Generic IP Camera Driver

Information Sheet for Crimson v3.0+

Compatible Devices

- IP cameras and web cameras where the static image is accessible through a web interface in either JPEG or bitmap formats.

Verified Device

- Axis P1214-E Network Camera

Overview

The Generic IP Camera driver provides to ability to retrieve static images from IP cameras that provide a web interface. These images can then be displayed using Crimson 3's camera primitives. In general, if it is possible to view the camera's current image as a JPEG or bitmap image via a web browser, then this driver will be able to retrieve the image as well.

The Generic IP Camera driver is only supported on Graphite series HMIs.

Configuration

IP Address

This field is for the IP address of the target camera.

Port Number

This is the field for the TCP port number that the camera is listening on.

Image URI

This field is the URI where the image is located on the camera's web server. This field may also include a query string to pass options to the camera. The exact format of the URI will depend on the model of camera. Consult the camera documentation for the appropriate value to use.

An example of the Image URI:

On the Axis camera tested, the default IP address is 192.168.0.90 and the image URI is /axis-cgi/jpg/image.cgi, as shown below:
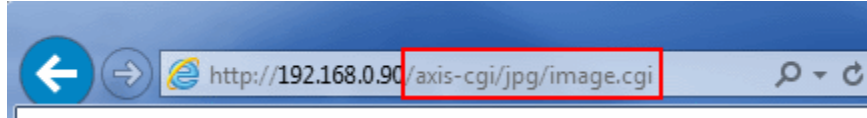
Therefore, "/axis-cgi/jpg/image.cgi" should be entered into the field as shown:
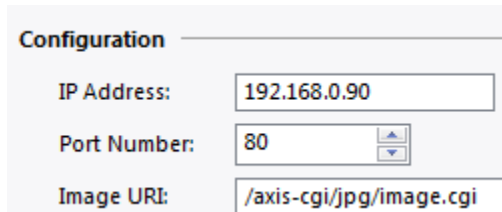

Image Format

This option determines what image format will be used to decode the image:

Auto-detect will attempt to determine the image type automatically and decode it appropriately.

JPG decodes images in the JPEG format.

BMP decodes images in the Windows Bitmap format.

Image Timeout

This field indicates the amount of time that the driver will wait for an image to be returned from the camera before assuming that there is a problem with the connection. Setting this field to zero will disable the image timeout.

Update Method

This field indicates the method by which the driver will request images. It has two possible values:

Periodic will request images as specified by the Update Period, described below.

On Demand will request images only when requested by calling the DevCtrl user function with function code 3. This process is outlined in more detail in the User Functions section below. Additional information can also be found in the Crimson 3.0 user manual.

Update Period

This field specifies the interval in seconds that the driver will wait between image requests. This field is only used when the update method is set to Periodic. Setting this field to zero will cause the driver to request images as quickly as possible.

Image Scaling

This field specifies a scaling factor for the image, given as a percentage. Scaling is only supported for images in the JPEG format. This field is only honored when the image format field is set to *JPG*.

Authentication

The driver supports basic and digest HTTP authentication. The choice between basic and digest is dependent on the desired security level as well as the types supported by the camera itself.

None disables authentication.

Basic authentication encodes the credentials, but does not encrypt or otherwise obscure them. Therefore, it is less secure than digest, but may be faster in some situations.

Digest authentication hashes the credentials so that they are not sent in clear text. This makes it more secure than basic, but it may be slower in some situations.

<u>Username</u>

If HTTP authentication is in use, enter the username for logging onto the camera into this field.

<u>Password</u>

If HTTP authentication is in use, enter the password for logging onto the camera into this field.

User Functions

The following functions are accessed by calling the DevCtrl function. DevCtrl is a function used by Crimson 3.0 in order to send device-specific commands.

The syntax for a call to DevCtrl is as follows:

```
Value = DevCtrl(Device Number, Function Number, Command-specific String)
```

The number to be placed in the Device Number argument to identify the device can be viewed in the status bar of the Communications category when the device name is highlighted. In the examples given below, 1 will be used as the device number.

The following commands are available by calling the DevCtrl function with the appropriate function number.

<u>Changing the target IP Address</u>

Function Number: 1
Return value: 1 if successful, otherwise 0.

This function is used to change the target IP address for the driver at runtime. The new IP address is given as a string formatted in dot-decimal notation.

Example:

```
Value = DevCtrl(1, 1, "192.168.1.100");
```

<u>Retrieving the Current IP Address</u>

Function Number: 1
Return value: the current IP address for the target device as an integer.

This function retrieves the current IP address that the target device is set to. The return value can formatted in dot-decimal by reading the return value into a tag that is formatted as an IP address. The command-specific string parameter is unused by this function.

Example:

```
IPAddress = DevCtrl(1, 2, "");
```

<u>Manually Trigger a Camera Update</u>

Function Number: 3
Return value: Always returns 0.

This functionality is only used when the update method is set to *On Demand*. When called, the driver will request a new image from the target camera. The return value and command-specific string are unused by this function.

Example:

```
DevCtrl(1, 3, "");
```

Cable Information

Ethernet – Standard Ethernet cable as appropriate for the IP camera.

Revision History

7/11/2016 – Created