What's new in Version 2.04 of the LabVIEW drivers for pco.cameras

type: Driver update summary

file name: WhatsNewIn204.docx

version: 2.04

release date: 03 September 2015

reference to: LabVIEW driver for pco.camera version 2.04, MA_DCLABVIEWver204.docx

author: MM

available to: All

content / history:

.

This document describes the features that are new to this release. For a detailed description of the full driver package, please see the LabVIEW driver for pco.cameras User Manual.
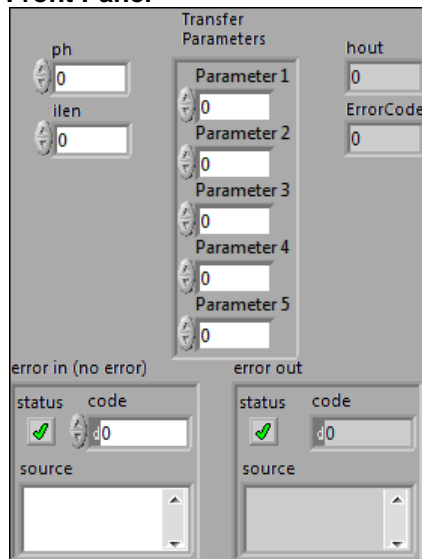
1.1. New in APIManagement.lib:

SetTransferParametersAuto.vi

Automatically sets the transfer parameters for a pco.edge 5.5. This is the recommended function when Soft-ROI is enabled. This function replaces G(S)etTransferParameter.vi and SetActiveLookupTable.vi.

Connector Pane
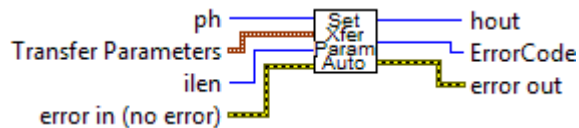
Front Panel

Controls and Indicators

ph Handle for the camera

error in (no error) The **error in** cluster can accept error information wired from VIs previously called.

ilen Total number of bytes in "Transfer Parameters" cluster: default is 0.

Transfer Parameters Interface-specific parameters to control the transfer of data from camera to PC. This can be left uninitialized to set the parameters automatically.
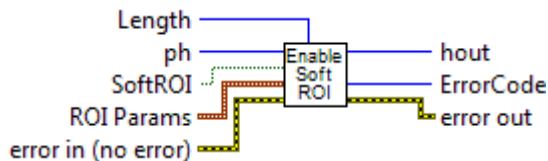

U64 **hout** Handle output

error out The **error out** cluster passes error or warning information out of a VI to be used by other VIs.
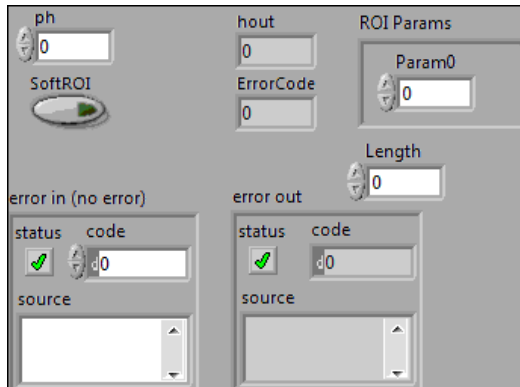
I32 **ErrorCode**

EnableSoftROIMode.vi

Enables Soft-ROI functionality for Soft-ROI capable interfaces. If it is necessary to get a smaller ROI-granularity (e.g. in x-direction it is only possible to set the ROI in steps of 160 pixels with a pco.edge 5.5) this function enables smaller granularity (e.g. a pco.edge 5.5 is reduced to 4 pixels in x-direction). If Soft-ROI is enabled it is recommended to use SetTransferParametersAuto.vi. This makes sure that the camera and interface are set to the correct transfer modes forSoft-ROI. GetTransferParameter.vi, SetTransferParameter .vi and SetActiveLookupTable.vi are replaced by SetTransferParametersAuto.vi . If SetTransferParametersAuto.vi is not used it is mandatory to ensure the correct setup of the transfer parameters (e.g. Soft-ROI is smaller than x=1920, but the camera ROI is bigger than x=1920 due to the granularity of the camera).

Connector Pane

```
  Length ────────┐
      ph ────────┤ Enable ├──── hout
 SoftROI ┄┄┄┄┄┄┤  Soft  ├──── ErrorCode
                 │  ROI   │
ROI Params ══════┤        ╞══ error out
error in (no error) ══════┘
```

Front Panel

```
  ph              hout        ROI Params
  [0]             [0]           Param0
 SoftROI        ErrorCode       [0]
  (●)            [0]
                               Length
                                [0]
error in (no error)    error out
status  code           status  code
 [✓] [0]               [✓]  [0]
source                 source
 [         ]           [         ]
```

Controls and Indicators

U64 **ph** Handle for the camera

error in (no error) The **error in** cluster can accept error information wired from VIs previously called

TF **SoftROI** Enables or disables Soft ROI mode in cameras that have this capability.

FALSE - Disable Soft ROI mode
TRUE - Enable Soft ROI mode.

I32 **Length** Length of the Soft ROI Params structure. Set to 0.

90a **ROI Params** ROI parameters for Soft ROI mode. Not currently implemented.

U64 **hout** Handle output

error **error out** The **error out** cluster passes error or warning information out of a VI to be used by other VIs.
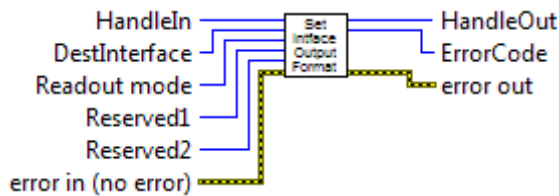
The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

I32 **ErrorCode**

1.2. New in BufferData.LLB

GetInterfaceOutputFormat.vi

Displays options for the data interface to the pco.cameras. This VI can be used to query the readout modes on the pco.edge, and various output format options on the HDSDI interface of the pco.dimax.
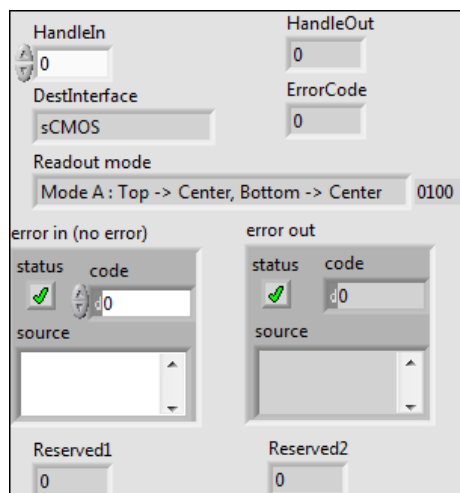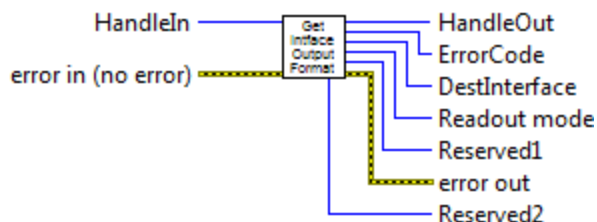
U64 **HandleIn** Handle to selected camera

error in (no error) The **error in** cluster can accept error information wired from VIs previously called.

DestInterface Selected interface type:

Reserved2 Reserved for future options

Reserved1 Reserved for future options

ErrorCode

HandleOut Handle for selected camera

error out The **error out** cluster passes error or warning information out of a VI to be used by other VIs.

Readout mode Data interface options

SetInterfaceOutputFormat.vi

Controls options for the data interface to the pco.cameras. This VI can be used to set the readout modes on the pco.edge, and control various options on the HDSDI interface of the pco.dimax.

**DestInterface** Select Interface type

| U16 | **Reserved2** Reserved for future options |

| U16 | **Reserved1** Reserved for future options |

| U64 | **HandleIn** Handle to selected camera |

| | **error in (no error)** The **error in** cluster can accept error information wired from VIs previously called. |
| U16 | **Readout mode** Data interface options. |

| I32 | **ErrorCode** |

| U64 | **HandleOut** Handle for selected camera |

| | **error out** The **error out** cluster passes error or warning information out of a VI to be used by other VIs. |

1.3. New in Sensor.llb

GetHWIOSignalDescriptor.vi

Provides details of the selected hardware signal, including signal names and I/O modes

Connector Pane

Front Panel

© PCO-TECH Support
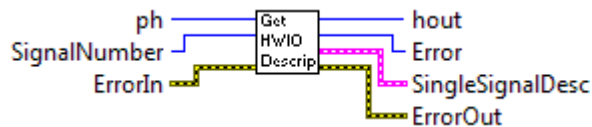
Controls and Indicators

`U64` **ph** Handle to a open camera connection

`ERR` **ErrorIn** The **error in** cluster can accept error information wired from VIs previously called.

`U64` **hout** Handle to a open camera connection (same as input handle)

`ERR` **ErrorOut** The **error in** cluster can accept error information wired from VIs previously called.

`DESC` **SingleSignalDesc** Information on the selected I/O signal

　`[abc]` **Names** Name given to this I/O signal, along with any alternates

　`U16` **SignalDefinitions** Flags for signal options. Option is available if bit is 1 (TRUE). Unlisted bits are reserved for future use.

　　Bit0: Signal can be enabled/disabled
　　Bit 1: Signal is status (output)

　　Other bits reserved for future use

　`U16` **SignalTypes** Flags for the selectable I/O options. Option is supported if bit is 1 (TRUE). Unlisted bits are reserved for future use

　　Bit0: TTL
　　Bit1: Higher than TTL voltage
　　Bit2: Contact closure
　　Bit3: RS485 differential signalling

　`U16` **SignalPolarity** Flags for level or transition triggering of I/O signal. Option is available if bit is 1 (TRUE). Unlisted bits are reserved for future use.

　　Bit0: Low level active
　　Bit 1: High level active
　　Bit 2: Rising edge active
　　Bit 3: Falling edge active

　`U16` **SignalFilter** Flags for signal filter options. Option is available if bit is 1 (TRUE). Unlisted bits are reserved for future use.
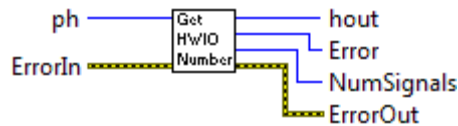
　　Bit0: Filter can be disabled.
　　Bit1: Medium-level filter availability (t > 1 us)
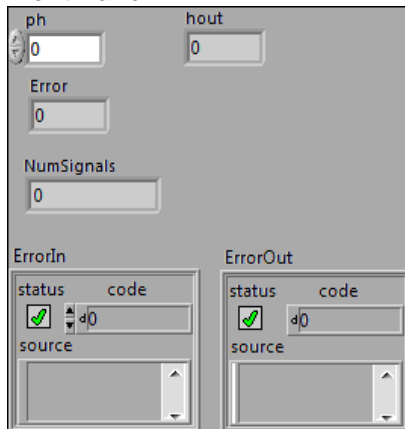　　Bit 2: High level filter available (t > 100 ms)

GetHWIOSignalCount.vi

Returns the number of Hardware I/O signals available.

Connector Pane

Front Panel

Controls and Indicators

ph Handle to camera

ErrorIn The **error in** cluster can accept error information wired from VIs previously called. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs.
Error

hout Handle out

**NumSignals** Number of Hardware Input/Output signals available for use.

ErrorOut The **error in** cluster can accept error information wired from VIs previously called.
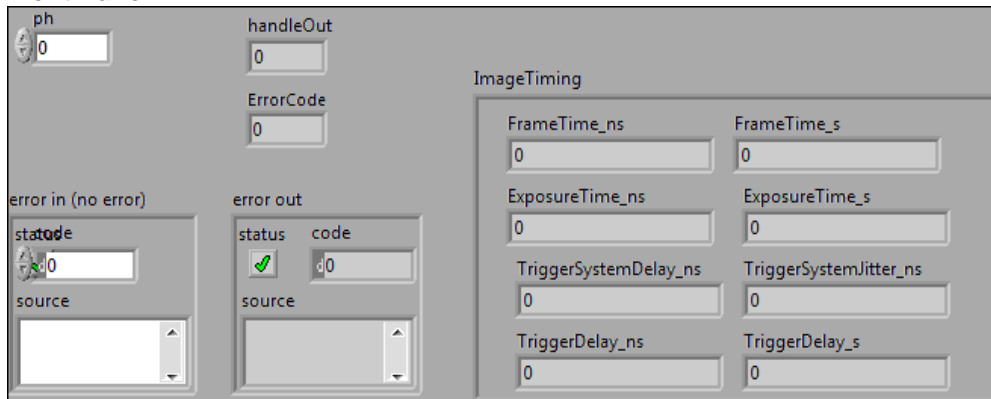
1.4. New in TimingControl.llb:
GetImageTiming.vi

Returns detailed timing information, including system delays and jitter. Can be used to make more accurate determination of the exposure timing.

Connector Pane
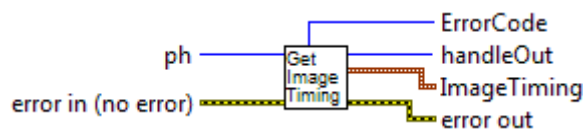
Front Panel

Controls and Indicators

`U64` **ph** Handle for the camera

`[===]` **error in (no error)** The **error in** cluster can accept error information wired from VIs previously called.

`[===]` **error out** The **error out** cluster passes error or warning information out of a VI to be used by other VIs.

`I32` **ErrorCode**

`U64` **handleOut** Handle to the camera

`>906` **ImageTiming** Detailed information about the exposure and delay timing, including internal delays and jitter.

    `U32` **FrameTime_ns** Time required, with FrameTime_s, to acquire one frame

    `U32` **FrameTime_s** Time required, with FrameTime_ns, to acquire one frame

    `U32` **ExposureTime_ns** This + ExposureTime_s = exposure time

    `U32` **ExposureTime_s** This + ExposureTime_ns = exposure time

    `U32` **TriggerSystemDelay_ns** Minimum internal trigger system delay in ns

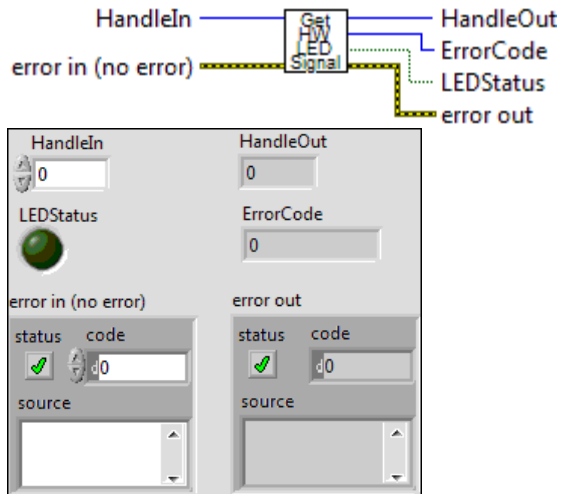   `U32` **TriggerSystemJitter_ns** Maximum possible jitter, +/- in ns

    `U32` **TriggerDelay_ns** This + TriggerDelay_s = total delay, including programmed and system delays

U32 **TriggerDelay_s** This + TriggerDelay_ns = total delay, including programmed and system delays

GetHWLEDSignal.vi

Gets the status of the LED at the back of the pco.edge camera

U64 **HandleIn** Handle to the requested camera

error in (no error) The **error in** cluster can accept error information wired from VIs previously called

U64 **HandleOut** Handle to the requested camera

error out The **error out** cluster passes error or warning information out of a VI to be used by other VIs.
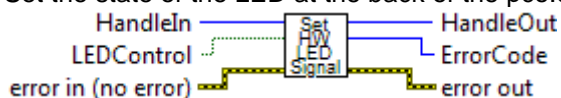
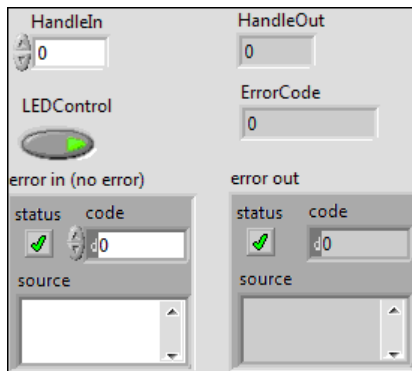**I32** **ErrorCode** 0 if no error, otherwise an error code will appear here

TF **LEDStatus** LED enabled state:

ON = Enabled
OFF = Disabled

SetHWLEDSignal.vi

Set the state of the LED at the back of the pco.edge camera

U64   **HandleIn** Handle to the requested camera

[...]   **error in (no error)** The **error in** cluster can accept error information wired from VIs previously called.

TF   **LEDControl** Enable or disable LED operation:
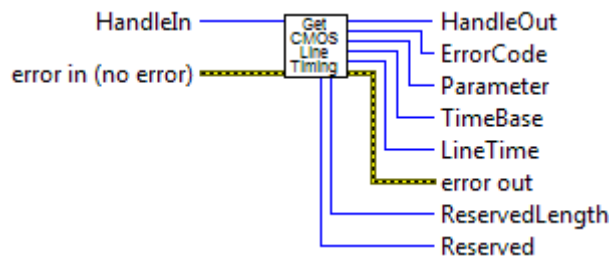
       ON = Enabled
       OFF = Disabled

U64   **HandleOut** Handle to the requested camera

[...]   **error out** The **error out** cluster passes error or warning information out of a VI to be used by other VIs.

I32   **ErrorCode** 0 if no error, otherwise an error code will appear here

GetCMOSLineTiming.vi

Sets the timing for an individual line in sCMOS cameras. This enables the camera scan rate to be matched to a moving source.

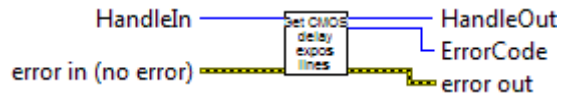|	
U64	**HandleIn** Handle to the selected camera
ERR	**error in (no error)** The **error in** cluster can accept error information wired from VIs previously called.
I32	**ErrorCode**
U64	**HandleOut**
ERR	**error out** The **error out** cluster passes error or warning information out of a VI to be used by other VIs.
U16	**Parameter** Displays the operation of the lime timing mode

CMOS_LINETIMING_PARAM_OFF = 0
CMOS_LINETIMING_PARAM_ON = 1 |
| U16 | **TimeBase** Time base (units of time) for the exposure setting.

0 - Nanoseconds
1 - Microseconds
2 - Milliseconds |
U32	**LineTime** Time of one line, in time base units.
U32	**Reserved** Reserved for future use
U16	**ReservedLength** Reserved for future use

SetCMOSLineTiming.vi

Sets the timing for an individual line in sCMOS cameras. This enables the camera scan rate to be matched to a moving source.
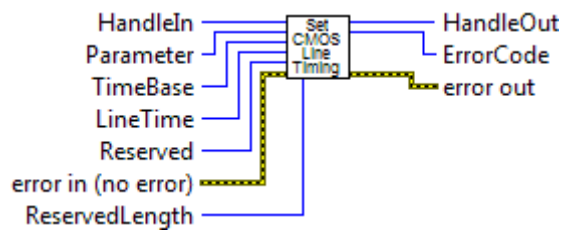
| U64 | **HandleIn** Handle to the selected camera |

| U16 | **Parameter** Controls the operation of the lime timing mode |

CMOS_LINETIMING_PARAM_OFF = 0
CMOS_LINETIMING_PARAM_ON = 1

| U16 | **TimeBase** Time base (units of time) for the exposure setting. |

0 - Nanoseconds
1 - Microseconds
2 - Milliseconds

| U32 | **LineTime** Time of one line, in time base units. |

| U32 | **Reserved** Reserved for future use |

| U16 | **ReservedLength** Length of reserved parameter list |

| | **error in (no error)** The **error in** cluster can accept error information wired from VIs previously called |
| I32 | **ErrorCode** |

| U64 | **HandleOut** |

| | **error out** The **error out** cluster passes error or warning information out of a VI to be |

used by other VIs.

GetCMOSLineExposureDelay.vi

This command returns the exposure and delay time for a frame. It is only valid when the line timing parameter is set to CMOS_LINETIMING_PARAM_ON.

U64	**HandleIn** Handle for the selected camera

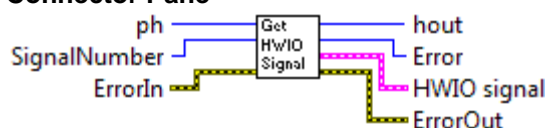| ⌷	**error in (no error)** The **error in** cluster can accept error information wired from VIs previously called
I32	**ErrorCode** 0 if no error, otherwise an error code appears here

U64	**HandleOut** Handle of the selected camera

⌷	**error out** The **error out** cluster passes error or warning information out of a VI to be used by other VIs.

U32	**ExposureLines** Number of lines required for exposure

U32	**DelayLines** Number of lines between consecutive frames

U32	**Reserved** Reserved for future features

U16	**ReservedLength** Length in bytes of the reserved structure

SetCMOSLineExposureDelay.vi

This command sets the exposure and delay time for a frame. It is only available when the line timing parameter is set to CMOS_LINETIMING_PARAM_ON.
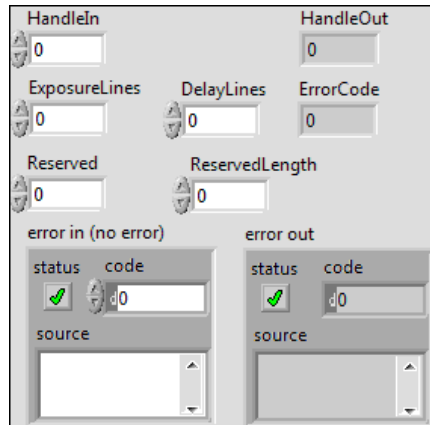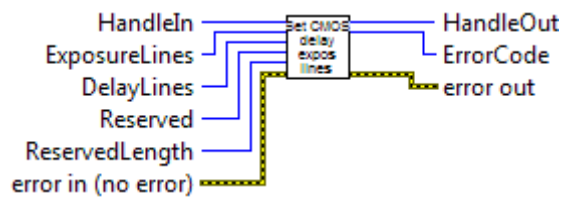
U64	**HandleIn** Handle for the selected camera	
U32	**ExposureLines** Number of lines required for exposure	
U32	**DelayLines** Number of lines between consecutive frames	
U32	**Reserved** Reserved for future features	
U16	**ReservedLength** Length in bytes of the reserved structure	
[error]	**error in (no error)** The **error in** cluster can accept error information wired from VIs previously called.	
I32	**ErrorCode** 0 if no error, otherwise an error code appears here	
U64	**HandleOut** Handle of the selected camera	
[error]	**error out** The **error out** cluster passes error or warning information out of a VI to be used by other VIs.	

GetHWIOSignal.vi

Provides details of the selected hardware signal, including signal names and I/O modes

Connector Pane

Front Panel

Controls and Indicators

`U64` **ph** Handle to a open camera connection

`ᴇᴇᴇ` **ErrorIn** The **error in** cluster can accept error information wired from VIs previously called.

`I32` **Error**

`U64` **hout** Handle to a open camera connection (same as input handle)

`ᴇᴇᴇ` **ErrorOut** The **error out** cluster passes error or warning information out

`ᴇᴇᴇ` **HWIO signal** Cluster of parameters describing the current configuration of the selected HWIO signal

  `U16` **SignalNumber** Number of the HWIO signal selected

  `TF` **Enabled** True if signal enabled, false if signal disabled

  `U16` **Type** Signal level

  `U16` **Polarity** Sense of signal voltage, or edge if edge sensitive

  `U16` **FilterSettings** Level of filtering on signal

---

author: MM   03.09.2015

© PCO-TECH Support

U16   **Selected** Indicates which variant of the signal is selected, if signal has more than one variant

SetHWIOSignal.vi

Controls the properties of the selected hardware signal, including modes and polarities

Connector Pane
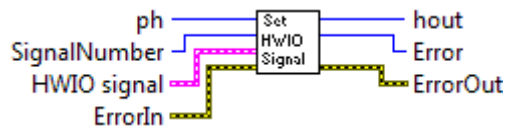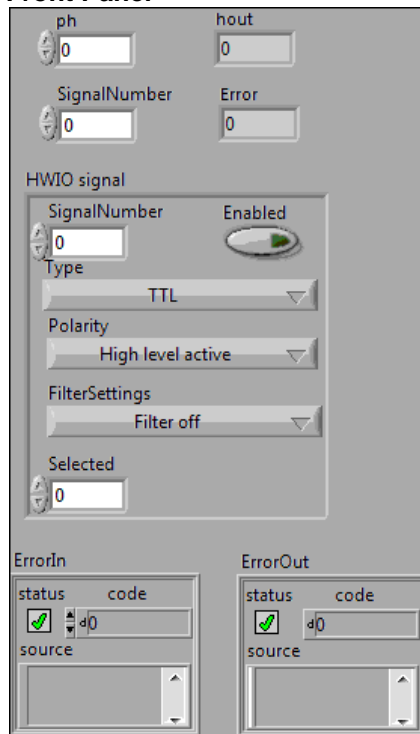
```
            ph ──────┐ Set ┌────── hout
  SignalNumber ──────┤HWIO├─────── Error
   HWIO signal ══════╡Signal╞═════ ErrorOut
       ErrorIn ══════┘
```

Front Panel

```
   ph              hout
  [0]             [0]

   SignalNumber    Error
  [0]             [0]

  HWIO signal
   SignalNumber    Enabled
  [0]             (   )
   Type
  [    TTL      ▽]
   Polarity
  [ High level active ▽]
   FilterSettings
  [    Filter off   ▽]
   Selected
  [0]

  ErrorIn                  ErrorOut
  status    code           status    code
  [✓]    [d0]              [✓]    [d0]
  source                   source
  [          ]             [          ]
```

Controls and Indicators

U64   **ph** Handle to a open camera connection

⊡   **ErrorIn** The **error in** cluster can accept error information wired from VIs previously called.

U16   **SignalNumber**

⊡   **HWIO signal** Cluster of parameters describing the current configuration of the selected HWIO signal

---

SignalNumber Number of the HWIO signal selected

Enabled True if signal enabled, false if signal disabled

Type Signal level

Polarity Sense of signal voltage, or edge if edge sensitive

FilterSettings Level of filtering on signal

Selected Selects which variant of the signal is used, if signal has more than one variant

Error

hout Handle to a open camera connection (same as input handle)

ErrorOut The **error out** cluster passes error or warning information out of a VI to be used by other VIs.

status The **status** boolean is either TRUE (X) for an error, or FALSE (checkmark) for no error or a warning.

The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

code The **code** input identifies the error or warning.

The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

source The **source** string describes the origin of the error or warning.

The pop-up option **Explain Error** (or Explain Warning) gives more information about the error displayed.

GetHWIOSignalTiming.vi

Queries the functionality on any hardware I/O signal that has multiple modes of operation.

ErrorIn The **error in** cluster can accept error information wired from VIs previously called.

HandleIn Handle to the selected camera

index Selects the HWIO signal to be queried

select On HWIO signals that have multiple functions, this selects the function of the signal. e.g. exposure out or busy out.

ErrorOut The **error out** cluster passes error or warning information out of a VI to be used by other VIs.

GetHWIOSignalTimingStructure Out

code Low level control code

length Length of message

index Index of signal queried

select Selected function of this signal

type Type of signal

parameter Timing details
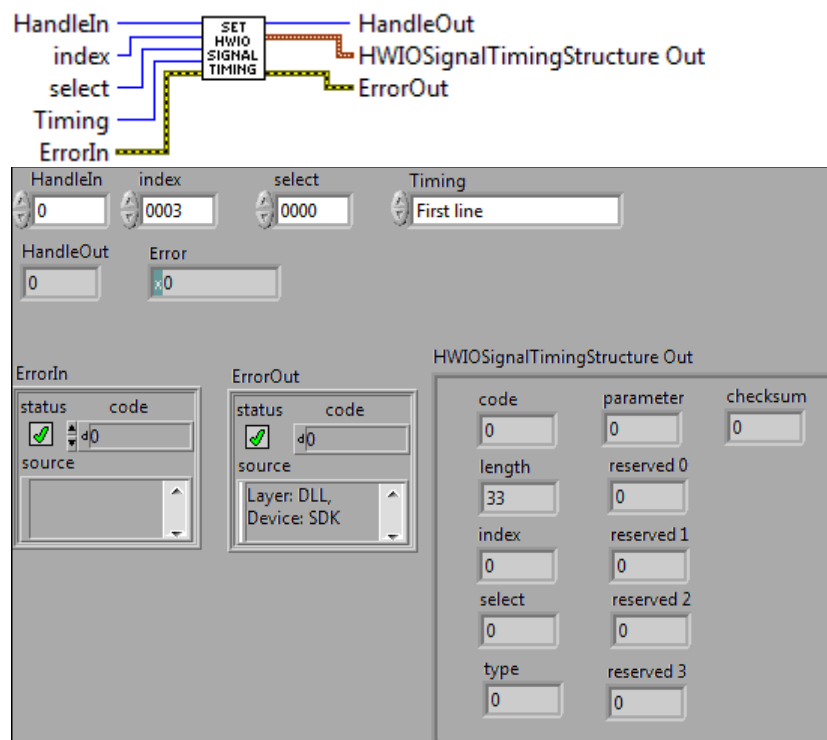
U32	**reserved 0**
U32	**reserved 1**
U32	**reserved 2**
U32	**reserved 3**
U8	**checksum**
U64	**HandleOut** Handle to selected camera

SetHWIOSignalTiming.vi

Selects the functionality on any hardware I/O signal that has multiple modes of operation This can be used to configure output signals on the pco.edge.

ErrorIn	The **error in** cluster can accept error information wired from VIs previously called.
HandleIn	Handle to selected camera
Timing	Selects the timing options for the HWIO signal
index	Selects the HWIO signal to be configured

`U16` **select** On HWIO signals that have multiple functions, this selects the function of the signal. e.g. exposure out or busy out.

`ErrOut` **ErrorOut** The **error out** cluster passes error or warning information out of a VI to be used by other VIs.

`Sub` **HWIOSignalTimingStructureOut**

  `U16` **code** Low level control command

  `U16` **length** Length of command message

  `U16` **index** Selected signal

  `U16` **select** Selected function

  `U32` **type** Type of signal

  `U32` **parameter** Timing options

  `U32` **reserved 0**

  `U32` **reserved 1**

  `U32` **reserved 2**

  `U32` **reserved 3**

  `U8` **checksum**

`U64` **HandleOut** Handle to selected camera

1.5. New In GeneralControlStatus.llb

GetFanControlStatus.vi

Returns information on fan set point and control mode. Indicates if fan is under automatic or manual control.
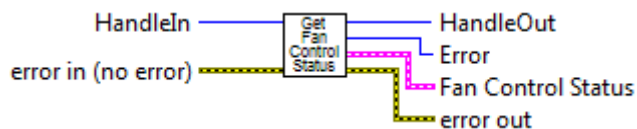
U64 **HandleIn**

error in (no error) The **error in** cluster can accept error information wired from VIs previously called. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs.

U64 **HandleOut**

error out The **error out** cluster passes error or warning information out of a VI to be used by other VIs.

Fan Control Status Contains information about the fan control status in the pco.edge.

> **TF** **Manual control** Indicates if the fan is available for manual control
>
> FALSE - Fan is under automatic control
> TRUE - Fan can be controlled manually.

> **U16** **FanMin** Minimum value for fan speed. This value is a percentage of the maximum fan voltage.

> **U16** **FanMax** Maximum value for fan speed. This value is a percentage of the maximum fan voltage.

**FanStep** Step size for the fan speed set point. This is a percentage of the maximum fan voltage.

FanSetPoint Sets the operating point for the fan. This is a percentage of the maximum fan voltage.

FanActual Actual operating point of the fan, as a percentage of the maximum fan voltage.

Reserved Reserved for future control options

SetFanControlStatus.vi

Controls the mode of fan opration, and sets the operating point of the fan in manual mode. Only pco.edge cameras with internal fan are supported.
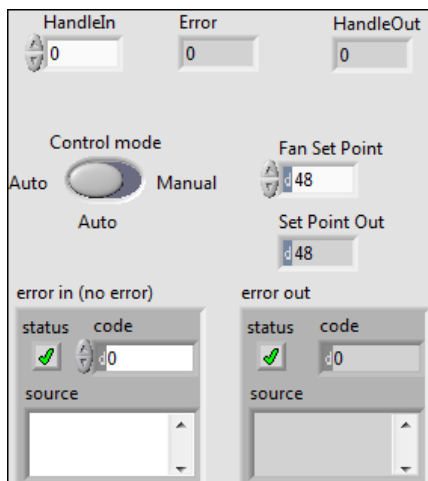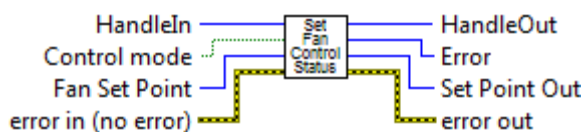
HandleIn

error in (no error) The **error in** cluster can accept error information wired from VIs previously called. Use this information to decide if any functionality should be bypassed in the event of errors from other VIs.

Control mode Switches the fan control mode from automatic to manual.

    FALSE = Automatic (default)
    TRUE = Manual

Fan Set Point In manual mode, this sets the fan speed. This input is ignored if the fan is in auto mode. The number represents a percentage of the maximum fan voltage. To find

the range of acceptable values, use GetFanControlStatus.vi.

`U64` **HandleOut**

`error` **error out** The **error out** cluster passes error or warning information out of a VI to be used by other VIs.

`U16` **Set Point Out** Current fan setpoint. The number represents a percentage of the maximum fan voltage. To find the range of acceptable values, use GetFanControlStatus.vi.