Scion Corporation

Scion 1394 LabView VI

Scion Corporation

Table Of Contents

Scion Corporation

Getting Started

Introduction

This manual describes the Scion 1394 LabVIEW VI's for Windows. The VI's support all Scion Monochrome and Color FireWire cameras. Included with this release are VI's for controlling all Scion FireWire Cameras packed into a library file (llb), example VI's for demonstrating how to capture images into LabVIEW, and a specialized DLL (sfwlv.dll) for interface support.

Please take a few moments to read through this manual before you begin using the VI's, as it should answer some questions that you may have. Please contact Scion Corporation should you encounter difficulty at any time, or if you have any questions.

System Requirements

To use the program, you need:
- A Pentium IV or higher processor for use with Microsoft Windows 2000, Windows XP or Windows Vista
- Scion Monochrome or Color FireWire camera
- 256 Mb of RAM (512 or more MB recommended)
- FireWire or IEEE-1394 port
- Scion FireWire Camera Library (sfwlib.dll)
- LabVIEW version 8.0 or higher
- NI-IMAQ 3.5
- Vision Development Kit Required for Grab abilities

Scion Corporation

Example VI's

Included are two example VI's that can be modified or used as a reference. They are located in the /Program Files/Scion Corporation/Scion 1394 LabView VI/ folder.

Snap Example.vi
This example VI will open the 1394 interface and a connected Scion Camera and capture a single image to the image window. The image can then be saved for later use. This example also shows how to use the Set_Camera_Param.vi and Set_Camera_Fparam.vi to modify the camera settings before the capture takes place.

Grab Example.vi
This example will open the 1394 interface and a connected Scion Camera like the above example, but will allow the image to continuously update until the stop button is selected. This shows how to use the Set_Camera_Param.vi and Set_Camera_Fparam.vi in a continuous capture environment. The image displayed always has a bit depth of 8 bits even if the bit depth is changed. This allows for a greater frame rate. But, when the stop button is selected the full bit depth image is then captured and displayed. This requires the Vision Development Kit to be installed to operate.

Included VI's

All of the camera control VI's are located in the sfwlv.llb library. All of the below VI's can be accessed by going to the "User Libraries" section of the "Functions" palette. This file can also be found in the LabVIEW folder in /user.lib/scion/. Below is a list and a short description of each camera control VI.

Open_1394_Interface.vi
Use this to open the FireWire interface to the camera. This will return an Interface Handle that can be used to open a connected camera.

Close_1394_Interface.vi
This will close the FireWire interface to the camera. Note that the Close_Camera.vi must be called before this.

Open_Any_Camera.vi
This will open the first Scion FireWire camera on the bus. The Open_1394_Inteface.vi must be called before this. A Camera handle will be returned which can be used to set the camera parameters and settings.

Close_Camera.vi
Use this to close the current camera.

Get_Camera_Param.vi
Returns camera configuration settings based on the input parameter. Gain, exposure, contrast, etc. can be read with this module.

Get_Camera_Param_Default.vi
Returns the default value of the selected input parameter.

Get_Camera_Param_Min.vi
Returns the minimum value of the selected input parameter.

Get_Camera_Param_Max.vi
Return the maximum value of the selected input parameter.

Get_Camera_Fparam.vi
This will return camera configuration settings based on the input parameter into a double value. Currently the Gamma value is the only supported parameter.

Get_Camera_Fparam_Default.vi
This will return the default value of the selected input parameter into a double value. Currently Gamma is the only supported input parameter.

Get_Camera_Fparam_Min.vi
This will return the minimum value of the selected input parameter into a double value. Currently Gamma is the only supported input parameter.

Get_Camera_Fparam_Max.vi
This will return the maximum value of the selected input parameter into a double value. Currently Gamma is the only supported input parameter.

Get_Camera_Info.vi
Retrieves camera information from the connected camera, such as CCD width, height and bit depth.

Get_Product_Desc.vi
Gets the product description of the camera.

Get_Product_Prefix.vi
This will return the product prefix or model number of the connected camera.

Get_Vendor_Desc.vi
Returns the vendor description of the connected camera. Generally this will return "Scion Corporation".

Get_Vendor_Prefix.vi
This will return the vendor prefix of the connected camera. Generally this will return "Scion".

Get_Param_Code.vi
This is a helper VI that will take the parameter key and value from the Get_Camera_Param modules and convert them to a readable string value.

Set_Camera_Param.vi
Use this to set the camera parameters or settings. Settings include gain, exposure, contrast, etc.

Set_Param_Code.vi
Helper VI that will take the parameter keys from the Set_Camera_Param.vi and convert them to a readable string.

Set_Camera_Fparam.vi
Loads the provided double parameter value into the camera. Currently the only fparam setting available is the Gamma.

Set_Fparam_Code.vi
Helper Vi that will take the fparam keys and convert to a readable string.

Snap_Image.vi
Use this to snap a single image from the connected camera. This will take an image from the camera and display the image in an IMAQ created image buffer. The Open_1394_Interface.vi and Open_Any_Camera.vi must be called before this.

Grab_Setup.vi
This will start the grab sequence. It will create buffers and pointers to be used in the Grab_Image.vi.

Grab_Image.vi

Use this in a while loop to get a continuous image on the screen. The image that is displayed will always be an 8-bit image. This allows the capture rates to be higher. When the while loop is stopped the Grab_End.vi will then take an image with the currently set bit depth.

Grab_End.vi

Use this after the Grab_Image.vi outside the while loop. This will take an image with the current settings and bit depth and display it to the screen.

Return_Code.vi

This is a return or error code module to track any errors encountered.

sfwlv.dll Reference

Below is a reference to the commands that are available in the sfwlv.dll. This is a reference for advanced users only.

SnapImg

The SnapImg function will take a snapshot image.

int SnapImg(

DWORD	Interface_Handle,
DWORD	Camera_Handle,
unsigned long	LVImagePtr,
int	LVLineWidth,
unsigned int	Width,
unsigned int	Height,
int	BitDepth

);

Parameters

Interface_Handle

> 1394 Interface Handle. Open_1394_Interface.vi will create this handle.

Camera_Handle

> Camera Interface Handle. Open_Any_Camera.vi will create this handle.

LVImagePtr

> Address for destination image buffer. Also known as the Pixel Pointer. This address is created from the IMAQ GetImagePixelPtr.vi.

LVLineWidth

> Width in pixels of image with border added. This is obtained from IMAQ GetImagePixelPtr.vi.

Width

> Width of the actual image in pixels without borders. This could be different than the CCD Width because of Binning and Preview.

Height

> Height of the actual image in pixels without borders. This could be different than the CCD Height because of Binning and Preview.

BitDepth

Bit depth of the image to capture.

Return Values

If the function succeeds, then function returns SLV_OK. If the function does not succeed one of the following codes is returned:

SLV_NOT_ENOUGH_MEMORY   Not enough memory to create buffer
SLV_FRAME_TIMEOUT   Frame capture timed out
SLV_ERROR_GETTING_FRAME   Camera could not complete frame

Remarks

SetUpGrab

The SetUpGrab function will create buffers and pointers for the Grab routine to follow.

int SetUpGrab(

DWORD	Interface_Handle,
DWORD	Camera_Handle,
unsigned int	Width,
unsigned int	Height

);

Parameters

Interface_Handle

> 1394 Interface Handle. Open_1394_Interface.vi will create this handle.

Camera_Handle

> Camera Interface Handle. Open_Any_Camera.vi will create this handle.

Width

> Width of the actual image in pixels without borders. This could be different than the CCD Width because of Binning and Preview.

Height

> Height of the actual image in pixels without borders. This could be different than the CCD Height because of Binning and Preview.

Return Values

If the function succeeds, then function returns SLV_OK. If the function does not succeed one of the following codes is returned:

SLV_NOT_ENOUGH_MEMORY   Not enough memory to create buffer

Remarks

Because of the method of capture in the Grab routines, this set of calls can only be used by a single camera. Multiple cameras are not supported simultaneously.

Grab

If placed in a LabVIEW While loop the Grab function will continuously grab frames to the image buffer.

int Grab(

DWORD	Interface_Handle,
DWORD	Camera_Handle,
unsigned long	LVImagePtr,
int	LVLineWidth,
unsigned int	Width,
unsigned int	Height,
int	BitDepth

);

Parameters

Interface_Handle

1394 Interface Handle. Open_1394_Interface.vi will create this handle.

Camera_Handle

Camera Interface Handle. Open_Any_Camera.vi will create this handle.

LVImagePtr

Address for destination image buffer. Also known as the Pixel Pointer. This address is created from the IMAQ GetImagePixelPtr.vi.

LVLineWidth

Width in pixels of image with border added. This is obtained from IMAQ GetImagePixelPtr.vi.

Width

Width of the actual image in pixels without borders. This could be different than the CCD Width because of Binning and Preview.

Height

Height of the actual image in pixels without borders. This could be different than the CCD Height because of Binning and Preview.

BitDepth

Bit depth of the image to capture.

Return Values

If the function succeeds, then function returns SLV_OK. If the function does not succeed one of the following codes is returned:

SLV_FRAME_TIMEOUT   Frame capture timed out
SLV_ERROR_GETTING_FRAME   Camera could not complete frame

Remarks

Because of the method of capture in the Grab routines, this set of calls can only be used by a single camera. Multiple cameras are not supported simultaneously.

GrabEnd

The GrabEnd ends the grab routines by releasing all buffers and memory addresses.

int GrabEnd(

DWORD Interface_Handle,
DWORD Camera_Handle

);

Parameters

Interface_Handle

1394 Interface Handle. Open_1394_Interface.vi will create this handle.

Camera_Handle

Camera Interface Handle. Open_Any_Camera.vi will create this handle.

Return Values

If the function succeeds, then function returns SLV_OK. If the function does not succeed one of the following codes is returned:

SLV_BUFFER_RELEASE_ERROR   Problems releasing buffers

Remarks

Because of the method of capture in the Grab routines, this set of calls can only be used by a single camera. Multiple cameras are not supported simultaneously.