# Acquiring Images from the AXIS Network Camera

Contents:



## 1.1 Introduction

In this module we'll acquire images from the AXIS network camera using the FRC Robot and Dashboard projects supplied with LabVIEW FRC. We'll also go over camera setup, connectivity, and the WPI Robotics Library Image Acquisition VIs.

This overview will explain the image acquisition functionality already included in the FRC projects, and get you started acquiring images. Please refer to the *Best Practices for Using the FRC cRIO Project Frameworks* tutorial if you are not already familiar with the FRC projects.

## 1.2 Hardware Setup

Configuration & Connecting the Camera to the FRC cRIO

Before connecting the camera to the FRC cRIO, the camera needs to be configured so that the Camera VIs can communicate with the camera. Complete the steps to configuring these settings using the Setup Axis Camera Tool by following the instruction found in the *LabVIEW Robotics Programming Guide for the FIRST Robotics Competition* manual, chapter 3.

To connect the camera to the cRIO, you must use a crossover Ethernet cable instead of a normal Ethernet Cable. A crossover cable is provided for you in the Kit of Parts (orange cable). Connect one end of the crossover cable to the AXIS camera and the other end to Ethernet port 2 on the cRIO. Then connect the power supply to the camera and make sure your cRIO has power.

LED Indicator

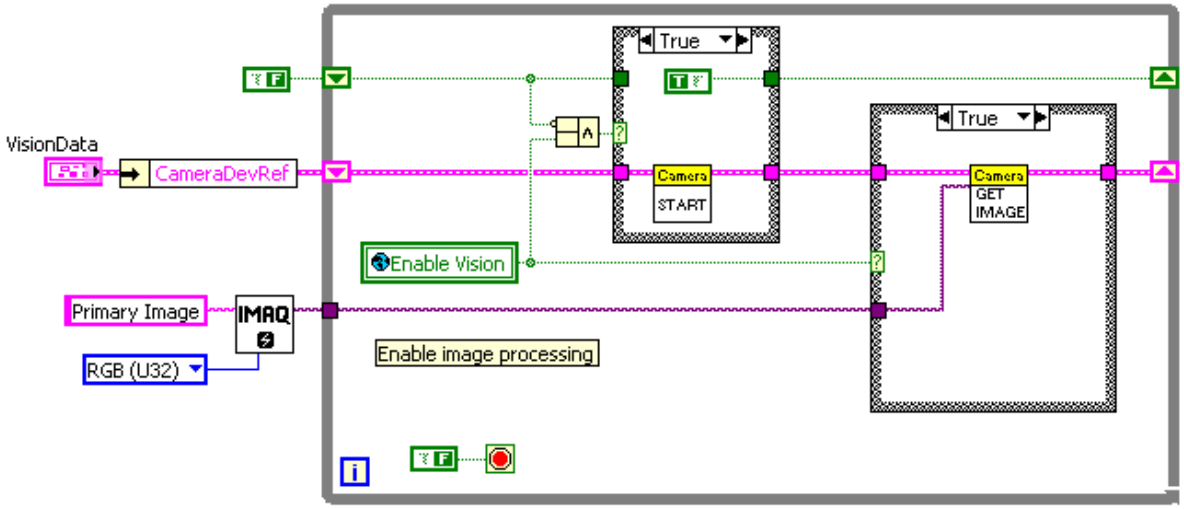
Once the camera has power, you will notice an LED indicator ring around the lens, and two LEDs on the back of the camera. These LED indicators help you determine the status of the camera. Below is a table that explains the different states of the indicators.

| | | |
|---|---|---|
| Network | Amber | Steady for connection to 10 Mbit/s network. Flashes for network activity. |
| | Green | Steady for connection to 100 Mbit/s network. Flashes for network activity. |
| | Red | Flashes rapid red, together with the Status indicator, for hardware error. |
| | Unlit | No connection. |
| Status | Green | Shows steady green for normal operation. Can be configured to flash green at intervals whenever the camera is accessed. See AXIS online help for more information. |
| | Unlit | When configured for “no flash” on camera access. |
| | Amber | Shows steady amber during reset to factory default or when restoring settings. |
| | Red | Slow flash for failed upgrade (see *Emergency Recovery Procedure* of AXIS 206 User Manual for more information) |
| Power | Green | Normal operation. |
| | Amber | Flashes green/amber during firmware upgrade. |

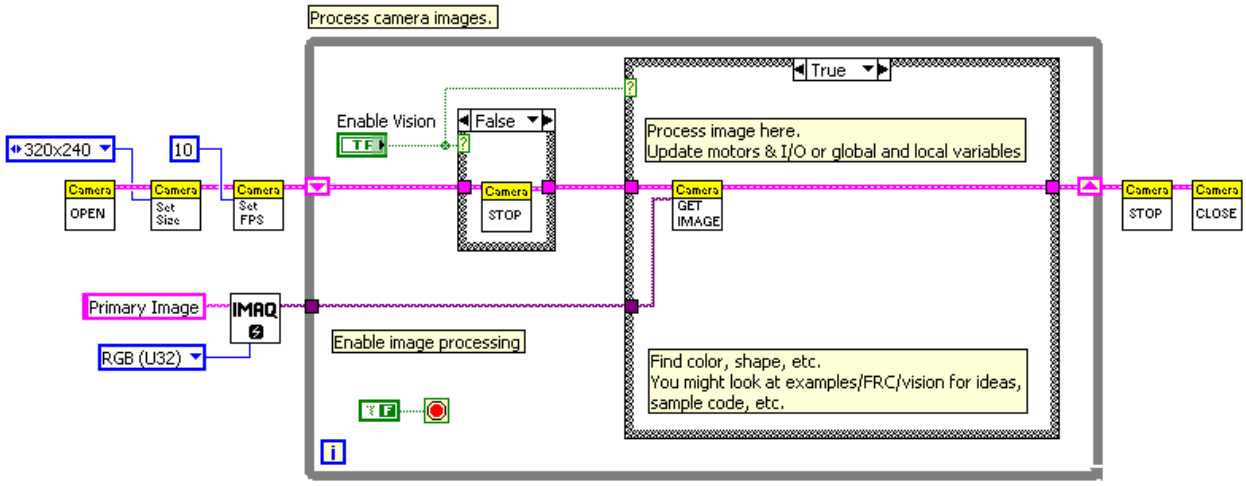
AXIS 206 User Manual, pg 10

### 1.3 Framework Vision Code

Start by creating a new FRC cRIO Robot project and a FRC Dashboard project from the LabVIEW getting started window. Open up the block diagrams for the Robot Main VI and the Dashboard Main VI. Depending on whether you have created an Advanced or Basic project will determine where the image acquisition code is located. If you select an Advanced project, the image acquisition code is located within the Vision subVI in the Robot Main VI.

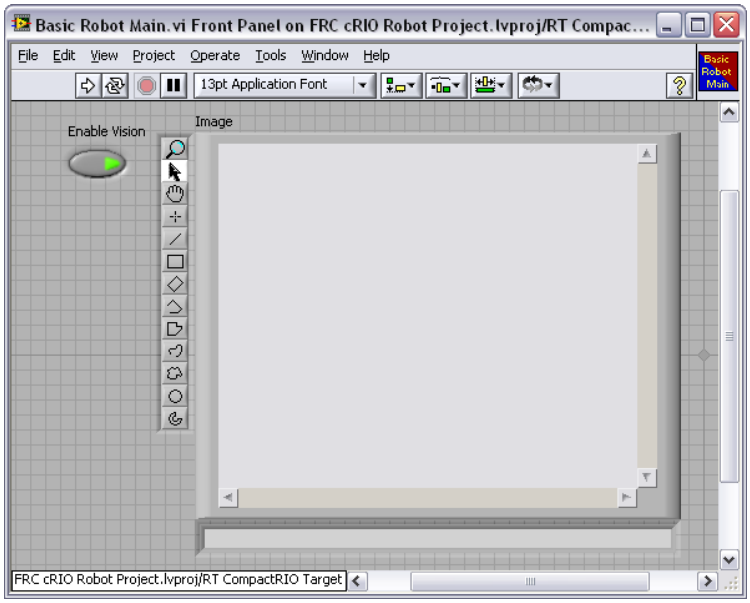


If you select a Basic project, the image acquisition code will be the second lowest loop in the Robot Main VI.

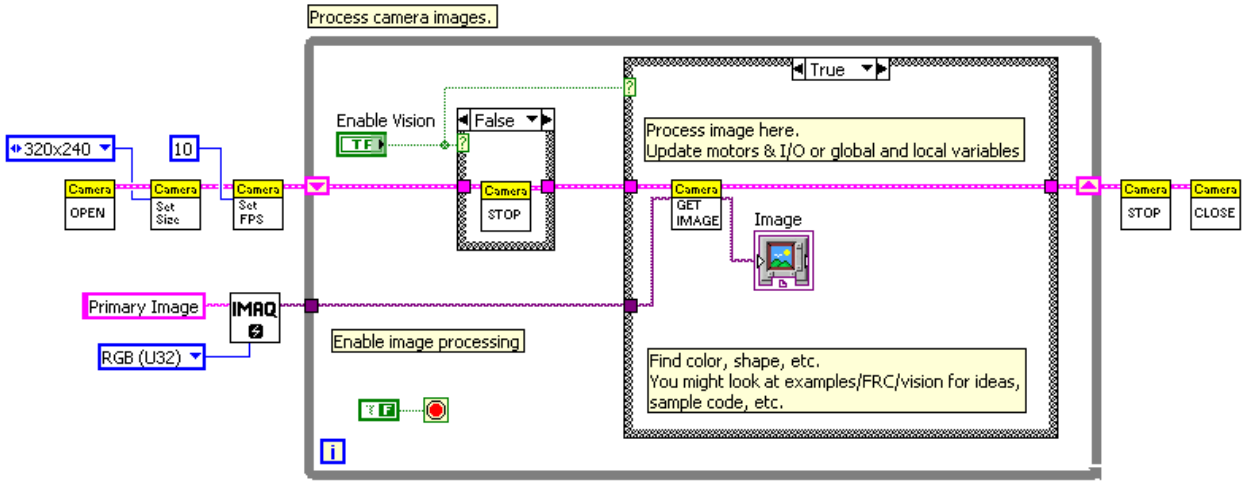



You will notice the code structure differs in organization but is identical in function. The major difference between the two projects is that in the advanced framework the camera initialization VIs are located within the Begin subVI , and then the relevant parameters are passed via a type definition to the Vision subVI. For this tutorial, we will focus on the code in the basic framework. The default vision code opens communication to the camera, sets parameters, and then starts acquisition and continuously acquires images as long as the Enable Vision Boolean is true.

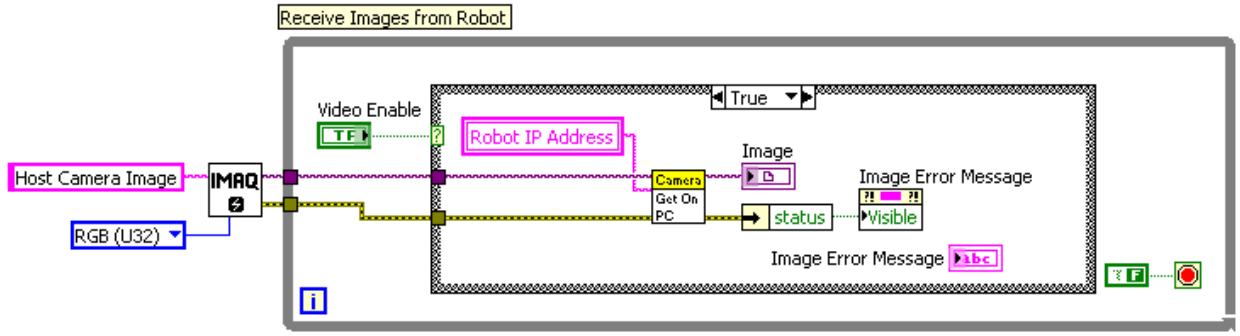
To display an image on the Front Panel of the Robot Main VI, right-click on the Front Panel to open the Controls palette. Drag and drop an Image Display located in the Vision Palette onto the Front Panel.



Next, switch to the Block Diagram and wire the Image Out output of the Get Image VI to the Image Display indicator. The image acquired from the camera will now display on the Front Panel of the Robot Main VI.




Since this code is running on the cRIO, it is recommended that you do not leave an Image Display on the Front Panel of the Robot Main VI. Instead, if you wish to see the images for troubleshooting and verification purposes, use the Dashboard Main VI to view the images. Just as there is code in the Robot Main VI to establish a connection to the camera and acquire images, there is code in the Dashboard Main VI to receive images acquired by your cRIO and display them on your laptop.

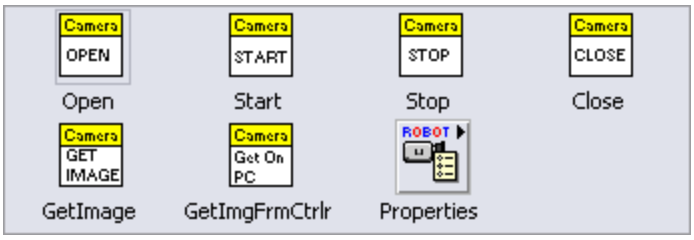



Now that we've walked through the default framework vision code, let's take a closer look at the Camera palette and the tools available to us for image acquisition.

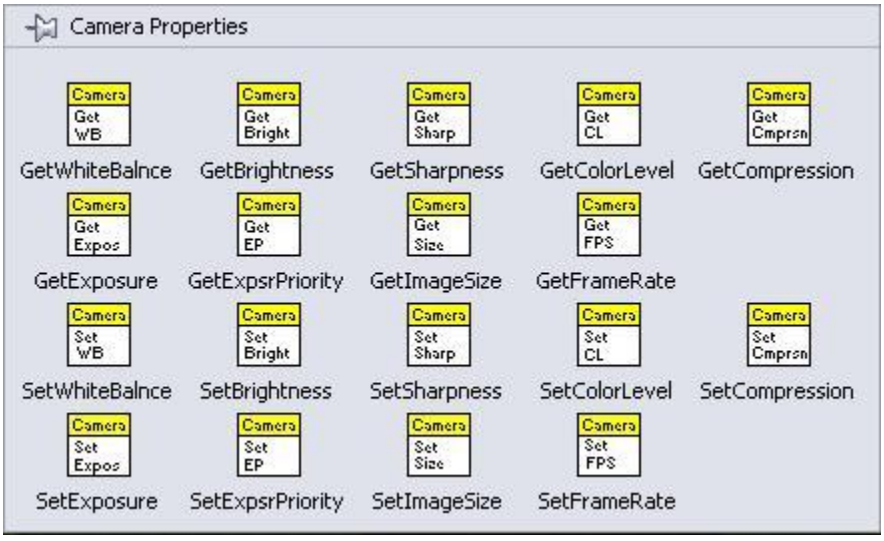
## 1.4 Image Acquisition Tools

Camera VI Palette

The Camera VIs are located within the WPI Robotics Library. To get to these VIs, right-click on the Block Diagram and navigate to **Programming»WPI Robotics Library»Camera**.




These VIs allow you to open a connection to a camera, start acquiring from the camera, get an image, stop acquiring, and close the connection to the camera. If you select Camera Properties, you will find two basic types of VIs: Get VIs and Set VIs. The Get VIs allow you to get the value of different properties of the camera, such as Brightness, Sharpness, Exposure, Image Size, and Frame Rate. The Set VIs allow you to set the values for these same properties.




For instance, the Set Exposure Priority VI allows you to prioritize either frame rate or image quality when the camera sets the exposure.

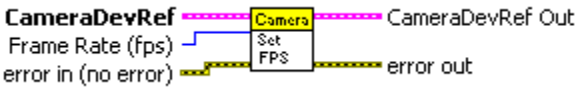

Giving more priority to frame rate will maximize frame rate at the expense of image quality, but giving more priority to image quality may decrease frame rate and increase motion blur. If you use this VI, the mode you choose will depend on the requirements of your program.

The Set Image Size VI is another useful VI.

This VI will allow you to specify the size of the image (in pixels) you want the camera to acquire. The default size is 320x240. You can also specify 640x480 and 160x120.

Frame Rate

Another important VI is the Set Frame Rate VI.




The frame rate is the rate at which images are taken from the camera, and is measured in frames per second (fps). The maximum frame rate for the AXIS camera is 30 fps (frames per second). The frame rate is equal the loop iteration rate for the loop containing the Get Image VI. The frame rate will depend on factors such as lighting, camera settings, and image size, as well as what other operations are performed in the loop containing the Get Image VI. As you add image processing and other operations to the image acquisition loop, you may see a decrease in camera frame rate due to the time it takes to process all the functions in your loop.

Auto Exposure and Auto Focus

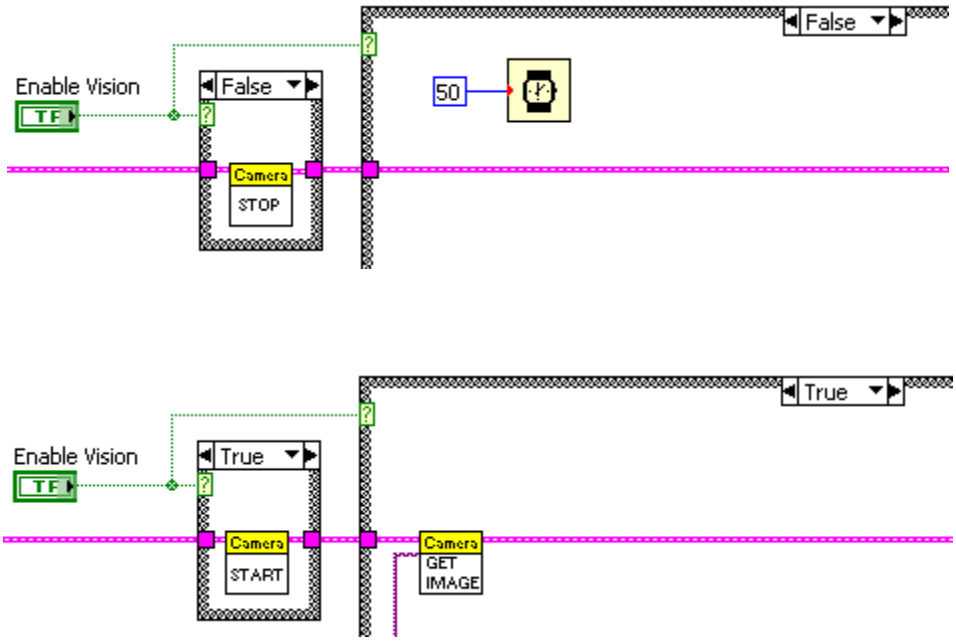
Some additional things to consider when using the AXIS camera are exposure and focus. The AXIS camera has an auto exposure feature. This means that the camera can automatically adjust exposure time as the lighting conditions change. If the brightness of the environment drops, the camera will lengthen exposure time to keep the picture brightness at an optimal level. If the camera is suddenly exposed to a bright light, the image will go white, but the camera will adjust the exposure after a few seconds to account for the change in lighting conditions.

Keep in mind that the auto exposure feature will affect the frame rate. If exposure time goes up due to dim light, then the frame rate will go down since it will take longer for the camera to acquire a frame. The AXIS camera does not have an autofocus feature. You must manually focus the camera using the focus ring located around the camera lens.

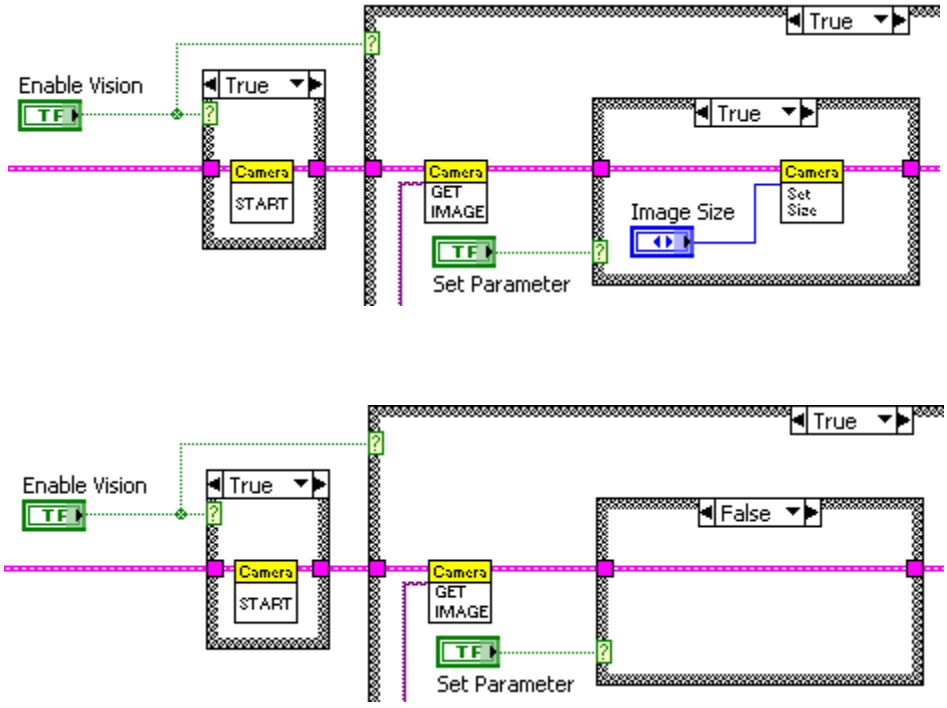
## 1.5 Changing Camera Parameters

When acquiring images, there are some initializations and parameters that you may need to set before you start an acquisition, such as image size, frame rate, or exposure. Most of the time, you will only need to set these parameters once, and thus you should set them outside the acquisition loop. If you put a camera configuration VI inside the acquisition loop, you will be setting that parameter every time the loop executes, adding processing overhead and communication to the camera, which will significantly slow down your loop. So what if you need to change settings multiple times during your program? How about fine tuning the parameters to find out the optimal settings for your application, or capabilities of the camera?

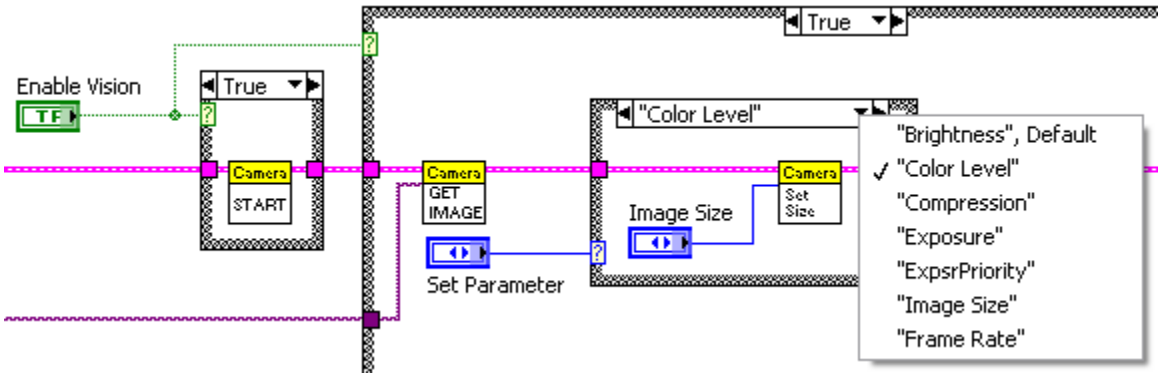
You can accomplish these things by using case structures inside the image acquisition loop to only set the camera parameters when you need to make a change. You can see this technique used with the *Enable Vision* Boolean in the framework vision code. If the *Enable Vision* Boolean is true, then the Camera Start VI is called, and then the Camera Get Image VI. If the *Enable Vision* Boolean is false, then the Camera Stop VI is called, and the loop waits 50 milliseconds before continuing.




We can add an additional case structure to the *True* case in which the Get Image VI resides to set additional camera parameters if a control on the front panel is enabled.



With this code, if the *Set Parameter* Boolean is True, then the value of the *Image Size* control is read and sent to the Camera. If the *Set Parameter* Boolean is False, then no parameters are changed, and the Camera Set Size VI is not called. In this manner, you can update camera parameters while your program is running without incurring the performance problems of always updating those parameters every time through the loop. This method is expandable in that you can make the decision to set a parameter based on a value in another part of your program, or use an enumerated control to select among multiple parameters to be set.




## 1.6 Image Processing

For information on how to add machine vision and image processing code to the Image loop, and how to use tools such as the Vision Assistant to rapidly prototype image processing functions, refer to the Image Processing in LabVIEW for FRC tutorial.

## 1.7 Conclusion

Congratulations! You now know how to set up and acquire images from your camera using the default LabVIEW FRC projects. With the overview you just saw of the available image acquisition and processing tools, you are now ready to explore more advanced uses for the images you are acquiring. With these tools, you'll be well positioned to leverage the power of vision to build a better robot!