alliance vision

LabVIEW toolkit

for Dalsa Sapera LT framegrabbers

version 7.4

LabVIEW™ Toolkit for Sapera LT™ compatible framegrabbers

Software Version 7.4
Document Version 4.0
March 2014

Alliance Vision
7 Avenue du meyrol
26200 Montélimar FRANCE
www.alliancevision.com

Conventions

[..]	Two numbers in square brackets and separated by two points represent a sequence of numeric values e.g. [1..100] means all the numbers from 1 to 100 included.
<Ctrl>	A keyboard symbol between two trangular brackets indicates a key board short cut. A key board short cut obtained by simultaneously pressing two or more keys at the same time is represented by the keyboard symbols separated by a ± sign e.g. **<Ctrl + Alt + Del>**
>	The » symbol leads you through nested menu items, dialog box options, and LabWindows function palettes to a final action. For example, a VI name followed by the sequence Vision Utilities » Image Management directs you to look for the VI in the Image Management subpalette of the Vision Utilities function palette.
	This icon appears next to helpful comments for the user.
	This icon appears next to helpful advice for the user.
	This icon will appear next important information giving one or many actions to be undertaken in order to avoid loosing data, or crashing the system, the software or equipment.
Bold	Bold text denotes items that you must select or click on, such as menu items and dialog box options. Bold text may also denote parameter names.
`monospace`	Text in this font are the possible variables and syntax examples to be used. This font is also used for the proper names text or characters that you should enter from the keyboard, sections of code, programming denotes of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, operations, variables, filenames and extensions, and code excerpts

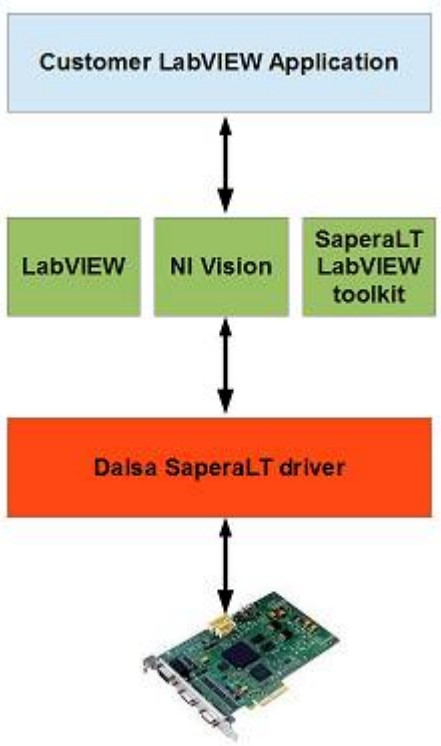
Contents

Chapter 1 : Overview

1.1 About the library

This manual contains the information you need to get started with the "SaperaLT LabVIEW Toolkit". To use this manual, you already should be familiar with at least LabVIEW programming environment and with Windows 7/8/XP/Vista operating systems.

The "SaperaLT LabVIEW Toolkit" is a software tool for interfacing some Teledyne DALSA™ image acquisition boards, with the National Instruments LabVIEW environment. This software will allows you to build LabVIEW applications based on Teledyne DALSA frame-grabbers.



The "SaperaLT LabVIEW toolkit" includes the following functionalities :

- enumerating the available compatible boards.
- Acquire and transfer images from a board to the PC host memory, using single-shot or continuous modes.
- Synchronize image acquisitions with external electrical trigger events.
- Display live images acquired from the board.
- Set/Get basic board parameters.

1.2 Hardware Requirements

- PCI-bus IBM PC or compatible with Pentium class or later processor
- One of the following Sapera LT compatible boards :
    - **X64 Series** (X64-CL, X64-iPro, X64-LVDS, X-64 Xcelera, X64-AN).
    - **PC2 Series** (PC2-Vision, PC2-Camlink, PC2-Comp).
    - **Xtium-CL**

1.3 Software Requirements

- One of the following supported operating system :
    - Microsoft Window 8®
    - Microsoft Window 7®
    - Microsoft Windows XP®

- Sapera LT 7.4 runtime libraries and boards low-level drivers (supplied)
- National Instruments LabVIEW 8.5 or later (for older versions support, please contact us).
- National Instruments "NI Vision Development Module", or "NI Vision Common Resources".

- The National Instruments "NI Vision Development Module" library is an image processing software add-on for LabVIEW.
- The National Instruments "NI Vision Development Module" library is recommended but not needed. If you do not have a full "NI Vision Development Module" license, you will, at least, have to install the "NI Vision Common Resources". The "NI Vision common resources" are a free subset of "NI Vision Development Module", available for free on the National Instrument web site.

Chapter 2 : Installation and Configuration

2.1 Sapera LT software and drivers installation

- **Launch "\Runtimes\SaperaLT\Teledyne Dalsa Sapera LT 7.40 RT with Camexpert.exe"**, supplied on the CD-ROM. (this installs all the necessary run-time libraries, and camera configuration files into the "C:\Program Files\Teledyne Dalsa\" folder).
- Shutdown the PC.
- Plug your frame-grabber in a PCI slot.
- Reboot the PC.
- Install the appropriate board low-level driver.
- Check the board is detected in the windows hardware device manager.
- Check the board is detected using Sapera "CamExpert" software tool.
- Using "CamExpert", setup your own camera/framegrabber configuration file, and save it as a ccf file.

2.2 Board and camera setup

Sapera LT applications need to load an appropriate acquisition configuration file before acquiring images from a camera. The user camera configuration files have the extensions **.ccf**.

The Sapera LT software supplies a set of predefined camera files for the most common camera types. The Sapera LT software also supplies a software called "CamExpert" which allows to create, modify and test **.ccf** camera files. Use "CamExpert" to generate a **.ccf** file.

Please consult the Sapera LT user manual for more information on how to use the "CamExpert" tool.

2.3 "Sapera LT LabVIEW Toolkit" Installation

- **Launch "\Runtimes\Microsoft\vcredist_x86.exe" (for 32-bit sytem),** or
  Launch "\Runtimes\Microsoft\vcredist_x64.exe" (for 64-bit sytem)
  This file installs necessary Microsoft runtime libraries.
- **Launch "\Sapera Toolkit x86 Setup.exe" (for 32-bit sytem)**, or
  Launch "\Sapera Toolkit x64 Setup.exe" (for 64-bit system).
  This file installs all the necessary LabVIEW libraries needed to interface your framegrabber ("Sapera.llb").
- The different LabVIEW libraries (one for each LabVIEW version) are installed under the following folder by default :

  "C:\Program Files\Alliance Vision\Sapera Toolkit\"
- To complete the installation, yit is possible to manually copy the LabVIEW library to the LabVIEW user libraries folder, in order to make the library functions accessible under the LabVIEW functions palette. The LabVIEW user libraries folder is usually :

  "C:\Program Files\National Instruments\LabVIEW x.x\user.lib\"

2.4 Quickstart and examples

The “SaperaLT LabVIEW Toolkit” includes demonstrations example Vi's, which are accessible in the LabVIEW functions palette. These Vi's would be a good introduction on using the "SaperaLT LabVIEW Toolkit". You can freely use or modify these Vi's in order to suit your needs.

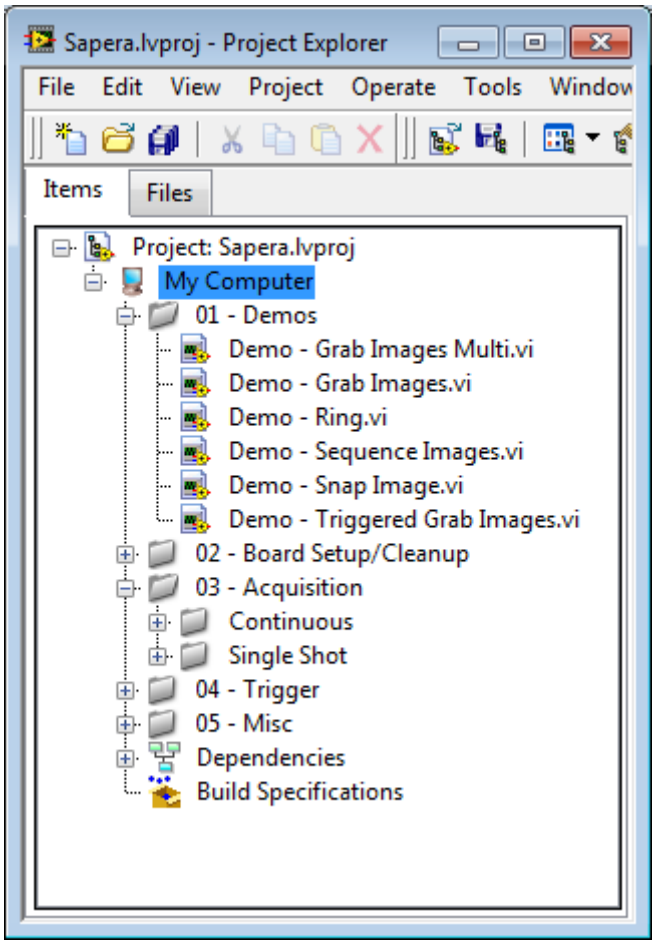
Demonstrations vi's supplied are :

- “Demo 1 - Snap image.vi” : (acquire one image in single shot mode)
- “Demo 2 – Sequence Images.vi” : (acquire n images in single shot mode)
- “Demo 3 – Grab Images.vi” : (acquire n images in continuous mode)
- “Demo 4 – Triggered Grab Images.vi” : (acquire n images in triggered continuous mode)
- “Demo 5 –Grab Images Multi.vi” : (acquire n images in continuous mode from two cameras)
- “Demo 6 – Ring.vi” : (acquire n images in continuous mode with a ring buffer)

Chapter 3 : Software Reference

3.1 LabVIEW Vi's reference

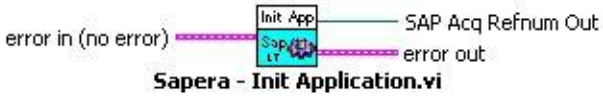
A LabVIEW project, is supplied along with the LabVIEW toolkit.



The LabVIEW toolkit Vi's falls into different categories:

- 01 – Demos: Exemple demonstrations vi's
- 02 – Board Setup/Cleanup: vi's for enumerating, initializing and setup the board.
- 03 – Acquisition: vi's for acquiring images using different modes.
- 04 – Trigger: vi's for enable/disable external or software trigger synchronization.
- 05 – Misc: miscellaneous vi's

3.2.1 Sapera - Init Application.vi



This vi initializes the library, and returns a Sapera refnum which can be passed to the other vi's of the library. You must call "Sapera – Exit Application" at the end of your application when you no longer need the Sapera Refnum.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

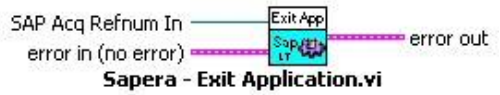
SAP Refnum Out is a unique refnum that identifies a library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera Exit application.vi"

3.2.2 Sapera - Exit Application.vi



This vi disposes a refnum which was previously return by "Sapera – Init Application.vi". You must call this function at the end of your application in order to release all resources associated with the library.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

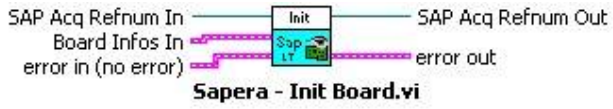
- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera Init Application.vi"

3.2.3 Sapera - Init Board.vi



Initializes a frame-grabber You must call "Sapera – Close Board.vi" when you no longer need to use the board.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Board Infos In is a LabVIEW cluster which contains necessary information concerning the board to initialize,

- **Board Name** is a string which contains the unique name of the board (Example: X64_1, for the first X64 board).
- **Channel** is a zero based number which identifies the board channel where the camera will be connected.
  (Example: if the board has four channels, **Channel** is a number between 0 and 3).
- **Configuration File** is a string which contains the full path-name to the camera configuration file (.ccf or .cca file), that contains the video signal settings, and the default acquisition parameters (Example : "C:\DALSA\Sapera\camfiles\user\foo.ccf").

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera Close Board.vi"

3.2.4 Sapera - Close Board.vi

Releases the current board resources and closes the board previously initialized by "Sapera – Init Board.vi".

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

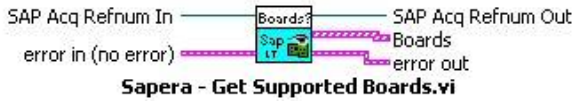
SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera Init Board.vi"

3.2.5 Sapera - Get Supported Boards.vi



Returns some useful information about all the supported boards and channels.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Boards is an array of LabVIEW cluster. Each element of the array contains useful information about one supported boards.

- **Name** is a string which contains the unique name of the board (Example: X64_1, for the first X64 board).
- **Serial Number** is a string which contains the unique serial number of the board .
- **Channels** is an array of string, each element in the array contains the name of a supported channel.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

3.2.6 Sapera - Get Channel Infos.vi

Returns information about the current selected channel, of the current board. This vi returns information concerning the video signal settings : horizontal and vertical resolution (pixels), bit-depth (bits/plane), Image type (mono8, mono16 or RGB32).

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

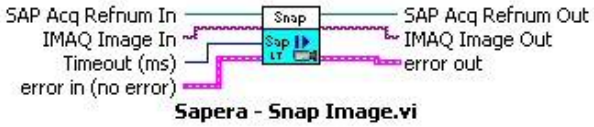
Channel Infos returns information concerning the video signal settings of the current channel.

- **ResX** is the horizontal resolution (pixels).
- **ResY** is the vertical resolution (pixels).
- **Bit-depth** is the bit-depth (bits).
- **ImageType** is the image type (supported image types are mono8, mono16 and RGB32).

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

3.2.7 Sapera - Snap Image.vi



Sapera - Snap Image.vi

Snap Image acquisition. The vi starts a single shot acquisition of a single image, and waits for the image acquisition completion. If the image has not been acquired before the defined timeout in milliseconds, the vi returns an error.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

NI Vision Image In is a refnum to a valid NI vision image. The NI Vision image must match the required type and size (see "Sapera – Get Channel Infos").

Timeout ms is a timeout delay in milliseconds.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

NI Vision Image Out is a refnum to a valid NI vision image. This is the same refnum as **NI Vision Image In**.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

3.2.8 Sapera - Sequence.vi

Sequence Image acquisition. The vi starts a single shot acquisition of a specified number of images, and waits for these images acquisition completion. If the images has not been acquired before the defined timeout in milliseconds, the vi returns an error.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Array Images In is an array of refnums to valid NI vision images. The NI Vision images must match the required type and size (see "Sapera – Get Channel Infos").
If the array contains 10 image refnums, the vi will then capture 10 images in a single shot.

Timeout ms is a timeout delay in milliseconds.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

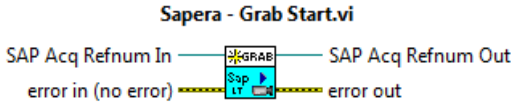
SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Array Images Out is an array of refnums to a valid NI vision images. This is the same refnums as **Array Images In**.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

3.2.9 Sapera – Grab Start.vi



Starts a continuous grab acquisition. The "Sapera LT driver for LabVIEW" will then continuously acquires images in a background thread. Then you can call "Sapera – Grab.vi", in order to copy the most recent image to a NI Vision image. You can also stop the continuous acquisition with "Sapera – Grab Stop.vi".

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

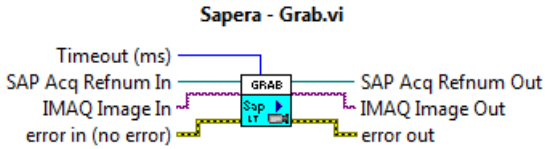
SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera - Grab Stop.vi", "Sapera – Grab.vi"

3.2.10 Sapera – Grab.vi



Copies the most recent image from a continuous grab acquisition, into a NI Vision image. Before using this vi, you must first have called 'Sapera Grab Start.vi". You can also stop the continuous acquisition with "Sapera – Grab Stop.vi".

Timeout defines the delay to wait before throwing a timeout error.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

NI Vision Image In is a refnum to a valid NI vision image. The NI Vision image must match the required type and size (see "Sapera – Get Channel Infos").

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

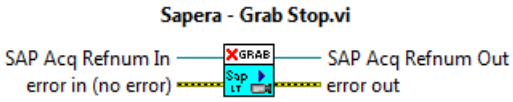
NI Vision Image Out is a refnum to a valid NI vision image. This is the same refnum as **NI Vision Image In**.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera - Grab Start.vi", "Sapera – Grab Stop.vi"

3.2.11 Sapera – Grab Stop.vi



Stops a continuous acquisition, previously started with "Sapera – Grab Start.vi".

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

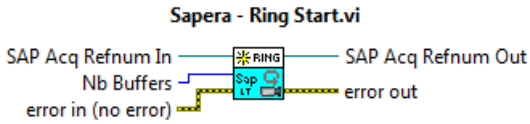
SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera - Grab Start.vi", "Sapera – Grab.vi"

3.2.12 Sapera – Ring Start.vi



Starts a continuous ring acquisition. The "Sapera LT driver for LabVIEW" will then continuously acquires images in a background thread, using multiples buffers. Then you can call "Sapera – Ring.vi", in order to copy one of the ring buffer to a NI Vision image. You can also stop the continuous acquisition with "Sapera – Ring Stop.vi".

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Nb Buffers define the buffer count of the ring for storing images sequentially.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

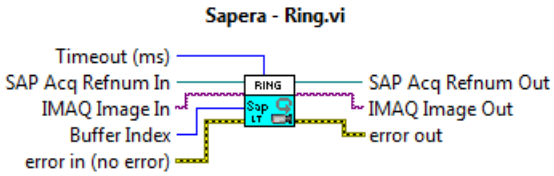
SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera - Ring Stop.vi", "Sapera – Ring.vi"

3.2.13 Sapera – Ring.vi



Copies the most recent image from a continuous ring acquisition, into a NI Vision image. Before using this vi, you must first have called 'Sapera Ring Start.vi". You can also stop the continuous acquisition with "Sapera – Ring Stop.vi".

Timeout defines the delay to wait before throwing a timeout error.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

NI Vision Image In is a refnum to a valid NI vision image. The NI Vision image must match the required type and size (see "Sapera – Get Channel Infos").

Buffer Index is a buffer index. The possible buffers indexes are:

- a number between 0 and "Number of Buffers" minus 1: returns the specified buffer.
- -1: returns the newest buffer in the ring.
- -2: returns the oldest buffer in the ring.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

NI Vision Image Out is a refnum to a valid NI vision image. This is the same refnum as **NI Vision Image In**.

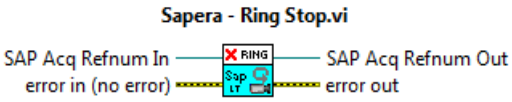
Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

Timeout defines the delay to wait before throwing a timeout error.

See also : “Sapera - Ring Start.vi”, “Sapera – Ring Stop.vi”

3.2.14 Sapera – Ring Stop.vi



Stops a continuous ring acquisition, previously started with "Sapera – Ring Start.vi".

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

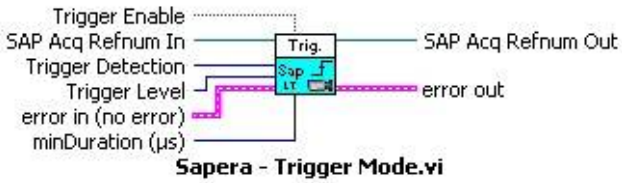
SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera - Ring Start.vi", "Sapera – Ring.vi"

3.2.15 Sapera – Trigger Mode.vi



Configures the current external trigger line. You must call this vi, before starting an acquisition in order to synchronize the acquisition with external electrical trigger events.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Trigger Enable enables(TRUE), or disables (FALSE) the external trigger event synchronization mechanism.

Trigger Detection configures the trigger detection method, supported values are :

- Detect Active low - Detect Active High - Rising edge - Falling edge

Trigger Level configures the trigger level parameter, supported values are :

- TTL, RS-422, LVDS

Min. Duration is the minimum duration of an external event in microseconds (µs). Default value is 0.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

status is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.

code is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.

source is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

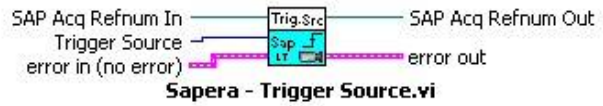
status is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.

code is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.

source is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera – Trigger Source.vi", "Sapera – Generate Software Trigger.vi"

3.2.16 Sapera – Trigger Source.vi



Sets the external trigger line, which will receive external trigger events.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previoulsy returned by "Sapera - Init Application.vi"

Trigger Source is the zero based index of the external trigger line. The maximum number of supported lines may depend of the board model.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera – Trigger Mode.vi", "Sapera – Generate Software Trigger.vi"

3.2.17 Sapera – Generate Software Trigger.vi

Generates a trigger event by software. This vi simulates an external trigger event.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previoulsy returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera – Trigger Mode.vi", "Sapera – Trigger Source.vi"

3.2.18 Sapera -Live Window.vi

Displays the live image from a continuous grab acquisition into a window. Before using this vi, you must first have called "Sapera – Grab Start.vi".

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previoulsy returned by "Sapera - Init Application.vi"

Show Window shows (TRUE), or hides (FALSE) the display window.

Window title is the string which will appears in the display window title bar.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Window Handle is the system window handle of the display window. This handle can be used with the Win32 API's in order to precisely configure the window appearance, positon, etc...
(see Win32 API's : MoveWindow, SetParent, SetWindowPosition, SetWindowLong, etc....)

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

3.2.19 Sapera – Select Video Source Dialog.vi

Displays a user friendly dialog box, which allows to easily select the board, channel and camera configuration file to use for the image acquisition. This vi then returns a "Board Infos Out" cluster, which can be directly connected to the "Board Infos In" connector of the "Sapera – Init Board.vi".

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previoulsy returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

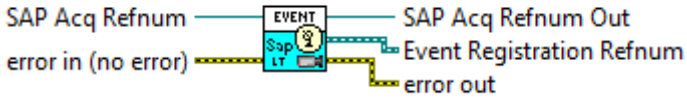
Board Infos Out is a LabVIEW cluster which contains necessary informations concerning the board to initialize,

- **Board Name** is a string which contains the unique name of the board (Example: X64_1, for the first X64 board).
- **Channel** is a zero based number which identifies the board channel where the camera will be connected.
  (Example: if the board has four channels, **Channel** is a number between 0 and 3).
- **Configuration File** is a string which contains the full pathname to the camera configuration file (.ccf or .cca file), that contains the video signal settings, and the default acquisition parameters (Example : "C:\Coreco\Sapera\camfiles\user\foo.ccf").

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

3.2.20 Sapera – Register Frame Grabbed Event.vi



Register a "Frame Grabbed" LabVIEW dynamic event. This is useful , if you want to use a LabVIEW event structure. Connect the "Event Registration Refnum" output to a LabVIEW event structure "dynamic event terminal". Then right click on the LabVIEW event structure, and add an event case for "*Dynamic <Buffer Index>: User event*". Then, each time a new frame will be available, SaperaLT will automatically generate this event, and that will callback code in the LabVIEW event stucture case.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previoulsy returned by "Sapera - Init Application.vi"

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

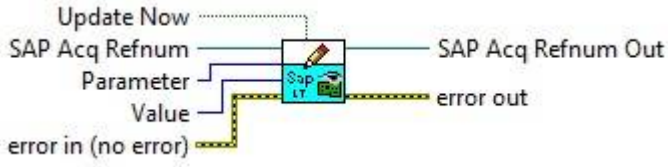
- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

SAP Refnum Out is a unique refnum that identifies a Sapera library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one subVI to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

3.2.21 Sapera – Set Board Parameter.vi



This vi sets a board parameter to the specified value.
Please note that some parameters should be not changed during an acquisition. Changing some parameters during an acquisition can cause unexpected behaviors.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previoulsy returned by "Sapera - Init Application.vi"

Parameter is a parameter identifier number. Please refer to the SaperaLT user manual, or to the vi online documentation for a complete liste of supported parameter id's.

Value is the new parameter value.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

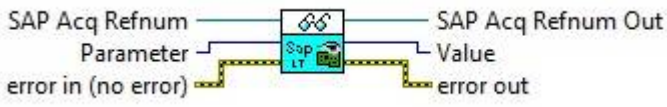
SAP Refnum Out is a unique refnum that identifies a library instance.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera Get Board Parameters.vi"

3.2.22 Sapera – Get Board Parameters.vi



This vi gets a board parameter value.

SAP Refnum In is a unique refnum that identifies a Sapera library instance. The SAP Refnum was previously returned by "Sapera - Init Application.vi"

Parameter is a parameter identifier number. Please refer to the SaperaLT user manual, or to the vi online documentation for a complete liste of supported parameter id's.

Error in (no error) is a cluster that describes the error status before this VI executes. If **error in** indicates that an error occurred before this VI was called, this VI may choose not to execute its function, but just pass the error through to its **error out** cluster. If no error has occurred, then this VI executes normally and sets its own error status in **error out**. Use the error handler VIs to look up the error code and to display the corresponding error message. Using the **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

Value is the parameter value.

Error out is a cluster that describes the error status after this VI executes. If an error occurred before this VI was called, **error out** is the same as **error in**. Otherwise, **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs to look up the error code and to display the corresponding error message. Using **error in** and **error out** clusters is a convenient way to check errors and to specify execution order by wiring the error output from one sub Vi to the error input of the next.

- **status** is TRUE if an error occurred before this VI was called, or FALSE if not. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code.
- **code** is a number identifying an error or warning. If **status** is TRUE, **code** is a nonzero error code. If **status** is FALSE, **code** is zero or a warning code. Use the error handler VIs to look up the meaning of this code and display the corresponding error message.
- **source** is a string that indicates the origin of the error, if any. Typically, **source** is the name of the VI in which the error occurred.

See also : "Sapera Set Board Parameter.vi"

Chapter 4 : Software Deployment

In order to redistribute your custom LabVIEW software, which uses "SaperaLT for LabVIEW" toolkit, you should redistribute and install the following files on the deployment machine:

- **Microsoft Visual Studio 2008 SP1 runtime**:
  "Runtimes\Microsoft\vcredist_x86.exe" for 32-bit operating system, or
  "Runtimes\Microsoft\vcredist_x64.exe" for 64-bit operating system.
  The "Microsoft Visual Studio 2008 SP1 Runtime" can be freely redistributed free of charge.
- **Sapera LT 7.4 Runtime with CamExpert**.
  The "SaperaLT Runtime" can be freely redistributed free of charge.
- **Alliance Vision SaperaLT LabVIEW toolkit runtime**:
  AVSapera.dll

The file AVSapera.dll is installed in the "C:\Windows\System32 " folder of the development machine. It is advise to copy it in the windows system folder of the deployment machine. Then the file "AVSapera.dll" must be registered on the deployment machine.

To register the file, you can use the following command :

```
regsvr32.exe AVSapera.dll
```

To unregister the file, you can use the following command :

```
regsvr32.exe /u AVSapera.dll
```

The "SaperaLT Toolkit for LabVIEW" library, is supplied with a runtime license which allows you to build and deploy one executable, if you want to deploy more executables based on the "Sapera Toolkit for LabVIEW" llibrary, please contact Alliance Vision. (sales@alliancevision.com)

Chapter 5 : Frequently Asked Questions

1. **How to send commands to a CameraLink camera, using the Alliance Vision LabVIEW Toolkit ?**
   It is not possible to send command directly from the Alliance Vision toolkit. However you can do this by mapping the framegrabber serial port to a Windows serial port.
   - Launch the "C:\Program Files\Teledyne Dalsa\Sapera\Bin\SapConf.exe" software
   - Map the framegrabber physical port to a Windows serial port (COM2, COM3, etc...).
   - Then use the LabVIEW Serial Vi's ("/Data Communication/Protocols/Serial") to send specific command to the camera.
   - Consult the camera technical manual, in order to get more informations about the camera commands and communication protocol.

2. **The toolkit doesn't work on a 64-bit operating system, I get an error "Dll failed to load".?**
   Probably, you haven't installed the whole 64-bit software tool chain. In order to use the Alliance Vision toolkit on a 64-bit machine, you must install :
   - Windows 64-bit operating system.
   - SaperaLT 64-bit driver.
   - LabVIEW 64-bit development platform.
   - Alliance Vision 64-bit SaperaLT LabVIEW toolkit ("Sapera Toolkit x64 Setup.exe").

3. **The toolkit doesn't install correctly on my machine, The "AVSapera.dll" failed to register?**
   - Log in with "Administrator" account or rights.
   - Disable the Windows User Accouht Control "UAC" in the Windows configuration Panel.
   - Please install the different softwares using the Chapter 2 procedure.
   - Install the softwares specified in Chapter 2, by doing a mouse right-click, then select "Install as Administrator".

Chapter 6 : Technical Support

Alliance Vision
7, avenue du Meyrol - BP184
26200 MONTELIMAR CEDEX
FRANCE

web: www.alliancevision.com

e-mail:
support@alliancevision.com
sales@alliancevision.com

Tel: +033 (0) 4 75 53 14 00
Fax: +033 (0) 4 75 53 14 04