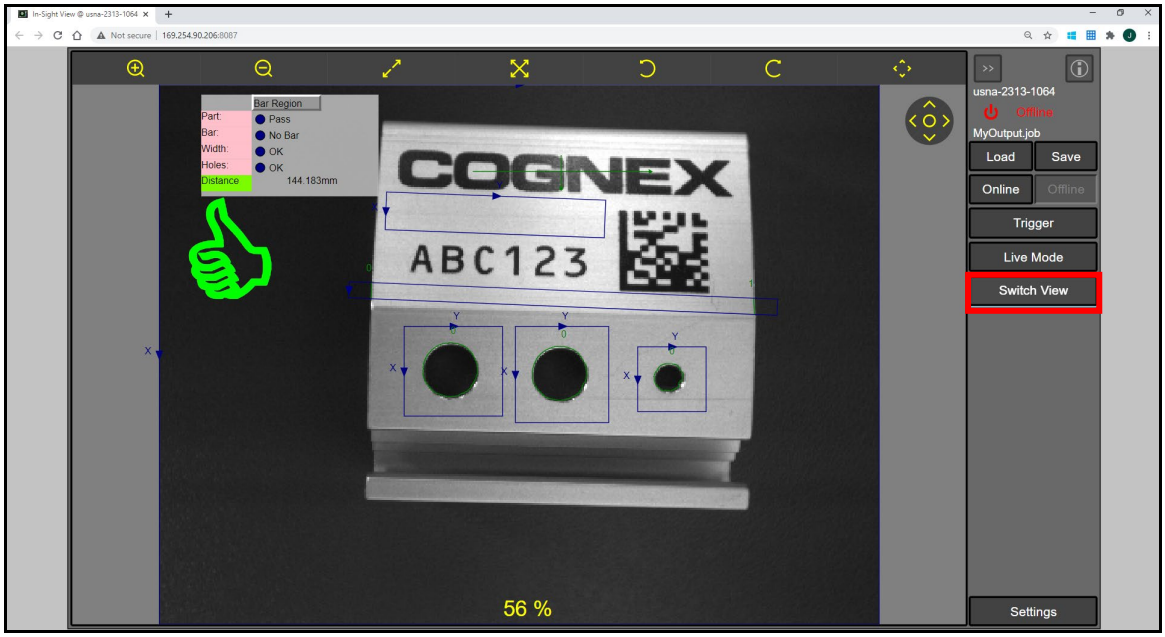
29. Right-click your sensor in the In-Sight Network pane and select **Open Web Browser View** from the fly-out list.



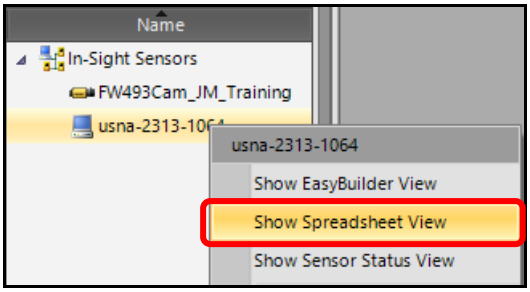
The **WebHMI** displays in a new web browser.

30. Disconnect from your In-Sight sensor.

31. Click the **Switch View** button to scroll through the various views.



32. Close the WebHMI.
33. Right-click your sensor in the In-Sight Network pane and select **Show Spreadsheet View** from the fly-out list to return to your spreadsheet.



34. Save the job.

Lab Exercise 9.3 – (if time allows)

- Add the **CountPassFail** tool, which is found in the Clocked Data Storage category, to the Custom View to determine the run rate for the result of each vision tool. The actual Count functions should not be seen in the view.

- Insert a Chart function to graph the **ExtractHistogram**'s Average value over time.

Lab Exercise 10.1 – Deployment and Finishing Applications

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

- Utilize the utilities available in In-Sight Explorer to finish deploying the application
- Use the VisionView to display the application

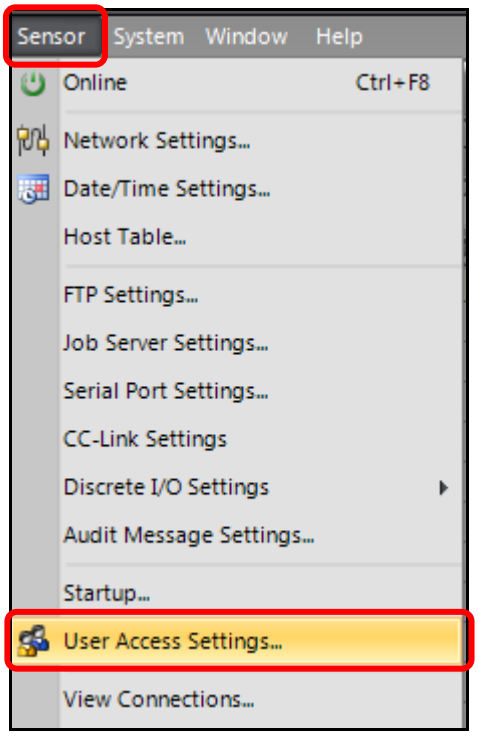
The Participant will utilize the following In-Sight Functions to successfully complete this exercise:

- User Access Settings
- Startup
- Report
- Backup
- Restore From

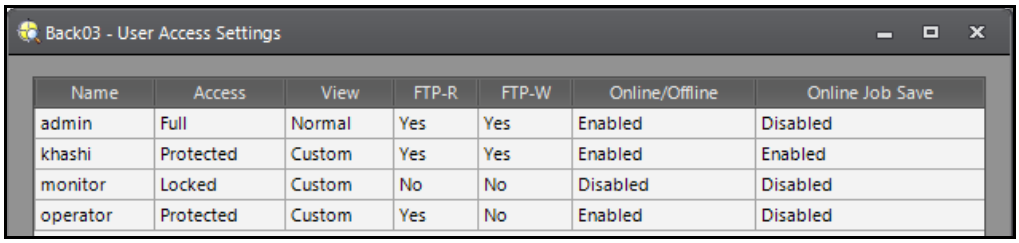
Follow the steps below to complete the lab exercise:

User Access Settings

1. Click on the **User Access Settings** link – this is found in the **Sensor** menu.

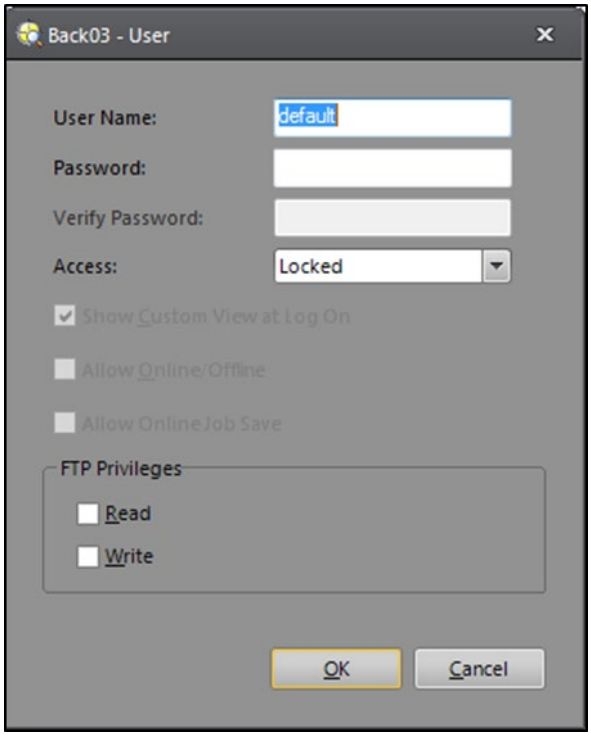


The **User Access Settings** display.



Name	Access	View	FTP-R	FTP-W	Online/Offline	Online Job Save
admin	Full	Normal	Yes	Yes	Enabled	Disabled
khashi	Protected	Custom	Yes	Yes	Enabled	Enabled
monitor	Locked	Custom	No	No	Disabled	Disabled
operator	Protected	Custom	Yes	No	Enabled	Disabled

2. Click the **Add** Add... button to create a new user.
   The **User** dialog displays.



3. Enter the new user's information in the appropriate fields and click the **OK** OK button twice.
   NOTE: *You can use any information that makes sense to you in these fields.*
4. After the new user is created, make sure that you can log onto your camera with the new user information. Then log back onto your camera as before (admin).
5. Test the new user information on another camera.

   Can you log in?

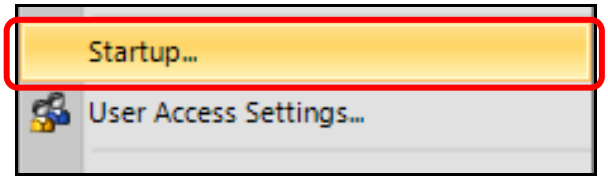
   _____

   What would you need to do to be able to log in?

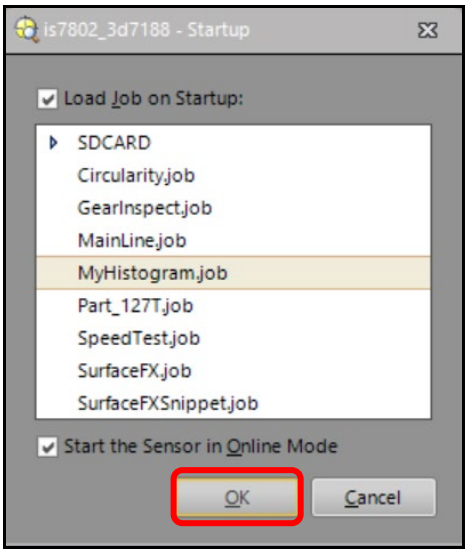
   _____

Startup

1. Click on the **Startup** link – this is found in the **Sensor** menu directly above the User Access Settings.



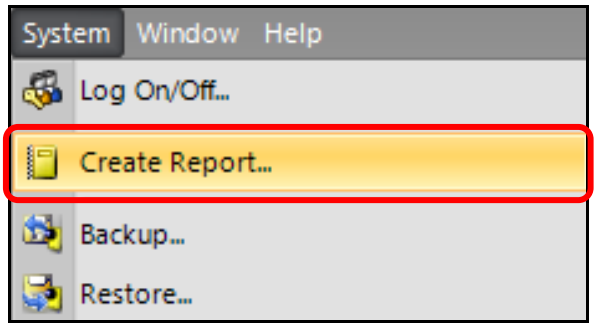
The **Startup** dialog displays.



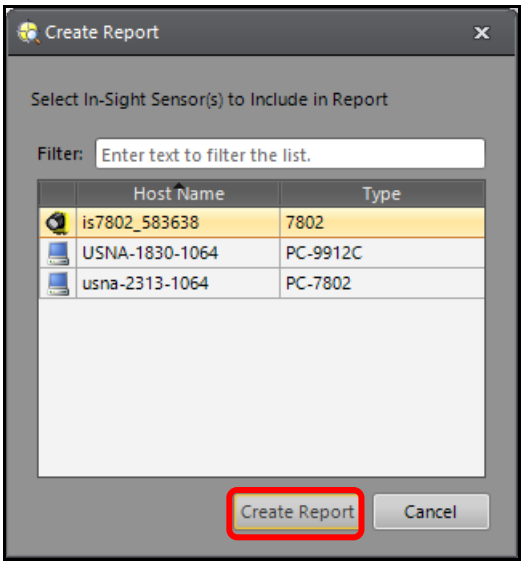
2. To have the camera automatically open your completed job and go online at startup, click on MyHistogram.job in the list and check the **Start the Sensor in Online Mode** checkbox. Then click the **OK**  button.

Report

1. Click on the **Create Report** link – this is found in the **System** menu.



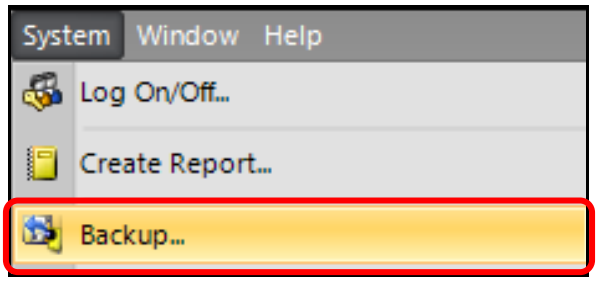
The **Create Report** dialog displays.



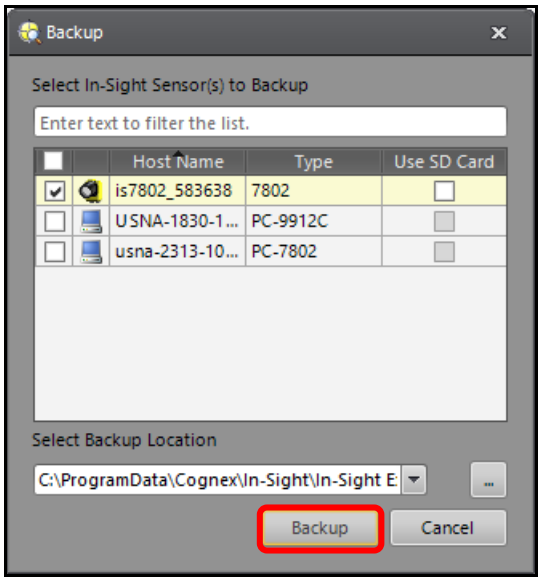
2. Select the In-Sight sensor to include in the Report and click the **Create Report** Create Report button.
3. When the Report is complete, open it and scroll through it to become aware as to what was saved.

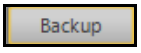
Backup

1. Click on the **Backup** link – this is found in the **System** menu.



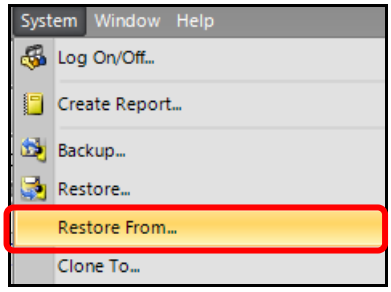
The **Backup** dialog displays.



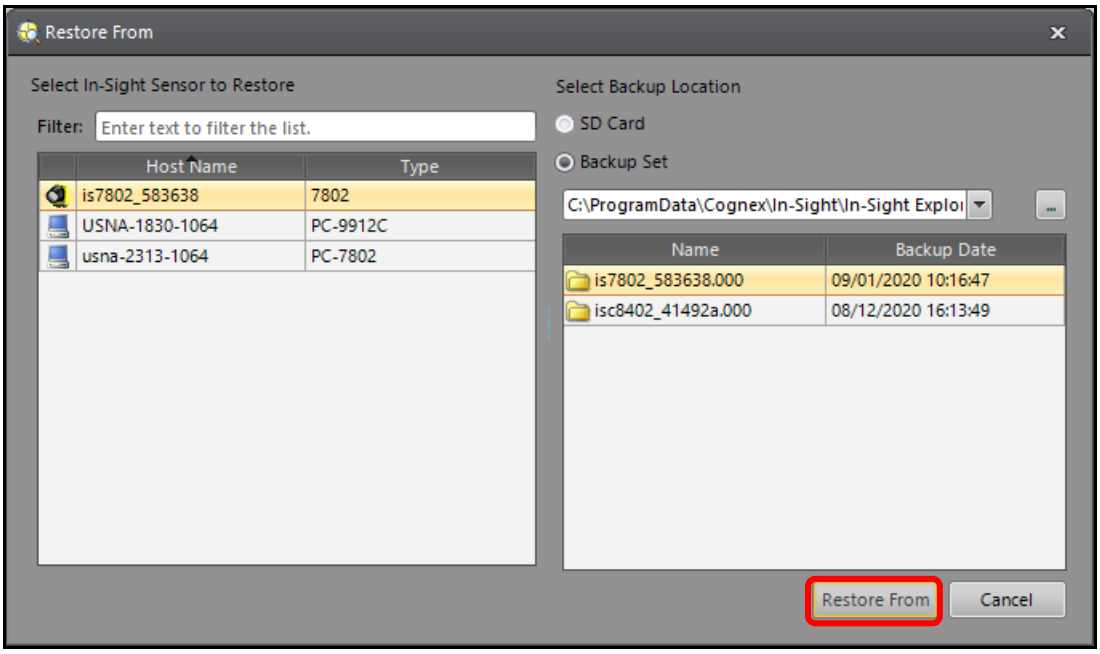
2. Select the In-Sight sensor to backup and click the **Backup**  button.
3. When the Backup is complete, change the I/O setting to something different. Now perform a **Restore From**.

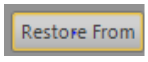
Restore From

1. Click on the **Restore From** link – this is found in the **System** menu.



The **Restore From** dialog displays.



2. Select the In-Sight camera to restore and then the most recent backup. Then click the **Restore From**  button.

3. When the Restore is complete, check the I/O settings again. What do they show?

VisionView (if time allows)

Go to a VisionView station or open the demo software. Walk through the intuitive steps for a few moments to become acquainted with the system. Log on to your camera to display it through VisionView.

In-Sight Spreadsheets Standard – Final Lab

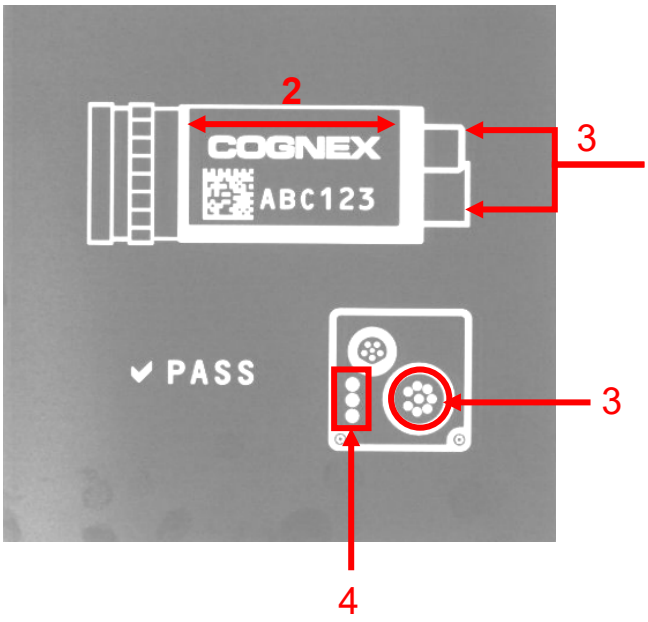
Lab Objective:

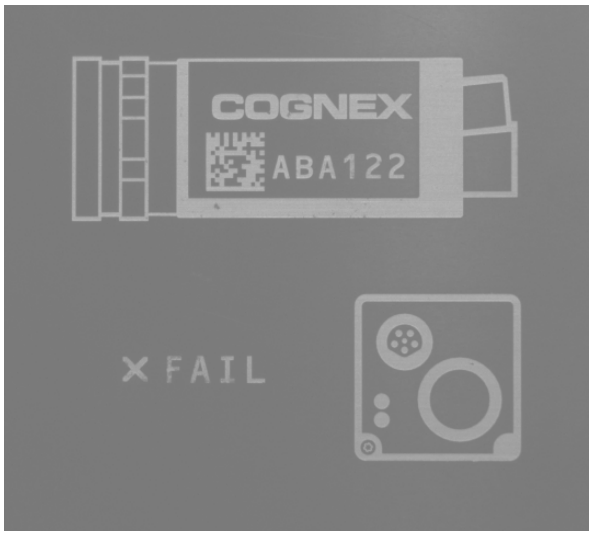
Your task is to create an In-Sight Spreadsheet inspection with a real part using what you learned during your In-Sight Spreadsheets Standard class.

All inspection tasks must be completed and the good/bad part must always pass/fail your inspection. Assume that the only defects on parts will appear as seen on the bad side of the plate, i.e., no other variations.

There are no requirements on *how* these tasks are to be completed. Be prepared to discuss your solution at the end of class.

Please record the tool used and where the tool is found within the Spreadsheet on the lines provided.





NOTE: *The numbers on the image refer to the corresponding lab steps.*

Inspection Tasks:

1. Consistently find the part in the image. Assume that the scale is constant and that the part can rotate, even to being upside down.

2. Measure the width of the part as shown in the picture above. The width of the camera body should be about 40-50 mm, depending on your setup. Your inspection should report the width in millimeters.

3. Check that the connectors have all pins present and that they are correctly installed.

4. Check that the part has the correct number of LEDs installed.

5. The camera must be triggered via a button function in the spreadsheet.

6. Create a Custom View that shows the status of all the inspection tasks and the status of overall inspection.